Soft-Core, Multiple-Lane, FPGA-based ADCs for a Liquid Helium Environment

Zikun Xiang*
University of Science and Technology of China

Tianqi Wang†
University of Science and Technology of China

Xi Jin¶
University of Science and Technology of China

Tong Geng‡
Boston University

Martin Herbordt‖
Boston University

Tian Xiang§
Suzhou graphichina technology co.ltd

ABSTRACT

System control and the collection of analog signals are fundamental tasks in many embedded computer systems such as are found in automotive, communication, and sensor network domains. Often latency is critical and FPGAs are an attractive alternative. Traditionally, external ADC (analog to digital converter) chips have been used for analog to digital signal conversion and transfer of the digital signals to the FPGA(s).

In large-scale quantum system experiments, the implementation of this classic control infrastructure is a challenge. In particular, the FPGA-based control system must work in a liquid helium environment at about 4°K. To improve the system's reliability, we need to integrate multiple lanes of soft core ADC directly into the FPGA.

In this paper we propose a method of building high-speed ADCs with time to digital converters (TDCs). The experimental results show that the ADCs can achieve a sampling rate of 100Msa/s with a 6-bit resolution for signals ranging from 0 to 3 V. In our design the ADC uses primarily the ISERDES logic of a Xilinx FPGA plus a small number of CLBs. Our design can integrate 24 lanes of soft-core ADCs into a Xilinx XC7A100t-2csg324 FPGA.

1 INTRODUCTION

Some of the most promising research areas in physics need cryogenic environments. These include condensed matter, superconductor, and, in particular, quantum physics. A goal in quantum physics is achieving quantum *supremacy* or *advantage*, i.e., the ability of quantum computing devices to solve problems that classical computers (in practice) cannot. Quantum computing is implemented in several different technologies, but most require working at deep-cryogenic temperatures.

With the vast amount of research dedicated to quantum computing in recent years, the search for systems built at cryogenic temperatures has intensified. As most quantum bits (qubits) are operating at sub-kelvin temperatures (temperature lower than 1°K), which are necessarily cryogenically cooled, there is a substantial temperature difference between the temperature of the qubits and that of the control electronics. This leads to two critical problems. First, the wiring into the qubits' cryogenic environment introduces a substantial heat load. And second, the added distance increases latency with a heavy impact on performance. Implementing control in the cryogenic environment (at 4°K) allows us to decrease both the thermal load and the physical distance to the qubits.

*e-mail: zk@mail.ustc.edu.cn
†e-mail:tqwang@mail.ustc.edu.cn
‡e-mail:tgeng@bu.edu
§e-mail:tian.xiang@graphichina.com
¶e-mail:jinxi@ustc.edu.cn
‖e-mail:herbordt@bu.edu

The implementation of a classical control infrastructure for large-scale quantum computers is challenging due to the need for integration and processing time [9, 10]. To date, some research has shown the feasibility of operating an FPGA at 77°K. The focus there is on cryogenic digital data links for physics experiments. Xilinx, Microsemi, and Altera/Intel FPGAs are fully or partially operational at 77°K. There are also several designs that use the FPGA operating at 4°K as quantum controllers [2]. In Xilinx FPGA product line, a Spartan 3 and (recently) an Artix 7 were shown to work at 4°K [3].

A recongurable cryogenic platform (in Figure 1) for the classical control of quantum processors has been proposed [8]. In this system, we need a DAC to send a wave as a control signal to manipulate the Quantum Processor. The processor then returns a signal carrying the information processed. Because of time and energy consumed in cooling cycles, the architectures should be reconfigurable. FPGAs or CPLDs are therefore appropriate [5].

At room temperature, FPGAs are often used as the digital controllers due to their performance, small size, integration, and flexibility [2, 6]. ADCs and DACs generally work with FPGAs as external devices. But if an external ADC chip is used, that part of the circuit is necessarily fixed and cannot easily be changed. If additional devices are required, the size of system increases requiring extra connectors and PCBs and overall system power. This direct approach is not always preferred, because it has limited scalability and flexibility and sometimes cannot achieve desired compactness.

A better approach in those cases is to integrate a hard ADC primitive into the FPGA. Xilinx includes an on-chip ADC (XADC) in some of its 7-series FPGAs. The main short-coming of this solution is that the ADC has fixed parameters and cannot be used in cryogenic platforms. Also, this approach takes away die area.

Another solution, that we explore here, is an ADC implemented directly into reconfigurable hardware, which can then be used as any other IP. The advantages of this approach include having the digital information directly available inside the FPGAs allowing faster processing of the digitized data. Moreover, an internal ADC can be interfaced easily and flexibly allowing the designer to adjust the number of ADCs to the requirements of the application. A soft core ADC in FPGA is proposed in [3]. In this architecture, an LVDS transceiver is used as a comparator and two time-to-digital-converters (TDCs) are used to convert the signals. Except for a small single resistor, needed to create the analog reference ramp, the ADC is implemented completely in the FPGA. In this way the number of I/O pins can be reduced.

The TDC core is the most important part of this design. Usually TDCs in FPGAs consist of a delay chain implemented with a carry chain. The delay of each link of the carry chain is 18 22ps, so high precision is possible. But carry chain logic is limited – when a large-scale system is needed, a single FPGA cannot meet the requirement. In comparison with ASIC ADCs, FPGA TDC-based ADCs have the usual disadvantages. However, ASIC ADCs have a performance limitation at cryogenic temperatures: there are significant changes in analog behavior of the various components.

978-1-5386-5989-2/18/$31.00 ©2018 IEEE

In this paper, we propose a new method of implementing TDC-based ADCs, which can reduce the temperature and voltage influence and substantially improve ADC integration. We use the ISERDES logic to implement a multi-phase clock TDC. The delay of the TDC is determined by the clock period phase. In our design, the only required is I/O; the number of ADCs we can integrate into a single FPGA scales with the number of I/Os.

The remainder of this paper is organized as follows. In section 2, we give background about the problem and provide motivation. In section 3, we review previous work on ADCs in FPGAs, both soft and hard core. In section 4, we present the architecture of our soft core ADC with ISERDES TDC. Finally section 5 reports the area consumption and performance of the design.

2 Background and Motivation

Building semiconductor quantum chips is one of the most important research projects in the world. Generally, the qubits operate at 20mK 100mK due to thermal noise, and the qubits sample driven by an external direct current (DC) power source cross a large temperature difference of about 300K [7]. In quantum systems (in Figure 1, the waves indicate logic messages like add and subtract. The combination of different amplitude, phase, and frequency can encode the compute messages. There are two signal components that work at room temperature: signal generation and signal analysis.

Signal generation consists of MW pulse AWG and microwave generator. The final signal is mixed by the microwave and DAC signal. The mixed signal (reference input signal) is transferred to the qubit system in 20mK across several attenuators and a DC block. The reference signal and the feedback signal from the qubit system are sent to the circulator. The return signal is transmitted to the signal analyzer.

Signal analysis consists of two ADCs, three filters, two mixers, and a microwave generator. The return signal comes from 20mk and passes through the high electron mobility transistor (HEMT), which works at 4K. Finally, the signal arrives at the signal analyzer at room temperature. Because of the attenuation in the reference signal input circuit, the signal must be amplified at room temperature. In this system, there is a long-distance transmission over a huge temperature difference. This adds clutter into the signal and make the system unstable. Use of filters is therefore significant. The signal is divided into I and Q and sent to two ADCs through two filters in each channel.

In order to build a feedback loop in this system, it takes a long time to transmit the signal making it difficult to correct the error. The coherent time of the qubit should be short enough to support data loopback. And the transmission and and calculation times are strictly limited. Thus a cryogenic digital or analog circuit, which receives data and sends control signals, can solve almost all of these problems.

In quantum computing system, each physical qubit always works at sub-1K temperature and requires a classical control loop capable of reading the qubit state and sending a signal to control system based on those decisions. If we want the loop system to be effective, the feedback loops should perform a complete correction cycle faster than the coherence time of the physical qubits. That time, however, depends on the qubit technology and system and varies from us to ns.

This requirement implies that the classical control systems need to be in the us for measuring, calculating, and correcting a physical qubit state message within those time frames to feed back the control signal. With the increasing scale of quantum systems, more data must be processed in a single time frame. So the system control units must be smaller enough physically.

The hardware infrastructure will most likely be implemented in a micro-architecture which works at cryogenic temperatures. So the hardware must be close to the qubits and easy to control. Because of the cryogenic environment, the hardware should be programmable and reconfigurable to reduce the time and power consuming cooling cycles. Due to the reconfigurable requirement, the reasonable choice is an FPGA or CPLD based on CMOS.

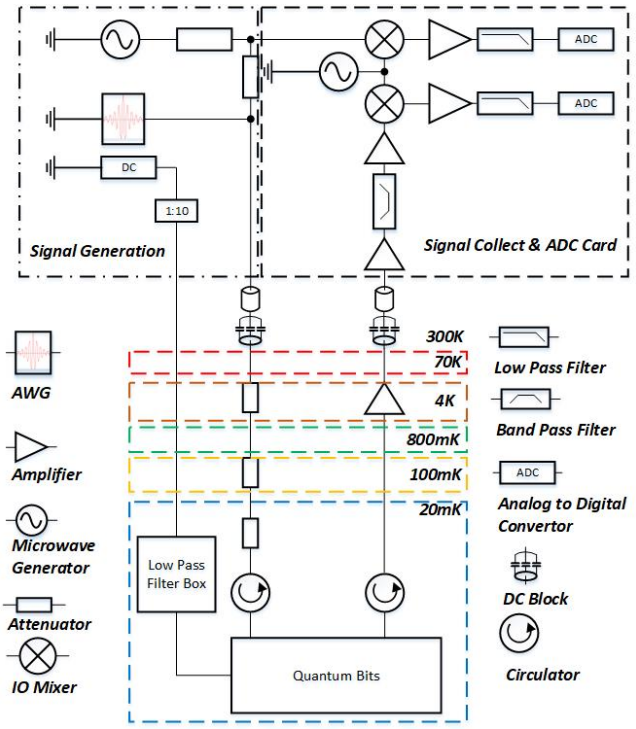

Figure 1: Overview of quantum experiments

The most important part of the system is sampling component including the ADC. In this paper, we focus on the architecture of the ADC based on FPGAs. We can use ADCs independently to measure and calculate the signal. Then the data, which we focus on or need, is sent to the host computer operating at room temperature. We can therefore understand the FPGA behavior by testing the signal transferred from the cryogenic condition.

To date, the ADCs based on FPGA utilize the general resource with multiphase clock TDCs or the carry chain with tapped TDCs. The tapped delay TDC has high time accuracy and flexibility. But it uses a carry chain to achieve high time accuracy which is limited to single FPGA. Also, the performance of the TDC varies sharply with temperature and voltage changes. To improve the level of integration, we propose a multiphase clock TDC with ISERDES to replace the original tapped TDC. Another advantage is that the ISERDES is stable with temperature and voltage changing, which makes the system operate better at cryogenic temperatures.

3 Related Work

The architecture of control electronics of quantum devices is proposed by Homulle, et al. [4]. The system consists of multiplexers and demultiplexers which could be integrated with qubits, amplifiers, attenuators, ADCs, DACs, converting mixers, oscillators, and reconfigurable digital logic. In our work, we are mainly concerned with the ADC architecture.

The quantum feedback system works as a loop. First, the analog signals coming from the qubits are converted to digital through the ADC on FPGA. Then the data can be analyzed and calculated to get the real qubit state and to measure the offset of the system. To reduce the resource consumption, the return signals are generated directly by the FPGA or DACs.

978-1-5386-5989-2/18/$31.00 ©2018 IEEE

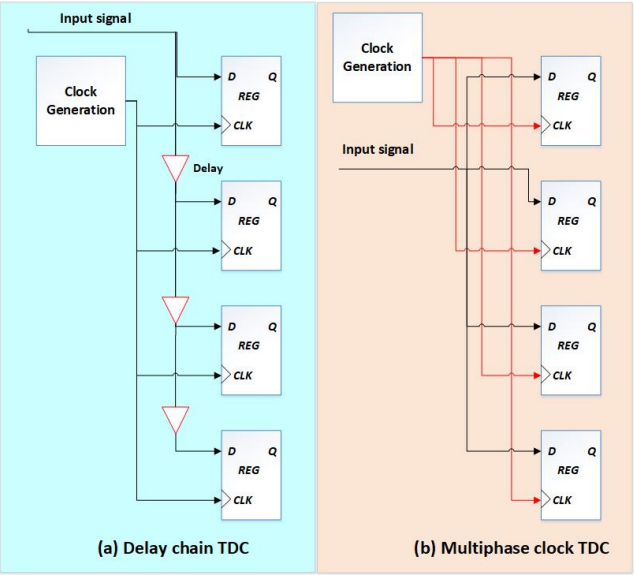

Figure 2: Comparisoin of two TDC architectures

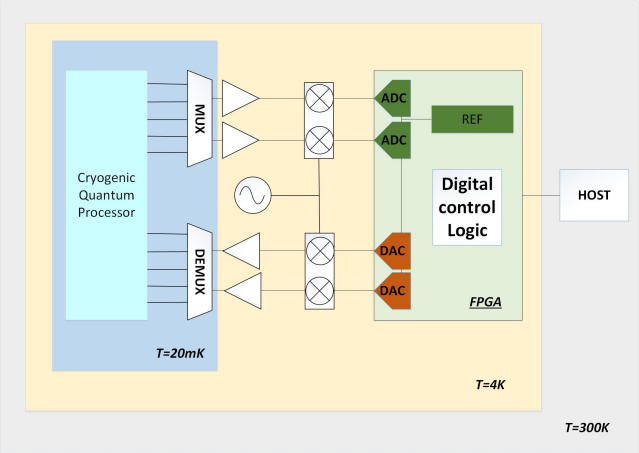

Figure 3: System Overview

There are several challenges to getting FPGAs to operate at cryogenic temperatures. First, is the effect of power dissipation on the cooling system. The power requirement is therefore more strict because of the limitations in refrigeration technology; added power makes the cryogenic environment unstable. Then, the digital integrated circuits must be carefully test in the extreme environment. Before the FPGA is tested at cryogenic temperatures, the platform must be tested in liquid helium directly. Finally, the cooling system power should be maximized.

We now describe how to implement the physical platform. The system needs a special printed circuit board (PCB) for cryogenic conditions. In the PCB, there are passive components and connectors around the FPGA. Those components must be able to reliably survive cooling cycles. Moreover, they need to work without performance degradation. The key is choosing a minimal number of discrete components certified for operation over a temperature range which is wider than industrial and military standards. The most important challenge is how to properly operate FPGAs at cryogenic temperatures. Since we have no control over its manufacture, the issue is one of characterization.

Previous work implementing ADCs on FPGAs is based on delta-sigma modulators [1]. In this architecture, the pulse is generated and compared with the analog signal using a feedback loop. However, this approach is only capable of detecting changes of input analog signal rather than its absolute value.

Another approach to implement ADCs with a TDC uses a slope scheme. The design of Wu, et al. [11] is implemented using a reduced number of external devices. The reference ramp is generated by the passive edge of RC an network (3 resistors and 1 capacitor). The comparators use the differential inputs of the FPGA and signals are digitized by TDCs. The performance of this approach is very limited, and the level of integration and compactness cannot be improved because of the number of components which are not implemented on the FPGA.

Homulle and Charbon propose an ADC solution using FPGAs [3]. They use an analog reference ramp to compare with the signal to be measured. Then the TDC gets the clock domain signal which is compared by the LVDS. They use a clock generator to divide the clock into six phase and then use these six clock signals in different phases to collect analog signal data. Combining the six results, they can achieve high speed and highly precise ADC [4]. They also tested this approach in a cryogenic platform, getting a good result.

TDC has been used for low cost, highly precise, high-speed converters in FPGAs. There are two ways to implement them; both are based on the same principle. The TDC has two sub-modules: the flip-flops used to sample the signal and counters to record the accurate time. The maximum frequency of the system limits the TDC sampling clock period. So the main problem designers have to face is how to reach a time resolution better than the clock frequency. To improve TDC performance, a modified sampling unit is used, rather than a single flip-flop, to reduce the least significant bits (LSBs).

In the first method, several flip-flops are driven by the same clock to sample the hit signal which has been delayed by the tapped delay chain. The arrival time of the signal to every flip-flop is same. The difference between each flip-flop is the delay time. By analyzing the output of all flip-flops, the data module can calculate the time between the start and stop signal. This architecture best minimizes RMS error of the TDC, it is only 10ps, but it is hard to implement in FPGAs due to the delay taps using the carry chain. The delay time is seriously affected by variations in temperature and voltage.

The second method uses a multi-phase clock architecture: the input signal is sampled with a flip-flop driven by a set of phase-shift clocks. Those clocks are generated by a phase-locked loop (PLL) inside the FPGA, so the skew of every clock is stable with temperature and voltage changes. As with the first method, output of the flip-flops indicates the time of the hit signal. This architecture uses distributed flip-flop in FPGA to build the ADC. And it requires that the routing delay of the clock to flip-flop be the same. However, this is hard to achieve with distributed flip-flops. The traditional multi-phase clock TDC is not good for use.

Compared with the carry chain TDC (see Figure 2), the multiphase clock TDC provides a higher level of integration. Xiang describes a new architecture of the multiphase clock TDC, which shows better RMS error compared with the traditional multiphase clock TDC [12]. Unlike traditional multiphase clock TDC, this architecture integrates the sampling unit in an I/O tile pin pair (positive and negative pins), instead of implementing it using general FPGA resources.

With the method mentioned above, our work can integrate more TDC in single FPGA. So it is possible to instance more soft-core ADC in the chip. This meet the requirement of quantum computing application.

978-1-5386-5989-2/18/$31.00 ©2018 IEEE

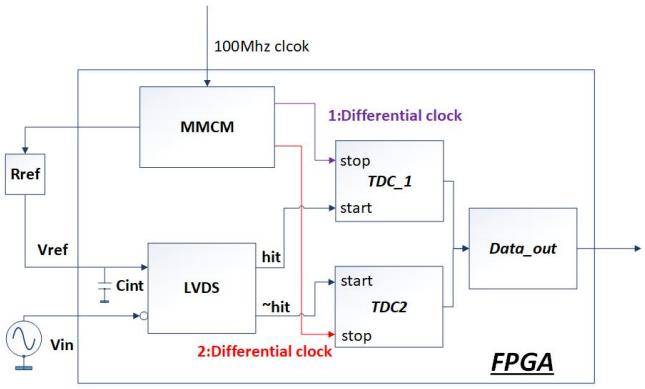

Figure 4: FPGA-based ADC

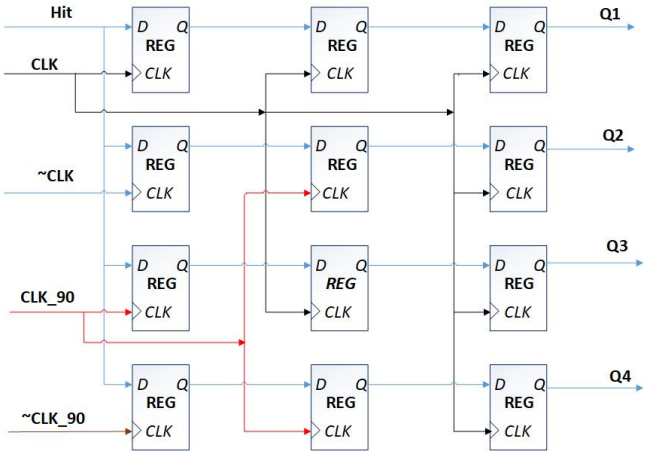

Figure 5: ISERDES flip-flop

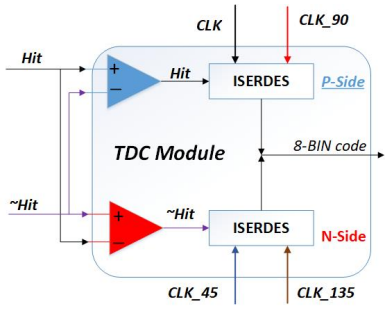

Figure 6: TDC Implementation

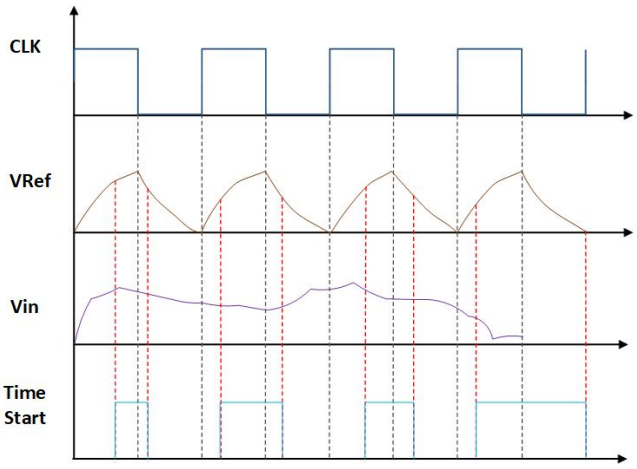

Figure 7: Timing diagram of ADC code. The reference clock signal at the top is used to generate VRef.

4 Architecture

4.1 ADC System Architecture

The architecture, which is depicted in Figure 4, is composed of an external resistor, a comparator, and a TDC module. The LVDS transceiver, acting as a comparator, can compare two analog signals with a pair of differential signals. The two TDCs are used for the time measurement. The mixed-mode clock manager (MMCM) provides multiphase clocks for TDC and Vout for the reference ramp. All the clocks in the TDC are generated by single phase-locked loop (PLL), so they share a same routing delay from PLL to TDC.

First, the MMCM receives an input 100 MHz clock (from an external source) and provides a high-frequency clock signal V_{out} that charges and discharges the RC circuit consisting of an external resistor R_{ref} and the input capacitance C_{int} of the LVDS input buffer. The frequency of V_{ref} determines the sampling rate.

Next, the LVDS input buffer is used to compare the analog input signal Vin with the reference analog signal. This is the same with the Homulle design [3]. But because of the new TDC approach, we use another LVDS input buffer to compare the signal exporting a pair of differential signals. So the differential signal '1' at the output of input buffer is generated as long as the reference voltage is higher than the analog input. The actual conversion to a digital code is performed by TDCs.

Two TDCs are added to measure the rising and falling edge of the compared signal. In our design, the multiphase clock TDC consists of a sampling unit and (shown in Figure 4) a data synchronization unit. The TDC operates at 800Mhz, enabling a pair of differential ISERDES blocks. We only use the I/O block to build the TDC so it reduces the overall FPGA logic utilization. With 1/8 clock period precision, we can cover a whole clock period. The ISERDES output is then latched and decoded to a binary value in the decode module. The decoder also runs at 800Mhz.

Using traditional way, the sampling rate for this system can not higher than the maximum clock frequency of FPGA. To reach a higher sampling rate, we use the multiphase clock method to build TDC module and the module works on the maximum clock frequency of FPGA chip. Each clock is routed to a resistor and changed to a reference ramp at the LVDS input. So the sampling rate can reach the clock frequency.

Outside of the FPGA is an external resistor R_{ref} used to generate the reference voltage. The system therefore has three inputs and two outputs. The waveforms representing the timing diagram of the system are depicted in Figure 7. The clock signal drives the overall operation of the circuit. It is used to capture input data and drives the output signal V_{out} connected to the RC network, generating a reference ramp.

4.2 TDC Structure

In the TDC, the sampling unit is based on the multiphase clock architecture and uses 8 equidistant phase-shifted clocks to collect the hit signal. The data synchronization unit then synchronizes the 8-BIN code to a slow clock domain. The different clocks are generated by the MMCM.

The sampling unit based on the I/O Tile is the most important part in TDC. So we use dedicated logic in the Xilinx I/O Tile, which

978-1-5386-5989-2/18/$31.00 ©2018 IEEE

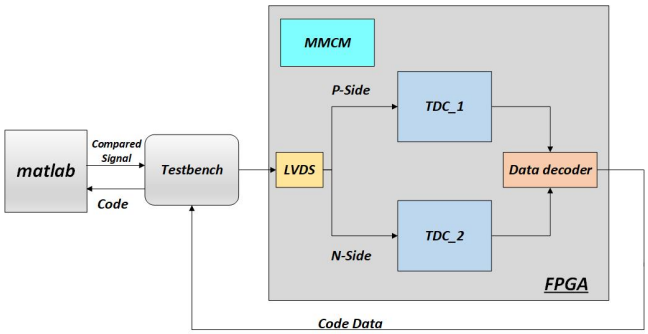

Figure 8: Verification-System

has two dedicated serial-to-parallel converters (ISERDESes). The system shown in Figure 6 consists of a differential input buffer with complementary output (IBUFDS_DIFF_OUT) and two ISERDES. The IBUFDS_DIFF_OUT connects the P and N pins on the FPGA and complementary outputs into FPGA (P-side and N-side). An ISERDES on P-side and N-side connect the respective hits.

There are two clock domains, the high frequency sampling clock and a slow clock for output. The frequency in sampling clock is 8 times faster than the slow clock. The sampling unit works in the sampling clock domain while the encoder works in the slow clock domain. The data synchronization unit crosses the two clock domains and encodes the 8-BIN code from the sampling clock to the data clock (slow clock domain).

The ISERDES is a dedicated serial-to-parallel converter with specific clocking and logic features. Using ISERDES rather than distributed flip-flop avoids the additional routing timing complexities [13]. The ISERDES is configured in oversample mode, which is used to capture two phases of DDR data. In this mode, the ISERDES has two clock input ports, CLKB and OCLK, with a 90° phase difference between them. It collects input data on both rising and falling edges of the two clocks [12].

Figure 5 shows how an ISERDES samples hit signals and synchronizes the data to the clock. Two ISERDES can be considered as 8 equidistant phase-shifted clocks TDC, with two 90° phase shifted clocks. The output code is the BIN code. Because the frequency of the sampling clocks is 8 times higher than the data clock, the resolution of the BIN code is 64 times better than the coarse time.

4.3 Implementation

The most important system component is the TDC. The input signal of the TDC is generated by an input differential buffer. The buffer compares the reference ramp generated by an RC circuit and the unknown analog signal and exports a differential hit signal to the TDCs. The time of signal delay to ISERDES is then equivalent. The value in the flip-flop block on the ISERDES is sampled by the multi-phase clock generated by a single PLL. The timestamp measured in this way is converted into BIN-code. However, the frequency of the clock is too high, so that the design needs to use a slow clock to drive the buffer to store the code. We then use a slow clock to send the data out.

The main reason for us to use ISERDES is that the delay of the input hit signal to the buffer is the same. So the problem of the multi-phase clock TDC is solved. The precision of TDC BIN width is decided by the clock precision. In our design, the BIN width is 156ps.

To add ISERDES to projects, we add a fixed I/O pin to collect the reference signal and the analog signal. To ensure performance, best performance is achieved by manually placing the ISERDES. This reduces clock domain crossings.

4.4 System Verification

To evaluate the ADC's and TDC's performance, we need a given analog signal whose frequency, phase, and voltage can be adjusted. The verification system is shown in Figure 8.

First, we simulate the system in Modelsim so the analog signal is generated by Matlab and driven by a 200Ghz clock. Because the frequency of drive clock is much higher than the sampling clock, we can use it to generate the data which should be compared by the LVDS.

We then use Modelsim to simulate the ADC system with the compared input signal. The input signal is driven by the 200Mhz clock signal. And the data generated by Matlab, which is comparative results between the V_{out} and analog signal, should synchronize with the first sampling clock. After the signal is sampled by the TDC, the 8 BIN-code is decoded and output to Matlab.

Finally, Matlab collects the data and plots the figure with the analog signal. We can then calculate the ADC parameters with the adjusted analog input.

5 Experimental Results

5.1 Experimental Setup

Several parameters are used to evaluate the performance of ADCs: differential nonlinearity (DNL), integral nonlinearity (INL), and signal to noise and distortion ratio (SNDR). DNL and INL determine the TDC performance.

DNL indicates the maximum deviation between measured digital code with the theoretical code, which can be calculated by a statistical test. For ADC, DNL test is used to measure the voltage code. For TDC, DNL test is used to measure timestamps BIN-width. In the TDC test, the timestamps of a large number of random hits are measured. The hits are then plotted in a histogram. Since the theoretical number is known, the DNL can be calculated. In the ADC test, non-linearities can be calculated in different ways. One effective way is to compare the ideal and the measured curves. Another is to select measurement points over the input range. A ramp that exceeds both the positive and negative ADC input range is generated. The results in each bin are summed to compute the ADC density.

INL indicates the maximum deviation between whole-system measured value and the real value. For TDC module, INL test is the time deviation at one measurement.For the whole ADC design, INL test is total voltage difference, representing how ADC works at one measurement.Usually, we add all the DNLs of each ideal code to calculate the INL. In simulatation, the main contributors to DNL and INL are delays of the ISERDES and the clock distribution. The placement of the delay line is essential to avoid the bad situation which exhibits higher non-linearities and lower resolution.

AC measurement is needed to calculate the SNR. We need to simulate an external fixed AC signal. Using an FFT, the frequency domain plots are created in Matlab showing the frequency components in the input signal. We can then measure the SNR from the intensity of the fixed AC signal and the noise. From the SNR, the effective number of bits (ENOB) can be calculated.

$$ENOB = \frac{SNR - 1.76}{6.02} \quad (1)$$

5.2 Utilization

Given the application for our ADC, a small low-cost FPGA is most appropriate (Artix7[13] XC7A100T-CSG324) to simulate whole system performance. The utilization report for this 24 channels ADC is listed in Table 1. The power consumption, reported by Xilinx XPower Analyzer, is 453mW.

5.3 Performance

The test system is simulated with Modelsim and Matlab. The sampling clock of the ADC is 800MHz, while the input signal is sampled

Table 1: Utilization Report

	Used	Available	Percentage
Num of Slice Reg	185	126800	0.15%
Num of Slice LUTs	122	63400	0.19%
Num of LUT-FF pairs	104	203	51%
Num of bonded IOBs	48	210	22%
Num of BUFG/BUFGCTRLS	7	32	21%
BRAM	0	240	0%
DSP Slice	0	270	0%

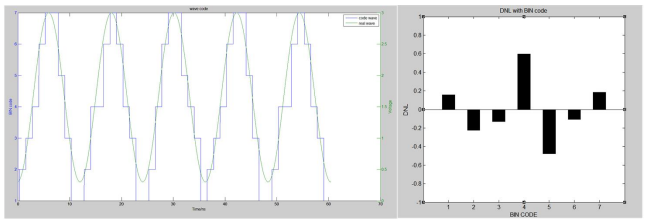

Figure 9: BIN and DNL result

with 8 clocks in one period. We use a density test to determine the DNL of ADC system (shown in Figure 9). Using an FFT (shown in Figure 10), in the frequency domain we can calculate the SNR and ENOB. The SNR is 24.3dB so the ENOB is 3.8bits.

5.4 Design Variations

This design shown can only meet the requirements of the target application shown. But a set of parameters can be varied, the most important of which are sampling rate, resources used, and voltage range.

Sampling rate: Reducing the sample rate to 100MSa/s makes the resolution 6 bit. A sample rate of 50 MSa/s adds an additional bit.

Resource utilization: To increase the resolution, our design need to divide clock phase into smaller pieces. This method requires more ISERDES primitive.

Voltage range: The input voltage range (0-3V) can be changed by the output drive strength and the external resistor value. Increasing the drive strength and decreasing the resistor value can improve the input voltage range.

6 Conclusion

In this paper, we propose a method of building high-speed ADCs with time to digital converters. Using the Xilinx FPGA's ISERDES logic resource, we can integrate 24 lanes of soft-core ADCs into single FPGA. While previous work only instance 6 channels of soft-core ADCs in the same FPGA. In our work, the ADC can achieve a sampling rate of 100Msa/s with 6-bit resolution.

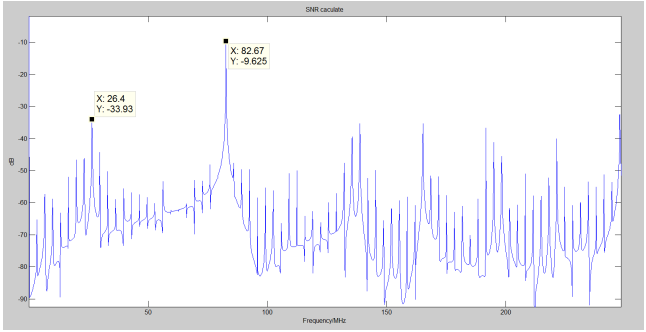

Figure 10: SNR result

References

[1] Integrated ADC for Altera Cyclone-IV Devices. 04 2011.

[2] P. Campagne-Ibarcq, E. Flurin, N. Roch, D. Darson, P. Morfin, M. Mirrahimi, M. H. Devoret, F. Mallet, and B. Huard. Persistent control of a superconducting qubit by stroboscopic measurement feedback. *Physical Review X*, 3(2):021008, 2013.

[3] H. Homulle, F. Regazzoni, and E. Charbon. 200 ms/s adc implemented in a fpga employing tdcs. In *Proceedings of the 2015 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays*, pp. 228–235. ACM, 2015.

[4] H. Homulle, S. Visser, and E. Charbon. A Cryogenic 1 GSA/s, Soft-Core FPGA ADC for Quantum Computing Applications. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 63(11):1854–1865, 2016.

[5] H. Homulle, S. Visser, B. Patra, G. Ferrari, E. Prati, F. Sebastiano, and E. Charbon. A reconfigurable cryogenic platform for the classical control of quantum processors. *Review of Scientific Instruments*, 88(4):045103, 2017.

[6] E. Mount, D. Gaultney, G. Vrijsen, M. Adams, S.-Y. Baek, K. Hudek, L. Isabella, S. Crain, A. van Rynbach, P. Maunz, et al. Scalable digital hardware for a trapped ion quantum computer. *Quantum Information Processing*, 15(12):5281–5298, 2016.

[7] P. Scarlino, D. van Woerkom, A. Stockklauser, J. Koski, M. Collodo, S. Gasparinetti, C. Reichl, W. Wegscheider, T. Ihn, K. Ensslin, et al. All-microwave control and dispersive readout of gate-defined quantum dot qubits in circuit quantum electrodynamics. *arXiv preprint arXiv:1711.01906*, 2017.

[8] F. Sebastiano, H. A. Homulle, J. P. van Dijk, R. M. Incandela, B. Patra, M. Mehrpoo, M. Babaie, A. Vladimirescu, and E. Charbon. Cryogenic cmos interfaces for quantum devices. In *Advances in Sensors and Interfaces (IWASI), 2017 7th IEEE International Workshop on*, pp. 59–62. IEEE, 2017.

[9] J. Sheng, C. Yang, and M. Herbordt. Towards low-latency communication on fpga clusters with 3d fft case study. *Proc. Highly Efficient and Reconfigurable Technologies*, 2015.

[10] L. Vandersypen and A. van Leeuwenhoek. 1.4 quantum computing-the next challenge in circuit and system design. In *Solid-State Circuits Conference (ISSCC), 2017 IEEE International*, pp. 24–29. IEEE, 2017.

[11] J. Wu, S. Hansen, and Z. Shi. Adc and tdc implemented using fpga. In *Nuclear Science Symposium Conference Record, 2007. NSS'07. IEEE*, vol. 1, pp. 281–286. IEEE, 2007.

[12] T. Xiang, L. Zhao, X. Jin, T. Wang, S. Chu, C. Ma, S. Liu, and Q. An. A 56-ps multi-phase clock time-to-digital convertor based on artix-7 fpga. In *Real Time Conference (RT), 2014 19th IEEE-NPSS*, pp. 1–4. IEEE, 2014.

[13] Xilinx. 7 Series FPGAs SelectIO Resources User Guide. Technical report, 10 2012.