Real-Time, Simultaneous Multi-Channel Data Acquisition System with No Time Skews between Input Channels

Fatemeh Zahedi and Zahra Zahedi
School of Electrical and Computer Engineering, Shiraz University, Iran
Email: {fatemeh.zahedi28, zahra.zahedi28}@gmail.com

***Abstract*—In this study, a simultaneous multi-channel data acquisition system is designed and implemented. The most significant feature of this system is that it can perform sampling of input channels without any time skews and additionally, it is a real-time system. The general architecture of this system is that the analog signals are received through multi-channel ADCs which are chained together to perform the digitalization of these inputs. Chaining of the ADCs allows the optimization of the number of lines on which the digitalized data is placed. The implemented system uses a Field Programmable Gate Array (Spartan-3, XC3S400 FPGA) and multi-channel ADCs (Texas Instruments ADS8556). This system has a typical SNR of 91.5dB and the calculated gain error for this system which is configured for analog voltage inputs of 0-5v, is equal to 0.0015%.**

***Index Term*—data acquisition, time skew, real time, simultaneous, FPGA, ADC**

I. Introduction

One of the most important parts of every branch of engineering is measurement engineering and using measurement instruments to achieve it [1].

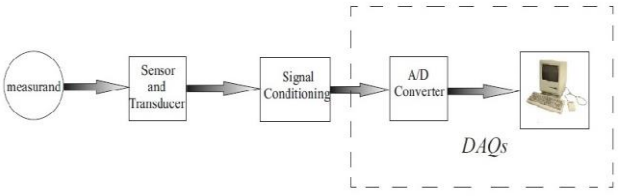

Figure 1. Instrument model. This figure shows different parts of an instrument. Notice how the last two parts are denoted to Data Acquisition Systems.

Fig. 1 is a simple example of an instrument model. As you can see, every instrument is consisted of different parts.

The last two parts are assigned to Data Acquisition Systems (DAQs) [2], [3]. As it is displayed in this figure, DAQs interface with sensors and transducers and digitalize analog output signals that come from these sensors to enable them to be processed and stored. Generally, DAQs are used for acquisition, processing, comparison and analysis of data [4].

DAQs are produced for various purposes and depending on their functionality, can be single-channel or multi-channel.

Multi-channel DAQs are required for most applications, for example, the digitalization of respiratory sound data at the posterior chest wall and air flow which are measured for the synchronization of pulmonary signal on the respiration cycle require the use of multi-channel DAQs [5].

The Positron Emission Tomography (PET) and Magnetic Resonance Imaging (MRI) systems use simultaneous multi-channel DAQs to retrieve data without affecting and conflicting with other inputs [6], [7]. This is a requirement in many similar cases but in cases such as ultrasound systems which acquire ultrasound data sequentially, one image at a time, in addition to the DAQs having to be multi-channel, real-time data acquisition also becomes a requirement [8].

The definition of real-time differs depending on different applications because each application has its own timing requirements which results in different definitions of the term"real time". In general, however, we define real-time data acquisition as the ability to gather data at the same rate as the user requires [9].

The presented design will cover most of the applications that require simultaneous multi-channel DAQs. Depending on the level of real-time data acquisition required, it can also cover a lot of such applications as well.

This design uses the Field Programmable Gate Array (FPGA) technology. This technology has many advantages including the wide range of supported DSP operations as well as the ability to upgrade without the need to replace hardware [4].

The FPGA used in this design is Spartan-3, XC3S400 which has 8,064 logic cells, 294,912 memory bits, 264 user I/O pins and 116 differential I/O pairs.

This design additionally uses a Texas Instrument, ADS8556 with features like 6 low power channels, 16 bit resolution and 450 KSPs to convert analog signals to digital.

Manuscript received July 1, 2012; revised November 21, 2014.

doi: 10.12720/ijsps.4.1.17-21

In later sections, we discuss materials and methods, system design and implementation, problems and solutions and finally results and discussions of the presented system.

II. Materials and Methods

A. Survey A/D Conversion

Recently, processing of the data retrieved from many operations are performed by Digital Signal Processing Systems. There for, these data should be digitalized first for these systems to be able to process them. This digitalization of data is performed by the ADC.

When we speak of simultaneous multi-channel A/D conversion, the goal is for the channels to sample data at the same time. One way to deal with this issue is to assign a separate ADC to each channel, but this solution is not very cost efficient not only because we will need one ADC for each channel, but because the power consumption of a system with this many ADCs is going to be very high.

Additionally, synchronizing all the ADCs is going to require specific methods.

Another common solution is to use a single ADC and an analog multiplexer which supports the sampling from all of the channels [4], [10]. This method also has its own flaws.

For example there is going to be a time skew between the channels when they are sampling and retrieving the data.

This system, uses a 6 channel ADC (ADS8556) to solve this issue and cause the sampling of the channels to be performed simultaneously without any time skew. The problem with this method is that there are a limited number of input channels and if more than 6 channels are needed to be synchronized, some special techniques need to be used.

It is worth mentioning however that, these techniques for synchronization of the channels is not going to be a difficult task.

B. DAQ System Structure

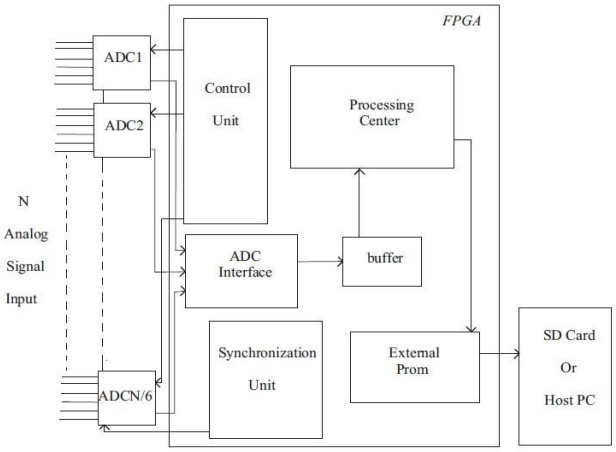

Figure 2. General system architecture and interfaces between its different units.

Fig. 2 shows the general structure of this DAQ system. As it is displayed in this figure, in this structure, N input channels are connected to N/6 ADCs. Sampling from each of the 6 channels belonging to each ADC is performed simultaneously only by using 3 pins (CONVST). The control unit performs tasks such as activation of the sampling process, configuration and modification of the Vref of each ADC and the range of the input signals and similar configurations including how data is placed on the output lines.

The ADC interface is consisted of a bus on which the data is placed and a clock which synchronizes the FPGA with the ADCs. This part is described in more detail in section III/A.

The synchronization unit is tasked with synchronization of the ADCs with each other. Synchronization is done by connecting the CONVST pins of the different ADCs together and connecting the last pin to the synchronization unit.

After the digitalized data is received by the FPGA, it is sent the processing center so that the necessary processing, comparison and analysis is performed on the data, and is placed on an external PROM which is connected to the FPGA, after which the data and the processing result is stored on an SD card or transferred to the host PC depending on the data requirements.

III. System Design and Implementation

The design and implementation of this system is consisted of a hardware and a software. The software is in the form of Verilog code for the FPGA. This code is generated using the SYSGEN toolbox which is a MATLAB add-on developed by XILINX.

This section covers the details about how the different parts of this system are designed.

A. ADC Interface and Control Unit

1) ADC Interface:

The ADC used in this design (ADS8556) is used with an internal clock with the frequency of 36MHz and is connected to the ADC interface to synchronize the FPGA with the ADC so that the sampling of the data, their placement on the bus and their retrieval are all performed based on this clock.

Using the BUSY pin we can monitor the starting time of the conversion, ending of the conversion and the time when the data of the 6 channels are latched on to the output registers.

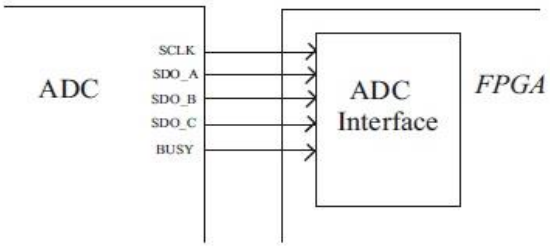

Figure 3. ADC interface connection. SDO_X, SCLK and BUSY pins are all connected to the ADC interface of the FPGA.

These parts are assigned to the ADC interface. Fig 3 details the connections between the FPGA and the ADCs as part of the ADC interface.

2) Control unit:

3 pins, sel_A, sel_B and sel_C, are connected to the Control Unit and if this unit sets all the pins to high, the data of the 6 channels will be transferred using the 3 output lines. If we need the data of the 6 channels to be transferred only on one output line, only the sel_A pin needs to be set to high.

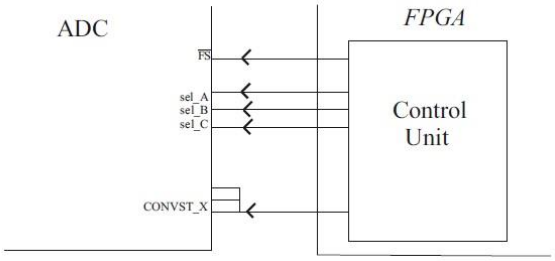

Figure 4. Control unit connection. The CONVST_X, Sel_X and $\overline{FS}$ pins of all of ADCs connect to the FPGA's control unit.

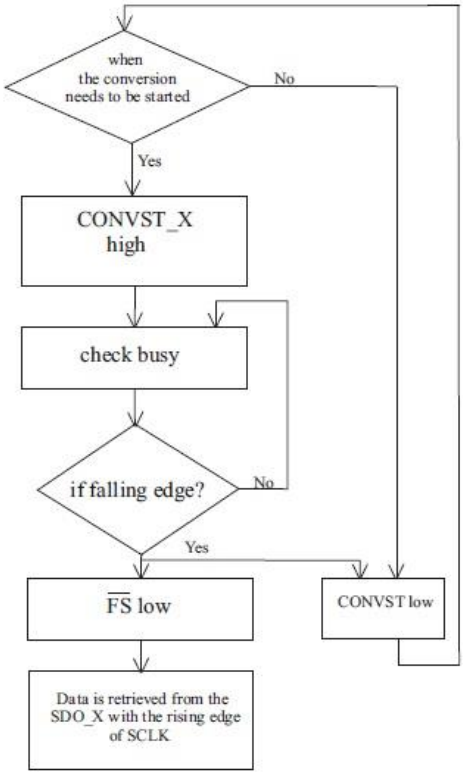

Figure 5. Flow diagram

Since it is aimed in this system that the sampling on all the channels occurs at the same time, therefore CONVST_A, CONVST_B and CONVST_C are connected to each other and only take up one pin on the FPGA. Rising edge in this pin will command the start of sampling. After the sampling is over, the data is placed on the line and setting the falling edge of the $\overline{FS}$ pin will then signal the beginning of the data transfer.

All of these commands are issued through the Control Unit.

Fig 4 shows in detail the connections between the ADC and the FPGA in regard to the control unit.

B. Timing and Flow Diagram

In the previous sub-section, the role of the ADC interface and Control Unit were explained individually. In this sub-section, we will explain how these two units work together.

The combination of these two units is displayed in Fig 5, from sampling to transferring of the digitalized data in a single cycle. Data conversion starts by setting the CONVST to high.

It should stay on high until data conversion is finished. During this time, the ADC interface checks the BUSY pin, and when it reaches the falling edge, $\overline{FS}$ and CONVST are both set to low by the control unit. By setting the $\overline{FS}$ to low, the data retrieval from the SDO_X begins on the first rising edge of SCLK. Fig. 6 shows the timing diagram of this process.

IV. Problems and Solutions

Designing and implementation of this system encountered some problems, the details and solutions of which will be covered in this sections.

A. The First Problem

Since the ADC has an internal clock, it is assumed at first that the entire FPGA should use this clock in which case other I/O pins will be in sync with the ADC because they also use the ADC clock. This would have been a good method if the clock generated by the ADC was perfect. In reality however, this clock does not have this characteristic and therefore using this clock for all of the FPGA will cause errors. On the other hand if the FPGA uses a crystal oscillation which creates a perfect clock, the problem will be solved. This clock should be a multiple of 36MHz. The internal clock of the ADC will be used solely to synchronize the ADC with the FPGA.

B. The Second Problem

It is claimed in this design that it is possible to have as many channels as needed whereas the FPGA has a limited number of I/O pins and so it can limit the number of input channels that can be incorporated in the system. This problem is resolved using Daisy Chain Mode. This mode can be activated by setting the DC_{EN} to high after which it is possible to chain the ADCs together. Fig. 7 shows this structure.

In this mode, the output of each ADC is placed on the DCIN-X of the next ADC and this next ADC will place the data on its DCIN-X pins onto its output lines in addition to its own output data and this chain continues until the last ADC transfers the data of all the ADCs to the FPGA using only the 3 lines of SDO-A, SDO-B and SDO-C.

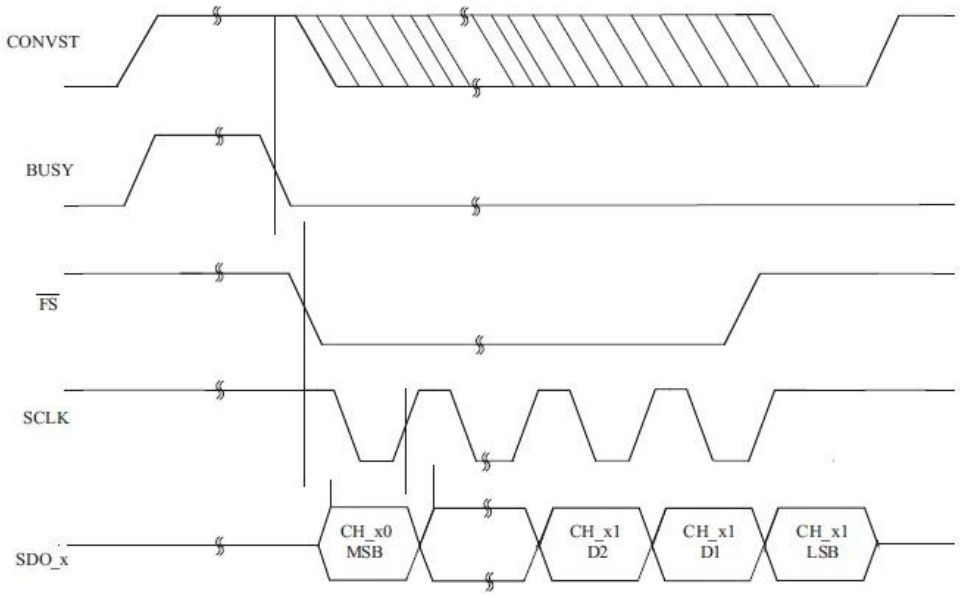

Figure 6. Timing diagram

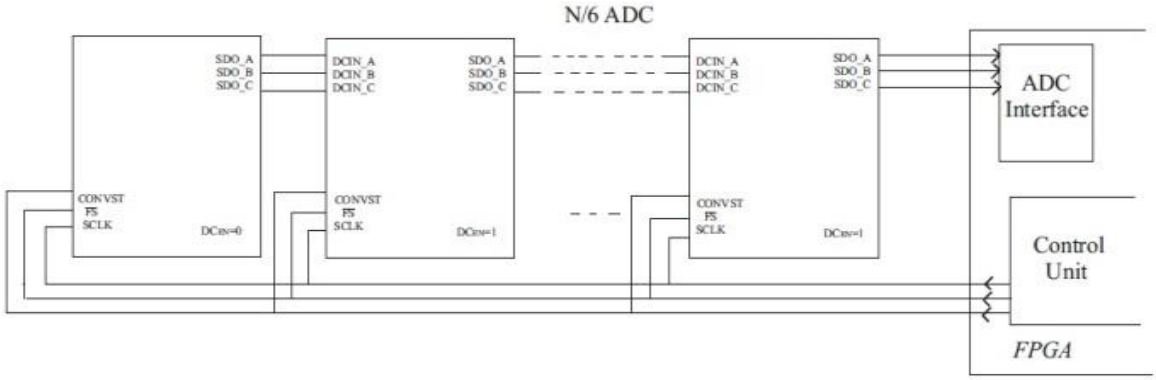

Figure 7. System structure with daisy chain mode

V. Results and Discussion

There are two important attributes related to DAQs: speed and resolution. These two attributes are usually inversely proportional to each other [11].

The presented system has a resolution of 16 bits and a speed equal to 450KS/s. This system is configured for unipolar inputs and the valid input voltage range is 0-5v. The resolution voltage (V_{res}) is 76.29 μv in which case the calculated gain error is equal to 0.0015% whereas its typical value for the used ADC is ± 0.25 with a unit of %FSR. The typical SNR for the used ADC is 91.5dB.

VI. Conclusion

The need to use simultaneous multi-channel DAQs in many applications indicates the importance and necessity of these systems.

The main purpose of this study is to design and develop a multi-channel data acquisition system which can sample data from multiple channels simultaneously and without any time skews which can also cover the real-time data acquisition and digitalization needs in many applications.

References

[1] F. A. Khan, Z. Hafeez, A. Mirza, and Q. Ain, "Design of FPGA based DAQ card using PCI express protocol," in *Proc. Multitopic Conference (INMIC)*, pp. 211-216, 2011.

[2] *The Measurement, Instrumentation, and Sensors Handbook*, J. G. Webster, CRC Press LLC, 1999.

[3] *Wiley Survey of Instrumentation and Measurement*, S. A. Dyer, ed. John Wiley and Sons, 2001.

[4] M. Abdallah and O. Elkeelany, "Simultaneous multi-channel data acquisition and storing system," in *Proc. International Conference on Computing, Engineering and Information*, 2009, pp. 233-236.

[5] I. Sen and Y. P. Kahya, "A multi-channel device for respiratory sound data acquisition and transient detection," *Engineering in Medicine and Biology 27th AnnuConf*, pp. 6658-61, 2005.

[6] H. P. Schlemmer, *et al*. "Simultaneous MR/PET imaging of the human brain: Feasibility study," *Radiology*, vol. 248, no. 3, 2008.

[7] M. S. Judenhofer, *et al*. "Simultaneous PET/MR images, acquired with a compact MRI compatible PET detector in a 7 Tesla magnet," *Journal of Radiology*, 2007.

[8] J. A. Jensen, O. Holm , L. J. Jerisen, H. Bendsen, *et al*. "Ultrasound research scanner for real-time synthetic aperture data acquisition," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 52, no. 5, 2005.

[9] J. Balent and T. Kerns, "Understanding real time for measurement and automation," *National Instruments White Papers*, 2008.

[10] M. Meurer and R. Raulesfs, "Enhancement of multi-channel a/d conversion by a code division multiplex approach," in *Proc. IEEE 6th International Symposium on Spread-Spectrum Technology and Applications*, 2000.
[11] Understanding New Developments in Data Acquisition, Measurement, and Control, Keithley, A Practical Guide to High Performance Test and Measurement.

Fatemeh Zahedi was born in Iran, in 1993. She is currently working toward the B.Sc. degree in Department of Electrical and Computer Engineering in Shiraz University, Shiraz, Iran. Her research interests include the role of haptics in training, learning improvement and motor control learning, teleopration and virtual environments, medical robotic and image guidance, and data acquisition systems.

Zahra Zahedi was born in Iran, in 1993. She is currently working toward the B.Sc. degree in Department of Electrical and Computer Engineering in Shiraz University, Shiraz, Iran. Her research interests include the role of haptics in training, learning improvement and motor control learning, control and computational learning, planning and scheduling, medical robotic and data acquisition systems.