Implementation of the onboard ADC and DAC on the Spartan 3E FPGA platform

A thesis submitted in partial fulfilment of the

Requirements for the degree of

Bachelor of Technology

In

Electronics and Communications Engineering

And

Electronics and Instrumentation Engineering

By

Satyaki Mascharak
Roll No- 108EC011
and
Arghyapriya Choudhuri
Roll No- 108EI033

Department of Electronics and Communication Engineering
National Institute of Technology, Rourkela

Implementation of the onboard ADC and DAC on the Spartan 3E FPGA platform

A thesis submitted in partial fulfilment of the

Requirements for the degree of

Bachelor of Technology

In

Electronics and Communications Engineering

And

Electronics and Instrumentation Engineering

By

Satyaki Mascharak

Roll No- 108EC011
and
Arghyapriya Choudhuri
Roll No- 108EI033

Under the guidance of

Prof Samit Ari

Department of Electronics and Communication Engineering
National Institute of Technology, Rourkela

NATIONAL INSTITUTE OF TECHNOLOGY ROURKELA

CERTIFICATE

This is to certify that the thesis entitled **“Implementation of the onboard ADC and DAC on the Spartan 3E FPGA platform”** submitted by Satyaki Mascharak (108EC011) and Arghyapriya Choudhuri(108EI033) in partial fulfilment of the requirements for the award of BACHELOR OF TECHNOLOGY Degree in Electronics and Communications Engineering and Electronics and Instrumentation Engineering at the National Institute of Technology, Rourkela (Deemed University) is an authentic work carried out by them under my supervision and guidance.

To the best of my knowledge, the matter embodied in the thesis has not been submitted to any other University/ Institute for the award of any degree or diploma.

Date: 14th May, 2012

Dr. Samit Ari
Assistant Professor
Department of Electronics and Communications Engineering
National Institute of Technology
Rourkela

ACKNOWLEDGEMENT

We avail this opportunity to express our indebtedness to our guide Dr. Samit Ari, Department of Electronics and Communications Engineering, National Institute of Technology, Rourkela, for his valuable guidance, constant encouragement and kind help at various stages for the execution of this dissertation work.

We are grateful to Mr. Vijay Sharma and Prof. Ayashkanta Swain , Department of Electronics and Communications Engineering, for providing valuable assistance and insight during the experimental process.

We also express our sincere gratitude to Prof. S.Meher, Head of the Department, Electronics and Communications Engineering for allowing access to valuable lab facilities in the department.

And last but not the least; we thank all those friends who helped us in the course of this entire dissertation work.

Satyaki Mascharak (108EC011) Arghyapriya Choudhuri(108EI033)

Department of Electronics and Communication Engineering,
National Institute of Technology, Rourkela
Rourkela-769008

ABSTRACT

The objective of this project is to first interface the onboard ADC and DAC available in the Spartan 3E FPGA platform, so that the real signals too can be processed by the FPGA board. Thus first of all, the ADC was interfaced and the results were observed via ChipScope Pro. Then the DAC was interfaced and checked if it was working or not. Finally both were operated together, where registers were used to store the values of the digital data obtained from the ADC and then sent to the DAC for the reconstruction of the original signal, which could be observed via a DSO.

ADC is a prime requirement whenever real-world signals come into play, hence interfacing the ADC is of great use and help in using the real-world signals for our use and further processing to extract vital information. DAC also aids in the said process similarly.

The basic aim being that a given input signal should output exactly or nearly exactly the given input signal after having it passed through the ADC and the DAC.

Keywords: Field Programmable Gate Array (FPGA), Analog to Digital Convertor, Digital to Analog Convertor, XILINX Spartan 3E FPGA platform, Flash PROM.

Contents

CERTIFICATE I

ACKNOWLEDGEMENT II

ABSTRACT III

LIST OF FIGURES VI

LIST OF TABLES VIII

Chapter 1: Introduction 1

1.1 Motivation 2

1.2 Objective: 4

1.3 Hardware and Software used 4

1.4 Testing Technology used 4

1.5 Applications 5

1.6 Literature Review: 5

1.7 Organization of the Report: 7

Chapter 2: Analog to Digital Convertor (ADC) 9

2.1 Introduction: 10

2.2 Xilinx Spartan 3E FPGA Starter Kit 14

2.2.1 The various peripherals available in the kit 14

2.3 ChipScope Pro: 16

2.4 Programmable Pre-Amplifier 17

2.4.1 Features of the pre-amplifier: 17

2.4.2. Communication between FPGA and the Preamplifier 19

2.4.3. Setting the Gain Values: 21

2.5 Working of ADC (LTC 1407A) 21

2.5.1. Communication between FPGA and ADC: 22

2.5.2. Controlling the ADC through VHDL coding: 23

2.6. Methodology: 24

2.7 Results: 26

2.7.1. Showing the output via the LEDs: 26

2.7.2. Showing the output via ChipScope Pro: 28

Chapter 3: Digital to Analog Convertor (DAC) 32

3.1 Introduction: 33

3.2 Applications: 34

3.3 Working of DAC: 34
3.4 Interface signals between the FPGA and the DAC: 37
3.5 Communication Protocol for the DAC: 37
3.6 Methodology: 38
3.7 Results: 39
Chapter 4: Conclusions and Future Work 41
4.1 Conclusion: 42
4.2 Future Work: 42

LIST OF FIGURES

Figure No.	Description	Page No.
Figure 1	Two Channel Analog Capture Circuit as on the Spartan 3E FPGA board	12
Figure 2	Block diagram of Successive Approximation type ADC	13
Figure 3	Diagram of the LTC1407A ADC chip	13
Figure 4	Detailed View for the Analog capture circuit, featuring the input pins to be used	14
Figure 5	A Dual, Matched Low Noise PGA (16-Lead SSOP Package)	18
Figure 6	Block Diagram of the circuit of LTC 6912-1 Programmable Gain Amplifier	19
Figure 7	Diagram for gain register settings via the A gain and the B gain	21
Figure 8	The Spartan 3E FPGA starter kit used	26
Figure 9	LED output glowing on application of 1.5V supply to the ADC, with a gain setting of ‘-1’.	27
Figure 10	ChipScope Pro output waveform on the application of 1.6V DC supply voltage to the ADC.	28
Figure 11	Screenshot of the ChipScope Pro Analyser showing the conversion process giving result after the second AD_CONV signal.	29
Figure 12	Zoomed view of Figure 10 showing a clear view of the results occurring after second AD_CONV signal only	30
Figure 13	Further zoomed view showing how exactly the values to $(2000)_{16}$ is getting generated.	30

Figure 14	.The ChipScope Pro analyser output waveform on the application of a sinusoidal signal to the ADC	31
Figure 15	Block diagram for the Digital to Analog convertor connection schematics	35
Figure 16	The inner circuitry of the LTC 2624 DAC chip	36
Figure 17	The Top view of the LTC 2624 DAC chip	36
Figure 18	The 32-bit protocol adopted by the on-board DAC	38
Figure 19	Ramp signal generated on passing an increasing pattern through the DAC	39

LIST OF TABLES

Table No.	Description	Page No.
Table 1	The allowable Gain settings along with the input voltage range and register values	21
Table 2	The results of the application of various voltages with various gain settings	28
Table 3	The results of the applied analog voltage to ADC and corresponding DAC output	39

Chapter: 1: Introduction

Today the world is moving towards a digital platform. Everything right from Cable television to our cell phone is getting digitized. But, at the same time, it is to be seen that all physical world signals available to us are still very much analog in nature. But their processing obviously happens in the digital domain. So for such a wide a variety of applications, it is of utmost importance that we have properly designed ADC's & DAC's. In an FPGA board we want to do a lot of operations on real world signals. Be it simple addition or be it complex transforms like Stockwell Transform we need to use the ADC and DAC very efficiently and frequently. Hence, the analysis and the results which follow in the project try to design an ADC and DAC and interface it with the real world signals.

1.1 Motivation

The primary motivation behind taking up this project is the large utility of the digital signals on which a lot of operations and transforms viz Short time Fourier transform, Stockwell Transform, Wavelet transform could be done, which are very tedious and time consuming jobs if the signals are in the analog domain. Simultaneously we must also remember that all digital signals cannot be used directly, as all the real world signals are more or less analog in nature. Hence it is of utmost importance that we are able to use both the ADC and DAC effectively; so that the analog signal first gets converted into digital one, then the processing is done on the digital signal, and finally the processed output is again converted into analog signal so that other systems may use it. An FPGA board, in this case Spartan 3e, was used and its ADC and DAC were interfaced with real world signals. For creating a real-time FPGA implementation of a variety of transforms ranging from the Fast Fourier Transform (FFT) to the Short Time Fourier Transform (STFT), one would need an ADC and a DAC so that the FPGA can interact with the real world signals. For this purpose, interfacing the ADC and DAC is of prime importance. The FPGA board used was Spartan 3E starter kit. This has an onboard ADC and an onboard DAC. Thus interfacing the onboard ADC and DAC with the analog signals and the consequent digital signals make it possible to analyze the real world signals.

While designing and interfacing of the ADC (the onboard ADC of Spartan 3E FPGA board), a simple state machine, with a number of states, was used. The properties of the on-board analog to digital convertor was studied and it was found that when the ‘AD_CONV’ signal goes ‘high’ after that only the analog to digital conversion starts taking place. The bus transaction within the FPGA board takes place via the SPI bus, shared with the ADC and the other peripheral devices available on the Spartan 3E FGPA board. When this ‘AD_CONV’ signal goes high, the Analog to Digital Convertor simultaneously samples both the analog channels (the on-board ADC is a dual channel ADC). The digital representation results, of this A/D conversion are not presented until the next time ‘AD_CONV’ is given to the ADC, as a result the digital representation of the analog data has a latency lasting for one cycle. The maximum sampling rate that is possible for the convertor being discussed is approximately 1.5 MegaHertz. The digital representation of the sampled analog values is shown as a 14-bit, two’s complement binary value by the ADC.

The values of the digitized data are stored in an array of registers, so that it can be used to from the registers for further processing. Then, after the processing of the values of the registers with the required functionality, the data is sent to the DAC so as to convert the digitized data into the corresponding analog signal. For this purpose the onboard DAC too uses the SPI bus for its communication. It can use either 24 bit or 32 bit protocol for the said conversion process. Thus the analog output would be obtained from the specified pins of the DAC depending on the DAC chosen to give the output.

1.2 Objective:

- The underlying objective of the present work is to interface the on-board ADC (Analog to Digital Convertor) and the DAC (Digital to Analog Convertor) with the real world signals.
- In doing so one can build a complete embedded dummy system that can take the real world signal and then convert it into digital format (via the ADC) .
- Then process that data, doing the required function (transforms etc. which would be performed by the FPGA itself), and then again convert the digital data obtained to analog form via the DAC.

1.3 Hardware and Software used

The hardware used was Xilinx Spartan 3E starter FPGA kit with onboard ADC (LTC 1407A) and DAC (LTC 2624). The software used was Xilinx ISE 10.1.

1.4 Testing Technology used

For testing purpose ChipScope Pro was used, as it helps to visualize the internal signals of the FPGA board, which can’t be seen with the help of an oscilloscope (as the output signals from the ADC are internal to the board and don’t have external interfacing). For visualizing the signals of the DAC, a DSO (Digital Storage Oscilloscope) was used, and a function generator was used to give the input signal to the ADC.

1.5 Applications

The various applications of ADC and DAC are as listed below:

- It finds great use in any kind of Digital Signal Processing Applications. A variety of analog signals provide users with information that is useful in one way or the other (provided appropriate processing is done). Applying various functions to those signals helps us to extract the exact information embedded in those signals.
- It is very useful when it comes to music and data recording, where sampling the data is of prime importance for optimum performance and hence they (ADC and DAC) come into the picture.

The most commonly used ADC is Successive approximation (SAR) ADC, in fact the on-board ADC chip used in the Spartan 3E starter FPGA kit is also of the type SAR.

It has helped not only in the fields of DSP but also in instrumentation and Communication and many more other fields.

1.6 Literature Review:

M. Vogels, B**.** De Smedt and G. Gielen [1] presented a paper on “Modelling and simulation of a Sigma-Delta digital to analog converter using VHDL-AMS”. According to this paper, the Sigma-Delta digital to analog converters (DACs) are less prone to circuit imperfections, than their Analog to Digital (A/D), counterparts because the DACs have their noise-shaping loop all in the digital domain, while the ADCs have the same in the analog domain. This being the case, still the basic low pass filter (analog part particularly) degrades, the overall performance of the system, especially in the case of multi-bit converters (when the conversion to take place is more than 1 bit). This paper provides an insight into a way of identifying and simulating the major noise and harmonics contributions of the system using VHDL-AMS.

P.W Jungwirth and A.D Poularakis [2] presented a paper on Dual Zero Crossing Analog-to-Digital Conversion. In this paper, it is discussed that in a conventional zero crossing analog-to-digital converter (single crossing analog to digital convertor), the zero crossing times of the input signal minus a reference signal are interpolated in order to get a good output from the ADC. A dual zero crossing analog-to-digital converter incorporates the zero crossings of the reference signal. The reference signal chain, consisting of a low resolution analog-to-digital converter followed by a software phase-locked loop based narrow band filter make up the dual crossing analog to digital convertor. The reference sinusoidal wave signal's bandwidth allows a narrowband digital filter to create an accurate estimate of the reference signal. Thus, according the said paper, the zero crossings of the input signal minus the reference signal trigger a digital latch thereby capturing the current output value of the estimated reference signal.

George Su´arez and Manuel Jim´enez [3] presented a paper on Behavioural Modelling of Sigma-delta Modulators using VHDL-AMS. It is of general conscience now-a-days that Sigma Delta Modulators form part of the core of today's mixed-signal designs as the cornerstone elements of oversampled analog-to-digital (A/D) and digital-to analog(D/A) converters. Although transistor-level simulation is the most accurate approach known for these components, this method (transistor level simulation) becomes impractical for complex systems due to the long computational time required. Thus behavioural modelling, if adopted, provides a viable solution to the design of the Sigma Delta Modulators.

JozefMihálov, VieraStopjaková [4] presented a paper on Implementation of Sigma-Delta Analog to Digital Converter in FPGA. Their paper presented implementation of a second-order Sigma-Delta Analog to Digital Converter (ADC) for audio band in field programmable gate array (FPGA) Xilinx Virtex 5.

1.7 Organization of the Report:

This thesis is divided into four chapters.

Chapter 2: In this chapter, we discussed the working of the on-board ADC. Also briefly discussed are the Xilinx Spartan 3E starter FPGA kit and the various peripherals available within the board. ChipScope Pro is discussed and how it helps in visualizing the internal signals of the FPGA via the JTAG chain is also discussed. The methodologies adopted to observe the output from the ADC is discussed. In one method, the output is observed via the on-board LEDs present and representing the bits of the digital data via the glow of the LEDs. In the second method being discussed in this chapter, the output digital data was obtained via the ChipScope Pro analyzer and the results are shown there.

Chapter 3: In this chapter, the on-board DAC present in the Spartan 3E starter FPGA kit was discussed. Also discussed is how the interfacing of this DAC is done and results obtained after the interfacing of the on-board DAC of the Spartan 3E FPGA starter kit are also discussed in this chapter.

Chapter 4: In this chapter, the conclusion to the project work is drawn and the future work possible in this area is discussed.

References:

[1]. M. Vogels, **B.** De Smedt and G. Gielen “Modeling and simulation of a Sigma-Delta

digital to analog converter using VHDL-AMS”, *IEEE/ACM International Workshop on Behavioral Modeling and Simulation, 2000. Proceeding,* 2000.

[2].P.W Jungwirth& A.D Poularakis. “Dual Zero Crossing Analog-to-Digital Conversion”, *Southeast Conference, Proceedings. IEEE*; 8-10 April, 2005.

[3]. George Su´arez and Manuel Jim´enez. “Behavioural Modelling of Sigma-delta Modulators using VHDL-AMS”, Electrical and Computer Engineering Department University of Puerto Rico at Mayaguez, 2005.

[4]. JozefMihálov, VieraStopjaková. “Implementation of Sigma-Delta Analog to Digital Converter in FPGA”, *Center of Excellence CENAMOST (Slovak Research and Development Agency),*2007

[5]. Acero, J., “VHDL Implementation of Sigma-Delta Analog to Digital Convertor”, *IEEE Transactions on Industrial Electronics,* Page(s):1843 – 1852, 2007.

[6] J. Candy and G. C. Temes, “Oversampling methods for AID and D/A conversion,” in *Oversampling delta-sigma converters,* pp. 1-25, IEEE Press, 1992.

[7] P. Ashenden, *The Designer’s Guide to VHDL*. Morgan Kaufmann Publishers, 1996.

[8] Bula, C.D., Jimenez, M., "Study of correlations among noise sources in ΣΔ modulator models", *Circuits and Systems (MWSCAS), 2011 IEEE 54th International Midwest Symposium on,* On page(s): 1 - 4

[9] V. T. Nguyen. "High-Pass Delta-Sigma Modulator and its Application to Time-Interleaved Delta-Sigma Converter". PhD dissertation, Ecole Nationale Supérieur des Télécommunications, 2004.

[10] K. Ishida and M. Fujishima. "Chopper-stabilized high-pass sigma-delta modulator utilizing a resonator structure". *IEEE Trans. on Circuit and Systems II,* 2003.

[11] V. T. Nguyen, P. Loumeau, and J. F. Naviner. "Analysis of High Pass filter in High Pass Sigma Delta Modulator". *IEEE Proceeding of Mixdes 2003,* 2003.

[12] P. R. Gray and R.G. Meyer. *Analysis and Design of Analog Integrated Circuits.* John Wiley and Sons, New York, second edition, 1993.

[13] E. Christen and K. Bakalar. "VHDL-AMS-A Hardware Description Language for Analog and Mixed-Signal Applications". *IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing,* Page no: 1263-1272, Oct. 1999.

Chapter 2: Analog to Digital Convertor (ADC)

2.1 Introduction:

From here on, detailing on how exactly the on-board ADC of the Spartan 3E FPGA board is interfaced for real world signals, is done.

The ADC used, that comes bundled in with the Spartan 3E FPGA starter kit, ready for being use, is manufactured by the Company LINEAR TECHNOLOGY LIMITED. The ADC chip used is LTC1407A.

Few of the salient features of this ADC chip are as follows:

- 3Msps (3 Mega Hertz sampling rate) Sampling ADC with Two Simultaneous Differential Inputs. In here what exactly is done that the sampling rate is divided in to two channels of 1.5 Msps each. Here Msps denotes the conversion rate of the ADC.

- 1.5Msps Throughput per Channel .This says about the efficiency of the ADC and how exactly and how efficiently it can quantize analog data into its digital counterpart.

- Low Power Dissipation: 14mW. It dissipates very low power and hence can be ideally used as chip level application. The hardware level implementation is only possible in a case where the power consumed is very less and it can switch itself into a low power consuming mode given the circumstances demanding so.

- 3V Single Supply Operation. It takes its inputs from a supply of 3V DC.

- ±1.25V Differential Input Range. The interim range of the ADC is 2.5 V (±1.25V). It can be manipulated in the way it has to be behaved. The input voltage range can be set by programmer by changing the gain parameters.

- Pin Compatible 0V to 2.5V Input Range Version (LTC1407A). Its pin is inherently compatible to an input range of 0V to 2.5V. Any range outside this is to be coded manually and then set and given to the chip.

- 2.5V Internal Band gap Reference with external overdrive. It’s internal Band gap i.e. the difference between its two states is around 2.5 V. It is driven externally from an external source.

- 3-Wire Serial Interface. It can communicate with the external data or the external world via serial media or serial bus (SPI in case of LTC 1407A). This makes communication very easy and simple.

- 80dB Common Mode Rejection at 100 kHz. This parameter is very much helpful whenever simultaneous sampling of both the channels of the ADC occur, the common mode rejection rejects the noise being generated and thus is helpful in getting the desired digital output.

Applications of this ADC chip:

- Telecommunications: In telecommunications, the signal conversion into digital form is of prime importance, as most of the processing is done in digital domain.
- Data Acquisition Systems: SCADA requires an efficient ADC for proper Data acquisition and data retrieving
- Uninterrupted Power Supplies: All UPS’s require an ADC for efficient battery usage and optimal power storage. Also for charging purposes it is the digital form that is prevalent and not the analog form. Hence ADC becomes a must for this purpose.

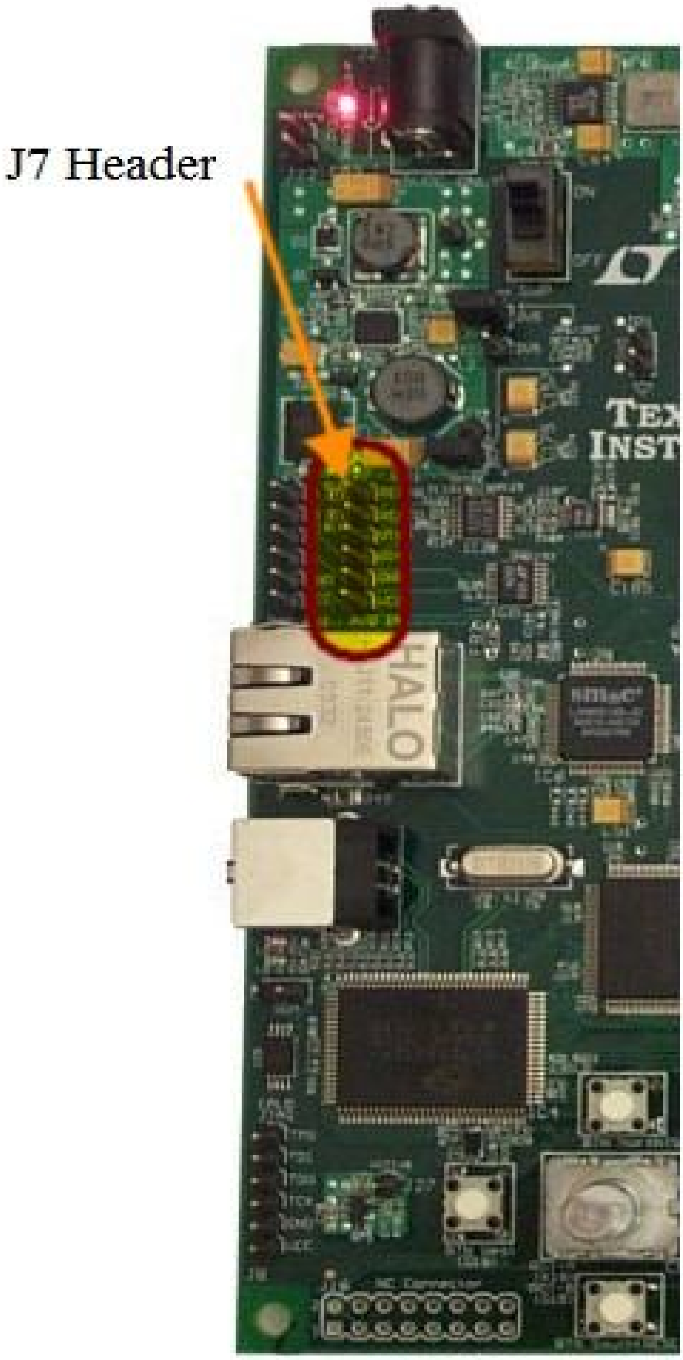


Figure 1: Two Channel Analog Capture Circuit as on the Spartan 3E FPGA board [5]

ADCs (Analog to Digital Convertor) are of various types. The one used for our purpose is the Successive Approximation Type ADC (SAR-ADC), where the main components include a DAC (digital to analog convertor), a clock, a comparator and a SAR register for storing the values of the digital data which comes after the comparator compares the values of the DAC with the analog input and outputs a ‘1’ or a ‘0’ depending on the condition. The block diagram can be shown as below:

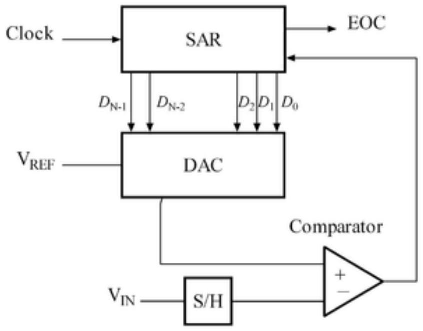


Figure 2: Block diagram of SAR- ADC.

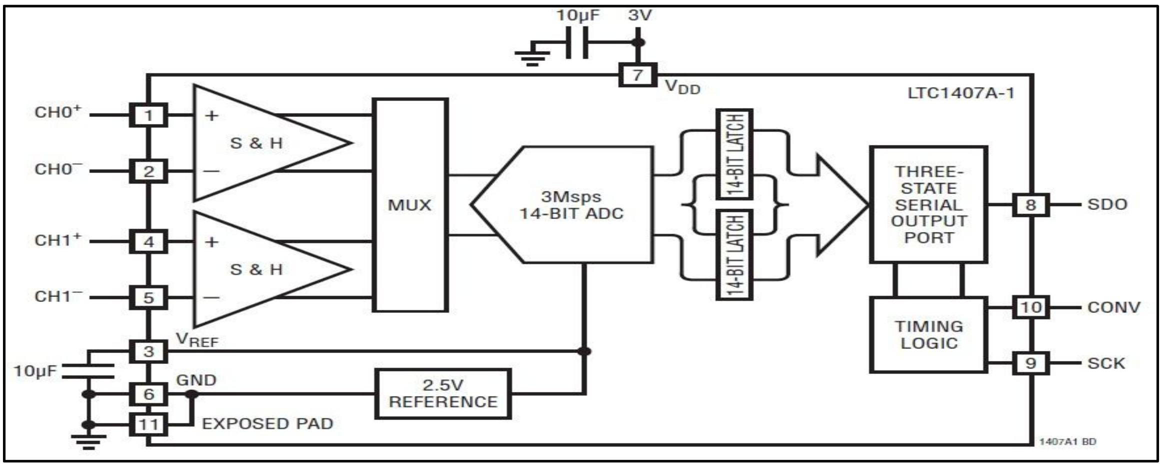


Figure 3: Circuit Diagram of the LTC1407A ADC chip [6]

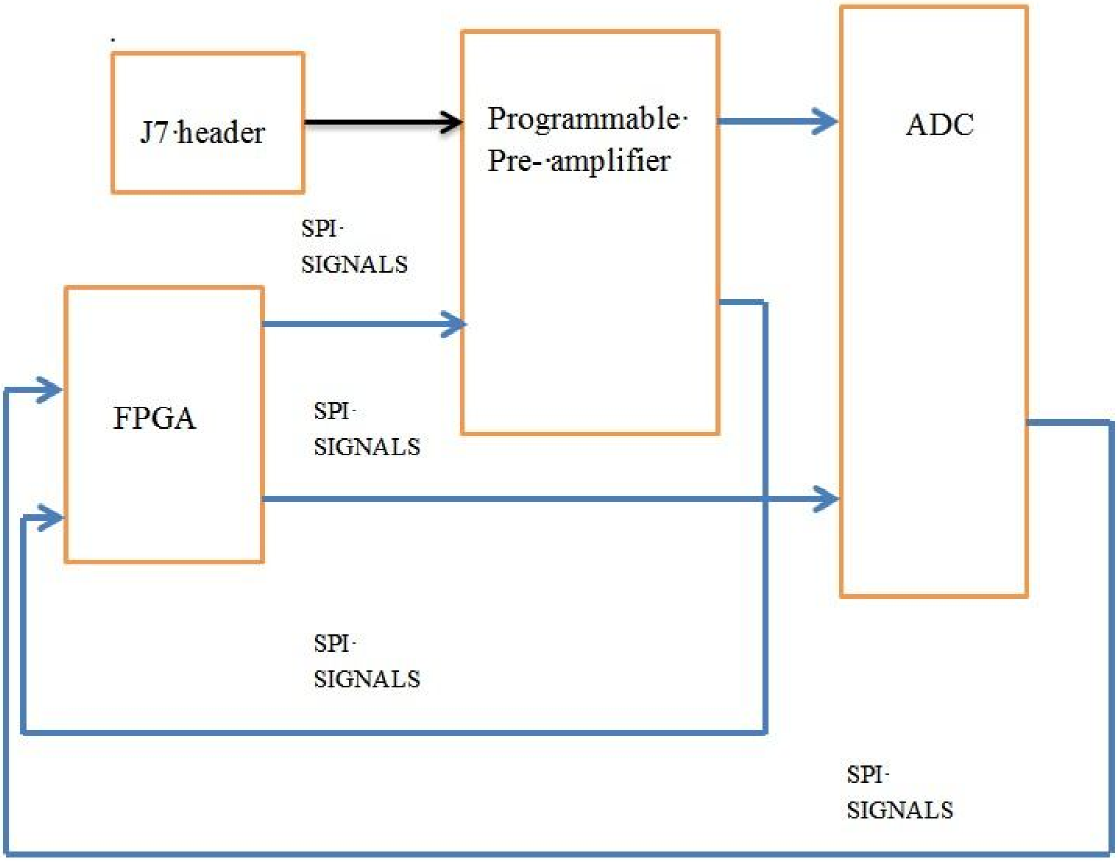


Figure 4: Detailed View for the Analog capture circuit, featuring the input pins to be used and the ADC chip [5].

2.2 Xilinx Spartan 3E FPGA Starter Kit

The Spartan 3E starter kit provides us the basic features as provided by the Spartan 3E FPGA. It also provides easy way to test the various programs in the FPGA itself, by dumping the ‘bit’ file into the FPGA and then observing the output.

2.2.1 The various peripherals available in the kit

The Spartan 3E FPGA board comes built in with many peripherals that help in the proper working of the board and also in interfacing the various signals to the board itself. Some of the peripherals included in the Spartan 3E FPGA board include:

- 2-line, 16-character LCD screen: This LCD screen can be interfaced with the various on-board signals of the FPGA to display various texts as desired by the programmer.
- PS/2 mouse or keyboard port: A PS/2 keyboard can be connected to the FPGA board and then depending on the key pressed the FPGA would do a variety of things, as programmed.
- VGA display port: This port can be used to display various encoded images via a screen. The image encoding would be done by the FPGA via the aid of the program and then the encoded image would be displayed on the screen.
- Two 9-pin RS-232 ports: This ports help in the transmission of serial data to and from the FPGA board.
- 50 MHz clock oscillator: This is the system clock which helps in giving the clock signal to the various events taking place within the FPGA and the various programs that require clock for their working, A Digital clock manager can also be used to reduce the frequency of the system clock so that is useful for various other purposes which need smaller clock frequency.
- On-board USB-based FPGA download and debug interface: The programmable file is dumped into the FPGA via the USB based download cable. Hence it is very much helpful in the testing of the programs whether they are working correctly or not.
- Eight discrete LEDs: The LEDs can be interfaced to glow when a particular output becomes high. Hence the LEDs can be interfaced to show the output of a single bit.

- Four slide switches and four push-button switches: These switches are used to give the inputs to the FPGA board. They can also act as the reset switches for the various programs.
- Four-output, SPI-based Digital-to-Analog Converter (DAC): It is the on-board DAC which is to be interfaced to give the analog output to the digital data values.
- Two-input, SPI-based Analog-to-Digital Converter (ADC) with programmable-gain pre-amplifier: It is the on-board ADC which converts the real world analog signals into digital values.

2.3 ChipScope Pro:

ChipScope Pro tool inserts logic analyzer, system analyzer, and virtual Input/Output (I/O) low-profile software cores directly into the design, allowing one to view any internal signal or node, including embedded hard or soft processors in the PC itself. Signals are captured in the system at the speed of operation and brought out through the programming interface, freeing up pins for the design. The signals are then captured and then with the help of ChipScope Pro analyser they are checked and tested if they tally with the desired results or not. With the increasing density of the FPGA devices, the impracticality of attaching test equipment probes to these devices under test also increases. Integration of key logic analyser and other test and measurement hardware components with the target design inside the supported Xilinx FPGA devices (Spartan 3E Starter kit in this case) make up the ChipScope Pro tool. This tool then communicates with the above components; and provides the designer with a good logic analyser solution. ChipScope Pro Analyzer supports a variety of download cables for communication between one’s computer and the devices in the JTAG boundary scan chain. Of these some are listed below:

- Platform Cable USB

- Parallel Cable IV

Platform cable USB is the one used in this case of Spartan 3E Starter kit. The program is dumped into the FPGA and then the analysis is done via ChipScope Pro. ChipScope Pro uses the JTAG chain for displaying the output waveforms on the computer screen. JTAG (Full form: Joint Test Action Group) is the one that was later standardized as IEEE 1149.1 (Standard Test Access Port and Boundary-Scan Architecture). The purpose of the development of the JTAG chain initially was for debugging and testing the PCBs (Printed Circuit Boards). Now-a-days it is also widely used for IC debug ports. Using this JTAG tool, the ChipScope Pro analyses the signals within the FPGA board and gives the graphical output.

2.4 Programmable Pre-Amplifier

The LTC6912-1 provides two independent inverting amplifiers with programmable gain. The purpose of the amplifier is to scale the incoming voltage on VINA or VINB so that it maximizes the conversion range of the ADC, namely 1.65 ±1.25V.

2.4.1 Features of the pre-amplifier:

- Channels with Independent Gain Control
- LTC6912-1: (0, 1, 2, 5, 10, 20, 50, and 100V/V)
- Offset Voltage = 2mV Max (–40°C to 85°C)
- Channel-to-Channel Gain Matching of 0.1dB Max
- 3-Wire SPITM Interface
- Extended Gain-Bandwidth at High Gains
- Wired-OR Outputs Possible (2:1 Analog MUX Function)
- Low Power Hardware Shutdown (GN-16 Only, 2µA Max at 2.7V)
- Single or Dual Supply: 2.7V to 10.5V Total

- Input Noise: 12.6nV/√Hz
- Total System Dynamic Range to 115dB
- 16-Pin GN (SSOP) or 12-Pin DFN Package Options

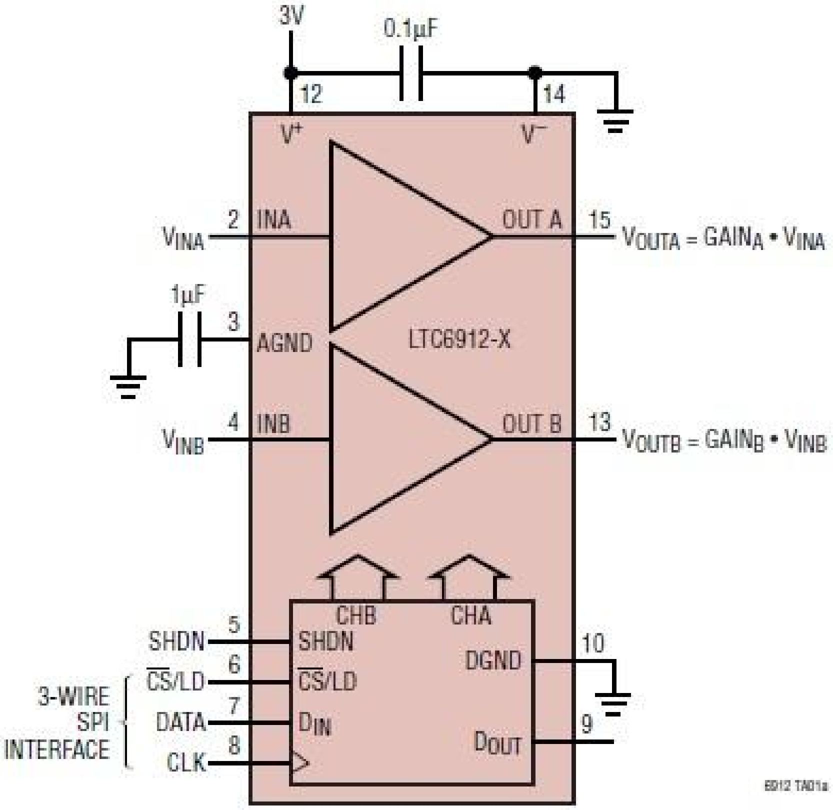


Figure 5: A Dual, Matched Low Noise PGA (16-Lead SSOP Package) [7]

The LTC6912 is a family of dual channel, low noise, digitally programmable gain amplifiers (PGA) that are easy to use and occupy very little PC board space. The gains for both channels are independently programmable using a 3-wire SPI interface to select voltage gains of 0, ‘-1’, ‘-2’, ‘-5’, ‘-10’, ‘-20’, ‘-50’, and ‘-100’V/V (LTC6912-1).The LTC6912 family of integrated chips consists of two matched inverting amplifiers with rail-to-rail outputs. When operated with unity gain, they will also process the rail-to-rail input signals. A half-supply reference, generated internally at the AGND pin, supports single power supply applications.

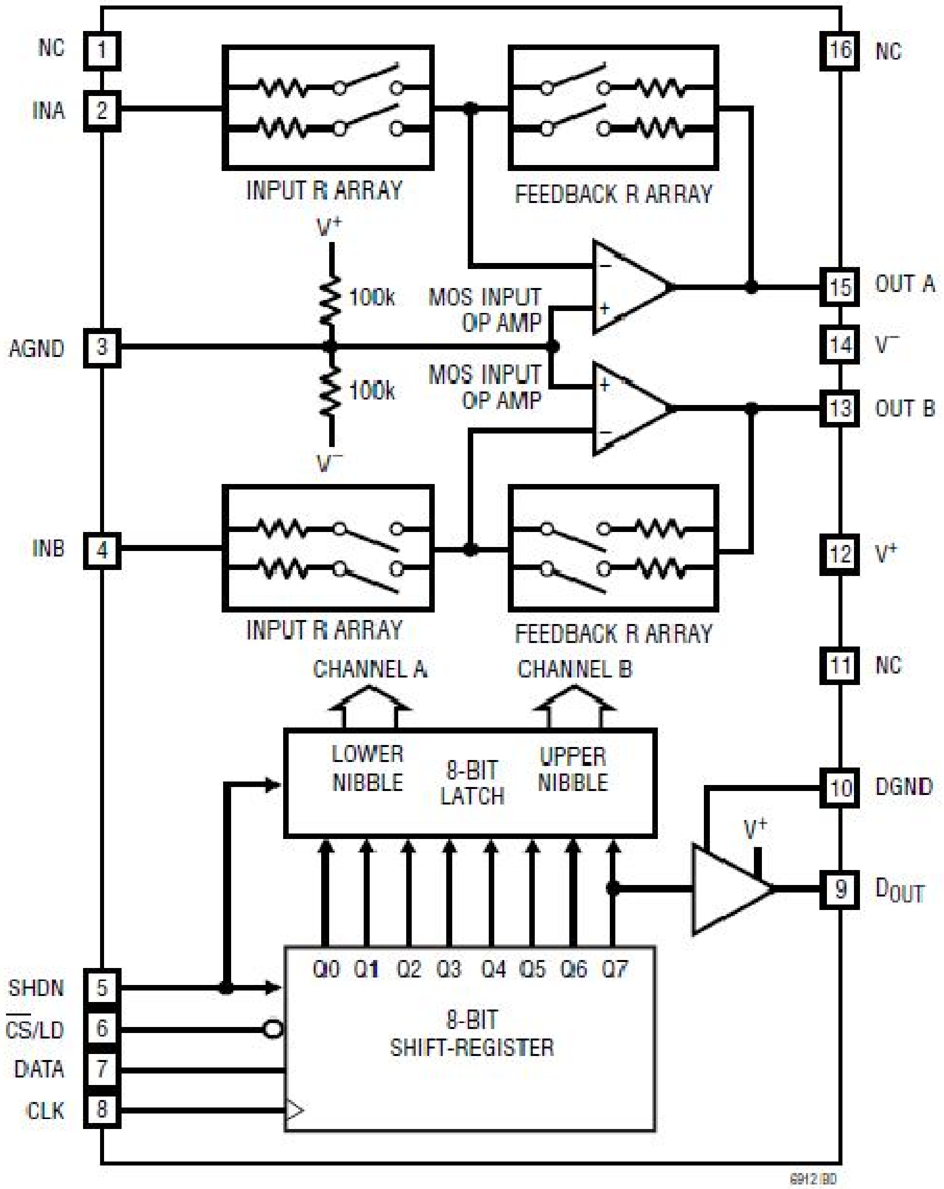


Figure 6: Block Diagram of the circuit of LTC 6912-1 Programmable Gain Amplifier. [7]

2.4.2. Communication between FPGA and the Preamplifier

Various interface signals are used for the interfacing of the amplifier with the FPGA. The amplifier uses the SPI bus for the communication between itself and the FPGA, and this SPI bus is shared between a variety of other peripheral devices like the ADC, DAC, Strata Flash and the Platform Flash. Hence the other interface signals to those devices must be suitably disabled so that proper interaction of the bus with the pre-amplifier takes place. Some of these are mentioned below:

(a) SPI_MOSI: This signal is found at the pin T4 of the Spartan 3E FPGA kit. This signal is directed from the FPGA to the AD. It presents serial data. MOSI stands for Master Output, Slave Input. It is mainly responsible for presenting the 8-bit programmable gain settings. Based on this pre-loaded gain settings the conversion of the analog value to digital value takes place. Also the input range to the ADC also depends on this gain setting.

(b) AMP_CS: This signal is found at the pin N7 of the Spartan 3E FPGA kit. This signal is directed from the FPGA to the AMP. It is an active low chip select signal. The amplifier gain is set when the signal returns high.

(c) SPI_SCK: This signal is found at the pin U16 of the Spartan 3E FPGA kit. This signal is directed from the FPGA to the AMP. It is basically the clock signal depending on which the gain setting is done. The SPI_MOSI signal sends one bit at a time at the rising edge of the SPI_SCK clock. Also AMP_DOUT signal, which is described below, echoes the gain setting beck to the FPGA at the falling edge of the SPI_SCK clock.

(d) AMP_SHDN: This signal is found at the pin P7 of the Spartan 3E FPGA kit. This signal is directed from the FPGA to the AMP. It is an Active high shutdown, reset signal.

(e) AMP_DOUT: This signal is found at the pin E18 of the Spartan 3E FPGA kit. This signal is directed from the AMP to the FPGA. This signal simply echoes the previous amplifier gain settings back to the FPGA. It can be ignored in most of the cases. It again presents data in serial data format.

2.4.3. Setting the Gain Values:

The gain setting is done by programming the programmable pre-amplifier via SPI_MOSI. The gain for each amplifier is sent as an 8-bit command word, consisting of two 4-bit fields. The SPI bus transaction starts when the FPGA asserts AMP_CS Low. The amplifier captures serial data on SPI_MOSI on the rising edge of the SPI_SCK clock signal.This data is sent with MSB first, i.e. B3 bit is sent first and then the rest in order to program the gain amplifier to the required settings. A pre-defined table sets the gain value according to the bits given to the gain register. The permissible gains and the corresponding settings are given in the adjoining table.

Table 1: The allowable Gain settings along with the input voltage range and register values [5]

Gain	A3	A2	A1	A0	Input Voltage Range (in Volts)	
	B3	B2	B1	B0		
0	0	0	0	0	-	-
-1	0	0	0	1	0.4	2.9
-2	0	0	1	0	1.025	2.275
-5	0	0	1	1	1.4	1.9
-10	0	1	0	0	1.525	1.775
-20	0	1	0	1	1.5875	1.7125
-50	0	1	1	0	1.625	1.675
-100	0	1	1	1	1.6375	1.6625

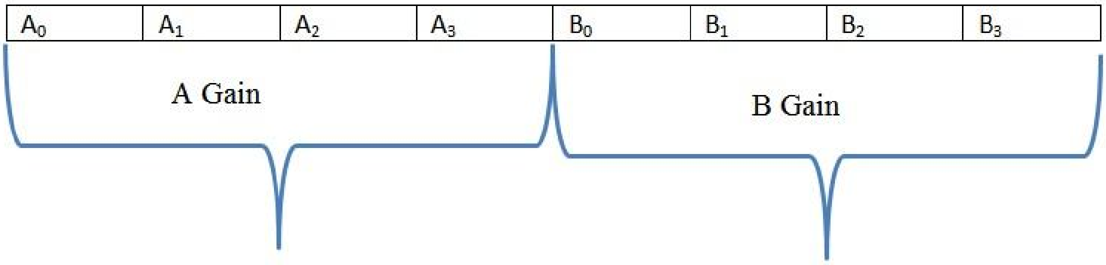


Figure 7: Diagram for gain register settings via the A gain and the B gain.[5].

2.5 Working of ADC (LTC 1407A)

ADC presents a 14-bit; two's complement digital output of the analog input. The input voltage given to the ADC depends on the programmable gain settings of the programmable pre-amplifier. The maximum input range is for the gain '-1' for which the input voltage range is 2.5V i.e. from 0.4V to 2.9V. The Analog to Digital Conversion formula [5] is given below:

$$D[13:0] = GAIN \times \frac{V_{IN} - 1.65V}{1.25V} \times 8192 \quad (1)$$

Here D [13:0] represents the 14 bit two's complement value of the analog input. It is output to the FPGA from the ADC via the SPI_MISO signal, as will be discussed later in the interfacing signals of the FPGA and the ADC. GAIN is the gain setting given via the programming of the gain register taking SPI_MOSI signal bit by bit. V_{IN} is the input voltage to the ADC. 1.65V is the reference voltage of the ADC. This is achieved by the voltage divider circuit provided in the ADC circuit (dividing the V_{cc} which is 3.3V). The range of the ADC used is ±1.25V. Hence the output is scaled by 1.25V. Also the output obtained is in 14 bit two's complement form and hence the output is scaled by 2^{13} or 8192. The inputs to the ADC may be driven differentially or as a single-ended input (VINA or VINB).Any unwanted signal that is common to both inputs of each input pair will be reduced by the common mode rejection of the sample-and-hold circuit. The input channels draw only one small current spike while charging the sample-and-hold capacitors at the end of conversion. For minimum acquisition time with high source impedance, a buffer amplifier must be used. LTC 6912-1 takes care of that buffer amplifier.

2.5.1. Communication between FPGA and ADC:

(a) AD_CONV: This signal is active high shutdown and reset signal. This signal marks the beginning of the conversion of the analog signal. It is an internal signal of the FPGA board, which can't be viewed with the help of a DSO (Digital Storage Oscilloscope). Pin P11 is responsible for this signal (AD_CONV). This signal has its direction from the FPGA to the ADC.

(b) SPI_MISO: This signal is the serial data output from the ADC chip to the FPGA board. It is the one that gives the digital representation of the sampled analog value as 14-bit two's complement binary value. It is again an internal signal and pin N10 is responsible for this signal. Thus the direction of this signal is from the ADC to the FPGA board.

(c) SPI_SCK: As described earlier this is the clock signal which plays an important role in the analog to digital conversion process and also sending the data from the ADC unit to the FPGA.

2.5.2. Controlling the ADC through VHDL coding:

The serial interface sends out the two conversion results in 32 clocks for compatibility with standard serial interfaces. Two 14 bit results are obtained from the ADC. The AD_CONV signal is not a traditional SPI slave select enable. Provisions should be made to provide enough SPI_SCK clock cycles so that the ADC leaves the SPI_MISO signal in the high-impedance state. Otherwise, the ADC blocks communication to the other SPI peripherals. The ADC tri states its data output for two clock cycles before and after each 14-bit data transfer (hence a total of 34 clock communication sequence is used).

The Common mode rejection allows users to eliminate ground loops and common mode noise by measuring signals differentially from the source. The SPI bus signals are shared by other devices on the board. It is vital that other peripheral devices are disabled when the FPGA communicates with the AMP or ADC to avoid bus contention. Whenever the AD_CONV signal goes high, the Analog to Digital Converter (ADC) simultaneously samples both analog channels. The results of this conversion are not presented until the next time AD_CONV is asserted on the SPI bus, thus creating a latency of one sample. The maximum sample rate is approximately 1.5 MHz. Hence by Nyquist Criterion, the frequency of the analog input must be less than half of the sampling frequency, i.e. the maximum input frequency that can be given to the ADC and can be converted into digital form without aliasing and other problems is less than 0.75 MHz.

2.6. Methodology:

The methodology adopted in the interfacing of the on-board ADC of the Spartan 3E FPGA platform kit is as mentioned below:

(a) First of all, the basic architecture and the design of the on-board ADC chip is understood and a state diagram is drawn to implement the same via the FPGA starter kit.

(b) While designing the state machine for the implementation of the ADC, the SPI_CLK clock cycles for which the signal needs to be active and the number of clock cycles needed by the AD_CONV signal to remain active so that proper analog to digital conversion takes place, is also taken care of.

(c) The VHDL code for the interfacing of the ADC with the real world signals is done, keeping in view the various interface signals that are available between the FPGA and the ADC. The main algorithm used in writing the program for the interfacing of the ADC is as mentioned below:

  (i) The SPI_MOSI signal is sent bit by bit with the MSB sent first, in order to program the gain register with the required gain settings.

  (ii) After the gain register has been programmed, and the required gain settings has been set, the analog to digital convertor starts conversion of the applied analog data after the assertion of the high AD_CONV signal.

  (iii) As dual channel conversion starts taking place simultaneously hence the total clock cycle required for the entire analog to digital conversion process in 34 clock cycles. The ADC goes to a tri-state before and after the digital conversion process and hence a total of 34 clock cycles is required for the entire ADC process.

(iv) Then after that 34 SPI_SCK cycles the AD_CONV signal again goes high and then the digital output is presented via SPI_MISO signal.

(d) The implementation constraints file is also coded for the above VHDL program with the pin number of the signal sources being the same as that given in the user guide of the Spartan 3E FPGA kit. This is done as per the user guide as the signals are internal and the pins, which give those signals as the output, are fixed while manufacturing the board itself.

(e) The VHDL code is then synthesized and then the programming file is generated, which can now be dumped into the FPGA kit via the platform cable USB using any Integrated Software Environment. The dumping of the said programming file (‘bit’ file) would interface the on-board ADC of the FPGA kit so that the ADC could process the real world signals and give the digital representation of the given analog input. The connection from the PC to the board was done via platform cable USB and thus the ‘bit’ file was dumped to the FPGA.

(f) Analog signal is then applied to the J7 header which is present for the interfacing of the real world signals with the FPGA kit. Then the results obtained are verified.

While displaying and verifying the output obtained from the ADC, two methods were used. In one the on-board LEDs were used to verify the result bit by bit; whereas in the other method, ChipScope Pro was used to verify the results of the analog to digital conversion process. The main handicap in this case of showing the output via LEDs was the availability of fewer LEDs than the required number of bits (The ADC gives a 14-bit digital output whereas there are only 8 on-board LEDs to display the results.). As a result the significant 8 bits were displayed via the LEDs and the rest of the other lower bits were ignored. Then the program was again run, this time the 8 significant bits that were obtained earlier were ignored and the rest 6 bits were displayed via the LEDs so that a complete picture of the conversion process was obtained.

In case of displaying the output via the ChipScope Pro, JTAG chain helps in getting the signals from the debug ports of the kit and displays the same via the ChipScope Pro analyser. The waveforms obtained via the ChipScope Pro are then analysed for verifying the results of the analog to digital conversion process.

2.7 Results:

2.7.1. Showing the output via the LEDs:

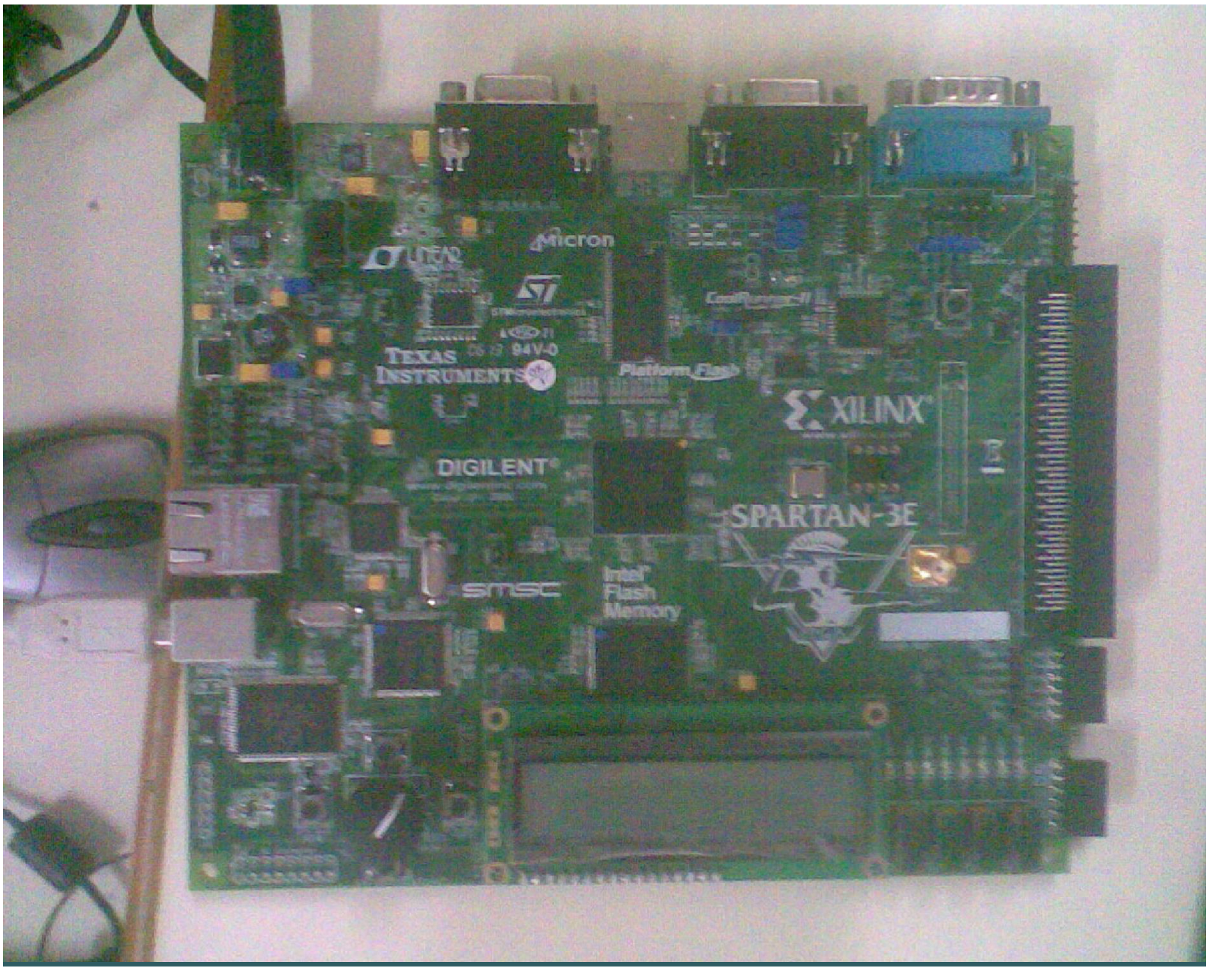


Figure 8: The Spartan 3E FPGA starter kit used. [5]

The main method was writing the VHDL program for the interfacing of the ADC with the real world signals and then interfacing the LEDs with the digital output of the ADC so that they would glow in accordance to the results output by the ADC. The ADC output was used to glow the on-board LEDs that are available in the Spartan 3E kit.

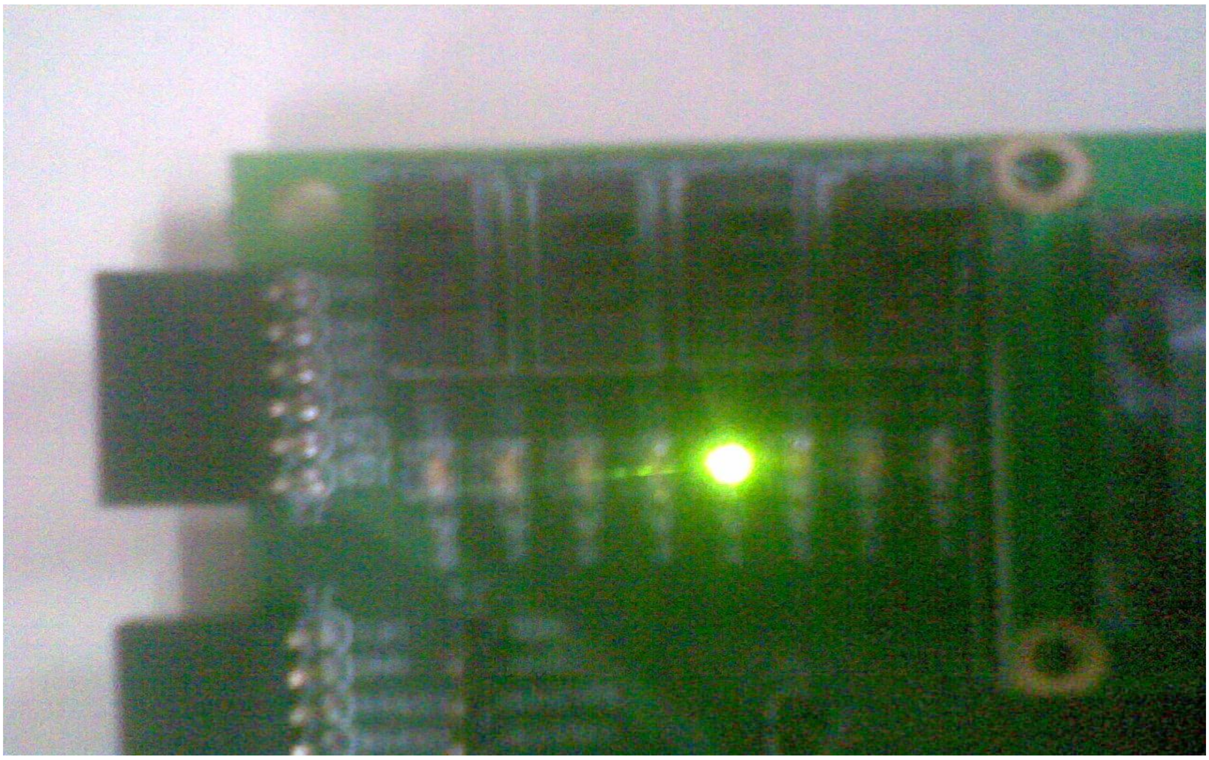


Figure 9: LED output glowing on application of 1.5V supply to the ADC, with a gain setting of '-1'.

In the case demonstrated above the voltage applied was 1.5V, and on application of the ADC conversion formula we get that the value of the 14 bit number should be something around $(983)_{10}$, but since here we are ignoring the last 6 bits and taking only the first 8 bits, hence the results can be approximated as somewhat close to the true value, depicting $(00010000000000)_2$ or $(1024)_{10}$ as the result.

Table 2: The results of the application of various voltages with various gain settings.

Gain	Analog Value	Digital o/p(actual)	Digital o/p(observed)
-1	1.5 V	00001111010111	00001111010110
-2	1.5 V	00011110101110	00011110101110
-2	1.6 V	00000110001111	00000110001101
-1	1.6 V	00000101001000	00000101001001

2.7.2. Showing the output via ChipScope Pro:

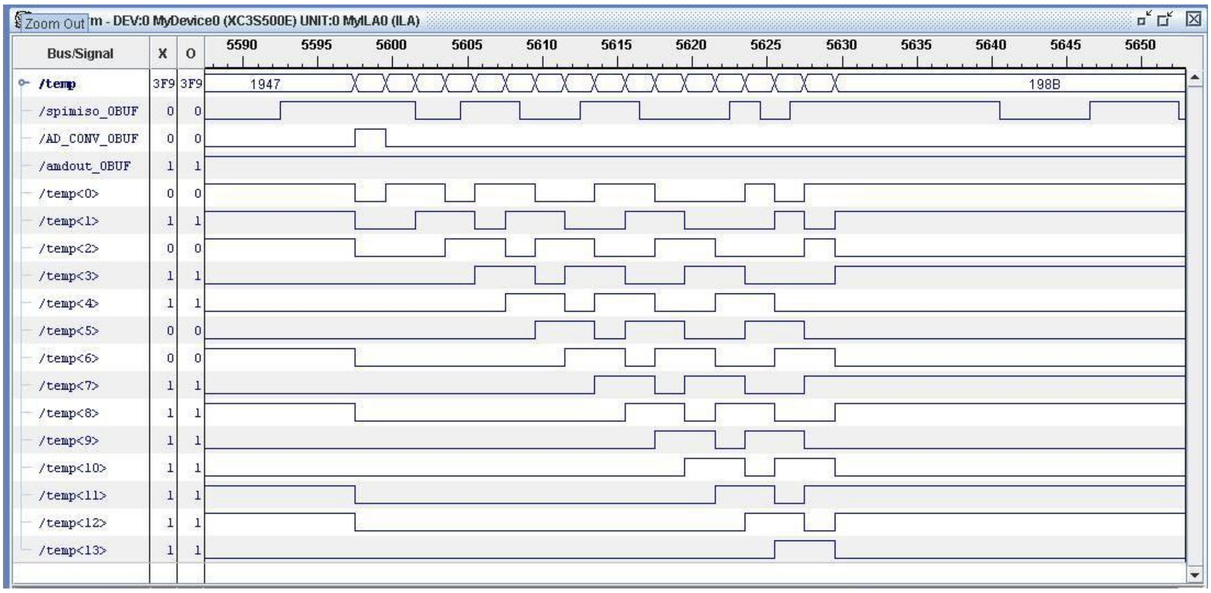


Figure 10: ChipScope Pro output waveform on the application of 1.6V DC supply voltage to the ADC.

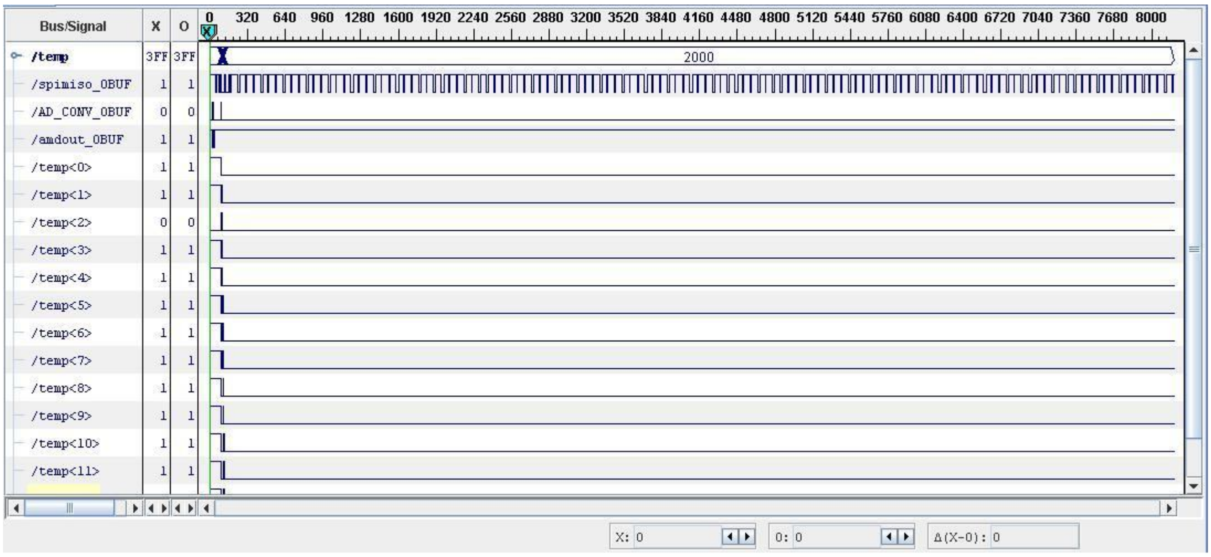
The AD_CONV signal is mainly responsible for the start and stop of the analog to digital conversion process. The 14 bit digital representation is shown as the ‘temp’ bus, with the individual bits represented below as temp<0> to temp<13>. As 1.6V DC supply was applied to the ADC, hence the number that would get generated would have been:

$$D[13{:}0] = (-20) \times \frac{1.6-1.65}{1.25} \times 8192 \text{ ....(2)}$$

$$\text{or, } D[13{:}0] = (1999)_{16}$$

As is evident from the ChipScope Pro output waveform, the value of the 14 bit representation is found to be $(198B)_{16}$ which is close to the actual value of $(1999)_{16}$ as determined mathematically.

Various other results that are obtained with the application of other voltages and different kinds of signals like the sine wave are depicted in the next figures.



Bus/Signal	X	O
/temp	3FF	3FF
/spimiso_OBUF	1	1
/AD_CONV_OBUF	0	0
/amdout_OBUF	1	1
/temp<0>	1	1
/temp<1>	1	1
/temp<2>	0	0
/temp<3>	1	1
/temp<4>	1	1
/temp<5>	1	1
/temp<6>	1	1
/temp<7>	1	1
/temp<8>	1	1
/temp<9>	1	1
/temp<10>	1	1
/temp<11>	1	1

Figure 11: Screenshot of the ChipScope Pro Analyser showing the conversion process givind result after the second AD_CONV signal.

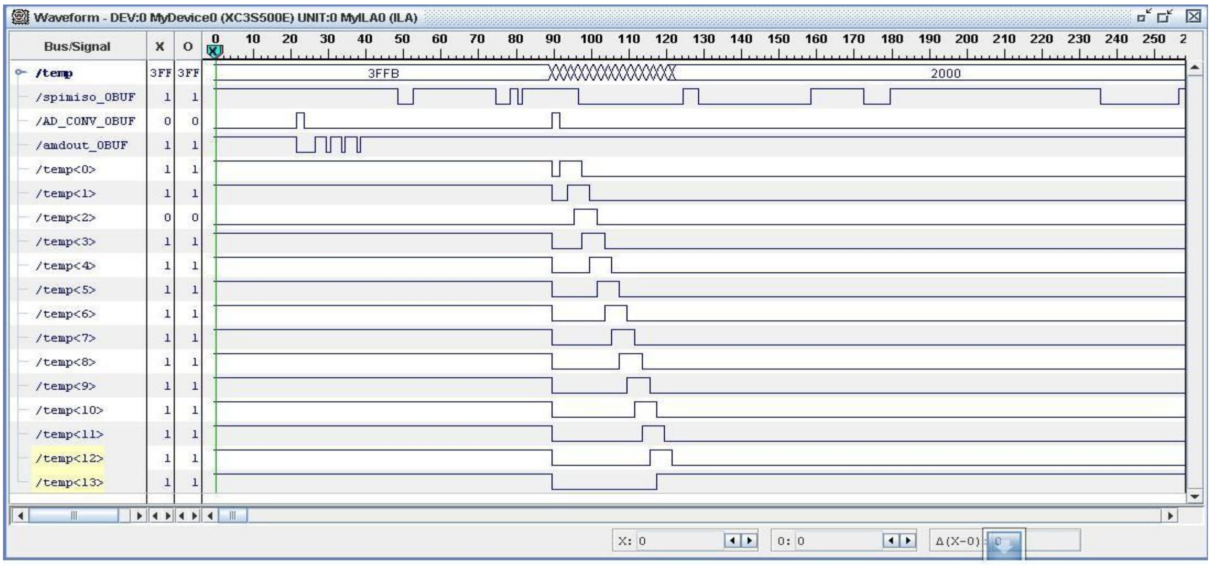


Figure 12: Zoomed view of Figure 10 showing a clear view of the results occurring after second AD_CONV signal only

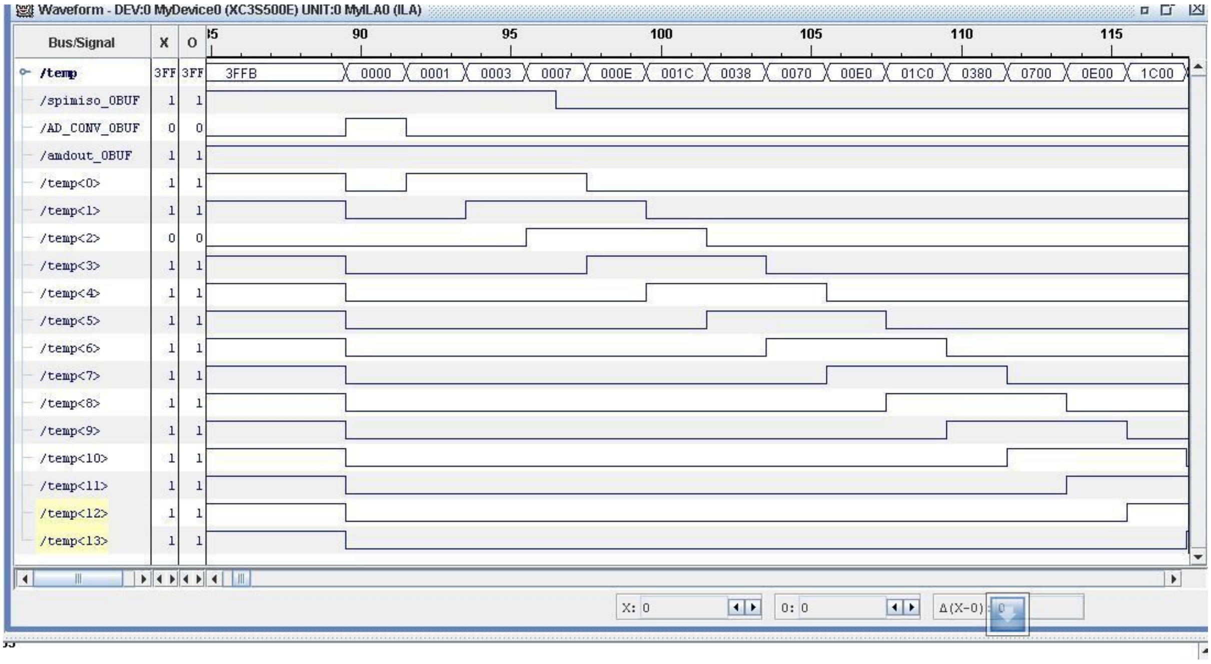


Figure 13: Further zoomed view showing how exactly the values to $(2000)_{16}$is getting generated.

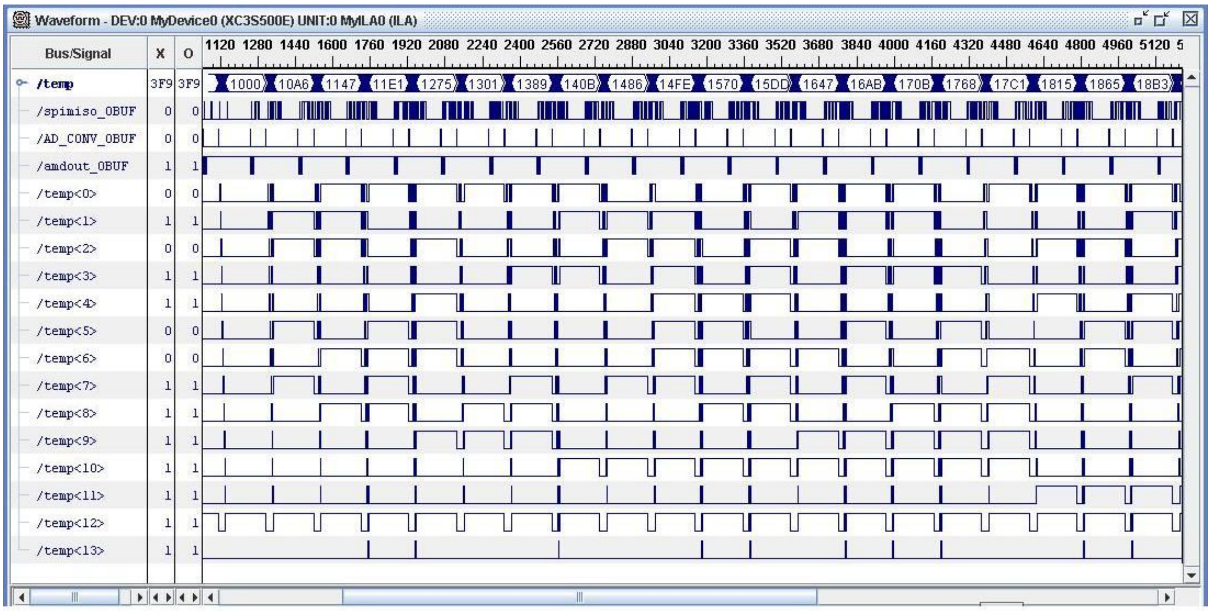


Figure 14: The ChipScope Pro analyser output waveform on the application of a sinusoidal signal to the ADC.

References:

[1] Castro-Lopez, R.; Ferandez, F.V.; Medeiro, F.; Rodriguez-Vazquez, A., "Behavioural modelling and simulation of ΣΔ modulators using hardware description languages," *Design, Automation and Test in Europe Conference and Exhibition, 2003* , pp. 168- 173, 2003

[2] Szermer, M.; Napieralski, A., "Modelling of the sigma-delta analogue to digital converters with application of VHDL-AMS," *Modern Problems of Radio Engineering, Telecommunications and Computer Science, 2004. Proceedings of the International Conference*, vol., no., pp.240-243, 28-28 Feb. 2004.

[3] Domenech-Asensi, G.; Ruiz-Merino, R.; Diaz Madrid, J.-A.; Neubauer, H.; , "Evaluation of VHDL-AMS models of a high performance ADC," *Industrial Electronics, 2007. ISIE 2007. IEEE International Symposium on* , vol., no., pp.1424-1429, 4-7 June 2007.

[4] Guelaz, R.; Desgreys, P.; Loumeau, P.; , "A sigma-delta bandpass ADC modelling in superconducting RSFQ technology with VHDL-AMS," *Specification, Verification and Design Languages, 2008. FDL 2008. Forum on* , vol., no., pp.90-93, 23-25 Sept. 2008.

[5] Xilinx, “Spartan-3E FPGA starter kit Board User Guide”, Internet: *http://www.xilinx.com/support/documentation/boards_and_kits/ug230.pdf/*, 20 Jan, 2011

[6] Linear Technology Limited, Datasheet of LTC 1407-1, Internet: *http://cds.linear.com/docs/Datasheet/14071fb.pdf/*

[7] Linear Technology Limited, Datasheet of LTC 6912, Internet: *http://cds.linear.com/docs/Datasheet/6912fa.pdf/*

Chapter 3: Digital to Analog Convertor (DAC)

3.1 Introduction:

Features of the on-board DAC (LTC 2624):

- Quad 12 bit DAC. It is a quadrature type 12 bit DAC .It means it has 4 independent DAC’s inbuilt within it. We can use one of them at a time or all of them at a time.
- Wide 2.5V to 5.5V Supply Range. It can accept its power supply over a range of 3 volts. But anything below 2.5 V and anything above 5.5 V is not acceptable.
- Low Power Operation: 250µA per DAC at 3V. It is a very low power consuming device. Hence it is ideal for chip level application
- Individual DAC Power-Down to 1µA, Max. The current is very low which passes through the DAC. Hence chances of it getting short circuited or getting overheated is very less.
- Ultralow Crosstalk between DACs (<5µV). There is hardly zero or very less crosstalk between the 4 internal DAC’s which work at tandem. This ensures that all 4 can work together without affecting the other.
- High Rail-to-Rail Output Drive (±15mA). The output swing is very high .It is approximately 30 mA. Such an output swing ensures that the DAC doesn’t reject any of the incoming input signals and that while conversion no bit is missed.
- Double Buffered Digital Inputs. The inputs that come to it are double buffered which means that they are twice relayed and then sent to the dac . It means that it acts like a master slave pair.
- Power-On Reset to Midscale. The DAC can at any stage be set to its midscale value without having to restart the entire process all over again.
- 16-Lead Narrow SSOP Package. The packaging is narrow and compact. This makes it usable anywhere and everywhere and on any platform.

3.2 Applications:

- Mobile Communications: Transmission of signals in the digital domain is a very costly affair. Hence the signals necessary for transmission, in case of mobile communication, is done via analog signals. But the processing of the signals is done in the digital domain only. Hence conversion of digital to analog signals is a must and here comes the importance of the DAC.
- Process Control and Industrial Automation: Digital signals are used for the various industrial automation purposes. But for transmitting those signals over large distances analog signals are necessary. And hence the digital signals need to be converted into the analog form, which is done via the DAC.
- Automatic Test Equipment: Any kind of automatic test equipment would require an analog component for the proper functioning of the device. Also in some case the signals are sent from the control room itself .Hence digital to analog signal conversion is necessary. Thus comes in the use of the DAC.

3.3 Working of DAC:

The reference voltage for DAC at outputs A and B is 3.3V whereas that for DAC at outputs C and D is 2.5V.The reference voltages themselves have a ±5%tolerance, so there will be slight corresponding variances in the output voltage. The DAC converts 12 bit unsigned number into the corresponding analog value. The voltage output equation [5] can be written as:

$$V_{OUT} = \frac{D[11:0]}{4096} \times V_{REFERENCE} \quad (3)$$

Where $V_{REFERENCE}$ = 3.3V for A and B; 2.5V for C and D.

Here as the DAC converts 12 bit unsigned number into the corresponding analog value hence the output is scaled by 2^{12} or 4096. And the result thus obtained is multiplied by $V_{REFERENCE}$ to get the desired analog output. SPI communication is of prime importance in case of DAC.

Some of the characteristics of the SPI bus that make it unique for this purpose are: it is full-duplex, synchronous and character oriented bus channel employing four-wire interface. The bus master transmits the serial data (SPI_MOSI) to the selected bus slave (selection done via address of the slave device) and drives the bus clock signal (SPI_SCK). At the same time, the bus slave also provides serial data (SPI_MISO) back to the bus master. This signal can be ignored.

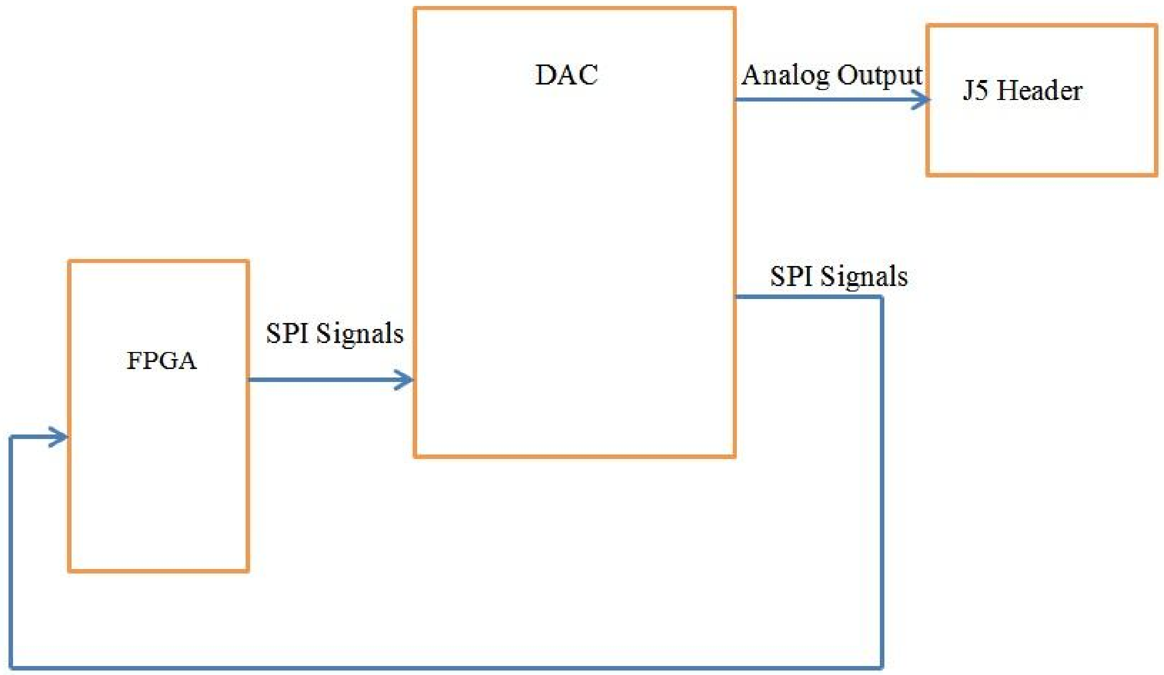


Figure 15: Block diagram for the Digital to Analog convertor connection schematics [5].

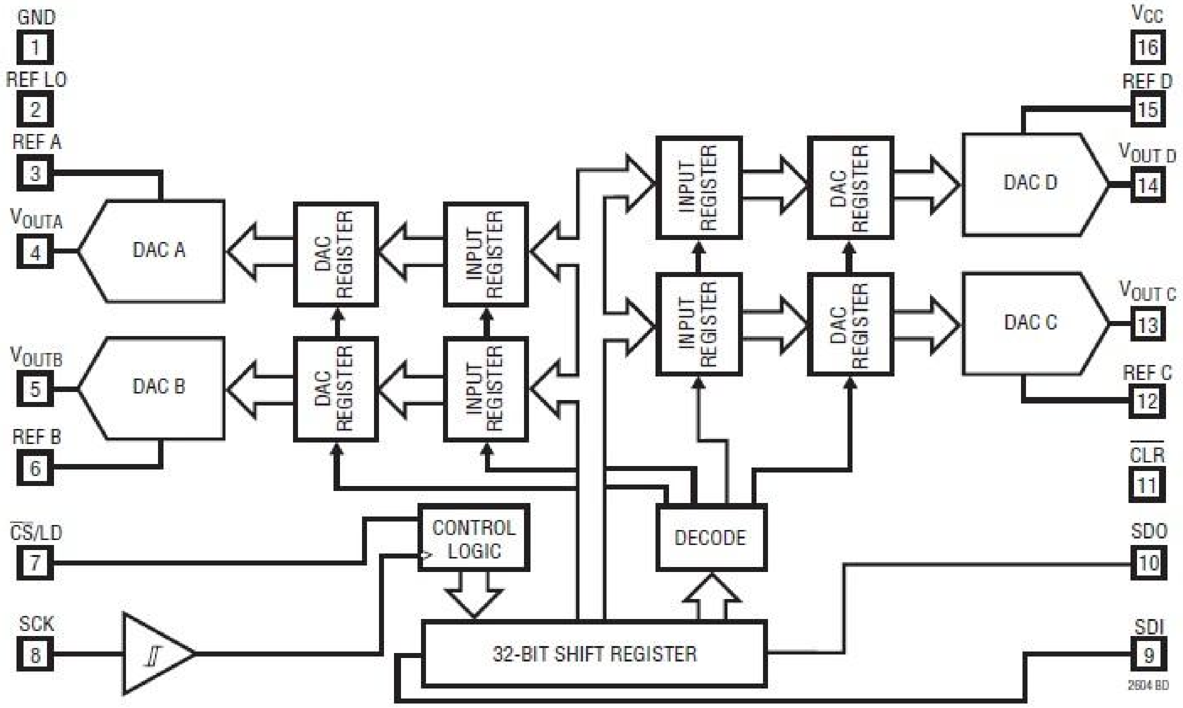


Figure 16: The inner circuitry of the LTC 2624 DAC chip. [1]

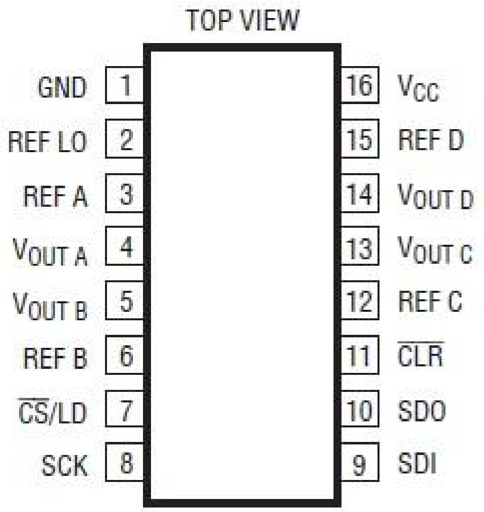


Figure 17: The Top view of the LTC 2624 DAC chip. [1]

3.4 Interface signals between the FPGA and the DAC:

The interface signals used for the interfacing of the DAC with the FPGA are:

(a) SPI_MOSI: It is the 12 bit unsigned number that gets converted into the analog value via the digital to analog convertor. It is the serial data which acts as the output from the master and input to the slave device. The direction of this signal is from the FPGA (master) to the DAC (slave). This signal originates at the pin T4 of the Spartan 3E FPGA board.

(b) DAC_CS: It is an active low chip select signal. The actual digital to analog conversion process starts when this signal returns high. As this signal drives the DAC process hence the direction of this signal is from the FPGA to the DAC.

(c) SPI_SCK: It is the bus clock signal that helps in conversion process.

(d) DAC_CLR: This is the asynchronous, active low reset input to the DAC given by the FPGA.

(e) SPI_MISO: This is the serial data that is input to the FPGA (master) from the slave (DAC) output.

3.5 Communication Protocol for the DAC:

The on-board DAC supports both a 24-bit and 32-bit protocol within the SPI bus.

• For 24-bit protocol:

- The CS/LD input is level triggered. When this input is taken low, it acts as a chip-select signal, powering-on the SDI and SCK buffers and enabling the input shift register.
- Data (Serial Data Input) is transferred at the next 24 rising SCK edges. The 4-bit command, C3-C0, is loaded first; then the 4-bit DAC address, A3-A0. And

finally the 16-bit data word. The data word comprises the 12-bit input code, ordered MSB-to-LSB, followed by 4 don’t-care bits.

- Data can only be transferred to the device when the CS/LD signal is low. The rising edge of CS/LD ends the data transfer and causes the device to carry out the action specified in the 24-bit input word.

• For 32-bit protocol, the FPGA first sends eight dummy or ‘Don’t care’ bits and then follows the same procedure as the 24-bit protocol.

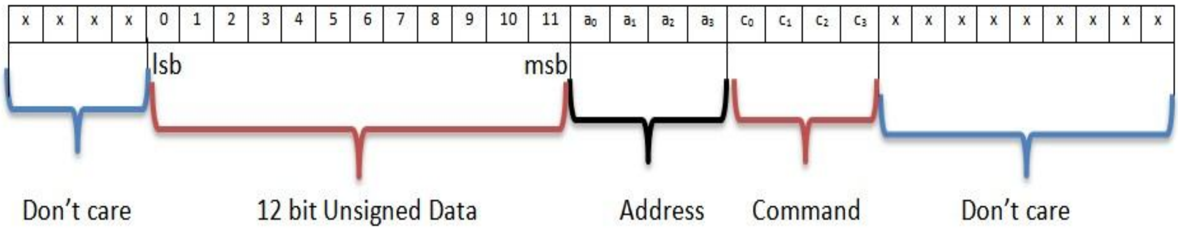


Figure 18: The 32-bit protocol to the on-board DAC (LTC 2624), showing the dummy bits, the 12 – bit unsigned number, the 4 – bit command, and the 4 – bit address for selecting the required DAC [5].

3.6 Methodology:

For interfacing the on-board DAC, the following methodology was used:

(i) A VHDL program is written and the 12 bit unsigned data is given by the program itself to the FPGA.

(ii) The FPGA outputs this 12 bit data to the DAC so that the conversion of the given 12 bit data into analog value by the DAC takes place.

(iii) The 4 bit command and the 4 bit address required to select the desired DAC is also given via the VHDL program.

(iv) The DAC then processes the incoming 12 bit data and depending on the reference voltage used (which again depends on the output DAC chosen via program) the DAC converts the digital data into analog value which can be observed via a Digital Storage Oscilloscope (DSO).

(v) For an increasing pattern given via the program, a ramp signal would be generated as was found as the output.

3.7 Results:

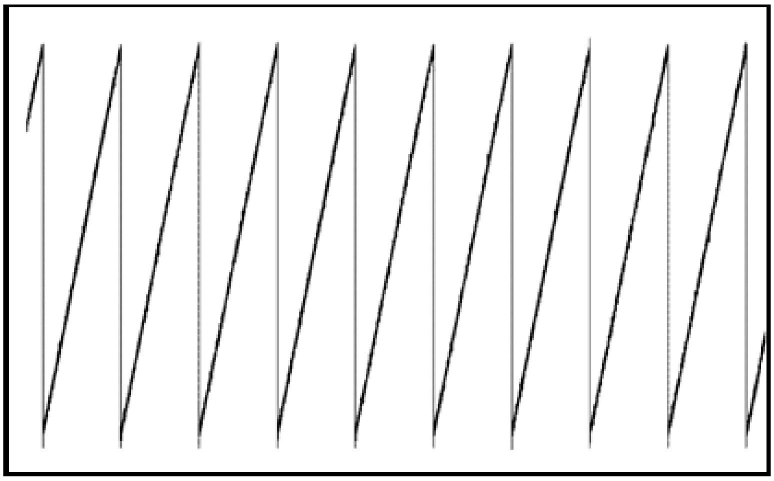


Figure 19: Ramp signal generated on passing an increasing pattern through the DAC.

Analysing the ramp signal, it was found that for the input data, output obtained at the chosen DAC (DAC at output A was chosen for the display of the waveform), was a good approximation of the analog signal.

Table 3: Table showing the results of the applied analog voltage to ADC and corresponding DAC output.

Analog Input given to the ADC	Gain	14 bit representation	Output observed via DAC
1.5V	-2	00001111010111	1.21V
1.6V	-2	00000110001111	1.18V
1.6V	-1	00000101001000	1.15V

The problem with the use of on-board ADC and DAC together in the Spartan 3E FPGA kit is that though the ADC converts the analog value into a 14 bit digital representation, the DAC is a 12 bit one, converting only 12 bits to the corresponding analog value. This results in the lower two bits getting lost; and hence the observation as tabulated above results. This can be overcome by using separate DAC for the conversion process, but then again a simply independent system can't be created. So this approximation is taken and the system is designed taking this into account.

References:

[1] Linear Technology Limited, Datasheet of LTC 2604 family, Internet: *http://cds.linear.com/docs/Datasheet/2604fd.pdf/*

[2] J. Candy and G. C. Temes, "Oversampling methods for AID and D/A conversion," in *Oversampling delta-sigma converters*, pp. 1-25, IEEE Press, 1992.

[3] Emi Miyata, Chikara Natsukari, Daisuke Akutsu, Tomoyuki Kamazuka, Masaharu Nomachi, Masanobu Ozaki, "Fast and flexible CCD-driver system using fast DAC and FPGA", *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, Volume 459, Issues 1–2, 21 February 2001, Pages 157-164.

[4] Magrath, A.J.; Clark, I.G.; Sandler, M.B.; , "Design and implementation of a FPGA sigma-delta power DAC," *Signal Processing Systems, 1997. SIPS 97 - Design and Implementation., 1997 IEEE Workshop on* , pp.511-521, 3-5 Nov 1997.

[5] Xilinx, "Spartan-3E FPGA starter kit Board User Guide", Internet: *http://www.xilinx.com/support/documentation/boards_and_kits/ug230.pdf/*, 20 Jan, 2011

[6] Cheung, R.C.C.; Pun, K.P.; Yuen, S.C.L.; Tsoi, K.H.; Leong, P.H.W.; , "An FPGA-based re-configurable 24-bit 96kHz sigma-delta audio DAC," *Field-Programmable Technology (FPT), 2003. Proceedings. 2003 IEEE International Conference on* , vol., no., pp. 110- 117, 15-17 Dec. 2003.

Chapter 4: Conclusions and Future Work

4.1 Conclusion:

- The onboard ADC & DAC of the Spartan 3e FPGA Board were properly interfaced with real world signal.
- Chipscope-Pro was used to analyze the internal signals.
- Because of the low power consuming capability of the ADC it can handle any kind of data.
- The ADC was analysed for a Sinusoid and a constant voltage supply
- The DAC was analysed and checked with a ramp input.
- Mathematically it was proved that the ADC output signal seen through Chipscope-pro and the value theoretically were almost same.

4.2 Future Work:

- The codes for ADC and DAC to be further used for working it at real time
- Fast Fourier Transform and Short Time Fast Fourier Transform to be implemented on it giving analog inputs, then using the ADC& DAC and giving the analog form of the respective transform
- It can be used for Register Storage for high Capacity data