

FPGA Implementation of Electrocardiography (ECG) Signal Processing

¹Sunil Kumar, ²Gurmohan Singh, ³Manjit Kaur

^{1,3} ACS Division, Centre for Development of Advanced Computing (C-DAC), Mohali, 160071, India.
s.k.pal@hotmail.com, manjeet@cdac.in

² DEC Division, Centre for Development of Advanced Computing (C-DAC), Mohali, 160071, India
gurmohan@cdac.in

Abstract

This paper describes the design and implementation of ECG signal processing on FPGA. Lot of research work has been carried out for implementation of ECG signal processing circuitry on FPGA. In this research work all the work is carried out using Xilinx Spartan-3E FPGA starter kit which is taken as a central processing unit. On board Programmable preamplifier, ADC and quad DAC on Spartan-3E FPGA kit have been employed for amplification, A/D conversion and D/A conversion respectively. To generate the necessary control signals for on board Pre-amplifier, A/D converter and D/A Converter, we have designed different HDL modules. The digital output of A/D converter is applied to the low pass FIR digital filter for processing the ECG signals. All the designed modules are integrated in a single TOP level entity. All modules are designed, functionally verified, synthesized, placed & routed using Xilinx 14.3i ISE tool. The functional verification has been performed using ModelSim simulator.

Keywords: ECG, VHDL, ADC, DAC, Pre-Amplifier, FIR Filter, Spartan-3E, FPGA.

1. INTRODUCTION

Electrocardiography (ECG or EKG) is an interpretation over a particular period of time of heart's electrical activity. The sensing electrodes are attached at particular points on human body non-invasively and the electrical activity is detected by front end analog capture circuit. The recording produced by this non-invasive procedure is termed as electrocardiogram (also EKG or ECG). ECG signal represents measure of rate and regularity of heartbeats along with position and size of the heart chambers. It also reveals if there is damage to heart, the effects of medicines or implanted devices like pacemaker used to regulate the heart. From the interpretation of ECG signal, one can also diagnose and measure presence of abnormal rhythms of the heart generally produced by serious damage to the conductive tissue carrying electrolyte imbalances or electrical signals. The Fig. 1 shows a sample of electrocardiogram (ECG) signal. The captured ECG signal is usually contaminated by noise. Therefore, it is important to filter the noise to prevent any mistakes in further analysis of the signal.

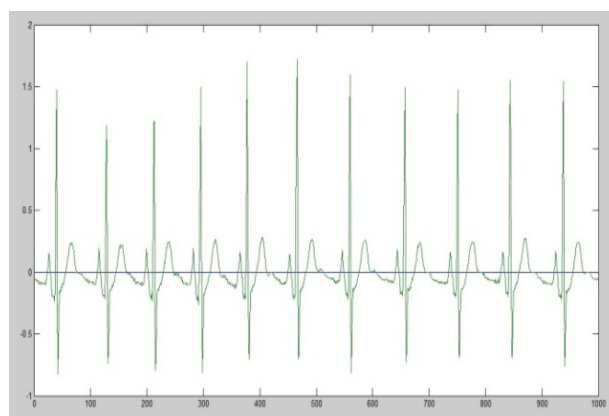


Figure 1: Sample of ECG signal

2. SYSTEM STRUCTURE

Field programmable gate arrays (FPGA) find numerous applications for rapid prototyping of digital circuits and systems such as video and audio processing, digital signal processing and digital systems designing and implementation as they offer low start up and implementation cost, fast manufacturing turnaround time, and ease of design modification [2]. Since most of the biological signals processing systems require that the hardware has large computing power and lower power consumption, FPGA meets those requirements efficiently by virtue of its parallel processing hardware resources and high performance. So, any biological signal processing system can be realized on FPGA [5]. To develop an ECG signal processing system on SPARTAN 3E FPGA, we have designed HDL based Pre-Amplifier control module, ADC control module, Low Pass FIR Filter, DAC control module, clock generating module and SPI Clock Interface for ECG data collection, Amplification, A/D conversion SPI communication and D/A conversion etc. The Fig. 2 shows a FPGA based ECG signal processing system.

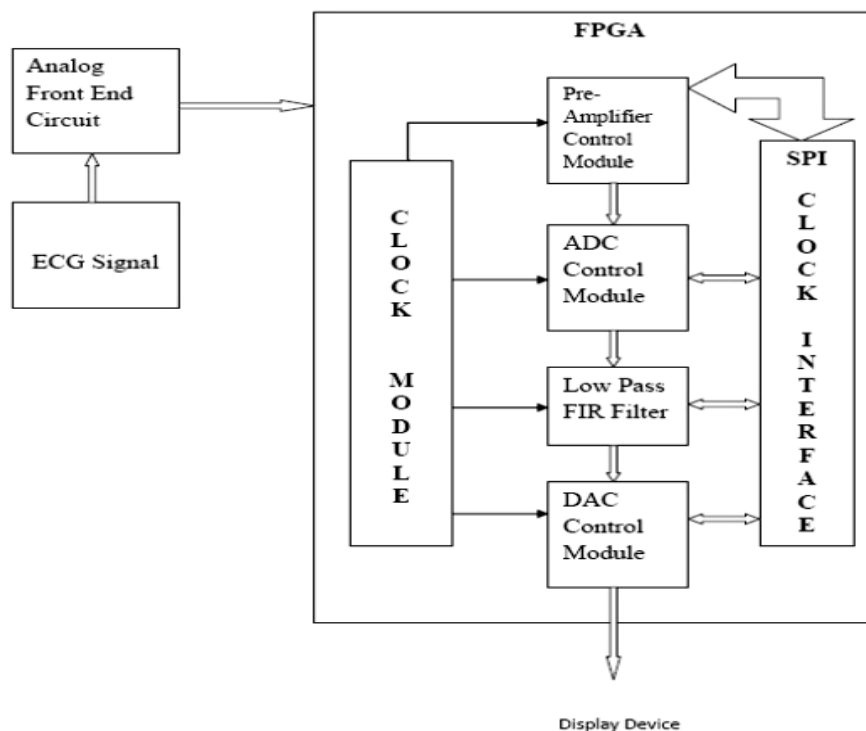


Figure 2: FPGA based ECG Signal processing system

2.1 Programmable Pre-Amplifier & ADC control module

For the amplification of ECG signal coming from analog front end circuit, we are using on board programmable preamplifier (LTC6912-1) chip of FPGA kit. The preamplifier comprises of two independent inverting amplifiers with programmable gain setting [3]. The Linear Technology (LTC1407A-1) ADC is fed with output of pre-amplifier. The ADC and pre-amplifier are either programmed serially or controlled by the FPGA.

Incoming ECG analog signals presented on the VINA or VINB inputs are amplified with reference to 1.65V. The reference signal of 1.65V is internally generated within this chip by a voltage divider circuit operating at 3.3V. The gain of each amplifier can be set from -1 to -100. The Table I show typical preamplifier gain settings with minimum & maximum input voltage range. The interfacing signals between pre-amplifier and the FPGA device are listed in Table II. The SPI bus is used to share 'SPI_SCK' and 'SPI_MOSI, SPI_MISO' signals with other devices. The active-low 'AMP_CSsignal' act as slave select input to the amplifier [3].

The Fig. 3 shows the SPI-serial interface with the amplifier. The gain of two pre-amplifiers is set as an 8-bit command word and serially sent to pre-amplifier via SPI_MOSI. The MSB bit (B3) is set first [3]. The amplified ECG output of Programmable preamplifier appears on the ADC chip which provides two channel ADCs. On application of AD_CONV signal, both analog inputs are sampled simultaneously. Table III lists the interface signals between the ADC and the FPGA device.

Table I: Programmable gain settings for Pre-Amplifier [3]

Gain	Input Voltage Range					
	A3	A2	A1	A0	Min	Max
	B3	B2	B1	B0		
0	0	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	1.587	1.712
-50	0	1	1	0	1.625	1.675
-100	0	1	1	1	1.637	1.662

Table II: Pre-Amplifier and FPGA Interface Signals [3]

Signal	FPGA Pin	Direction	Description
SPI_MOSI	4	FPGA → AD	Serial data: Master Output, Slave Input. Presents 8-bit programmable gain settings.
AMP_CS	N7	FPGA → AMP	Active-Low chip-select.
SPI_SCK	U16	FPGA → AMP	Clock for synchronization
AMP_SHDN	P7	FPGA → AMP	Active-High shutdown, reset
AMP_DOUT	E18	FPGA → AMP	Serial data. Echoes previous amplifier gain settings.

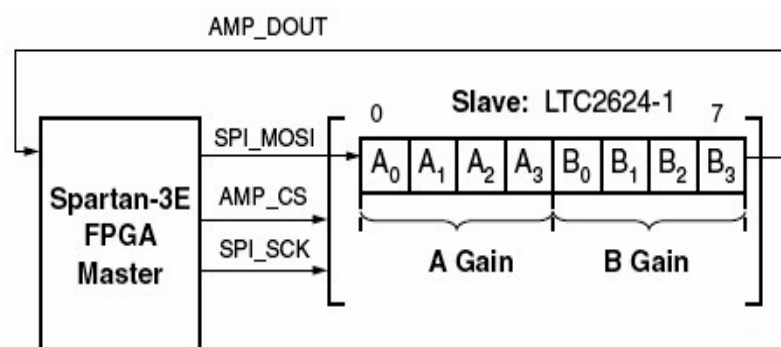


Figure 3: SPI Serial Interface to Amplifier [3]

Table III: ADC Interface Signals [3]

Signal	Pin	Direction	Description
--------	-----	-----------	-------------

FPGA			
SPL_SCK	U16	FPGA → ADC	CLOCK
AD_CONV	P11	FPGA → ADC	Active-High Shutdown and reset
SPL_MISO	N10	FPGA → ADC	Serial data: Master Input, Serial Output.

The SPI bus signals are also shared by other devices available on the FPGA board. It is very important to note that when FPGA device communicates with the ADC or AMP, other devices are disabled to avoid bus contention. Table IV provides the signals and logic values required to disable the other devices. For blocking the DC value in analog input ECG signal here we generally use greater than 1μF DC blocking capacitor and then connect AC signals to VINA or VINB via DC blocking capacitor [11].

Table IV: Disable Other Devices on SPI Bus [3]

Signal	Disabled Device	Disabled value
SPI_SS_B	SPI Serial Flash	1
AMP_CS	Programmable pre-amplifier	1
DAC_CS	DAC	1
SF_CEO	Parallel Flash PROM	1
FPGA_INIT_B	Platform Flash PROM	1

2.2 Designing of a low pass digital filter to remove noise in the electrocardiogram (ECG) Signal

The outcome of ADC chip will be a 14-bit, digital value of amplified ECG signal. It is important to filter the noise to prevent any mistakes in further analysis of the signal. To remove noise in the ECG signal, we had designed a low pass digital filter in VHDL. There have some specific advantages of using FIR Filter as a low pass digital filter to remove the noise in ECG signal. The main advantages are:-

- 1) FIR digital filters are simple to design and its structures are easy to implement.
- 2) FIR filters are guaranteed to be stable due to all zero structures and have linear phase.
- 3) FIR filters also have low sensitivity to filter coefficient quantization errors which will ease in hardware implementation.
- 4) FIR digital filters present equal delay to all frequencies. These results in no phase distortion and the inputs will be delayed by a specific delay period when it is transmitted to the output. In digital communication, a filter with constant group delay is highly desirable.

2.2.1 FIR Filter Design

The designing of a FIR filter can be defined by the following equation:

$$Y[n] = \sum_{i=0}^N b_i x[n - i] \quad (1)$$

Here, 'y[n]' is the filter output, 'x[n]' in the input signal, 'b_i' is the filter coefficients and 'N' is the filter order. The higher the value of N results in higher precision but makes filter design more complex. The Fig. 4 shows basic block diagram of a digital filter.





Figure 4: Basic block diagram of a Digital filter

Most of the idealized frequency response has discontinuities or abrupt jump at the boundaries between bands. Hence, the impulse response becomes non-causal and unstable. To avoid these problems, it is necessary to truncate the ideal impulse response by allowing some deviation from the ideal response. The deviation includes the pass band cutoff frequency (W_{pass}), stop band cutoff frequency (W_{stop}), pass band deviation (δp) and stop band deviation (δs). Fig. 5 shows the transfer function versus frequency specification of a low pass filter for a low pass filter.

The equation (2) can be used to estimate the filter order of an equiripple filter for a low pass filter with transition width (Δf), pass band ripple (δp) and stop band ripple (δs).

$$N = \frac{-10 \log (\delta p \delta s) - 13}{14.6 \Delta f} \quad (2)$$

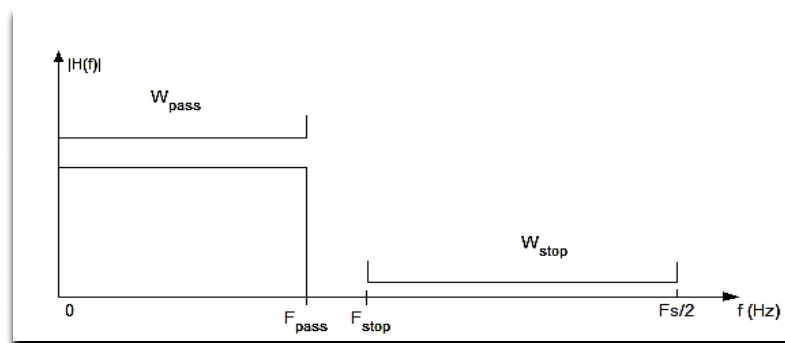


Figure 5: Transfer function versus frequency of a low pass filter

2.2.2 FIR Filter Specifications

The design specifications for a FIR digital filter include:

- 1) Characteristics of the filter (low, high or band pass)
- 2) Pass band frequency
- 3) Stop band frequency
- 4) Sampling frequency
- 5) Number of order
- 6) How to implement the filter
- 7) Design Constraint (cost, resource limitation)

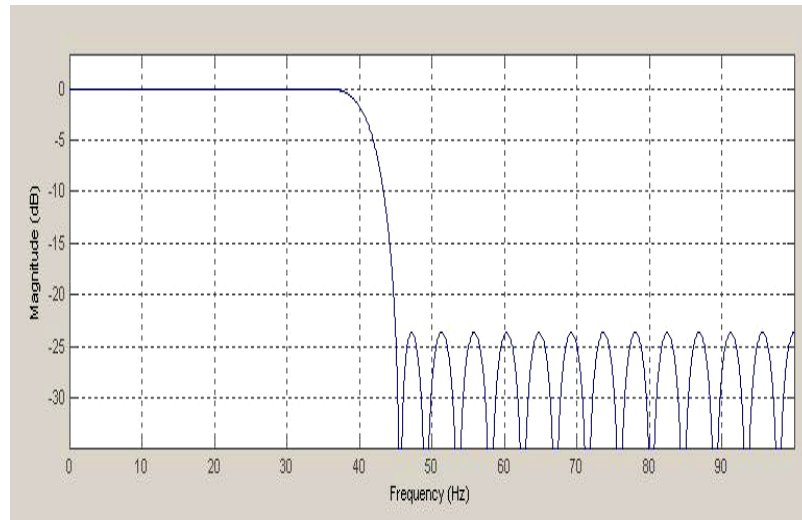


Figure 6: Magnitude response of a low pass filter with order 50

The Fig. 6 shows the magnitude frequency response of low pass filter obtained using the MATLAB FDA tool command. By asserting the desired value of stop band frequency, pass band frequency and sampling frequency, MATLAB will generate the appropriate filter order required. For simplification, the filter coefficients are generated using the MATLAB FDA tool. Table V shows the design specifications of designed FIR low pass filter.

Table V: Specifications of designed FIR Low Pass Filter

pass band deviation (δp)	0.07
stop band deviation (δs)	0.0001
stop band frequency (wp)	45Hz
Pass band frequency (ws)	30Hz
Sampling frequency (fs)	200Hz
Order (N)	50

2.2.3 Realization on Filter Structures

Block diagram can be used to represent the computational algorithm of a FIR filter. Multipliers, adders and registers are used to represent the basic building block diagrams. These basic block elements and their equivalent signal flow diagrams are shown in Fig. 7.

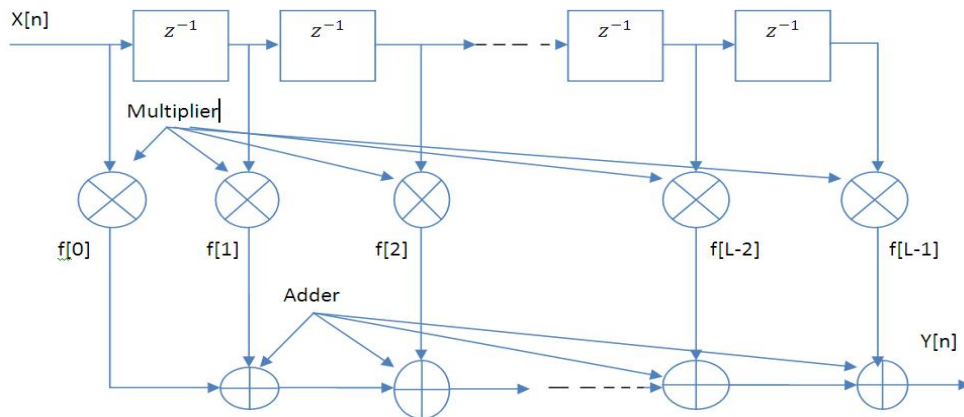


Figure 7: Structure of a FIR filter of order L

Here $x[n]$ is the input data, $y[n]$ is the output data and $f[0], f[1], f[2], f[L-2], f[L-1]$ are the unit impulse responses. Based on the equation of a FIR filter, a basic block diagram can be drawn. This is to simplify in the hardware architecture design [8].

2.2.4 Conversion of Coefficient to Integer

To design an ideal filter, it requires an infinite word length of filter coefficients. However, it is not possible to do so. Hence, an appropriate approach to solve this problem is to round off the filter coefficients to an X-bit representation. In order to minimize the hardware used in this project, the filter coefficient can be quantized to 8, 16 or 32-bit either in signed or in unsigned integer.

2.2.5 FIR Low Pass Filter Design Using MATLAB

The Fig. 8 represents the ECG signal plotted in time domain. It is easier to analyze noise region of a signal by plotting it in the frequency and time domain. The Fig. 9 shows the noisy ECG signal filtered using the designed low pass FIR digital filter.

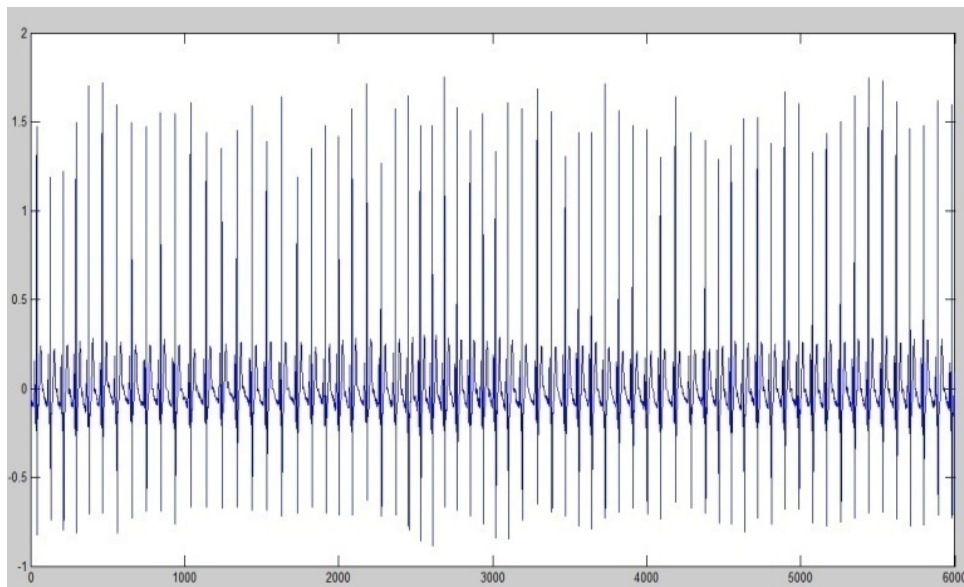


Figure 8: ECG signals with noise plotted in time domain

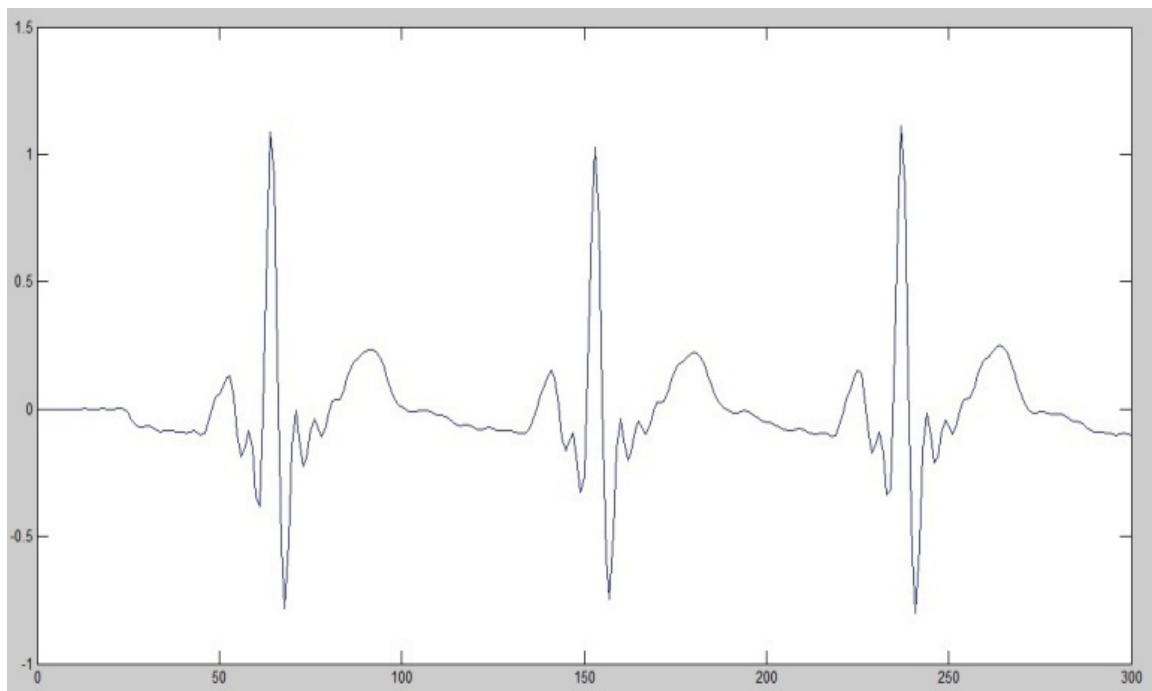


Figure 9: Filtered ECG signal using MATLAB

2.3 DAC control module

The Spartan-3E FPGA board contains an SPI-compatible, 4-channel, serial DAC. This is a Linear Technology (LTC2624) quad DAC having 12-bit unsigned resolution [3]. The filtered digital ECG signal from low pass FIR filter will pass through DAC for reconstruction in its analog format that which we can directly send and displayed on CRO or any other display.

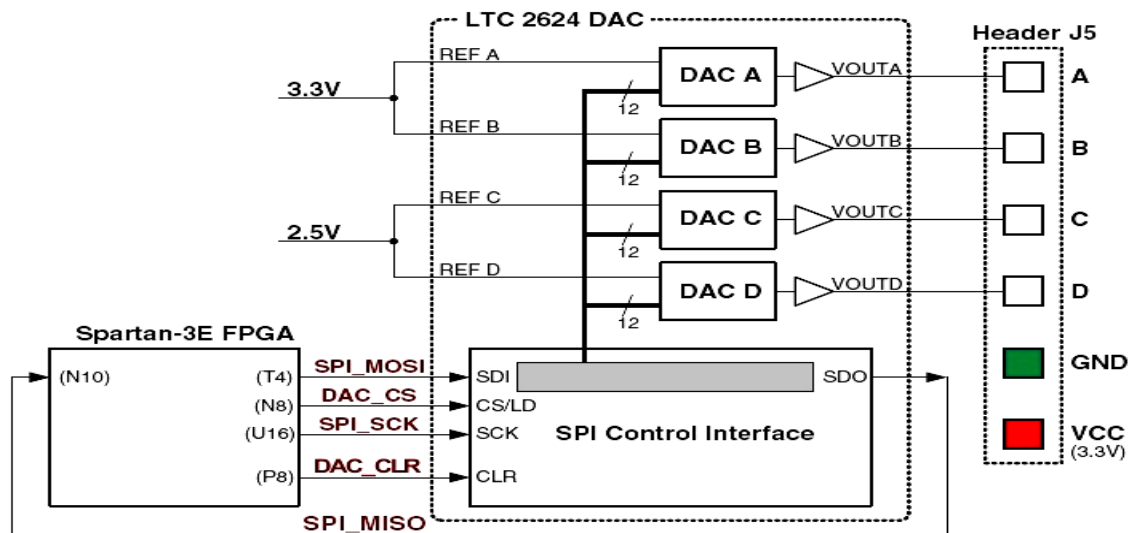


Figure 10: Digital-to-Analog Connection Schematics [3]

The FPGA device as shown in Fig. 10 employs SPI to transmit digitized values to the four DAC channels [3].

Table VI: DAC Interface Signals

Signal	FPGA Pin	Direction	Description
SPI_MOSI	T4	FPGA → DAC	Serial data: Master Output, Slave Input
DAC_CS	N8	FPGA → DAC	Active-Low chip-select. Digital-to-analog Conversion starts when signal returns High.
SPI_SCK	U16	FPGA → DAC	Clock
DAC_CL	P8	FPGA → DAC	Asynchronous, active-Low reset input
SPI_MISO	N10	FPGA → DAC	Serial data: Master Input, Slave Output

Table VI lists the interface signals between the FPGA and the DAC. The signals shared with other devices using SPI bus are ‘SPI_MISO’, ‘SPI_MOSI’, and ‘SPI_SCK’. The ‘DAC_CS’ signal represents active-Low slave select input to the DAC. The ‘DAC_CLR’ signal is asynchronous reset active-low input to the DAC. The Fig. 11 shows the SPI communications protocol needed to interface with the on board DAC. The DAC supports 32-bit protocol is shown [3].

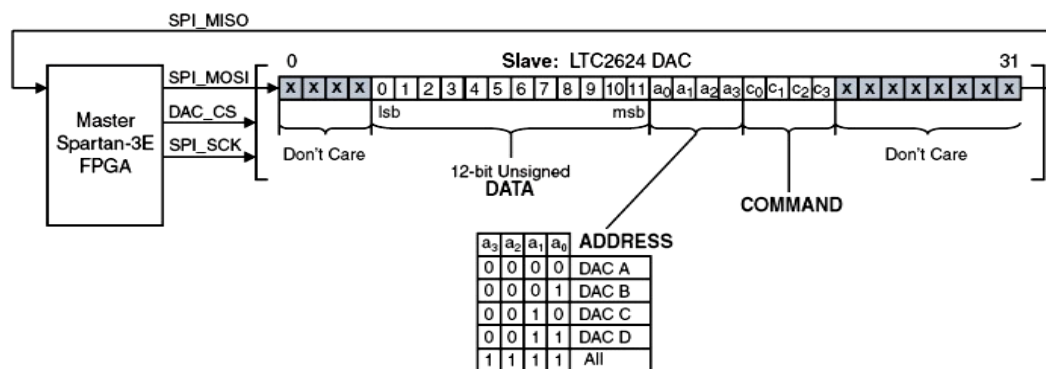


Figure 11: SPI Communications Protocol to DAC [3]

At the start, FPGA device transmits eight “don’t care” bits, thereafter 4-bit command is sent. The widely used command with the FPGA device is “COMMAND [3:0] = 0011”. This command updates immediately the selected DAC output with the specified data value. After this command, using a 4-bit address field the FPGA device selects one or all the DAC output channels. Then FPGA transmits first 12-bit unsigned ECG data value on one clock tick event and remaining bit of DAC on the next clock event that the DAC converts to an analog value on the selected output. At the end, the remaining 4 don’t care bits pad the 32-bit command word [3]. It is very important to note that other devices are disabled during communication between FPGA and DAC to avoid bus contention. Table VII lists logical values and signals required to disable the other devices.

Table VII: Signal and Logical values to disable other devices

Signal	Disabled Device	value
SPI_SS_B	SPI serial Flash	1
AMP_CS	Programmable pre amplifier	1
AD_CONV	ADC	0
SF_CE0	Flash PROM	1
FPGA_INIT_B	Flash PROM	1

3. SIMULATION RESULT

Each module of ECG signal processing system is described in VHDL, functionally verified, synthesized and then implemented on FPGA. Then all modules were integrated into a signal top level entity *ECG_top_level_module1*. The top level entity is again functionally verified, synthesized and implemented on FPGA. Xilinx Spartan-3E FPGA board is used for implementation of ECG signal processing system. The Xilinx ISE 14.3 design suite is used for HDL level design entry, synthesis and bit-stream file generation. Model-Sim simulator is used for verification of functionality of individual modules as well as top level *entity ECG_top_level_module1* software. The RTL schematic of top level entity is shown in Fig.12.

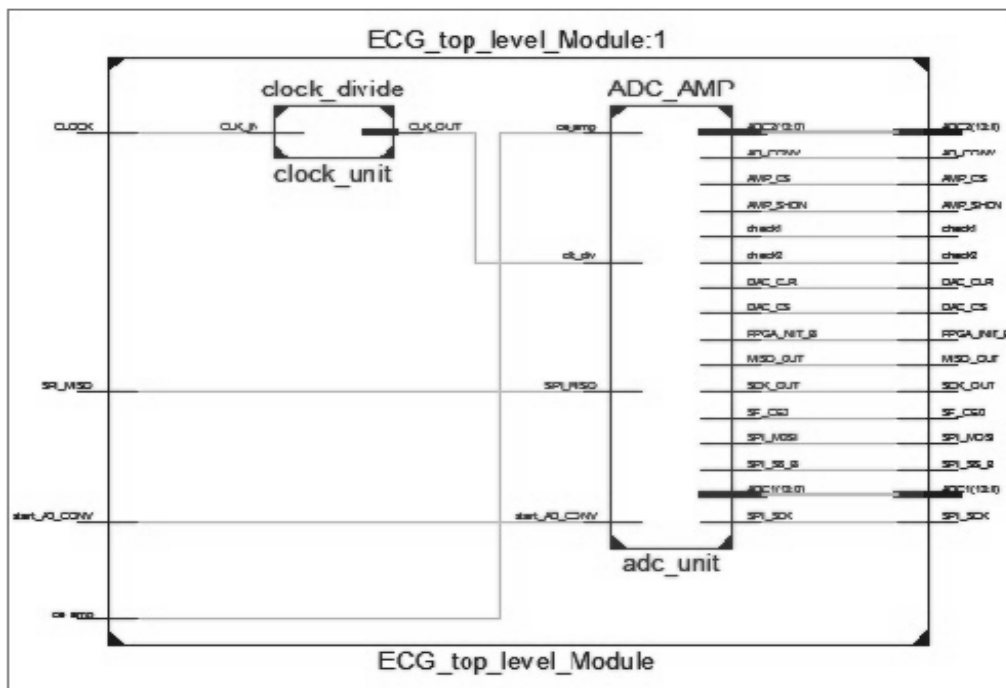


Figure 12: RTL Schematic of ECG_top_level_module

The simulation waveforms of designed Pre-Amplifier, ADC and DAC control module and 50th order Low pass FIR filter are shown in Figs 13 and 14 respectively.

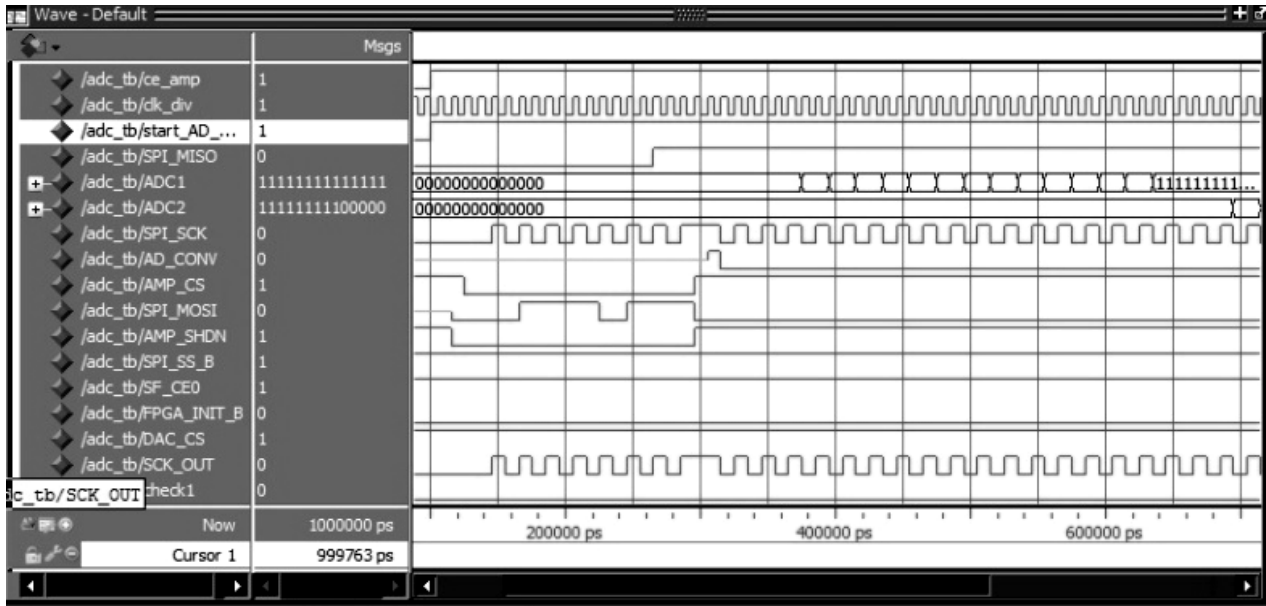


Figure 13: Timing Simulation of ECG_top_level_module

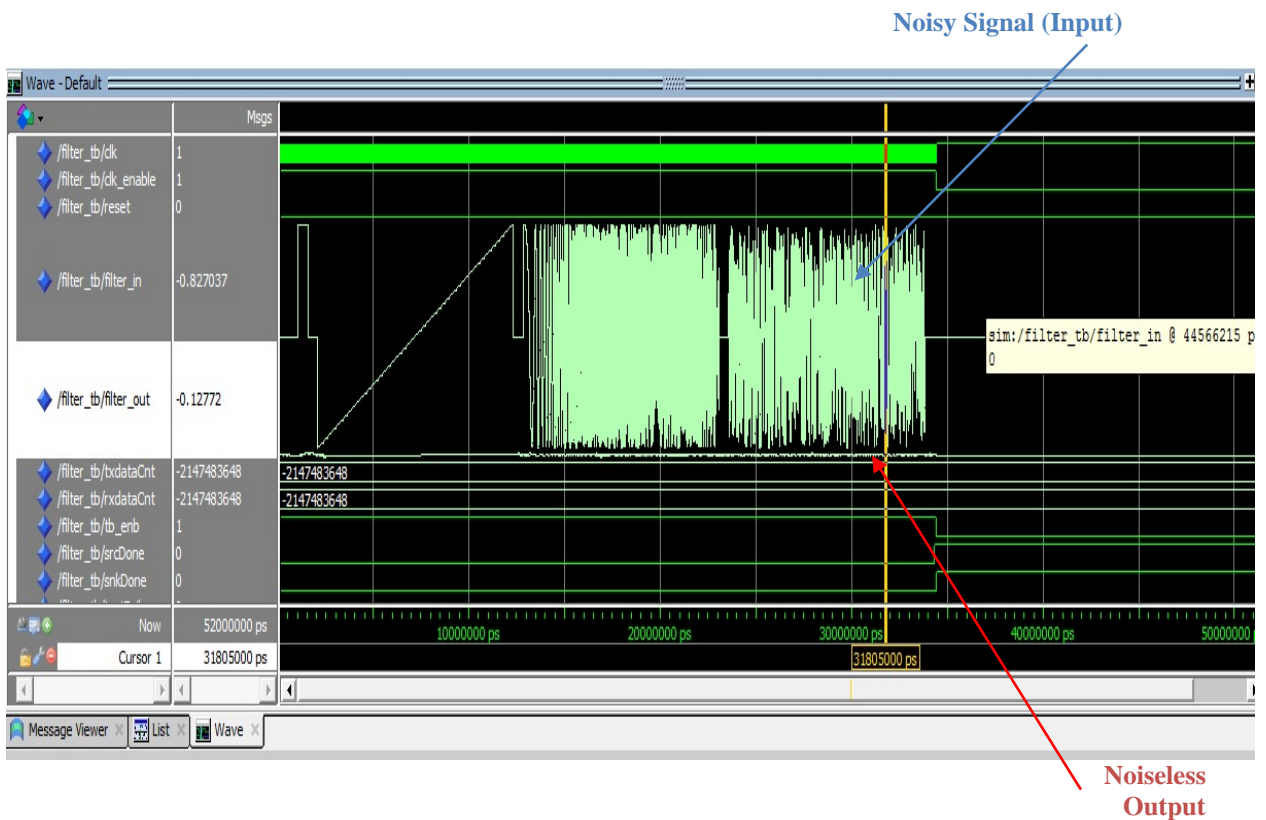


Figure 14: Timing Simulation of 50th order Low Pass Digital FIR Filter

The configured *ECG_top_level_module* on FPGA is firstly tested and verified using a function generator by giving some type of known analog signals such as sine, rectangular and saw tooth wave forms and their corresponding output plotted on oscilloscope. The Fig.15 shows plotted output on oscilloscope when we have given square wave as input from function generator.

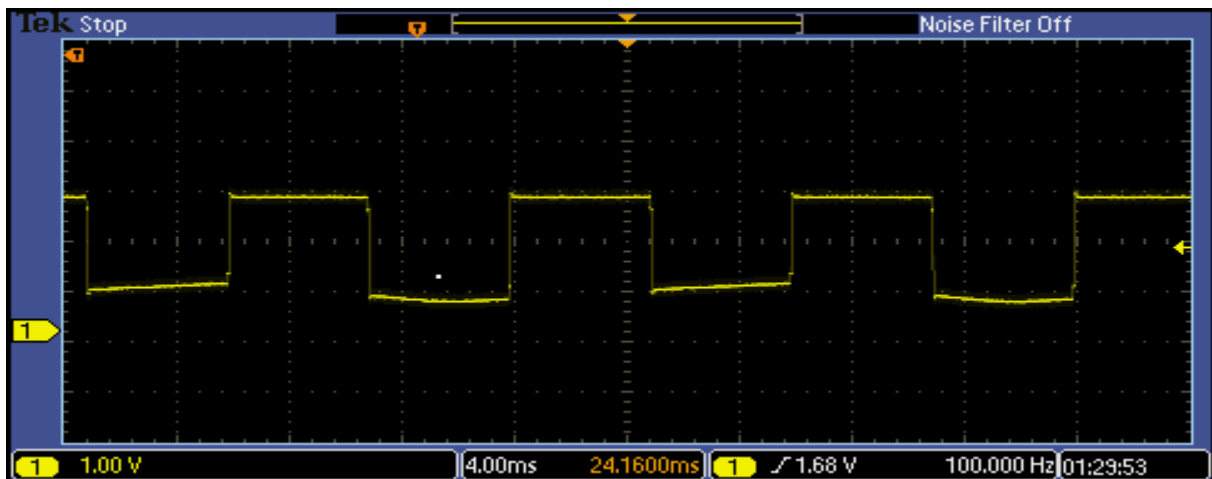


Figure 15: Plotted rectangular wave on oscilloscope

The implemented *ECG_top_level_module* on FPGA has been verified by giving a real time ECG signal from an ECG simulator via an analog capture circuit i.e. an instrumentation amplifier to the 6-pin ADC header (J7) on FPGA and the corresponding ECG output signal extracted from 6-pin DAC header (J5) port of FPGA kit. The Fig.16 shows the plotted ECG waveform on oscilloscope.



Figure 16: Plotted ECG Signal on Oscilloscope

4. CONCLUSION

Since Existing ECG monitoring systems are usually bulky in nature and normally situated in a hospital and can only be used within one hospital. Some Mobile ECG signal processor has been implemented but they are so expensive and there haven't any facility of reprogram ability and design change. In this research work, we have implemented a real-time Electrocardiogram (ECG) monitoring system completely based on Field Programmable Gate Arrays (FPGA). All the required modules for ECG signal processing are implemented on FPGA through VHDL programming language. The FPGA chip is taken as the central microprocessor for collection, filtering and transmission of real-time ECG signals. This research work will be very useful for designing of a real time signal processing, for fast implementation, low turnaround time, reduction of designing costs, and due to reprogrammable feature of FPGA there will be ease of design modification.

REFERENCES

- [1] Volnei A. Pedroni , “Circuit Design with VHDL”, *MIT Press Cambridge, Massachusetts London, England*, Edition -2004.
- [2] Yongming Yang , Xiaobo Huang , Xinghuo Yu, “Real-Time ECG Monitoring System Based on FPGA” in *proceedings of the 33rd Annual Conference of the IEEE Industrial Electronics Society (IECON)*, Nov. 5-8, 2007, Page(s): 2136 – 2140 ,Taipei, Taiwan , DOI: 10.1109/IECON.2007.4459886.
- [3] Xilinx,”Spartan-3E starter kit board user guide” Xilinx Ltd, p.69-82, 2008.
- [4] Hyperlink: http://www.xilinx.com/support/documentation/boards_and_kits/ug230.pdf
- [5] Desai, Vaibhav, "Electrocardiogram (ECG/EKG) using FPGA", *SAN JOSÉ STATE UNIVERSITY, Master's Projects*. Paper 238, 2012
- [6] Charuyuphan Chareonsak, Farook Sutta, Yu Wei, and Xiong Bing “Design of FPGA hardware for a real-time blind source separation of Fetal ECG signals” in *proceedings of 2004 IEEE International Workshop on Biomedical Circuits & Systems*. 1-3 Dec. 2004, Page(s): S2/4 - 13-16. DOI: 10.1109/BIOCAS.2004.1454135 .
- [7] Bruno Santos Pimentel Joao, Hiliirio de Avila Valgas Filho, Rodrigo Lacerda Campos, “FPGA Implementation of a DCT-Based Digital Electrocardiographic Signal Compression Device” in *proceedings of IEEE 14th Symposium on Integrated Circuits and Systems Design, Universidade Federal de Minas Gerais*, 0-7695-1333-6/01 2001, Page(s): 44 – 49, DOI: 10.1109/SBCCI.2001.953002 .
- [8] Fábio Fabian Daitx, Vagner S. Rosa , Eduardo Costa and Paulo Flores “VHDL Generation of Optimized FIR Filters”, in *proceedings of 2008 IEEE International Conference on Signals, Circuits and Systems*, 7-9 Nov. 2008, Page(s): 1 – 5, DOI: 10.1109/ICSCS.2008.4746944 .
- [9] Chia-Ching Chou, Wai-Chi Fang, Hsiang-Cheh Huang , “A Novel Wireless Biomedical Monitoring System with Dedicated FPGA-based ECG Processor” in *proceedings of 2012 IEEE 16th International Symposium*, 4-6 June 2012, Page(s): 1 – 4, DOI: 10.1109/ISCE.2012.6241744 .
- [10] Roger Woods Queen’s University, Belfast, UK John McAllister Queen’s University, Belfast, UK Gaye Lightbody University of Ulster, UK Ying Yi University of Edinburgh, UK, “FPGA-based Implementation of Signal Processing Systems” , A John Wiley and Sons, Ltd., Publication, first edition - 2008. ISBN 978-0-470-03009-7.
- [11] Linear Technology 6912-1, Dual programmable gain Amplifier Data sheet. Linear Dual channel on board ADC (LTC1407-1) data sheet.
- [12] MAX1297 ADC and AD620 instrumentation amplifier data sheet.
<http://www.datasheetcatalog.org/datasheet/maxim/MAX1295-MAX1297.pdf>
<http://m.analog.com/mt/www.analog.com/en/specialtyamplifiers/instrumentationamplifiers/ad620/products/product.html>

