

Optimized Reconfigurable ASIP of FIR Filter using FPGA

Amruta S. Vhatkar^{A*} and P.C.Bhaskar^A

^ADepartment of Electronics Technology, Shivaji University,, Kolhapur, M.S. (India)

Accepted 10 May 2014, Available online 01 June2014, Vol.4, No.3 (June 2014)

Abstract

Reconfigure ability denotes reconfigurable computing capabilities of a system, so that its behavior can be changed by reconfiguration with improvements in capacity & performance. Application specific instruction set processor is bridge between ASIC & DSP. This research works based on Xilinx System Generator which support for runtime reconfiguration which further controls ASIP.FIR (Finite Impulse Response) filter is configured for different taps & methods.1-D signal i.e., ECG (Electrocardiogram) is taken for filtration, which having Gaussian noise also known as EMG noise. A generalized instrumentation signal which needs Filters are also taken for computation. This proposed work is implemented on Spartan xc3s500e-4fg320 board and pin N17, H18 are configured as reconfigurable switch which will give facility for applying different type of filter at run time. The results of different filter design techniques are analyzed using Matlab simulation tool and performance is evaluated based on signal to noise ratio.

Keywords: ECG, FIR filters, ASIP, signals to noise ratio (SNR), FPGA, Reconfigurable.

1. Introduction

In today's scenario computer and electronics world need a reconfiguration facility because of cost effective approach. Generally we are used to two different ways of performing computation one is hardware computations which normally having e-waste after it gets damage and second is software based computations which provide simulations result so programmer will go for appropriate hardware implementation. Highly optimized resources are given by application-specific integrated circuits (ASICs), for quickly performing critical sections i.e., tasks but it is permanently configured for one application. One Time Programmable (O.T.P) terminology is best suitable for ASICs. Software computing provides flexibility to change applications on a fly and perform a variety tasks, but when hardware implementation of applications, which allows to performs different tasks as user need is not possible by ASICs. Revolutionary devices such as Field-programmable gate arrays (FPGAs) blend the benefits of software and hardware. An FPGA based system can be reprogrammed many times because computations are programmed into the chip, but they are not permanently frozen at a time of manufacturing process.

The principal difference when FPGAs compared to ordinary microprocessors is the ability to make substantial changes to the data path itself in addition to the control flow (Matthias Alles *et al*, 2010). Three forms of reconfiguration can be done:

1. Processor is reconfigured at load time (static reconfiguration)

2. The algorithm flow is changed at pre-determined points or phases of a program execution (quasi-static reconfiguration)

3. Reconfiguration can occur at arbitrary time during program execution.

Application specific instruction processors (ASIPs) are allow for programming with flexibility for a application oriented tasks under speed, accuracy and power constraints (Paolo Meloni *et al*, 2012). This leads to several works to develop the tools for analyzing the given application and determine the optimal instruction set which maximizes the performance. Whereas ASIC is specially designed for one behavior, it is difficult to accommodate any changes at a later design stage. In contrast, programmable processors can be easily adapted to different applications by changing only the programs. It is the reason that ASIPs are widely accepted in numerous systems. For designing ASIP there are two approaches (Rashid Muhammad) first one is design from scratch, where new instructions are designed for target application and second is customizing instruction set (i.e., may instructions are grouped together & form a valid block set) for existing predefined configurable processor.

The electrocardiogram (ECG) is a 1-D signal as time-varies signal reflects by the ionic current flow which causes two make heartbeat means contraction and relaxations of heart muscles. ECG is obtained by recording the potential difference between two electrodes placed on pulsating area of the skin (Dr. Ana-Maria Zagrean).Biomedical signals like ECG or EEG are very critical signal processes. A feature extraction with précised computation is again a challenging task. Physician makes

*Corresponding author: Amruta S. Vhatkar

diagnosis of patient by visual assessment so it is very necessary to have a noise free signal track.

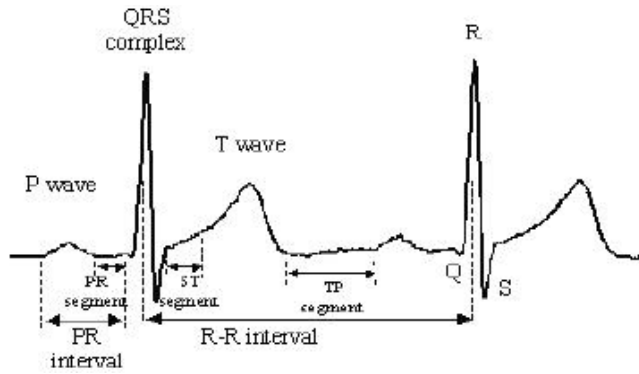


Figure1: Typical ECG Signal.

The Figure1 shows an example of normal ECG trace, which consists of P wave, QRS complex and T wave. . ECG signal generally corrupted by various kinds of noise like Power line interference, Electrode Contact noise, Motion Artifact, Muscle contraction, Base line wanders and many more. Also instrumentation applications need a noise free signal so parameters get extracted (Yun-Chi Yeha,c et al 2008).

In this paper, the main focus is to performance evaluation of system which is providing reconfigure ability. By taking 1-D signals and applying different FIR filters on them performance of system is evaluated.

2. FPGA Implementation

For implementation of reconfigurable FIR filter on FPGA, the distributed arithmetic method is chosen. The system generator tool enables the use of the Math Works model based design environment Simulink for FPGA design. So the filter design model is developed with the use of Xilinx System Generator (XSG) tool as follows (Matthewownby et al, 2002).

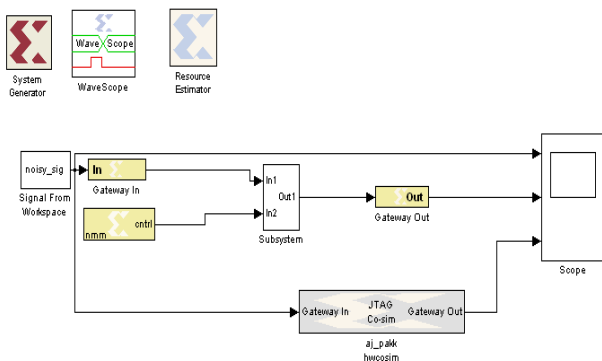


Figure 2: FPGA implementation of Reconfigurable FIR filters design with System Generator Tool.

Above figure having subsystem block in which filter is configured for performing different tasks. Cntrl is a system generated block by hardware co-simulation, which is key element in design of system and it allow to run time reconfigure with help of hardware switches.

RTL Schematics

The proposed design RTL schematic is shown in Figure 3.

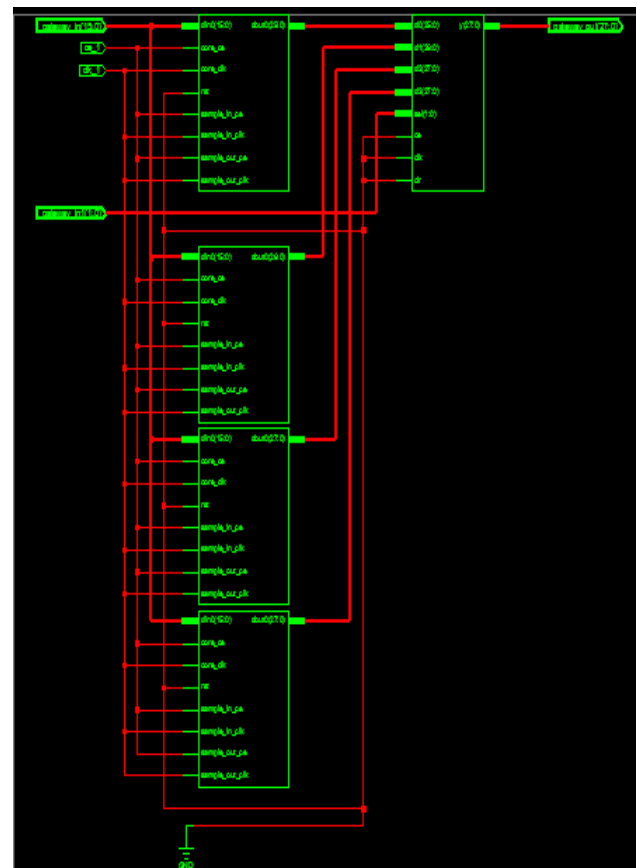


Figure 3: RTL schematic for reconfigurable FIR filter design.

3. Evaluation Parameters

There are two important parameters, to check the filtration of signal.

A. Power Spectral Density (PSD)

The Period gram power spectrum estimate represents the distribution of the signal power over frequency. From the spectrum the frequency content of the signal can be estimated directly. The Period gram is an estimate of the PSD of the signal defined in a sequence form as $[X_1 \dots X_n]$. Period gram uses N point FFT mechanism to compute the power spectral density.

B. Signal to Noise Ratio (SNR)

SNR is a parameter used to quantify and compare the performance of Filtering system and also determine the noise level from signal. The expression used to calculate SNR is as follows:

$$SNR = 10 \log_{10} (\text{variance } (S_o) / \text{variance } (S_o - S_f))$$

Where, S_o = original Signal S_f = filtered signal.

4. Results & Discussions

MATLAB Simulation results of the designed filters to filtered 1-D signal with Gaussian noises are obtained as follows:

Figure 4 shows filtration of ECG signal which having SNR -5.9184 at input side, after applying 40 taps FIR filter of Kaiser window it filters out noise and SNR is -0.3236 at output side. In figure 4 different taps are applied at run time and for last samplings 40 taps filter is applied by configuring N17 and H18 switch (i.e., 11)

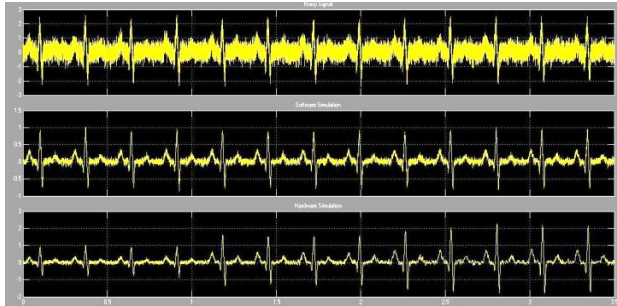


Figure 4: Output waveforms of software and hardware simulation for different taps on ECG.

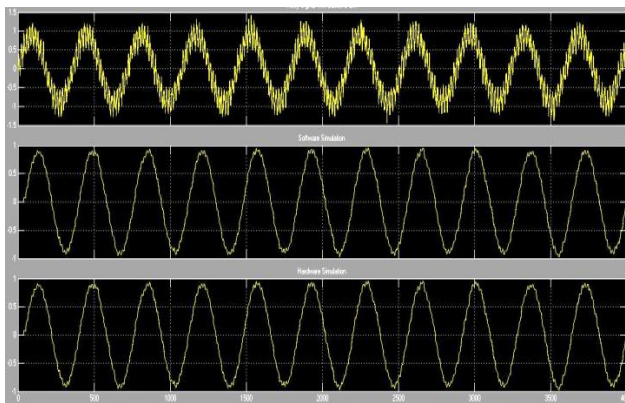


Figure 5: Output waveforms of software and hardware simulation.

Table 1 Simulation results of different filter design techniques.

Sr. No	Filters Type	Order	SNR at input	SNR at output
1.	Rectangle	10	10.18	2.75
2.	Kaiser	10	10.18	2.68
3.	Equiripple	10	10.2	-0.1
4.	Kaiser	40	10.2	-0.36
5.	equiripple	20	10.48	-0.32
6.	kaiser	20	10.48	0.55
7.	hamming	30	10.48	0.8
8.	equiripple	30	10.48	0.2
9.	blackman	30	10.48	0.6

Again general signals which are used in instrumentation just like balancing shaft or multi functioning op-amps which gives a linear noise variations (Seema Rana *et al*, 2007). Assume in shaft balancing machine because of

linear vibration output signal became noisy then it is necessary to filter out it. For checking reconfigure ability of proposed system signals are generated with Gaussian noise and by different configuration of filter output signals are evaluated.

Table 2 Resources estimation with Xilinx FPGA devices (Alfredo Rosado-Muñoz et al, 2011)

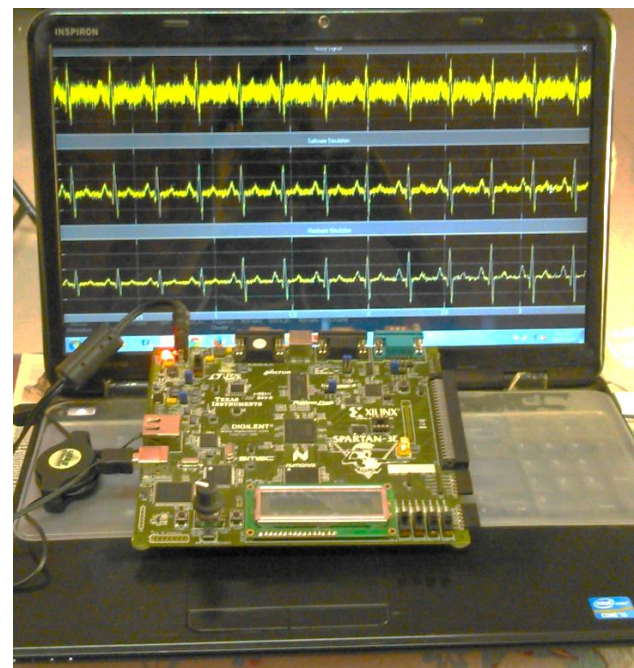
FPGA Chip	xc3s500e -4fg320	xc4vfx12 -12ff668
Slices used	4415(94%)	7998(73%)
4 Input LUTs	8099(86%)	5233(47%)
Mini Period	5.295 ns	2.595ns
Max Frequency	188.857 MHz	385.349 MHz

aj_pakk_cw Project Status (04/30/2014 - 10:32:50)			
Project File:	aj_pakk_cw.ise	Current State:	Synthesized
Module Name:	aj_pakk_cw	• Errors:	No Errors
Target Device:	xc3s500e-4fg320	• Warnings:	2381 Warnings
Product Version:	ISE 10.1 - WebPACK	• Routing Results:	
Design Goal:	Balanced	• Timing Constraints:	
Design Strategy:	Xilinx Default (unlocked)	• Final Timing Score:	

aj_pakk_cw Partition Summary			
No partition information was found.			

Device Utilization Summary (estimated values)				
Logic Utilization	Used	Available	Utilization	
Number of Slices	4415	4696	94%	
Number of Slice Flip Flops	8099	9312	86%	
Number of 4 input LUTs	5975	9312	64%	
Number of bonded I/Os	35	232	15%	
Number of GCLKs	1	24	4%	

Fig. 5 Resource utilization report of reconfigurable FIR filters design.



As per the above results, FPGA implementation of reconfigurable FIR filter with different configuration technique requires less hardware resources, less time (5.295 ns) and less power consumption (91mW) than any other design methods.

Conclusion

In this paper we have successfully implement a reconfigurable FIR filtering system and performance

evaluation is done for different filtering techniques and taping. This research work is carried out in Xilinx System Generator which is excellent tool to design filters. The generation of VHDL descriptions from unified modeling language of the filters which reduce dramatically time which will require for generating a solution. For different tappings of Kaiser window we achieve run time reconfigure ability and proposed methodology is practically implemented on Xilinx starter kit *xc3s500e-4fg320*.

References

- Hanan M Hassan, Karim Mohammed and Ahmed F Shalash,(2012),Implementation of a reconfigurable ASIP for high throughput low power DFT/DCT/FIR engine *EURASIP Journal on Embedded Systems* , A Springer Open Journal.
- Matthias Alles, Timo Vogt , Christian Brehm, Norbert, (2010)FlexiChaP adynamically reconfigurable ASIP for channel decoding for future mobile systems, *Dynamically reconfigurable system* ,Springer .
- Paolo Meloni, Sebastiano Pomata, Giuseppe Tuveri, Simone Secchi, Luigi Raffo and Menno Lindwer (2012),Enabling Fast ASIP Design Space Exploration: An FPGA-Based Runtime Reconfigurable Prototyper , *Hindawi Publishing Corporation, International Journal of Reconfigurable Computing*, Article ID 580584
- Rashid Muhammad, Ludovic Aprville, and Renaud Pacalet, Evaluation of ASIPs Design with LISATek *System-on-Chip laboratory LabSoC*, GET/ENST Sophia-Antipolis, France
- Ana-Maria Zagrean, Practical Notes of the Physiology Department II , *Carol Davila Univ. of Medicine and Pharmacy, Bucharest* , Coordinator, 2nd Year English Module.
- Yun-Chi Yeha,c, Wen-June Wanga (2008),QRS complexes detection for ECG signal: The Difference Operation Method, *Elsevier Ireland Ltd*.
- Matthewownby, WagdyH .Mahmoud (2002),A Design Methodology for Implementation DSP with Xilinx System Generator for Matlab, IEEE.
- Seema Rana, Kirat Pal (2007), A Multifunction Filter for Instrumentation Applications, *Journal of Physical Sciences*,vol. 11.
- Alfredo Rosado-Muñoz, Manuel Bataller-Mompeán, Emilio Soria-Olivas, Claudio Scarante, and Juan F. Guerrero-Martínez (2011), FPGA Implementation of an adaptive filter robust to impulsive noise, two approaches IEEE transactions on industrial electronics,vol.58
- B.S.Mahant, L.P.Thakare (2013), Implementation of 64-Point Reconfigurable FFT Processor for ASIP Architecture *International Journal of Advanced Computer Research*, volume-3