FPGA Based Design & Implementation of Low Power FIR Filter for ECG Signal Processing

Rupali Madhukar Narsale, Dhanashri Gawali and Amit Kulkarni

Abstract— **WSN is composed of a large number of sensor nodes with multi-hop networking capability that are widely deployed for a wide variety of applications such as health monitoring systems, environment monitoring, interactive user interfaces, etc. Sensor systems must utilize the minimal possible energy while operating over a wide range of operating scenarios. Signal processing is an essential component of wireless sensor network node. One of the most widely used operations in this is finite-impulse response (FIR) filtering. Multiplication is at the core of FIR filter and saving power at the multiplication level can significantly impact the lifetime of a sensor node.**

In this, we propose design of low power FIR filter and implementations of the same on FPGA. MATLAB is used for an Equiripple or Remez Exchange (Parks-McClellan) technique of FIR filter design. A notch FIR filter is designed for the removal of power line interference in ECG signal which is a very important step in the pre-processing stage of ECG signal. Direct-form approach in realizing a digital filter is considered. This approach gives a better performance than the common filter structures in terms of speed of operation, cost, and power consumption in real-time. The minimum power achieved is 0.073w in FIR filter based on DA to 626 taps, 8 bits inputs and 9 bits coefficients. The proposed FIR filter has been designed using VHDL, simulated, synthesized and implemented using Xilinx ISE Artix-7 series FPGA and power is analyzed using Xilinx XPower analyzer.

***Index Terms*— ECG, Equiripple FIR Filter, Distributed Arithmetic (DA), FPGA.**

I. INTRODUCTION

Wireless sensor networks (WSN) have got researchers attention in recent years because those can be used in wide range of applications. Wireless Sensor Network could contain hundreds of sensors that collect and some cases pre-process data before it is send to central node for final processing. In the most cases sensors are deployed to remote location without capability to replace battery. This means that one of the key elements for distributed sensors is long lifetime covering both reliability and energy efficiency because the battery limits lifetime of the sensors. Due to the large number of nodes that may be deployed and the long required system lifetimes, replacing the battery is not an option. Sensor systems must utilize the minimal possible energy while operating over a wide range of operating scenarios. A long node lifetime under diverse operating conditions demands power-aware system design [1]. Design methodology of LSIs which contains digital wireless communication systems addresses to low power and low energy system implementation, because of the growing need for longer battery lifetime and the problem of power supply. Signal processing in wireless sensor network has a vast range of applications. Finite Impulse Response filtering (FIR), Infinite Impulse Response filtering (IIR), and Kalman Filtering find applications in object tracking, environmental monitoring and many other applications [2].

In this paper, we focus on a digital FIR (Finite impulse response) filter, because the digital filter has a large power consuming components, such as adders and multipliers. Traditionally, direct implementation of a K-tap FIR filter requires K multiply-and-accumulate (MAC) blocks, which are expensive to implement in FPGA due to logic complexity and resource usage [3]. To resolve this issue, we present distributed arithmetic (DA), which is a multiplier-less architecture. Distributed arithmetic is an important algorithm for DSP applications. It is based on a bit level rearrangement of the MAC "multiply accumulate" operation to replace it with set of addition and shifting operations. Our motivation for using DA is its extreme computational efficiency. The advantages are best exploited in circuit design, but off-the-shelf hardware often can be configured effectively to perform DA. By careful design one may reduce the total gate counting a signal processing arithmetic unit by a number seldom smaller than 50 percent and often as large as 80 percent [4].

Digital filters are typically used to modify or alter the attributes of a signal in the time or frequency domain. The most common digital filter is the linear time-invariant (LTI) filter. An LTI interacts with its input signal through a process called linear convolution, denoted by $y = f * x$ where f is the filter's impulse response, x is the input signal, and y is the convolved output. The linear convolution process is formally defined by:

$$y[n] = x[n] * f[n] = \sum_{k=0} x[n]f[n-k] \quad = \sum_{k=0} f[k]x[n-k]. \qquad (1)$$

LTI digital filters are generally classified as being finite impulse response (i.e., FIR), or infinite impulse response (i.e., IIR). As the name implies, an FIR filter consists of a finite number of sample values, reducing the above convolution sum to a finite sum per output sample instant. An FIR with constant coefficients is an LTI digital filter. The output of an FIR of order or length L, to an input time-series x[n], is given by a finite version of the convolution sum given in (1), namely:

Manuscript received May, 2014.

***Rupali Madhukar Narsale**, Electronics Dept., MIT Academy of Engineering, Pune, Pune, India, Phone/ Mobile No.9158141492.*

***Dhanashri Gawali**, E&TC Dept., NBN Sinhgad School of Engineering Pune, India,*

***Amit Kulkarni**, Electronics Dept., MIT Academy of Engineering, Pune, India, Phone/ Mobile No.9373222881,*

ISSN: 2278 – 7798

$$y[n] = x[n] * f[n] = \sum_{k=0}^{L-1} f[k]x[n-k]. \quad (2)$$

Where, f [0] ≠ 0 through f [L-1] ≠ 0 are the filter's L coefficients. They also correspond to the FIR's impulse response. For LTI systems it is sometimes more convenient to express in the z-domain with

$$Y(z) = F(z)X(z) \quad (3)$$

Where, F (z) is the FIR's transfer function defined in the z-domain by

$$F(z) = \sum_{k=0} f[z]z^{-k} \quad (4)$$

The L^{th} -order LTI FIR filter is graphically interpreted in Fig.2. It can be seen to consist of a collection of a "tapped delay line," adders, and multipliers. One of the operands presented to each multiplier is an FIR coefficient, often referred to as a "tap weight" for obvious reasons. Historically, the FIR filter is also known by the name "transversal filter," suggesting its "tapped delay line" structure [5],[6].

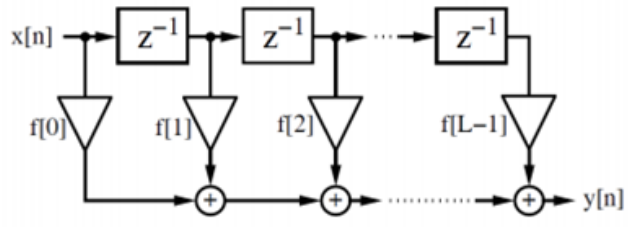

Fig. 1: FIR filter in the transposed structure

Power consumption can be reduced by a combination of several techniques. Pipelining and parallel processing can be used to reduce power consumption by reducing supply voltage. Power consumption can be reduced by reducing effective capacitance which can be achieved by reducing the number of gates or by algorithmic strength reduction where the number of operations in an algorithm is reduced. Power can also be reduced by reducing memory access. The single most effective means to power-consumption reduction is clock gating where all functional units which need not compute any useful outputs are switched off by using gated clocks. Use of multiple-supply voltages and a simultaneous reduction of threshold and supply voltages are also effective in reducing power consumption. Most power-reduction approaches apply to dedicated, Programmable or Field Programmable Gate Array (FPGA) systems in a dual manner. Optimizations of the speed and power consumption of digital filters can be achieved by using dedicated operators instead of general ones whenever possible. Multiply and Accumulate (MAC) is an important unit of DSP systems. It decides the power consumption and speed of operation of DSP systems [7].

The techniques which are used to achieve low power consumption in VLSI-DSP applications span a wide range, from algorithm and architectural levels to logic, circuits and device levels. Power optimization approaches at the high level are significant since research results indicate that higher levels of abstraction have greater potential for power reductions. The most power savings came from the two extremes, system level & gate level power optimization [8].

II. SOFTWARE IMPLEMENTATION

Digital filters play a crucial role in digital signal processing, e.g., biomedical signal processing. There are two main types of digital filters: finite impulse response (FIR) filters and infinite impulse response (IIR) filters. The transfer function of FIR filter is,

$$H(z) = \sum_{k=0}^{N} h(k)z^{-k} \quad (5)$$

Where, *h (k),* k=0, 1, 2, 3…, N are the filter coefficients [9]. There are three well known methods for FIR filter design

- The window method
- The Frequency sampling technique
- Optimal filter design method

The simplest technique is known as "Windowed" filters. This technique is based on designing a filter using well-known frequency domain transition functions called "windows". The use of windows often involves a choice of the lesser of two evils. Some windows, such as the Rectangular, yield fast roll-off in the frequency domain, but have limited attenuation in the stop-band along with poor group delay characteristics. Other windows like the Blackman, have better stop-band attenuation and group delay, but have a wide transition-band (the band-width between the corner frequency and the frequency attenuation floor). Windowed filters are easy to use, are scalable. An Equiripple or Remez Exchange (Parks-McClellan) design technique provides an alternative to windowing by allowing the designer to achieve the desired frequency response with the fewest number of coefficients. This is achieved by an iterative process of comparing a selected coefficient set to the actual frequency response specified until the solution is obtained that requires the fewest number of coefficients [10].

Present paper deals with design and development of digital FIR Equiripple filter. The filtering technique suggested is simply FIR Type-1 filter. The application identified to demonstrate the idea is ECG signal processing. The basic ECG has the frequency range from 0.5Hz to 100Hz. This ECG gets corrupted due to different kinds of the artifacts. The different types of artifacts are Power line interference, motion artifacts, base line drift and instrumental noise. Due to these types of the artifacts ECG gets corrupted and correct information is not transferred to the cardiac specialist. The care must be taken to nullify the artifacts to avoid wrong diagnosis. Certain type of the noise may be filtered directly by time domain filters using signal processing techniques or digital filters. Different researchers are working on noise reduction in the ECG signal [11].

In the present work notch filter is designed. An ECG signal containing power line noise is applied to this filter. An ECG signal from the database is loaded into the MATLAB®. Various signals such as original ECG signal and ECG signal added with power line noise are examined in time domain and frequency domain. Suitable design parameters for different digital filters are chosen based on the frequency of

ISSN: 2278 – 7798

noise. Thereafter, the ECG signal with noise is passed through the designed digital filter. Finally, the filtered ECG signal is investigated again in time domain and frequency domain. The quantitative properties of the designed and implemented filters are investigated by comparing their spectra before and after filtration [9].

III. HARDWARE IMPLEMENTATION

The proposed FIR architecture is shown in Fig.1. As it can be seen, it consist of functional units (FU), adder stage and only one shift and accumulate stage without any multipliers. We have implemented a 676-tap filter and included the multiplexors after 26 delay stages in the FIR to show the added benefit of this FIR to be configured as a convolution filter often used in physiological monitoring applications. Assuming the tap coefficient to be 8-bit wide, a standard *M*-tap transverse FIR filter equation 2 can be modified to:

$$y(n) = \sum_{m=0}^{M-1} h(m)x(n-m) = \sum_{m=0}^{M-1} [\sum_{k=0}^{7} x(n-m)h_k(m)2^k] \qquad (6)$$

The square term in equation 6 can be implemented by using shift registers and adders. The term h_k is a one bit data `0' or `1', and is the weight of the coefficient. The resulting architecture based on equation 6 contains same *M* number of shift- add-accumulate blocks as multipliers in conventional FIR (Fig.1). This can be simplified further to equation 7 for area critical implementation resulting in the following:

$$y(n) = \sum_{k=0}^{7} [\sum_{m=0}^{M-1} x(n-m)h_k(m)]2^k \qquad (7)$$

This results in area efficient architecture because the term inside the square bracket reduces from 16-bits to 8-bits. For a M-tap filter, a transverse filter with multipliers will contain 2*M shift registers, M multipliers and (M -1) adders, while the proposed filter will contain 8 *M AND gates, 16 *M shift registers and (M-1) adders. As shown in Fig.2, the 676-tap filter consists of 676 functional units, an adder stage and one add-accumulate block.

As shown in Fig.2 the functional unit (FU) is the core of the architecture and is defined in the square bracket term in equation 7. Each FU is capable of one partial product. In every clock cycle, one 8-bit partial product is calculated. So a complete 8-bit sample would be delivered once in every eight clock cycle. The 676 functional unit outputs 5408-bits of partial product every clock cycle this is one eighth of the sample. The partial product of each of the functional unit is fed to the adder stages that sum up the 676 partial products. The adder stages are 16-bit wide instead of 32-bit, which again reduces area. Coefficient bits are shifted left in each clock cycle so that the partial product is ANDed from most significant bit to least significant bit as shown in Fig.3. To avoid overflow, a 32-bit wider adder structure (with 16-bit half adder and a 16-bit full adder) in the shift, add-accumulate stage is implemented. The left shift in the accumulator and the add takes care of the weight associated with the left shift of the coefficient data. A shift operation is done in the accumulator by tying the least significant bit to `0' to adjust the weight of the coefficients. This process is continued 8 times till one filtered sample or convolved data is obtained. The new data is loaded after every eight clock cycles. A simple 8-bit shift register is implemented to generate the control signal once every 8 clock cycles for loading or shifting of the input data. The critical path (or longest path) of the design is the dotted line marked in the Fig.3 which is clearly shorter than the transverse structure. We assume that the data input is done directly and completely avoids any buffering stages in the FIR.

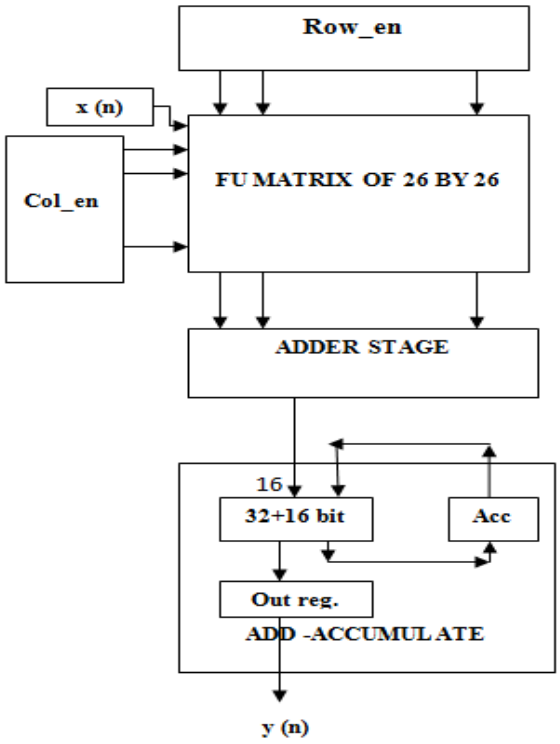

Fig.2: FIR architecture for 676 tap filter

For understanding purpose proposed architecture for 9 tap filter is shown in fig.3 [12].

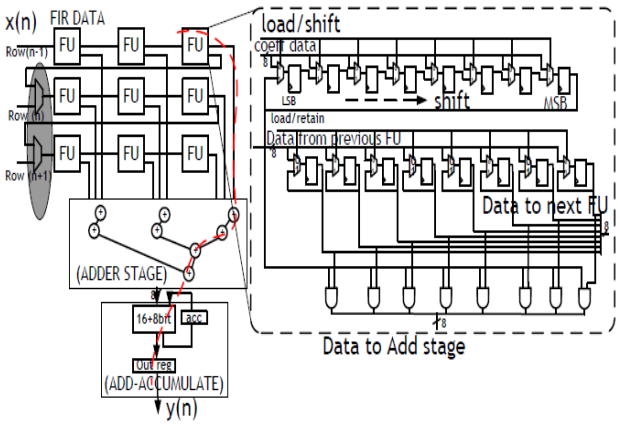

Fig.3: FIR architecture for 9 tap filter.

ISSN: 2278 – 7798

IV. TESTING & RESULTS

We have been designed the filter first in MATLAB in order to check the feasibility of the specifications in MATLAB. We got the desired results in MATLAB. Then the filter with the desired specifications has been designed in VHDL and simulated in Modelsim software and after that burned on FPGA kit.

A. Implementation of The Equiripple Notch Filter:

Since the goal of this paper is to design and implement digital filter for the processing of noises and to evaluate the performance of the filters, each noise signal is first modelled and simulated. Power line interference consists of 50/60 Hz and its harmonics which can be modelled as sinusoids and a combination of sinusoids with amplitude up to 50 % of the peak- to-peak of ECG amplitude. Fifty Hertz power line noise is simulated using the MATLAB. The frequency of power line is 50 Hz.

$$N(t) = A \times \sin(2 \times \pi \times f \times t) \qquad (8)$$

Where, N (t) is the power line noise, A is the amplitude and f is the frequency of power line. ECG signal from database and ECG signal superimposed with modelled power line interference in time domain are plotted as shown in Fig.5. Also, the Fast Fourier Transform (FFT) of the signals is illustrated in Fig.5. It can be clearly seen that the nature of power line interference, 50 Hz sinusoid is obvious in the power spectrum. From the information gained by plotting the FFT of ECG signals in the frequency domain, the design and implementation of digital filter are proceeded to remove and reduce the noises [9].

Digital notch filter is mostly the first choice for rejecting the specific frequency of the signal. The FIR notch filter is designed (45-55Hz) to suppress the 50 Hz power line noise. The sampling frequency is 500 Hz.

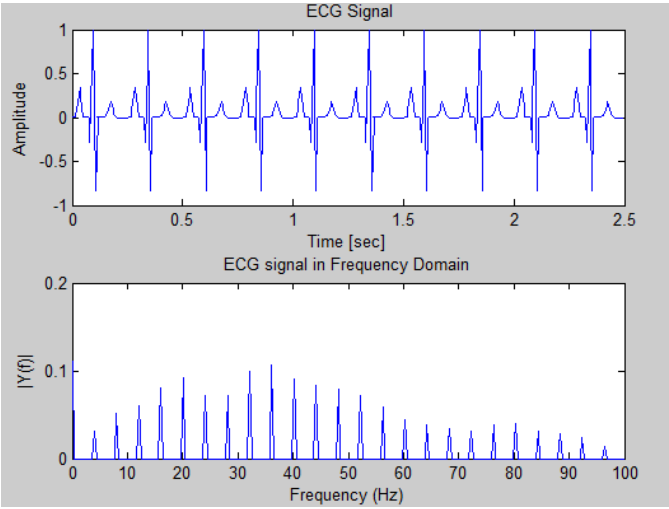

Fig. 4: ECG signal in Time & Frequency domain

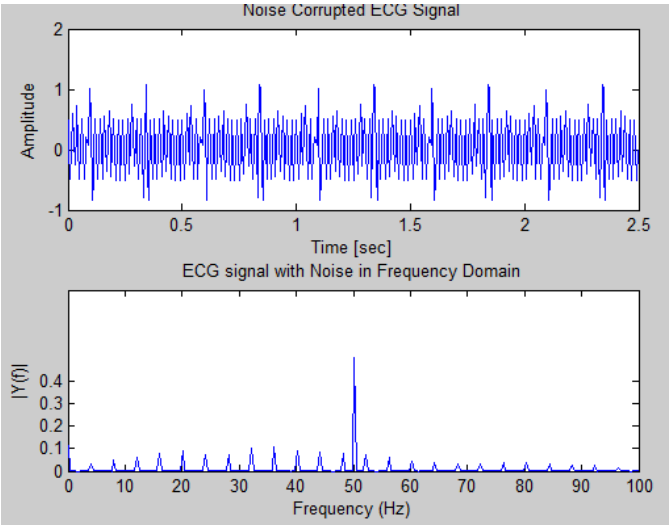

Fig. 5: ECG Signal corrupted with noise before filter in Time & Frequency domain

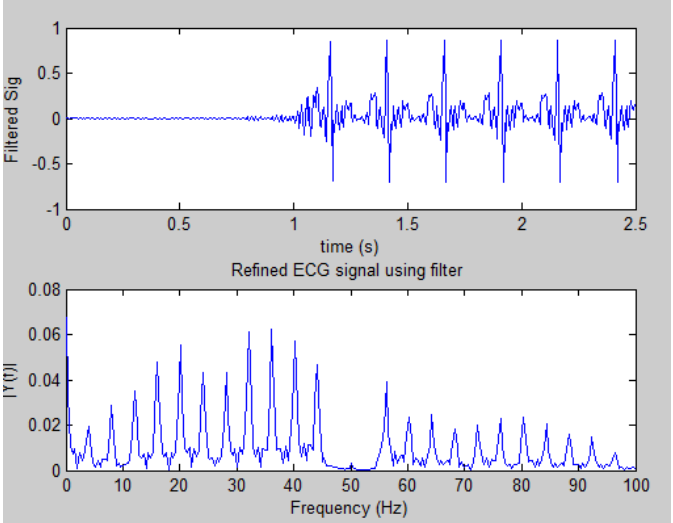

Fig. 6: ECG Signal corrupted with noise after filter in Time & Frequency domain

B. Hardware Implementation of FIR Filter

Field Programmable Gate Arrays (FPGAs) can be reprogrammed as many times in order to achieve the desired result. The major design benefit in this lies in the ability to test designs that might work. Prior to the development of the FPGA, the fabrication process can be quite expensive and very time consuming. The use of FPGAs in the design process allows the more design flexibility, and reducing a cost and developing time. If the design fails after being tested on a FPGA, the designer can simply rework the design and download it again to the FPGA. Use of an FPGA would thus eliminate the loss in development time caused by a faulty initial design, as well as giving the designer knowledge of whether or not the design works.

The VHDL code of the digital FIR filter has been simulated in Modelsim and the following waveforms obtained. Now this VHDL code is used to generate the circuit using Xilinx synthesis tool for notch filter design and main circuit block is shown in fig.7. The minimum power achieved is 0.073w in FIR filter based on DA to 626 taps, 8 bits inputs and 9 bits coefficients.

ISSN: 2278 – 7798

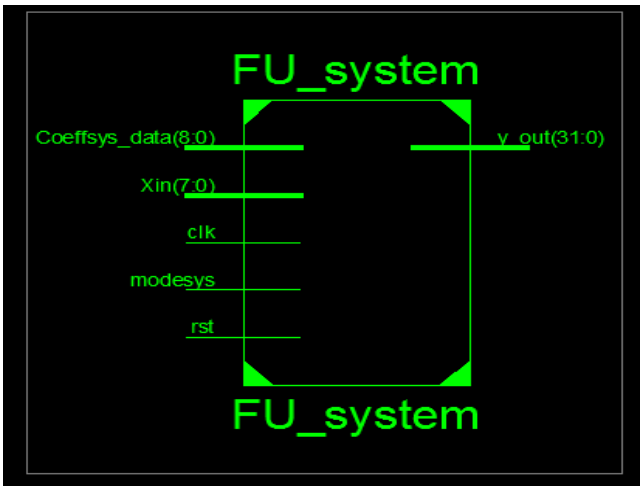

Fig.7.Top level RTL Schematic

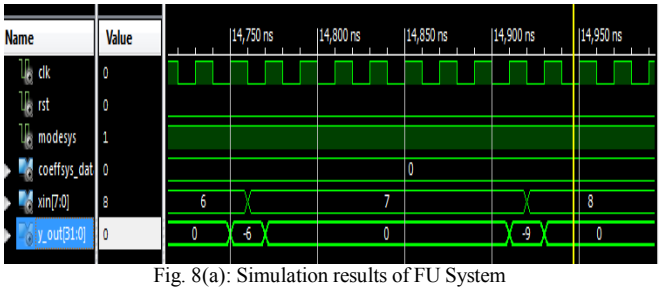

Fig. 8(a): Simulation results of FU System

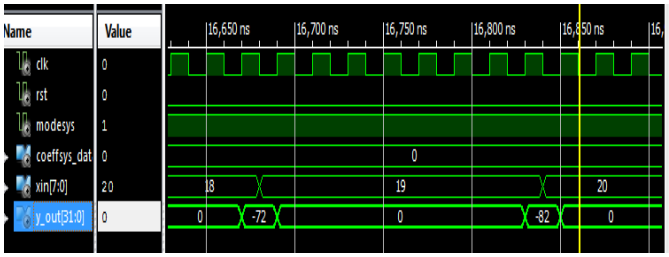

Fig. 8(b): Simulation results of FU System

Device Utilization Summary (estimated values)			
Logic Utilization	Used	Available	Utilization
Number of Slice Registers	15639	269200	5%
Number of Slice LUTs	81805	134600	60%
Number of fully used LUT-FF pairs	8152	89292	9%
Number of bonded IOBs	52	285	18%
Number of BUFG/BUFGCTRLs	2	32	6%

Fig.9.Device utilization summary

V. CONCLUSION

The present work introduces the digital filtering method to cope with the noise artifacts in the ECG signal. FIR filters based on Equiripple design has been designed and implemented. The Equirriple notch FIR filters are applied on the ECG signal. The results of the implementation show that filter removed the noise specifically meant for it to filter.

The implementation of DA architectures for FIR filter has been designed using VHDL, simulated, synthesized and implemented using Xilinx ISE Artix-7 series FPGA. The design has been verified using Modelsim with VHDL code and power consumption is analyzed using Xilinx software. The results of software implementation compared with hardware implementation.

ACKNOWLEDGMENT

We would like to thank Prof. P. V. Suryawanshi (E&TC department, MITAOE), for providing guidelines on design aspects of FIR Filter using MATLAB. Also we thank Electronics department, MITAOE for providing their support and valuable inputs which helped improve our work.

REFERENCES

[1] Rex Min, Manish Bhardwaj, Seong-Hwan Cho,Eugene Shih, Amit Sinha, Alice Wang, Anantha Chandrakasan, "Low-Power Wireless Sensor Networks",*IEEE*, pp. 205-210, Jan. 2001.

[2] Ahmed Abdelgawad, Sherine Abdelhak, Soumik Ghosh and Magdy Bayoumi, "A Low-Power Multiplication Algorithm for Signal Processing in Wireless Sensor Networks", *IEEE*, pp. 695-698, 2009.

[3] M. Keerthi , Vasujadevi Midasala, S Nagakishore Bhavanam, Jeevan Reddy K,"FPGA Implementation Of Distributed Arithmetic For FIR Filter", *International Journal of Engineering Research & Technology (IJERT)*, Vol. 1, Issue 9, pp.1-8, Nov.2012.

[4] Stanley A. White, "Appliction of Distributed Arithmatic to Digital Signal Processing: A Tutorial Review", *IEEE ASSP Magazine*, pp. 4-19, July-1989.

[5] Bahram Rashidi, Bahman Rashidi, Majid Pourormazd," Design and Implementation of Low Power Digital FIR Filter based on low power multipliers and adders on xilinx FPGA", *IEEE*, pp. 18-22, 2011.

[6] A. Renuka Narasimha1, K. Rajasekhar, A. Sujana Rani, "Implementation of Low Area and Power Efficient Architectures for Digital FIR Filters*", IOSR Journal of VLSI and Signal Processing (IOSRJVSP)*,Vol. 1, pp. 22-28, Sep.-Oct. 2012.

[7] Prof. Gopal S.Gawande, Dr.K.B.Khanchandani, T.P.Marode, "Approaches to Design & Implement High Speed-Low Power Digital Filter: Review", *International Journal of Computing & corporate Research*, Vol. 2, Issue 1, Jan. 2012.

[8] Francis G. Wolff, Michael J. Knieser, Dan J. Weyer, Chris A. Papachristou, "High-Level Low Power FPGA Design Methodology", *IEEE,* pp. 554-559, 2010.

[9] Aung Soe Khaing and Zaw Min Naing, "Quantitative Investigation of Digital Filters in Electrocardiogram with Simulated Noises", *International Journal of Information and Electronics Engineering*, Vol. 1, No. 3, pp. 210-216, November 2011.

[10] Mahesh S. Chavan, Ra.Agarwala, M.D.Uplane, " Design and implementation of Digital FIR Equiripple Notch Filter on ECG Signal for removal of Power line Interference", *WSEAS Transactions On Signal Processing*, Vol. 4,Issue 4, pp. 221-230, April 2008..

[11] Neeraj kumar, Imteyaz Ahmad, Pankaj Rai, "Signal Processing of ECG Using Matlab", *International Journal of Scientific and Research Publications*, Vol. 2, Issue 10, pp. 1-6, Oct. 2012,.

[12] Biswajit Mishra and Bashir M. Al-Hashimi, "Subthreshold FIR Filter Architecture for Ultra Low Power Applications", *Springer Berlin Heidelberg*, pp. 1-10, 2009.

ISSN: 2278 – 7798

Ms. Rupali Narsale She is currently pursuing her M.E. with specialization in VLSI & Embedded Systems under Pune University. She has received her B.E. degree from Pune University in Electronics & Telecommunication Engineering in 2008.

Ms. Dhanashri Gawali received Bachelor degree in Electronics engineering from Pune University in 2001, Master degree in Electronics-VLSI engineering from Bharati Vidyapeeth University in 2005. She is pursuing Doctorate in Electronics and currently working as an Assistant Professor at an engineering institute affiliated to Pune University, India. She is member of ISTE, VLSI Society of India, ACEEE and IACSIT. Her area of interest includes hardware design using reconfigurable platform and its applications in various domains such as computing, communication and control.

Mr. Amit Kulkarni He completed his M.Tech.in Embedded System. He is currently working as Assistant Professor at an engineering institute affiliated to Pune University, India. He is member of ACEEE. His paper is published on xDSL national conference.

ISSN: 2278 – 7798