Energy Efficient FIR Filter for Biomedical Application using FPGA

Foisal Ahmed[1], Md. Fokhrul Islam[2]

[1]Department of Electrical and Electronic Engineering, Prime University, Dhaka-1216, Bangladesh,

[2] Department of Electrical and Electronic Engineering, Islamic University of Technology (IUT), Organization of Islamic Cooperation (OIC), Board Bazar, Gazipur-1704, Bangladesh.

faisal_ete@yahoo.com, fokhrul@iut-dhaka.edu

Abstract

This research work proposes a low power Finite Impulse Response (FIR) digital filter system using Field Programmable Gate Array (FPGA) to filter biomedical signals. A major issue regarding such implementations is associated with computational complexity that may hinder its practical application. Several researches that aim to overcome this problem have been found by using various designs like booth multiplier, distributive arithmetic design, reduce adder graph (RAG), common sub-expression elimination method etc. This work has achieved the target of power line frequency of 50Hz noise cancellation. It is shown that the proposed design using RAG can give faster speed and require less hardware resources than that for the Direct Structure-I design. The performance of the proposed method is also compared with several existing research works to study its efficacy and it is shown that proposed design can achieve nearly 2% area delay product and 43% less power consumptions in implementation of a FIR low-pass filter. The designed filter is also tested on ECG waves as example of biomedical signal from samples of the MIT-BIH Arrhythmia database corrupted with power frequency noise. The entire system has been implemented on the ALTERA DE-II FPGA education board by synthesizing Verilog HDL using Quartus II tool.

Key words: FPGA, FIR, Low Pass Filter, Power Line Noise, RAG, Verilog HDL.

Introduction

Digital filter gives the new shape and dimension in the modern technology day by day. Finite Impulse Response (FIR) filters are the most basic digital signal processing system components. It is at any rate the frequency with a strictly linear phase. There is no input to output feedback, which is also a stable system [1]. The direct form architecture is the simplest way to implement the FIR filter in realization. Recently, many researchers are using FPGA platform for implementing the FIR filter for designing purpose [2] and [6].

Presently, reducing the implementation complexity of FIR filter in FPGA has been focused a lot of research. To design any FIR filter, the multiplier is one of the main restrictions which evaluate the performance of the desired filter. In FIR filter, the multiplication operation is performed between one particular variable and many coefficients and known as the multiple constant multiplications (MCM). The algorithms proposed earlier to implement this MCM for an efficient FIR filter design can be categorized in two main groups: 1) graph based algorithms and 2) common sub-expression elimination (CSE) algorithms.

In real time applications of FIR filter whose coefficients sometimes dynamically changing makes problem in FPGA and ASIC implementation. In [3] and [7] a method has been proposed efficient constant multiplier architecture based on vertical-horizontal binary common sub-expression elimination algorithm (VHBCSE) to minimize this problem. Their technique is capable of reducing the average probability of use or the switching activity of the multiplier block adders by 6.2% and 19.6%. They have also showed better improvement in power consumption and delay in the FIR implementation by using VHBCSE algorithm. Similarly, problem in implementing the FIR filters whose coefficients change during runtime can be overcome by using distributed arithmetic (DA)-based [4]. In that case, they have used lookup tables (LUTs) for implement and found nearly 68% - 58% less area-delay product and 78% -59% less energy per sample than DA-based systolic structure. They

have reviewed the basic design approaches applicable for the design synthesis of hardware efficient FIR filter. One familiar approach is to encode the tap coefficients of such filter in the form of signed power-of two and thus the operation of multiplication is substituted by simple addition and shifting. By using graph based algorithm [8], the author presents some solutions in implementing the MCM. Thus, Graph based designed has been proposed in this work to implement the FIR filter for filtering the biomedical signal to evaluate the performance efficiently.

Transpose Structure for Implementation of an FIR Filter

Practically, FIR filters are employed in filtering problem upon maintaining the linear phase characteristics. Basically, an FIR with constant coefficients is an LTI digital filter the response of FIR filter of length L, to the discrete time input signal x[n] is given by a finite series of the convolution sum given by

$$y[n] = x[n] * h[n] = \sum_{k=0}^{L-1} x[k]\, h[n-k]$$

In Z-domain, it sometimes more convenient to use as Y[Z] = X[Z]H[Z], where H[Z]is the transfer function defined by $H[Z] = \sum_{k=0}^{L-1} h[k]\, z^{-k}$

The L^{th} order LTI FIR filter is shown in (Figure -1) in the figure the combination n of adders and multipliers are referred as "tapped delay line" and the filter coefficients are known as "tap weight". Conventionally, FIR filter is known as transversal filter.

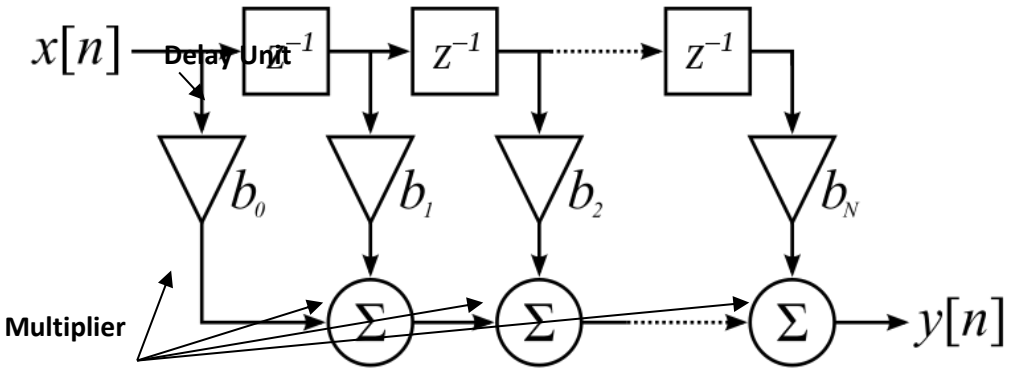


Figure -1: Structure of Direct form of FIR filter of length N.

Another preferred model of FIR filter is transposed structure which is made by some variation of the direct model. The main advantages of such model is that we do not need an extra shift register for x[n] also can achieve high throughput without need of an extra pipeline stage for the adder of the product.

By the virtue of modern technology, computer aided design (CAD), digital filter can be designed easily. Design means to determine the filter coefficients which meet all filter specifications of desired filter. As example, MATLAB signal processing tools box helps us in designing the FIR filter efficiently keeping all of filter specifications.

Another important issue in designing high performance filter means that the designed filter has high implementation complexity and high cost due to higher order. Researchers are concentrating on these issues to develop high performance filter with less complexity. Several FIR filter design techniques exist to overcome this problem. Some well-known design such as window design [9], Parks-McClellan design [10], minimum phase equiripple design [11] and interpolated design. However, every design has some trade-off with one another. At present, numerous works has been carried out to develop hardware efficient FIR filter which are less area and

power consumption. Discuss in earlier section on this paper as some design like CSE, DM, MCM, and Multiplier less filter shown better results on this issue. By using another popular approach graph dependency structure has been taken consider to design in this work. In this methodology, the tap coefficients of such filter are used as in the form of sum of signed powers of two and therefore the multiplication steps are replaced by ordinary addition and shifting.

The idea of graph representation of multiplier block was initially introduced by Bull and Horrcks [12] and successively this area has been developed with some other research [13]-[14]. Actually, applying this technique the numbers of adders are being reduced in representing the constant multiplier. As for example the constant 45 can be shown two ways which represent the fewer adders then the canonic sign digit (CSD). (Figure-2a) shows 45x computed as ((x-4x)-16x) + 64x by the help of three adders. On the other hand, in (Figure-2b) and (Figure-2c) representing finest way when only required two adders using Graph based approach.

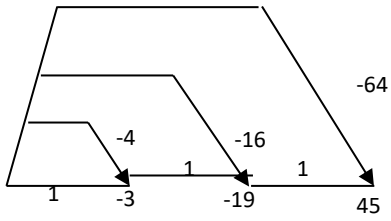


Figure-2a

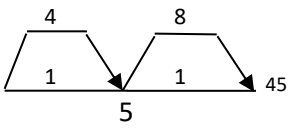


Figure-2b

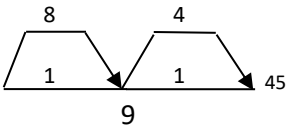


Figure-2c

Proposed Reconfigurable Transposed based FIR filter for FPGA Implementation

Firstly, the filter co-efficients are derived by using MATLAB signal processing tool box named FDA (Filter design and Association). The designed filter is being made such a way that which maintain all filter specifications accurately. FIR filter can be implemented by using adders, multipliers and registers. For optimization purpose, the number of multipliers should be reduced. It is not area efficient design, if using lot of multipliers. As our proposed filter has 15 orders, it requires minimum 16 multipliers for efficient computation. But implementing 16 multipliers in FPGA board is not an appropriate way as it is not an area efficient. Furthermore, FPGA board has limited number of hardware resources. Therefore, the constant multiplication processes are decomposed into addition and shifts method. This process eventually reduces the complexity. This work proposes the architecture of all coefficients for FIR filter using the n-dimensional reduced adder graph (RAG) algorithm which potentially remove the hardware complexity both in area and power consumption. It realizes the following formula described below:

1) All coefficients are being converted into some constant number. After then these number is to be changed into 8-bit integer number.

2) The set of all coefficients are completely synthesized using RAG algorithm and represent only adders and shift to obtain the gain of minimum adder cost efficiently.

Cost estimation for the coefficients using RAG algorithm:

Design process

C [0] = 3 = $2^1 + 2^0$

C [1] =-16=16 = 2^4

C [2] = 77 = $2^6 + 2^3 + 2^2 + 2^0$

C [3] = 128 = 2^7

Graph Based Designed

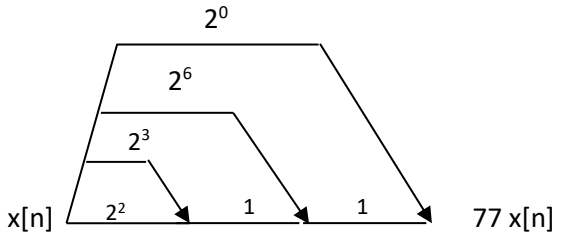


Figure-3a

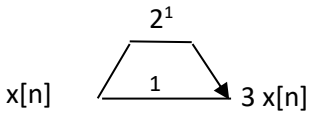
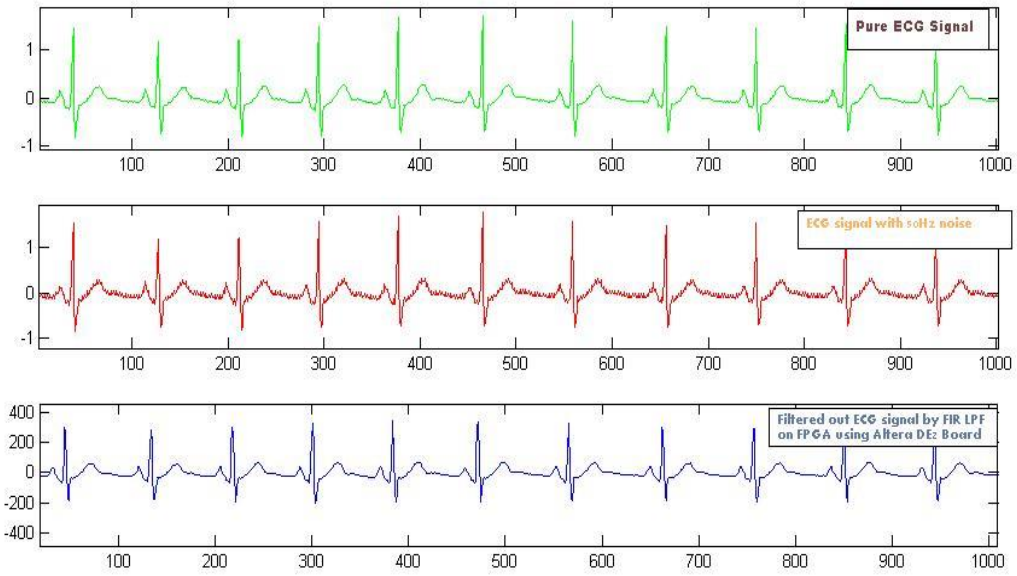


Figure-3b

Result and Comparison

Our design is being equipped to 11 bit and 19-bit adders and 8-bit multipliers via verilog HDL code by using Quartus-II synthesis tool and implementing on Altera FPGA DE board and device Family named Cyclone II; Model no: EP2C5F256C8. Also, power is being analyzed using Quartus-II power analyser. As our designed filter is for the processing of biomedical signal, therefore some ECG sample data has been collected from the MIT-BIH [15] for testing purpose shown in Figure 4.



(Table-1) shows firstly comparison between the direct structures and proposed structure. After then (Table-2) shows the power consumptions, number of LUTs, number of slices and FFs and the type of device that has used in different articles.

Table-1: Resources Utilization of the Hardware of the proposed designed

Resources	Used	Percentages (%)
Total Logic elements	125 out of 4,608	3%
Total combinational functions	93 out of 4,608	2%

Dedicated logic registers	110 out of 4,608	2%
Total pins	21 out of 158	13%
Total memory bits	0 out of 119,808	0%
Embedded Multiplier 9-bit elements	0 out of 26	0%

To compare the proposed designed, we have used Direct Structure-I filter of the same order shown in (Figure - 5) without using RAG algorithm on the same order FIR LPF.

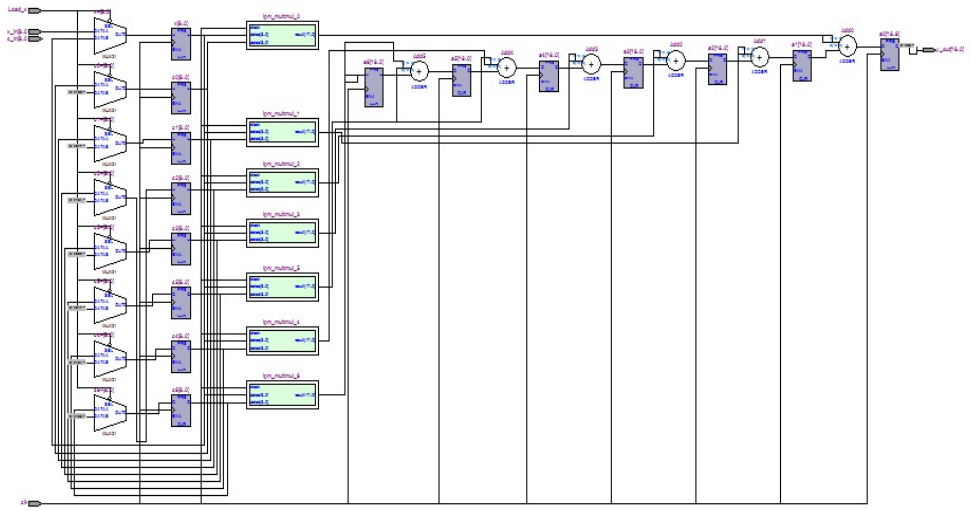


Figure-5: RTL view of Direct Structure-I FIR filter of the same order.

In (Figure-6) and (Figure-7) shows the performance chart between the proposed designed using RAG and the Direct Structure-I without using RAG algorithm.

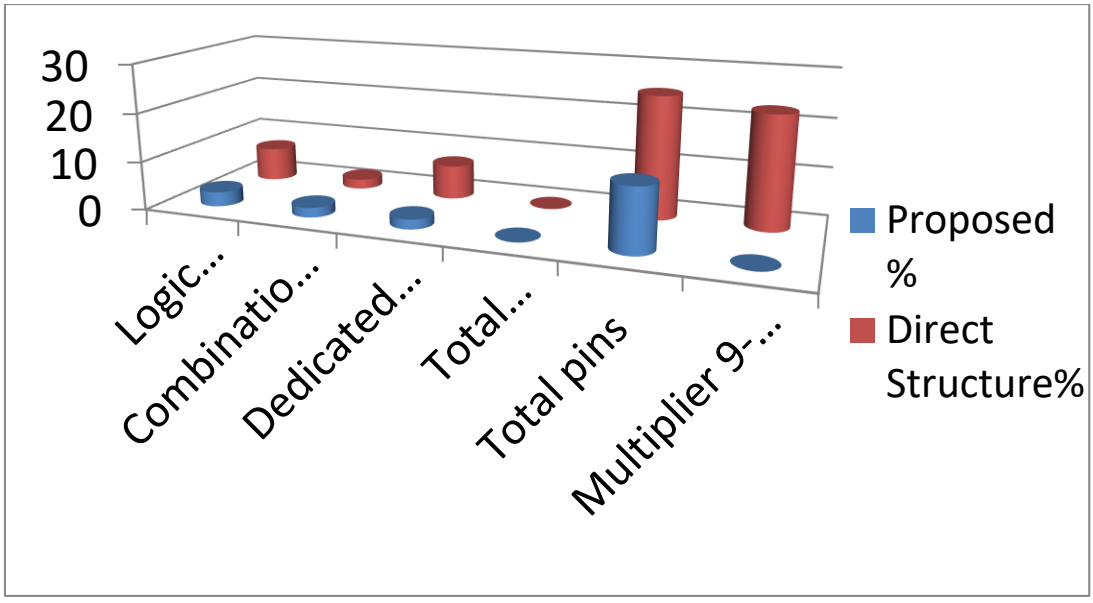


Figure -6: Comparison Chart between proposed and Direct Structure-I filter.

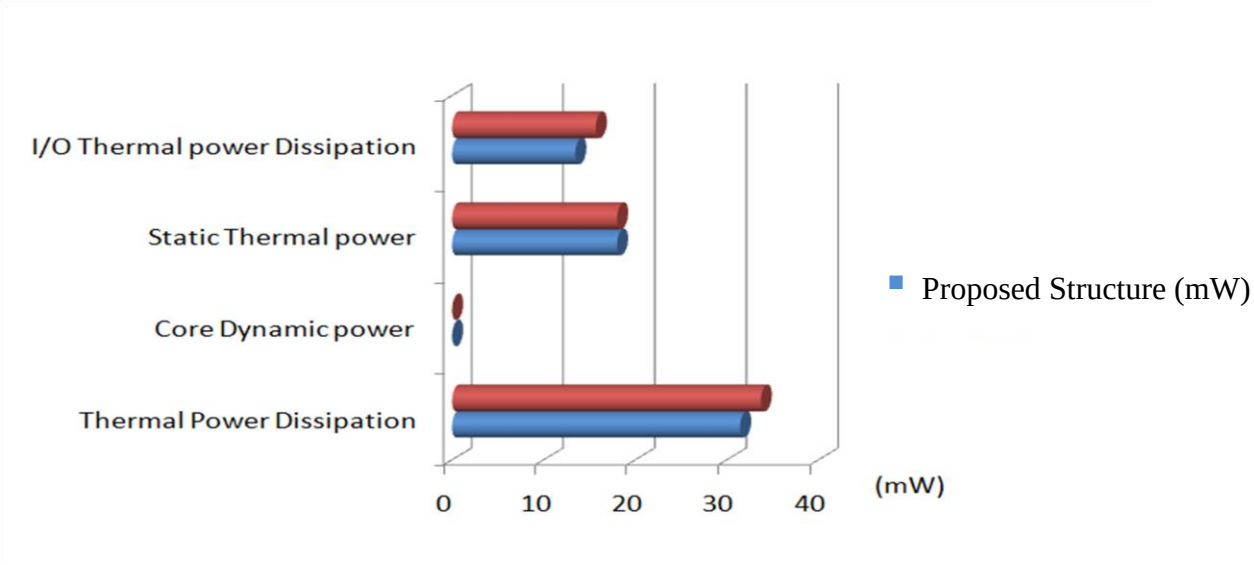


Figure 7: Comparison Chart of power summery between Proposed and Direct Structure-1.

Table-2: Comparison of Resources Utilization between proposed designed with the reference [2]

Reference-[2]				**Proposed Design**			
Model	**Resources**	**Used**	**Percentages (%)**	**Model**	**Resources**	**Used**	**Percentages (%)**
	Total logic elements	13,777 out of 18,752	73%		Total logic elements	125 out of 4,608	3%

Altera DE II FPGA	Total combinational functions	13,611 out of 18,752	73%	Altera DE II FPGA	Total combinational functions	93 out of 4,608	2%
	Dedicated logic registers	3,532 out of 18,752	19%		Dedicated logic registers	110 out of 4,608	2%
	Total memory bits	131,072 out of 239,616	55%		Total memory bits	0 out of 119,808	0%
	Embedded Multiplier 9-bit elements	20 out of 52	38%		Embedded Multiplier 9-bit elements	0 out of 26	0%
	Bit length	8	---		Bit length	8	---

Now, the utilized resources of this proposed work are compared with the other related works [2] and [5] in the (Table-2) and (Table-3) respectively.

Firstly, the proposed filter is compared with the related work [2] in (Table-2) where using same designed cutoff frequency 50/60Hz and similar 8-bit coefficients but little more sampling frequency 360Hz.

Secondly the proposed filter is compared with another related work [5] in (Table-3) where used same designed cutoff frequency 50/60Hz and similar 8-bit coefficients but more sampling frequency 500Hz. They have used notch FIR filter of (45-50) Hz for suppressing the noise.

Table -3: Comparison of Resources Utilization between proposed designed with the reference [5]

Reference-[5]				**Proposed Design**			
Model	**Resources**	**Used**	**Percentages (%)**	**Model**	**Resources**	**Used**	**Percentages (%)**
Xilinx ISE Artix-7 FPGA	Number of Slice Registers	15,639 out of 26,9200	5%	Altera DE II FPGA	Number of Slice Registers	125 out of 4,608	3%
	Number of Fully LUT-FF	8,115 out of 89292	9%		Number of Fully LUT-FF	93 out of 4,608	2%
	Number of bonded IOBs	52 out of 285	18%		Number of bonded IOBs	21 out of 158	13%

	Number of BUFG	2 out of 32	6%		Number of BUFG	0 out of 119,808	0%
	Bit length	9	---		Bit length	8	---
	Power Consumption (mw)	73	---		Power Consumption (mw)	31.45	---

Now the (Figure-8) shows the comparison chart of hardware resources of the proposed designed filter with the references [2] and [5].

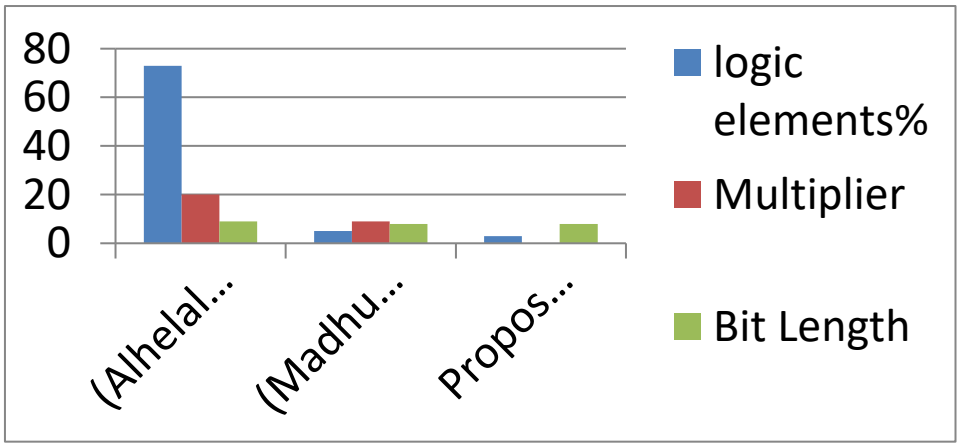


Figure-8: Comparison chart of hardware resources of the proposed design with the references [1]-[2].

So, this optimized low power FIR filter can be used for filtering the ECG signal in any small portable smart devices where hardware resource is very much concerned.

Conclusion

The work has proposed a reliable design of FIR digital filter for extracting the pure biomedical signal with using minimum power and hardware cost. Using this filter, power frequency can be removed effectively from ECG, EEG, and PPG etc. signal. The usage resources of this filter are so low enough that the proposed reconfigurable FIR filter can be easily used in any portable device such as smart phone or TAB. It is shown that proposed design can achieve nearly 2% area delay product and 43% less power consumptions in implementation of a FIR low-pass filter than other research work.

References

1. Mohapatra, B. and Mohapatra, R. (2016). Performance Analysis on Frequency Response of Finite Impulse Response Filter. Procedia Computer Science, 79, pp.729-736.
2. Alhelal, D., Aboalayon, A. I. K., Daneshzand, M. and Faezipour, M. (2015). FPGA-Based Denoising and Beat detection of the ECG signal. Transaction on Systems, Applications and Technology Conference (LISAT), Long Island.

3. Hatai, I., Chakrabarti, I. and Banerjee, S. (2015). An Efficient Constant Multiplier Architecture Based on Vertical-Horizontal Binary Common Sub-Expression Elimination Algorithm for Reconfigurable FIR Filter Synthesis. IEEE Transactions on Circuits and Systems I: Regular Papers, pp.1-10.
4. Park, S. and Meher, P. (2014). Efficient FPGA and ASIC Realizations of a DA-Based Reconfigurable FIR Digital Filter. IEEE Transactions on Circuits and Systems II: Express Briefs, 61(7), pp.511-515.
5. Madhukar, N. R., Gawali, D. and A. Kulkarni. (2014). FPGA Based Design & Implementation of Low Power FIR Filter for ECG Signal Processing. International Journal of Science, Engineering and Technology Research.V3(6).
6. Kumm, M., Fanghänel, D., Möller, K., Zipf, P. and Meyer-Baese, U. (2013). FIR filter optimization for video processing on FPGAs. EURASIP Journal on Advances in Signal Processing, 2013(1), p.111.
7. Chandra, A. and Chattopadhyay, S. (2016). Design of hardware efficient FIR filter: A review of the state-of-the-art approaches. Engineering Science and Technology, an International Journal, 19(1), pp.212-226.
8. Manoj M, and Imtiaz R, (2016). FIR Filter Design using Multiple Constant Multiplication (MCM) Method with Modified Ripple Carry Adder. International Journal of Engineering Research and, V5(03).
9. Engelberg, S. (2008). Digital signal processing. 1st ed. Berlin: Springer.
10. McClellan, J., Parks, T. and Rabiner, L. (1973). A computer program for designing optimum FIR linear phase digital filters. IEEE Transactions on Audio and Electroacoustics, 21(6), pp.506-526.
11. Mathworks.com. (2019). Designing Low Pass FIR Filters - MATLAB & Simulink Example - MathWorks United Kingdom. [online] Available at: https://www.mathworks.com/help/dsp/examples/designing-low-pass-fir-filters.html [Accessed 1 March. 2019].
12. Bull, D. and Horrocks, D. (1991). Primitive operator digital filters. IEE Proceedings G Circuits, Devices and Systems, 138(3), p.401.
13. Dempster, A. and Macleod, M. (1995). Use of minimum-adder multiplier blocks in FIR digital filters. IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, 42(9), pp.569-577.
14. Voronenko, Y. and Püschel, M. (2007). Multiplierless multiple constant multiplication. ACM Transactions on Algorithms, 3(2), p.11-es.
15. Physionet.org. (2019). *PhysioBank Databases*. [online] Available at: http://www.physionet.org/physiobank/database/#ecg [Accessed 1 March 2019].