Journal of Cardiovascular Disease Research

ISSN: 0975-3583,0976-2833 VOL12,ISSUE05,2021

HIGH PERFORMANCE, LOW POWER ARCHITECTURE OF 5-STAGE FIR FILTER USING MODIFIED COMPRESSOR WITH MONTGOMERY MULTIPLIER

Ata. Kishore Kumar[1], Afreen Kubra[2], Samreen Fiza[3], Dr. Sandhya Tatekalva[4], G. Tirumala Vasu[5]

[1]Electronics & Communications Engineering Department, KMM Institute of Technology and Science, Tirupati, Andhra Pradesh, India

[2,3,5]Electronics & Communications Engineering Department, Presidency University, Bangalore, Karnataka, India.

[4] Department of Computer Science, S. V. University, Tirupati, Andhra Pradesh, India.

[1]drkishore1609@gmail.com, [2]afreenkubra96@gmail.com, [3]samreenfiza236@gmail.com, [4] dr.sandhyasatish@gmail.com, [5]gt7hillsrit@gmail.com

ABSTRACT

The digital world continues to witness an unprecedented growth in view of the technological advancements in the field of Digital Signal Processing (DSP). The increased usage of digital applications along with the tremendous evolution of Very Large Scale Integration (VLSI) technology over a few epochs has led to the development of enhanced algorithms and architectures for DSP systems that augur to meet the demands of a wide variety of applications in this expanding horizon of the signal processing sector. Finite Impulse Response (FIR) digital filter is the most potent and frequently used component in various signal processing and image processing applications. Since the intricacy of implementation grows with the filter order and the precision of computation, real-time realization of these filters with desired level of accuracy and less area-delay-power complexity is a challenging task.

Multiplication is one of the most extensively used arithmetic operations in a wide range of applications, such as multimedia processing and artificial neural networks. For such applications, multiplier is one of the major contributors to the energy consumption, critical path delay and resource utilization. These effects get more pronounced in FPGA-based designs. However, most of the state-of-the-art designs are done for ASIC-based systems. Further, few FPGA-based designs that exist are largely limited to unsigned numbers, which require extra circuits to support signed operations. To overcome these limitations for the FPGA-based implementations of applications utilizing signed numbers, this paper presents an area-optimized, low-latency and energy-efficient architecture for accurate signed multiplier. Based on these objectives, many algorithms and architectures have been proposed for FIR filters. The goal of this project is to demonstrate the work method of a simple First order FIR filter by showing the results in a Vector Waveform simulation. FIR filter is a digital filter. Time delay gives the information of speed. FIR filter which has minimum time delay gives better response than others. Area utilization and time delay are two important factors to design any filter. The important advantage of FIR filter on FIR filter is its implementation efficiency. FIR filters require less number of orders as

ISSN: 0975-3583,0976-2833 VOL12,ISSUE05,2021

comparison to meet same specification. FPGA provides more logic flexibility and the power consumption is low. FPGA is a semiconductor device containing programmable logic components and programmable interconnects.

Keywords: Astute Compressor, Digit-Serial Montgomery Multiplier, FPGA, HON_LPWM strategy.

1. INTRODUCTION

An Integrating all the blocks to construct a model in a single package has become the need of the day. Field Programmable Gate Arrays (FPGA) has become most popular among all the integrating techniques available. Both logic design and DSP based applications are supported by FPGA's. Lot of research on FPGA's are carried out to accelerate DSP applications. Complicated system demands faster speed as well as quick and fast system performance. These could be done by optimizing a circuit to a single problem. To achieve high throughput, DSP systems dictate hardware intensive solutions. If 40 million instructions per second (MIPS) is the operating speed of a DSP chip, the bandwidth requirement is lower as compared to 500 KHz. Therefore DSP processors are unfit for implementation of many stages of communication system. FPGA's are reprogrammable for various equipment capacities with no expansion in the execution cost as connected with custom ICs and ASIC's. Configurability and parallelism are the real points of interest of FPGA's. Size/adaptability of the rationale cell, speed and the directing engineering are central point for a FPGA based outline. The rationale cell executed on FPGA can be a basic transistor or an unpredictable structure like chip. FPGA's have two execution approaches.

One is through SRAM based LUT and the second using multiplexer based rationale components. Multiplexer based execution is by and large quicker than SRAM based outline, yet possess marginally more prominent region. Commotion and other undesirable sign parts of a sign can be evacuated by computerized channels. They work on a limited accuracy computerized representation of a sign. Simple signs are changed over to advance by testing, quantizing and coding. A small special purpose digital computer then processes the obtained digital signal and converts the input sequence into a desired output sequence. Advanced preparing of simple signs has a few points of interest. The PC can be time shared for a few uses and the computerized usage expense is impressively lower than that of its simple partner. Advanced channel's exactness depends just on the word length, quantizing interim and the testing rate. Instead of RLC segments, the advanced channels utilize basic components like adders, multipliers, shifters and postponement components. Subsequently they are not influenced by elements like temperature, warm float and others that influence the simple channel. Advanced channels can be altered by basically changing the PC calculation as opposed to simple framework which must be physically remade. The execution of limited move band FIR channels requires impressively more number of juggling operations and equipment parts, for example, multipliers, adders and postponement components. To accomplish the same frequency detail, FIR channels require higher number of taps contrasted with FIR channels. To actualize FIR channels, extensive

ISSN: 0975-3583,0976-2833 VOL12,ISSUE05,2021

number of multiply and Accumulate (MAC) squares are required. The operational velocity of FIR channel is by and large a bottleneck for high channel throughput. This makes them unacceptable for FPGA usage. To stay away from MAC operations, various Multipliers less methods are being utilized. Among all the accessible strategies, Distributed Arithmetic (DA) is most appropriate for FPGA execution. DA utilizes Look-Up Tables, adders and movement registers to play out the separating operation. In this system LUT's are pre developed for all the conceivable blends of inputs and the channel coefficients.

They are conjured while performing channel operation. In the present work an effective and upgraded Distributed Arithmetic based strategy for rapid reconfigurable outline and execution of FIR channels is created. In practical applications of digital signal processing, changing the sampling rate of a signal is a major challenge to be addressed. Multi-rate systems employing multiple sampling rates in the processing of digital signals are popular in DSP. The work started with the understanding of the basic working principle of FIR filter, the key components required for the design of a filter, the area and design complexity involved in the usage of multipliers in the filter. After a broad literature survey, multiplier less technique such as Distributed Arithmetic Algorithm was found as an efficient method for design of FIR filter.

As a First step of research work, implementation of FIR filter using system generator was done. Verification of behavioral functionality of the developed model was done. Implementation of FIR filter using Xilinx tool and verification of behavioral functionality of the developed model was completed. Also analysis for memory utilization for the filter was made. Basic FIR filter based on Distributed Arithmetic was designed and implemented on FPGA. In a conventional DA, as the contents of the Look up Table is fixed, if there is a need for change in the filter coefficients, whole of the architecture on FPGA, had to be changed which was again a major setback. To overcome the mentioned setback, improvement or modification on the existing algorithm was done which resulted in Reconfigurable Distributed Arithmetic architecture where the filter coefficients could be changed at the runtime without any change in the architecture of the filter. Comparative analysis of the developed design is provided to show that the implemented design can achieve the best results in terms of area and power utilization. Application area of the implemented FIR filter is also discussed.

2. LITERATURE SURVEY

Back EMF detection is important aspect of the sensor less control of BLDC motor. In this paper, the BEMF detection using FPGA based digital low pass filtering of the line to line voltage using Least *P*th-norm FIR filter has been realized. Lower switches of the legs are fed with PWM and upper switches are fundamentally controlled. The HON_LPWM strategy is used for the speed control as well as for the BEMF detection. Various quantities, e.g., percentage current ripple, current magnitude, line to line back EMF, speed, and switching signals were analyzed for the case of varying voltage with fixed duty as well as for varying duty cycles with fixed DC voltage. It has been analyzed that the back EMF detection with this method has some limitations on duty cycle because intermittent up and down peaks in filtered BEMF waveform become high

ISSN: 0975-3583,0976-2833 VOL12,ISSUE05,2021

as we increase or decrease duty out of certain band. During the implementation of BEMF based sensor less techniques, these up and down peaks in BEMF waveforms have to be removed. For this, further improved filtering has to be searched so as to assure proper wave shaping with least delays in line to line BEMF zero crossings for sensor less application. Whole system has been implemented using Xilinx based System Generator using Vivado 2014.4 design suite for FPGA based programming in connection with MATLAB/Simulink and WAVECT controller.

Simulation results and hardware results have been verified. The application of FPGA-based FIR filtering to increase the usable bandwidth of a piezoelectric transducer used in optical phase locking was also proposed. We experimentally perform system identification of the interferometer with the cross-correlation method integrated on the controller hardware. Our model is then used to implement an inverse filter designed to suppress the low frequency resonant modes of the piezoelectric transducer. This filter is realized as a 24th-order FIR filter on the FPGA, while the total input–output delay is kept at 350 ns. The combination of the inverse filter and the piezoelectric transducer works as a nearly flat response position actuator, allowing us to use a proportional–integral (PI) control in order to achieve stability of the closed-loop system with significant improvements over a non-filtered PI control. Finally, because this controller is completely digital, it is straightforward to reproduce. Our control scheme is suitable for many experiments that require highly accurate control of flexible structures.

Application of FPGA-based FIR filtering to increase the usable bandwidth of a piezoelectric transducer used in optical phase locking was proposed. We experimentally perform system identification of the interferometer system with the cross-correlation method integrated on the controller hardware. Our model is then used to implement an inverse filter designed to suppress the low frequency resonant modes of the piezo-electric transducer. This filter is realized as a 24th-order FIR filter on the FPGA, while the total input-output delay is kept at 350ns. The combination of the inverse filter and the piezo-electric transducer works as a nearly-flat response position actuator, allowing us to use PI control in order to achieve stability of the closed-loop system with significant improvements over non filtered PI control. Finally, because this controller is completely digital, it is straight forward to reproduce. Our control scheme is suitable for many experiments which require highly accurate control of flexible structures.

In this paper, a digital FIR filter is proposed for the fast detection of EEG signal to smooth and compress the signal. This paper intends to design a digital FIR filter based on FPGA to get faster biomedical signals specially EEG signals. For this purpose, a high order FIR filter is introduced to make EEG signal noise free, less costly and simple. For hardware usage, FPGA load up is utilized which is a mix of various logic gates which offers economical and enduring administrations. Processor based implementation of DSP algorithms often result in poor performance as the architecture is being sacrificed to achieve the ease in programmability. This has prompted designers to use different hardware platforms, where in the architecture can be specifically developed to achieve the requisite performance. Very recently FPGAs have featured as a viable platform for realization of different DSP algorithms. However, owing to increased performance requirements, the traditional LUT based FPGAs fail to achieve the desired results.

ISSN: 0975-3583,0976-2833 VOL12,ISSUE05,2021

This has prompted vendors to improve the quality and quantity of the underlying FPGA fabric. One such improvement has been the inclusion of special DSP macro blocks in modern FPGAs. These assist in fast and optimal realization of different arithmetic operations like addition, multiplication etc. In this paper, we consider the realization of Direct and Transposed FIR filter structures using these DSP blocks. Detailed experimentation using Xilinx Spartan-6 FPGAs shows a subsequent improvement in performance when compared to traditional realizations. The FIR filters are one among the digital filters which are widely proposed in field programmable gate array implementations. This paper presents the design of 4-tap and 8-bit fast low-pass FIR filter design under FPGA background using hardware description language (HDL). This design leads many applications like biomedical signals, pattern recognition, and image processing and communications fields. The main attention of this FIR filter is focused towards the noise and performance constraints. In light of FPGA to accomplish FIR filter, not just considered the fixed capacity DSP-explicit chip constant, yet in addition the DSP processor adaptability. The blend FPGA and DSP innovation can further improve integration, increment work speed and framework abilities. The cardiovascular attack is a more dangerous than other diseases and it is measured by ECG (Electro cardiograph) signals which is like a noisy signal in real time, especially in the field of telemedicine environment. The noisy ECG signals have more motion artifacts, electrical interference, etc. An adaptive filtering approach based on Discrete Wavelet Transform and an artificial neural network is proposed to reduce the noise in ECG signal. The quality of de-noised signal is improved by SVM algorithm. This suggested approach can successfully take out a broad scope of noise and our method achieve up to almost 82% improvement on the SNR of de-noised signals. The MATLAB simulation results shown clearly about the improvement of ECG signal with SNR value. The Filter Bank Multi Carrier (FBMC), Generalized Frequency Division Multiplexing (GFDM), Universal Filtered Multi-Carrier (UFMC), and Filtered-Orthogonal Frequency Division Multiplexing (f-OFDM) techniques are proposed for their suitability to 5G requirements. On one side f-OFDM show very good out of band emission characteristic as well as it promises flexibility requirement of 5G. For enhanced Mobile Broad Band (eMBB), it provides a natural extension to simple OFDM. In addition to the acceptable performance achieved through applying f-OFDM, study of hardware cost associated with its implementation is very important which is missing in previous research work. FIR filtering is a key processing block in f-OFDM and differentiates it from its predecessor OFDM waveform. Hence, this paper is aimed at real time hardware implementation of a flexible hardware architecture related to filtering part of Filtered-OFDM transmitter. In this regard, two flexible solutions for f-OFDM FIR filter are presented. These both schemes are modelled in Verilog HDL, which are then synthesized and implemented on Virtex-7 FPGA. Finally, a comparison of these solutions is also provided. Both models provide the flexible novelty solutions for FIR filtering of f-OFDM waveform as well as these architectures shows an outstanding 5 times decrement in hardware utilization in terms of adders and multipliers as compared to previous filtering technique used in UFMC.

ISSN: 0975-3583,0976-2833 VOL12,ISSUE05,2021

3. EXISTING SYSTEM

To derive the transfer function H(z) of a system, digital filter design process is generally used. Digital filters are derived using two approaches: Finite Impulse Response (FIR). FIR filters do have feedback associated with them whereas IIR filters doesn't. As mentioned earlier, these filters (FIR Filters) are used generally in the linear time invariant (LTI) systems. In real practice, the impulse response of these filters is almost zero or approaches to zero at higher values and is neglected after a certain point. However the physical systems which give raise to FIR or FIR responses is not similar.

Analog electronic filters are composition of components like resistor, inductors and capacitors and sometimes linear amplifiers and the combination of all the above mentioned is generally called as FIR filters. On the other contrary, discrete-time filters, which are based on a tapped delay with no feedback associated with them, are necessarily the FIR filters. The memory elements in the analog filters are capacitors or inductors generally and their internal state never completely relaxes following an impulse. But in the another case, after an impulse has occurred at the end of the tapped delay line, then the system has no further memory of that impulse and has to be returned to its initial state. The impulse response is exactly zero beyond that point. A digital filter is programmable and its operation is actuated by a program stored in the processor's memory. So, the digital filter can simply be changed without affecting the circuitry (hardware). It is easily designed, tested, implemented on a general-purpose computer or workstation and handle low frequency signals accurately. For the design of digital filter, the system function H (z) or the impulse response function h (z) must be satisfied.

The digital filter structure can be implemented by hardware or software. In order to implement the specified system the required basic operations are addition, delay and multiplication by a constant. Generally, different structures give different results. The most common methods for realizing digital linear systems are direct, cascade, parallel forms. Recursive filters use long convolution and achieve long impulse response in a very efficient manner. These filters employ a very fast processing speed as they can process the signal very rapidly. The impulse response of these two filters is generally decaying exponentially, whereas there are digital filters, which uses the convolution process named as FIR filters. The FIR filter's methodology is to convert the specifications of the digital filter into a low pass analog prototype filter specification. It can determine the transfer function Ha(s) of the analog low pass filter. In this paper we have designed a 64-Tap low pass FIR Butterworth with 48 KHz and 10.8 KHz as sampling frequency (Fs) and cut-off frequency (Fc) respectively. The magnitude and the phase response of the same are discussed below:

4. PROPOSED METHOD

In the world of smart technology, data security is a crucial entity for every individual. The techniques for securing data are in high demand and became challenging for the designers to come up with efficient ways based on the requirements. The expanding areas of networks and security demand more number of efficient algorithms for use in cryptography. Many researchers

ISSN: 0975-3583,0976-2833 VOL12,ISSUE05,2021

have done phenomenal work and proposed various architectures with the advancements. Encryption is an essential criterion in cryptography for data handling and processing. When explicit information from one environment is to send to the other, then encryption of the data is necessary for security purposes. Cryptography provides pavement for this restraint. It deals with encryption and decryption of the data such that the fundamental information transfer from the sender to the recipient remains confidential. The cryptographic algorithms which are used in advance systems have motive principles of authentication and data secrecy. Performing modular multiplication is one of the core operations in cryptography.

4.1 Conventional Digit-Serial Montgomery Multiplier

The conventional Montgomery multiplier is constructed for m=6 and d=3, where m is the digit size of each input and d is the bit size. The first stage in computing the results of the multiplier is the processing stage. The partial product generation for the given multiplicand and multiplier takes place in this stage with the combination of And-network and Adder-network. The multiplicand and the multiplier are of 6-bit length each. The individual bit representation of multiplicand, multiplier and modulus N.

The complete Montgomery multiplication algorithm takes place in 2 cycles. In the first cycle of the algorithm, 6 bits of multiplicand P and first 3 bits of multiplier Q are given as the inputs. In the second cycle, the next 3 bits of the multiplier Q will be acting as the inputs. To choose the bits of multiplier according to the cycle, 2:1 multiplexers are used. For each bit of the Q, a multiplexer is used and based on the selection line, the required bits are chosen. If the selection line is high for all the 3 multiplexers, first 3 bits of Q i.e., (q0, q1, q2) will be propagated as inputs. If the selection line is low, the next 3 bits of Q i.e, (q3, q4, q5) are propagated as inputs.

In the first cycle of the algorithm, all the carry bits which are given as the inputs to the processing element are considered as zeros i.e, C=0. 6 bits of multiplicand and first 3 bits of the multiplier generate partial products as shown in eq. (4). The outputs of the processing element are the Sum and Carry bits of full adders that are present in the adder network, and they will be driving the next stage. P = (pm−1, pm−2.....p0) Q = ((qm−1, qm−2.....qm−d), (qd−1, qd−2.....q0)) N = (nm−1, nm−2.....n0) C = 0 (3) C = X d i=0 mX−1 j=0 pj qi + C (4) The second stage is the division stage. An array of full adders performs the division operation in the algorithm. Bits of N operates as one of the inputs to the full adders and the other inputs are Sum and Carry bits of the adders involved in the successive stages. The outputs of adders after performing the division operation are the Carry bits from C0 to C6 and Cc1 to Cc6. The Carry bits will be acting as inputs to the final stage of the design. 3 D-flip-flops in which 2 are of 6 bit input size and 1 single D-flip-flop are used to store Carry bits generated from the division cell and to give them back as inputs to the processing element in the second stage.

The usage of D-flip-flops in the circuit reduces the critical path delay. A D-flip-flop works under the presence of a clock signal and operates with a delay in input by one clock cycle. After the generation of Carry bits, the second cycle starts and in the second cycle of the multiplier, the next 3 bits of multiplicand and the Carry bits from the D-flip-flop will repeat the

ISSN: 0975-3583,0976-2833 VOL12,ISSUE05,2021

process and produces the final output of the multiplication from the compressor. The output produced is of 6 bit length which is equal to the bit size of the given inputs to the multiplier.

4.2. Modified Montgomery Multiplier

The modified Montgomery multiplier consists of 3 stages as shown in Fig. (1). the first two stages perform the similar operation as in the conventional design. The reduction cell in the conventional design is replaced with the proposed 12:6 Astute Compressor, making the design into a modified one. This modification resulted in area and power efficient multiplier by reducing the complexity involved in the usage and design of multiplexers in the reduction cell of the conventional design and by declining the number of transistor count.

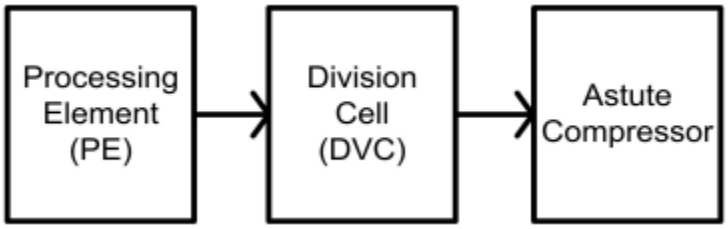

Fig. 1: Block Diagram of Modified Montgomery Multiplier

The modified Montgomery multiplier design works for any 6 bit length input P and Q, but the value of N is restricted to 3 and 7. The results produced by the multiplier are accurate. Using 45 nm technologies in the Cadence Virtuoso tool, the design of modified Montgomery multiplier with a combination of CMOS and PTL logic resulted in low power and area efficient design. The proposed astute compressor lessened the number transistors to a notable amount.

4.3 Proposed 12:6 Compressor: Astute Compressor

The compressor is a circuit that is used to depreciate the area by down turning the transistors count and reduce the overall power and delay for more reliable performance of a design. This paper presents a 12:6 Astute Compressor which compresses 12-bit input to 6-bit output as shown in Fig. (3). the compressor consists of a Half Adder (HA) and a series of Modified Full Adders (MFA).

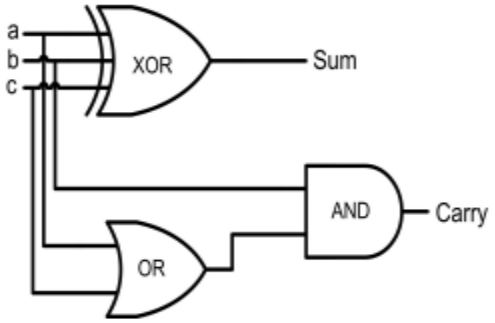

Fig. 2: Modified Full Adder (MFA)

ISSN: 0975-3583,0976-2833 VOL12,ISSUE05,2021

The modified full adder works for all the input combinations except for the input combination of 101 which is binary equivalent of 5. All the adders are connected in series and the Carry of each adder will be propagated to the next successive adder as one of the inputs. The other inputs to the adders are the previous stage Carry bits C0 to C6 and Cc1 to Cc5. Propagation of the Carry bits to the next successive adder take place till the end. The final adder in the compressor consists of only Sum part and the Carry part is neglected for the exact output of 6 bit length.

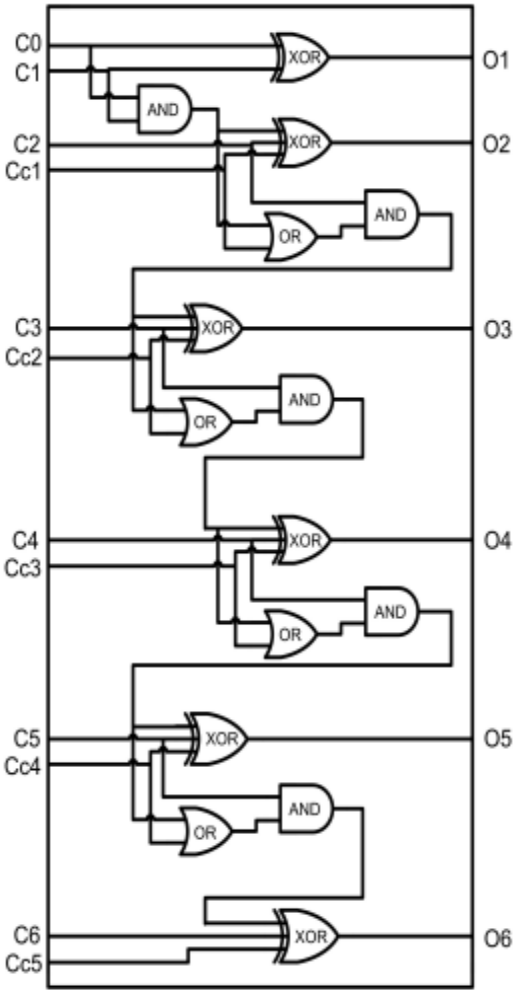

Fig. 3: Proposed 12:6 Compressor:Astute Compressor

In the epoch of propelling technology, processing of the signal to establish proper communication plays a pivotal role. To have better communication, one must take many factors into account. Usage of filters in digital signal processing can reduce the complexity and helps to achieve better performance in various factors such as noise cancellation, removal of unwanted signals, and proper attenuation. Restoration and separation of the signals while signal processing are the two important undertakings by a filter. The classification of filters depends on whether it is an analog or digital filter. The digital filters can be classified into Finite Impulse Response (FIR) filters and Infinite Impulse Response (IIR) filters.

ISSN: 0975-3583,0976-2833 VOL12,ISSUE05,2021

The analog filters can be classified into Butterworth filter, Chebyshev filter etc. A filter can be extended to any number of stages based on the requirement. The complexity depends on number of stages present. More number of stages results in high complexity filter design. With increase in complexity, the cost of the filters will also increase. Therefore, the design of higher order filters with reduced complexity and cost is on high demand in the market. A 5^{th} order low pass FIR filter is adopted for the implementation of the proposed multiplier in this paper. FIR filters are mainly used in the applications where there is a need for linear phase response. In general, FIR filters are stable filters. To design a FIR filter, one must choose the design method. Three methods are known for the design of FIR filter namely, 1) Frequency sampling techniques 2) window methods 3) optimal filter design methods

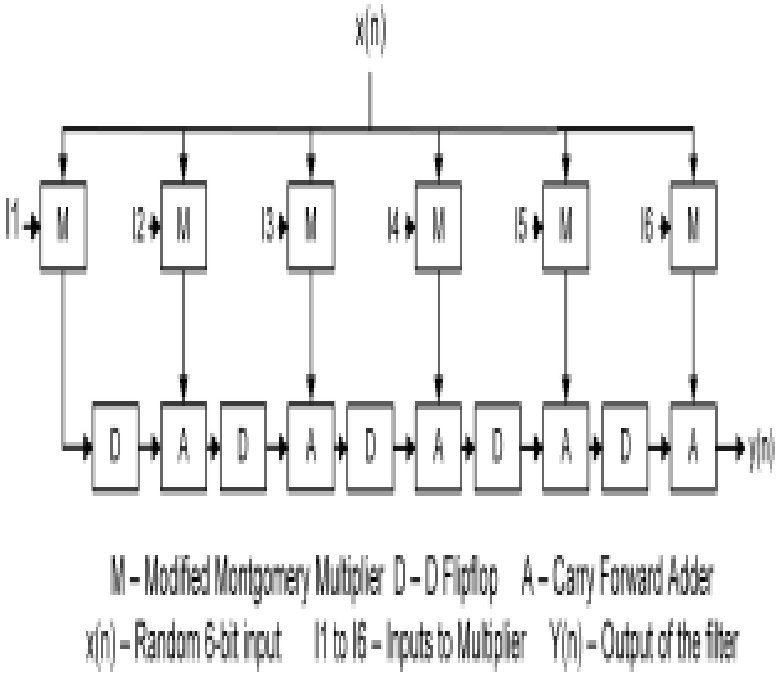

Fig. 4: 5th Order Low Pass FIR Filter

$Y(n) = I1.x(n)+I2.x(n-1).....+I6.x(n-5)$ (5) The general equation of the filter. The filter design consists of multipliers, D-flip-flops, and a carry-forward adder as the main elements as shown in Fig. (4). D-flip-flop is used in order to reduce the critical path delay of the filter. The carry forward adder which is used in the filter consists of a half adder and full adders as its functional elements in which carry of each adder is propagated to the next adder in series. The inputs to the adder are the multiplier output and the D-flip-flop output each of 6-bit length each. The adder produces a 6-bit output, and the final output of the circuit is the output of last adder present in the filter. The complete filter design has been done with 4254 transistor count and the filter produced an average power of 216 uW and a delay of 1.09 ns

ISSN: 0975-3583,0976-2833 VOL12,ISSUE05,2021

5. SIMULATION RESULTS

5.1 Proposed results

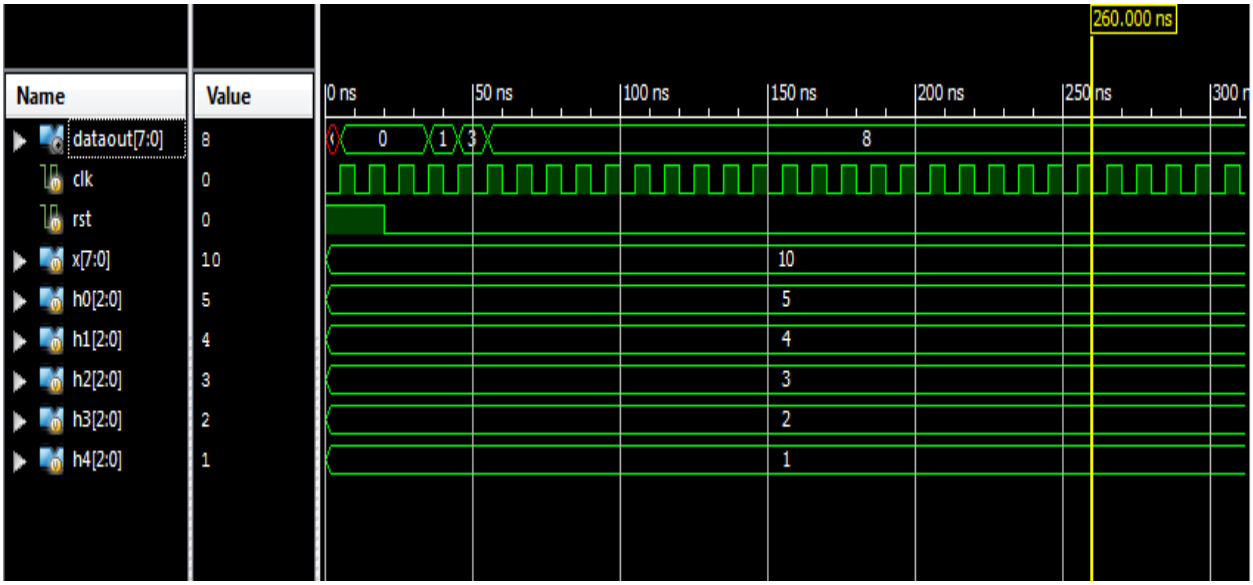

Figure 5.1 Simulation outcome

The figure 5.1 represents the simulation outcome of 5 stage fir filter. Here, Ho, h1, h2, h3 and h4 are the impulse inputs, x is the data input and resultant outcome is stored into data out.

Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slice Registers	32	408000	0%
Number of Slice LUTs	116	204000	0%
Number of fully used LUT-FF pairs	22	126	17%
Number of bonded IOBs	33	600	5%
Number of BUFG/BUFGCTRLs	1	32	3%

Figure 5.2: Design summary

Figure 5.2 represents area utilization by the proposed method; out of available 408000 slice registers only 32 are used. Out of available 20400 slice look up tables only 116 are used, so the area consumption is reduced.

ISSN: 0975-3583,0976-2833 VOL12,ISSUE05,2021

```
Data Path: h0<1> to dataout<7>
                              Gate     Net
  Cell:in->out      fanout    Delay    Delay  Logical Name (Net Name)
  ----------------------------------------  ------------
   IBUF:I->O             11    0.000    0.475  h0_1_IBUF (h0_1_IBUF)
   LUT3:I0->O             3    0.043    0.466  B1/C1/F2/Mxor_sum_xo<0>211 (B1/C1/F2/Mxor_sum_xo<0>2
   LUT5:I1->O             6    0.043    0.573  B1/C1/F2/Mxor_sum_xo<0>23 (B1/C1/F2/Mxor_sum_xo<0>2)
   LUT6:I0->O             2    0.043    0.546  B1/C1/F1/b_cin_AND_2_o1 (B1/carry1)
   LUT5:I0->O             3    0.043    0.299  B2/C2/F1/cout1 (B2/carry2)
   LUT5:I4->O             3    0.043    0.299  B2/C3/F1/cout1 (B2/carry3)
   LUT5:I4->O             3    0.043    0.299  B2/C4/F1/cout1 (B2/carry4)
   LUT5:I4->O             3    0.043    0.299  B2/C5/F1/cout1 (B2/carry5)
   LUT5:I4->O             3    0.043    0.466  B2/C6/F1/cout1 (B2/carry6)
   LUT5:I1->O             1    0.043    0.428  d2<6>1 (d2<6>)
   LUT4:I1->O             1    0.043    0.550  dataout<7>9 (dataout<7>9)
   LUT6:I0->O             1    0.043    0.279  dataout<7>10 (dataout_7_OBUF)
   OBUF:I->O                   0.000           dataout_7_OBUF (dataout<7>)
  ----------------------------------------
  Total                        5.454ns (0.473ns logic, 4.981ns route)
                                       (8.7% logic, 91.3% route)
```

Figure 5.3 Time summary

Figure 5.3 represents time utilization by the proposed method, and the proposed method consumes total 5.45ns of path delays, which includes 0.473ns logical delays and 4.981ns of route delays, respectively.

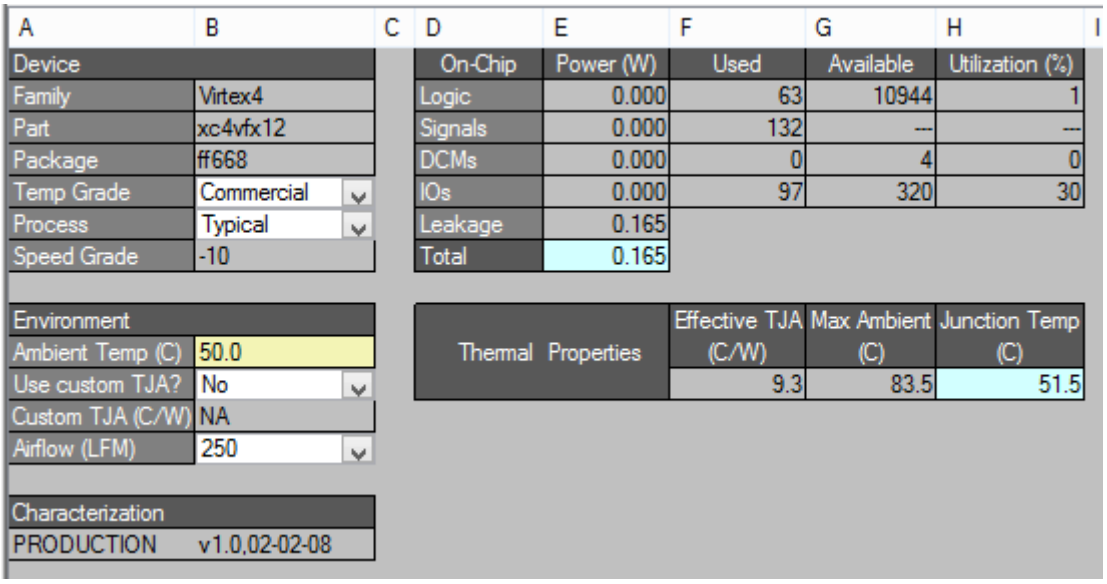

Figure 5.4: Proposed power summary

Figure 6.4 represents power utilization by the proposed method, and the proposed method consumes total 0.165 watts of power, respectively.

ISSN: 0975-3583,0976-2833 VOL12,ISSUE05,2021

6. CONCLUSIONS

In this work, we have explored the possibility of realization of block FIR filters in transpose form configuration for area delay efficient realization of both fixed and reconfigurable applications. A generalized block formulation is presented for transpose form block FIR filter, and based on that we have derived transpose form block filter for reconfigurable applications. We have presented a scheme to identify the MCM blocks for horizontal and vertical sub expression elimination in the proposed block FIR filter for fixed coefficients to reduce the computational complexity. Performance comparison shows that the proposed structure involves significantly less ADP and less EPS than the existing block direct-form structure for medium or large filter lengths while for the short-length filters, the existing block direct-form structure has less ADP and less EPS than the proposed structure.

REFERENCES

[1]. Soni, Umesh Kumar, MalothNaresh, and Ramesh Kumar Tripathi. "FPGA Based Speed Control and Back EMF Extraction from Line Voltages Using FIR Digital Filters for BLDCM." Advances in Power and Control Engineering. Springer, Singapore, 2020. 41-62.

[2]. Okada, Masanori, et al. "Extending the piezoelectric transducer bandwidth of an optical interferometer by suppressing resonance using a high dimensional FIR filter implemented on an FPGA." Review of Scientific Instruments 91.5 (2020): 055102.

[3]. Okada, Masanori, et al. "Extending the PZT bandwidth of an optical interferometer by suppressing resonance using a high dimensional FIR filter implemented on an FPGA." arXiv preprint arXiv:1912.10707 (2019).

[4]. Afreen Kubra, Ata. Kishore Kumar, G. Tirumala Vasu, Samreen Fiza, “Toll-Gate Payment Automation with RFID and SMS Alerts”, Journal of Critical Reviews, Vol 06, Issue 06, 2019.

[5]. Mahabub, Atik. "Design and implementation of cost-effective FIR filter for EEG signal on FPGA." Australian Journal of Electrical and Electronics Engineering (2020): 1-9.

[6]. Khurshid, Burhan, et al. "Achieving Performance Speed-Up in FPGA Based FIR Filters Using DSP Macro Blocks." 2019 6th International Conference on Signal Processing and Integrated Networks (SPIN). IEEE, 2019.

[7]. Sudharsan, R. Raja, and J. Deny. "Field Programmable Gate Array (FPGA)-Based Fast and Low-Pass Finite Impulse Response (FIR) Filter." Intelligent Computing and Innovation on Data Science. Springer, Singapore, 2020. 199-206.

[8]. Balachandran, G., and Praveen Kumar Gupta. "FPGA–Based Electrocardiography Signal Analysis System using (FIR) Filter." International Journal 8.1 (2020): 17-19.

[9]. Shahbaz, Muhammad Mateen, AasimWakeel, and Bahram Khan. "FPGA Based Implementation of FIR Filter for FOFDM Waveform." 2019 2nd International Conference on Communication, Computing and Digital systems (C-CODE). IEEE, 2019.

ISSN: 0975-3583,0976-2833 VOL12,ISSUE05,2021

[10]. Khan, Mansoor, and Shahrukh Agha. "Least squares linear phase FIR filter design and its VLSI implementation." Analog Integrated Circuits and Signal Processing (2020): 1-11.

[11]. Zhang, Ning, et al. "FPGA-Based Implementation of Reconfigurable Floating-Point FIR Digital Filter." International Conference in Communications, Signal Processing, and Systems. Springer, Singapore, 2019.

[12]. Samreen Fiza, Afreen Kubra, Ata. Kishore Kumar, G. Tirumala Vasu, "Bundle protocol implementation for satellite applications", JOURNAL OF CRITICAL REVIEWS, VOL 06, ISSUE 06, 2019.

[13]. Katru, Raju, and M. Chandrasekher. "FPGA Based Parallel Filters for Failure Recovery." 2019 3rd International Conference on Recent Developments in Control, Automation & Power Engineering (RDCAPE). IEEE, 2019.

[14]. G. Tirumala Vasu, Samreen Fiza, Afreen Kubra, Ata. Kishore Kumar, "Home Automation using IoT", International Journal of Advanced Science and Technology Vol. 29, pp. 6611 – 6617, 2020.

[15]. Afreen Kubra, RK Karunavati, Priyanka Das, Yogesh Prasad, Waghmare Rahul Gautam, "Design of A Reconfigurable Digital Modulator For High Bit Rate Data Transmission," 2020 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT), 2020, pp. 1-6, doi: 10.1109/CONECCT50063.2020.9198620.

[16]. Iqbal, JL Mazher, and T. Manikandan. "FPGA-Based Reconfigurable Architectures for DSP Computations." Advances in Smart System Technologies. Springer, Singapore 587-594.

[17]. Nayak, Seema, ManojNayak, and PankajPathak. "A Review on FPGA-Based Digital Filters for De-Noising ECG Signal." Sensor Network Methodologies for Smart Applications. IGI Global, 2020. 1-24.

[18]. Datta, Debarshi, and HimadriSekharDutta. "Efficient FPGA Implementation of FIR Filter Using Distributed Arithmetic." Energy Systems, Drives and Automations. Springer, Singapore, 2020. 151-160.

[19]. Ata. Kishore Kumar, G. Tirumala Vasu, Samreen Fiza, Afreen Kubra, "VLSI Implementation of Block Matching Motion Estimation Algorithm", International Journal of Advanced Science and Technology, Vol. 29, No. 04, (2020), pp. 11615 – 11625.

[20]. Samreen Fiza, Afreen Kubra, Ata. Kishore Kumar, G. Tirumala Vasu, "Bundle protocol implementation for satellite applications", Journal of Critical Reviews, Vol 06, Issue 06, 2019.