International Journal of Advances in Engineering and Management (IJAEM)
Volume 4, Issue 6 June 2022, pp: 954-960 www.ijaem.net ISSN: 2395-5252

Design of Fir Filter Using Yavadunam Sutra Algorithm Based High Speed Vedic Multiplier

Mrs.S.Manjubharathi[1], R.Kalaiselvi[2], N.Keerthana[3]

[1]*Asst.Prof Department of ECE & KRISHNASAMY COLLEGE OF ENGINEERING AND TECHNOLOGY*
[2,3]*Department of ECE & KRISHNASAMY COLLEGE OF ENGINEERING AND TECHNOLOGY*

Submitted: 01-06-2022 Revised: 10-06-2022 Accepted: 15-06-2022

ABSTRACT

Immeasurable analysis is used in the realm of communication and signal processing applications in today's world. The FIR filter is commonly used in signal processing applications to improve signal quality. The speed of the multiplier unit engaged in any processor's activity determines the system's performance. Because multipliers are essential components of the FIR filter scheme. Its performance has influenced the way the FIR filter system is implemented. In addition, because to enormous technological advancements, several methodologies such as arrays and Vedic algorithms have been developed to speed up multiplier computations. Due to the critical route present in partial products, the problem of speeding up operation and resource usage of hardware using all standard approaches must be optimised using the proposed method. The implementation and execution of a FIR Filter design employing the Yavadunam sutra algorithm are presented in this work. Various multiplier algorithms, such as the Yavadunam sutra method and array multipliers, are used to investigate the FIR filter. The FIR filter is used to analyze latency, and the suggested Yavadunam sutra algorithm is used to distribute and reduce area and power. The entire design is designed in VHDL for Cyclone II FPGA in Quartus 8.1. The FIR Filter based on the Yavadunam sutra algorithm has a short time delay of 14.246 ns and uses less logic cells than existing FIR Filters.

I. INTRODUCTION

Convolution and correlation are used in the FIR filtering process in DSP applications in general. Filtering is the process of extracting useful data from a signal. This is accomplished by conducting weighted summations in the filter's supplied input signals, such as digital video and audio signals, during transmission. After removing the undesirable signal known as noise, the signals with useful information were saved with only their essential information. A number of mathematical processes are applied to a sampled discrete-time signal in order to condition it for any changes. The convolution approach is used to calculate the filter's response for the provided signal k (n). Some multiplication methods employ adders to perform their operations. Two architectures are shown in this paper. For assessing their performance, sequential and parallel micro programmed FIR filters with Wallace tree and Vedic multipliers are used to investigate delay and critical path on ASIC implementation

FIR filters based on the carry-save-array multiplier (CSAM) and Wallace tree multiplier (WTM), also used in DSP to speed up operations and adaptive filter applications, are implemented with a resource allocation algorithm for higher performance and comparable power consumption with pre-emptive tasks.

To increase the system's performance, a FIR filter was constructed utilising the Yavadunam sutra algorithm. He structured the problem in such a way that it took less time to compute than the built-in MATLAB function and was faster than the other methods. The FIR system uses linear convolution as one of the primary approaches among the numerous algorithms. We provide a high-speed effective multiplier for the FIR Filter based on old Vedic mathematical equations in this research. So we compute discrete linear convolution using Yavadunam sutra algorithm approaches.

In terms of power, delay, and area, the proposed Yavadunam sutra method is determined to be substantially more efficient than other standard algorithms. The Yavadunam sutra algorithm denotes "proportionality," which consists of both a working and theoretical index (base) that is extremely suitable for values that are far away from the index, which is impossible to achieve with other techniques. Using a Vedic algorithm like the Yavadunam sutra algorithm and an array multiplier, these estimated results are taken into account for power, latency, and area. Yavadunam sutra, a Vedic algorithm, has been found to be both fast and effective. This strategy is used in the FIR design methodology to improve filtering functions and remove unwanted noise. As a result, system efficiency is increased with less resources and less processing.

II. VEDIC MULIPLIERS

A multiplier is a crucial component in the development of digital systems. In the literature, several algorithms for implementing rapid multipliers have been documented. Another approach for creating an efficient multiplier is to use the VEDIC multiplication algorithm. The VEDIC multiplier is discussed in this paper. In VEDIC mathematics, there are three methods for multiplying. Only one way is generic and may be used in all scenarios, while the other two are for specific situations. Urdhva-Triyakbhyam is the main Vedic multiplication algorithm. It is a universal multiplication formula that can be used in any multiplication situation. Vertically and crosswise is the literal translation.

The VEDIC multiplier is used to multiply two operands by multiplying them vertically and crosswise and then adding the results. The two operands 46 and 33 can be used to understand this multiplication procedure. The operands 33 and 46 can be written as 33 = (310 + 3) and 46 = (410 + 6 respectively. (36) + 403 + 306 + 3040) can be used to represent the multiplication (4633). Figure 1 depicts this multiplication.

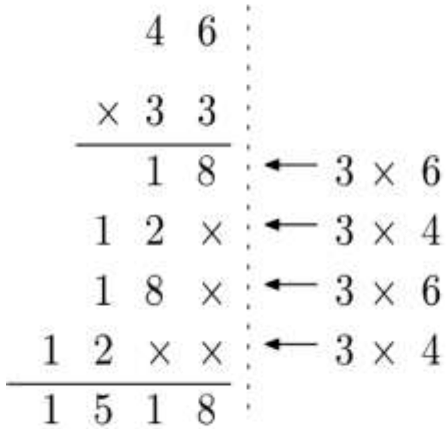


Figure 1: VEDIC Multiplication

Similar way, this multiplication algorithm can be adopted to implement faster binary multiplier. A 4-bit binary multiplication is shown below in Figure 2

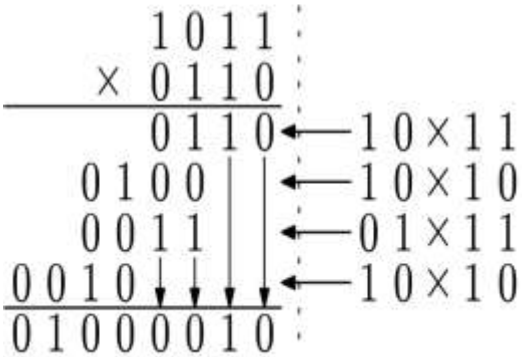


Figure 2: VEDIC multiplication for 4-bit data width.

The VEDIC multiplier is a good fast multiplicative algorithm option. In comparison to other techniques, VEDIC multipliers reduce both hardware and delay. Figure 3 depicts a 2-bit multiplier. Only two HA blocks and four AND gates are used in this circuit.

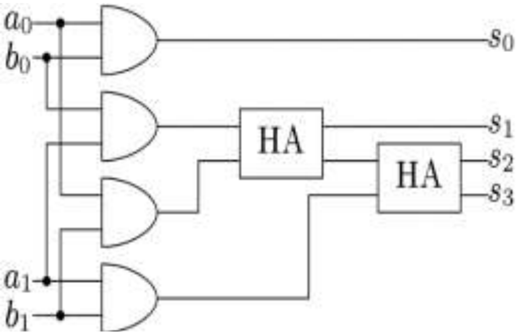


Figure 3: VEDIC multiplier for 2-bit data width

VEDIC multiplier for 4-bit data width is shown in Figure 3.4. This structure is achieved using four 2-bit multipliers. Here three Add blocks are used. These blocks can be implemented using high speed adders like conditional sum adder, carry look ahead adder or carry select adder as shown in the post fast addition. In this Figure4 the circles represent the concatenation block. For example, the two bits from the wire are connected to the output directly

.

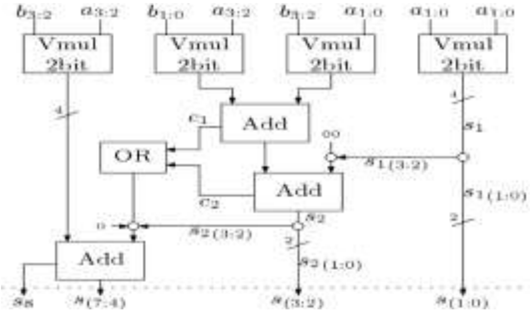


Fig 4: VEDIC multiplier for 4-bit data width.

III. PROPOSED SYSTEM

Among basic arithmetic operations, Multiplication demands more processing time and seek complex hardware. Multipliers with low latency and minimum power dissipation are preferred to design an optimized circuit. In our proposed, Yavadunam sutra algorithm is been employed for multiplying any two numbers. By using Yavadunam sutra algorithm based multiplier we have designed a Finite Impulse Response (FIR) filter. Multiplier based on the Yavadunam sutra algorithm provides reduced time delay and logic elements.

In this approach, we design a FIR filter by implementing a proposed Yavadunam sutra multiplier algorithm. The conventional direct form structure with 8-Tap FIR filteruse 8 numbers of multipliers, 7 numbers of additions for a single sample. The conventional FIR design uses delay elements, which adds delay to the signal by certain value by multiplying the values of the previous steps with the corresponding coefficient. The same filter can be reduced by using Yavadunam sutra multiplier for calculating product between inputs with the coefficient by replacing normal multiplication. The main advantage of this proposed method is the usage of fewer resources utilized to accomplish the same task by reducing area, delay, and power effectively.

YAVADUNAM SUTRASUTRA ALGORITHM

This segment conveys the principles and algorithm of Yavadunam sutra followed by some example. We can multiply any no. using Yavadunam Sutra whether it is in decimal or binary. Let's take example of two decimal numbers using the Yavadunam sutra.

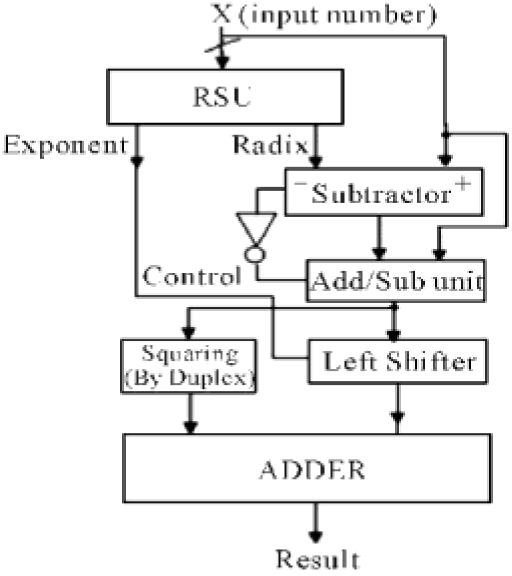


Fig 5 Algorithm Depiction

Take an example of 104 and 112. So X=104 and Y =112 The important observations are:

- Take the Nearest base value of 10N which is 100 in this example.
- Find the deficiency by sutra acting the number from its base value.

So, D1 = X - 10N = 104-100= 4 D2 = Y - 10N = 112-100=12

- Now take LHS = (X+Y) - 10N + Carry
- Thus, LHS= (104 + 112) -100 + 0 =116
- And RHS is taken by multiplying both the deficiency D1*D2. Thus, RHS = 4*12 = 48
- Now RHS= LHS&RHS which is 11648 in this case.

Thus, Answer of multiplying 104 and 112 is 11648.

This method is proved successful only for the numbers close to basevalues. Hence this multiplier has the objective for the achievement of effective area and power. The following section conveys the proposed Vedic multiplier using Yavadunam principles. As we have done multiplication in Yavadunam sutra we can also do it in binary but for binary inputs, base value will be in power of 2 (i.e. 2N), where N is no. of bits in the input. So using this method we can multiply any number of any ranges in quickly. Here in this case, deficiency is computed by taking two's compliment of two inputs taken in binary. Based on the input values there are three modes of operation. Let's take two binary numbers be X and Y. If both X and Y is greater than 2N-1 then it is Mode 1. If both X and Y are less than 2 N-1 then it is considered Mode 2. If one of the inputs is greater and one of the inputs is less (i.e. X>2 N-1 and Y 2 N-1) then it is considered as mixed called Mode3. Hence, there are three modes – mode1, mode2 and mode3. The Algorithm for Yavadunam multiplier, if inputs are binary contains N bit multiplier and Algorithm is as follows.

Algorithm

Let's take two binary digits be X and Y.

Step 1: Find the deficiency considered as D1and D2 which is computed by taking the two's complement of X and Y respectively.

Step2: Now we have to multiply both the Deficiencies D1 and D2 Using the N bit multiplier.

Step 3: The RHS of product XY is derived by taking the least N bits of product of deficiencies D1and D2.

Step 4: MSB N bits of the multiplier is taken from the RHS and it is further adder deficiencies that we derived in Step 1.

Step 5: The LHS of the product is computed by adding the deficiencies D1 and D2. Depending on the value of the input the adder output is takenas such or two's complement.

Step 6: The carry of the RHS is adder. The sign of the adder changes based on the given inputs. If both the inputs are greater than 2N-1, Then LHS = [2N – (D1+D2)] + carry bits from RHS If both the inputs are lesser than 2N-1, then LHS = (D1+D2) - carry bits from RHS. If both the inputs are mixed, then LHS = [2N – (D1+D2)] + carry bits from RHS.

IV. DESIGN CRITERIA

The FIR order simplifies the creation of FIR filters for add-on and modification operations and the primary objective of this strategy is to reduce the sum of additional and subtraction transactions [19]. The downside of this approach is that the coefficient cannot be modified until the filter design has been defined. Due to this, difficulties in FIR filter programming this method is not acceptable. Though the above mentioned difficulties can be solved by the use of Vedic Mathematics with previous techniques.

FIR FILTER

In FIR filter, length (N) of the filter is denoted as Tap. In this project 4-Tap Filter is designed and it consists of 3 adders, 3 delays and 4 multipliers. The multipliers and delay employed here are of 8 bit and adder is of 16 bit. Input X_n of the FIR filter is 8 bit and the output Y_n is 16 bit. Equation for FIR Filter is given by

$$Y[n] = \sum_{K=0}^{N-1} h[k].X[n-k]$$

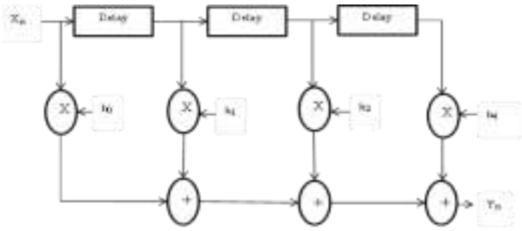


Fig 6 FIR Filter

RIPPLE CARRY ADDER DESIGN

In a ripple carry adder, carry out of each full adder is the carry in of the next most significant full adder. Carry bits get rippled into the next stage hence it is named as ripple carry adder.

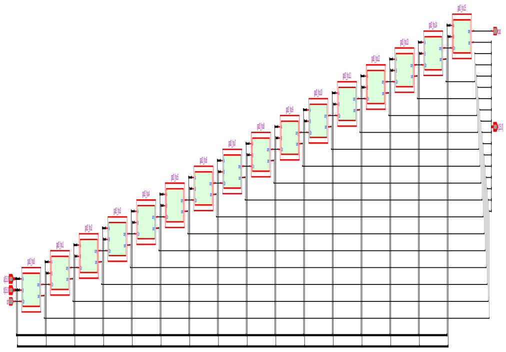


Fig 7 Ripples Carry Adder Design

YAVADUNAM SUTRASUTRA ALGORITHM BASED MULTIPLIER

Multiplier designed based on Yavadunam sutra algorithm. It consists of four 4-Bit multiplier and three 8-Bit Ripple carry adder.

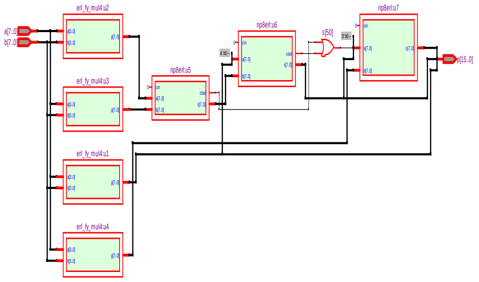


Fig 8. Multiplier

D FLIP-FLOP

Positive Edge triggered D Flip-flop is implemented with clock and reset, output is changed only during the positive edge of the clock

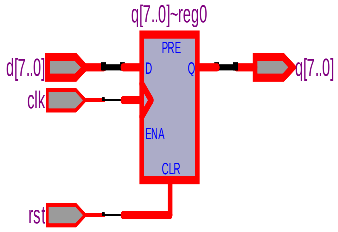


Fig 9. D Flip Flop

V. PROJECT IMPLEMENTATION

First, the input X_n is applied as the input to FIR filter it is then multiplied with the impulse response $h_{n.}$ in this project h0=0,h1=1,h2=2,h3=3 is predefined in the program and the multiplied valued is then added using 16 bit adder.

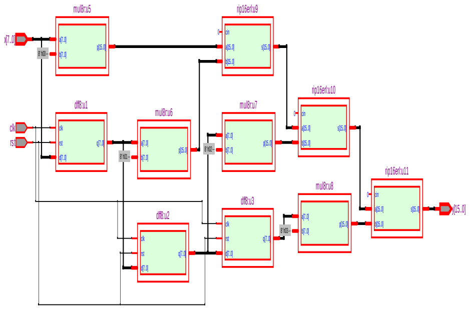


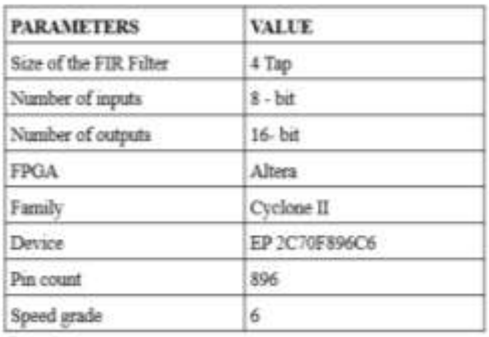
Fig 10 Project Design

Using linear convolution, the direct form structure of the FIR filer can be determined. All sequences k(n) and j(n) are finite in the FIR filter, and thus linear convolution is finite. The convolution will give the output sequence y(n) with an input x of length P with a filter of length Q. Vedic Mathematics contains several multiplication algorithms from which Yavadunam Sutra has been selected. Through using this Sutra, a modified Yavadunam Sutra is proposed to be used for the implementation of FIR filters. This approach further improves the Vedic multiplier by using strong adders. The approximate 8-Tap FIR filter is generated using a multiplier and adders for a similar sample of inputs.

To get the following specifications, a low-pass FIR filter must be constructed: Type of filter-low pass filter Sampling Frequency(s)-2000 HZ transition/Cut-off Frequency(fc)-460 HZ Length of filter (M) – 8 Ft = Cut off or transition frequency / sampling Frequency Hz. (1) The frequency of transformation (ft) is calculated based on Equation (1). Ft = 460/2001 = 0.23 Hz The rectangular frame truncates the normalized frequency response. Unless the generic transfer method weights the range, the coefficients inside the length of the window will be maintained and the period will be omitted. There are several windows, including a rectangular one, a barlet, a hanging one, a hamming one, a black maan, and a Kaiser with a rectangular frame. You can measure the right window feature using a rectangular window system.

The simulation parameters are shown in Table

Table 1 Simulation Parameters



PARAMETERS	VALUE
Size of the FIR Filter	4 Tap
Number of inputs	8 - bit
Number of outputs	16- bit
FPGA	Altera
Family	Cyclone II
Device	EP 2C70F896C6
Pin count	896
Speed grade	6

A digital low-power FIR filter is designed using the proposed improved Vedic multiplier. This method is comparable to the Yavadunam sutra's construction of FIR filters, but it simplifies adder functionality by incorporating new techniques into the multiplier process. For any complexity of filter arrangement, using the modified Vedic multiplier is a simple and efficient operation. The input value k(n) is multiplied by the vector coefficient j(n) and the correct resultant number is summed using the proposed multiplication technique.

Using the modified Yavadunam sutra, the proposed FIR Filter architecture based on the 8-Tap Direct Form-I can be simply translated to FPGA applications. The set number of inputs is represented by the vector k(n). The j(n) function represents the nth FIR filter coefficient for KSA, the two numbers to be used for this modified version of the Yavadunam sutra Vedic multiplier. To reduce the operation of the FIR filter, the modified Yavadunam sutra multiplier can be used to determine the filter coefficient j(n) and the input variable k(n). Instead of simply multiplying the larger inputs, the filter conducts multiplication operations by adding the appropriate amount acquired from W.B, resulting in a faster process. Furthermore, KSA uses incomplete commodities to increase speed. Finally, the unit multiplication and addition perform a bitwise addition with the required adjustments.

VI. SIMULATION RESULTS

OUTPUT OF FIR FILTER

Output of the FIR filter is verified by applying different values. For the input n=3. Output is obtained as,

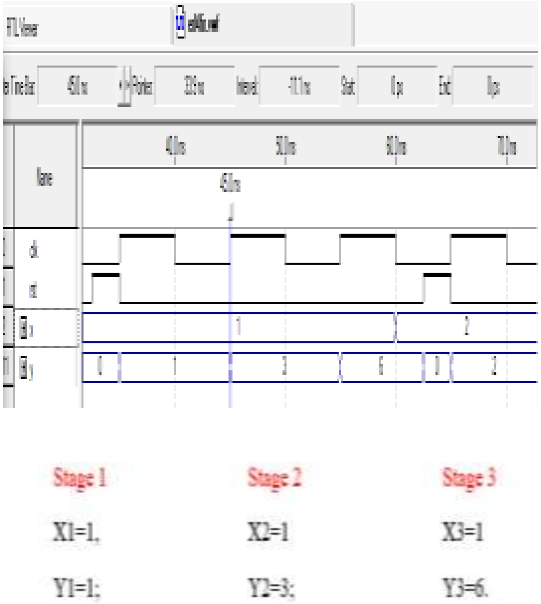
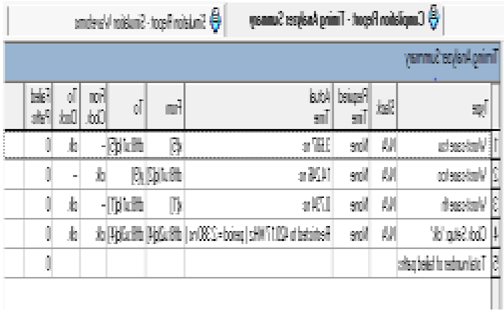


Fig 11 Output of FIR filter

TIMING ANALYSIS

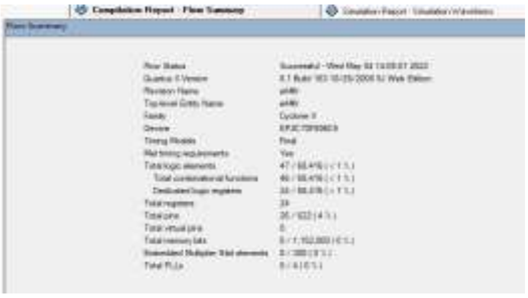

Compilation Report - Timing Analyzer Summary
Simulation Report - Simulation Waveforms

Timing Analyzer Summary

	Type	Slack	Required Time	Actual Time	From	To	From Clock	To Clock	Failed Paths
1	Worst-case tsu	N/A	None	[illegible] ns	[illegible]	[illegible]	--	clk	0
2	Worst-case tco	N/A	None	14.246 ns	[illegible]	[illegible]	clk	--	0
3	Worst-case th	N/A	None	[illegible] ns	[illegible]	[illegible]	--	clk	0
4	Clock Setup: 'clk'	N/A	None	Restricted to [illegible] MHz (period = [illegible] ns)	[illegible]	[illegible]	clk	clk	0
5	Total number of failed paths								0

Fig 12 Timing Analysis

SIMULATION REPORT


Compilation Report - Flow Summary
Simulation Report - Simulation Waveforms

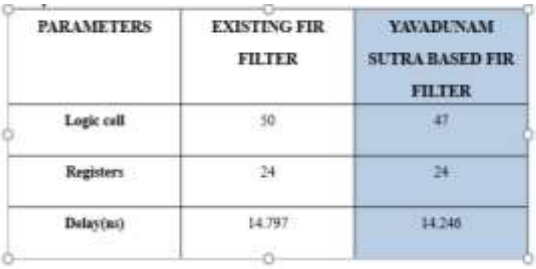
Flow Summary

Flow Status	Successful - Wed May 04 13:03:07 2022
Quartus II Version	8.1 Build 163 10/28/2008 SJ Web Edition
Revision Name	[illegible]
Top-level Entity Name	[illegible]
Family	Cyclone II
Device	EP2C70F896C6
Timing Models	Final
Met timing requirements	Yes
Total logic elements	47 / 68,416 (< 1 %)
Total combinational functions	46 / 68,416 (< 1 %)
Dedicated logic registers	24 / 68,416 (< 1 %)
Total registers	24
Total pins	26 / 622 (4 %)
Total virtual pins	0
Total memory bits	0 / 1,152,000 (0 %)
Embedded Multiplier 9-bit elements	0 / 300 (0 %)
Total PLLs	0 / 4 (0 %)

Fig 13 Simulation Report

Performance comparison of various fir filters

Simulation results shows that the Yavadunam sutra algorithm based FIR Filter has low time delay of 14.246ns and it consumes less number of logic cells when compared to the existing FIR filter structures.



PARAMETERS	EXISTING FIR FILTER	YAVADUNAM SUTRA BASED FIR FILTER
Logic cell	50	47
Registers	24	24
Delay(ns)	14.797	14.246

Fig 14 Performance Comparison of FIR filter

VII. CONCLUSION

In this project, a FIR filter is developed using the Yavadunam sutra squaring algorithm, and the results are compared to those of other FIR filter designs. The entire design is designed in VHDL for Cyclone II FPGA in Quartus 8.1. The FIR Filter based on the Yavadunam sutra algorithm has a low time delay of 14.246 ns and uses less logic cells than the conventional FIR Filter. Because the Vedic multiplier can conduct quick computing operations, it allows for more powerful DSP systems. The present architecture's processing performance is 20% faster than the conventional FIR filter, according to simulation data. The Vedic multiplier-based FIR filter also uses half as much energy as a regular array multiplier-based FIR filters. In addition, when compared to a standard array multiplier-based FIR filter, the suggested FIR filter takes around 30% less space.

FUTURE SCOPE

The algorithm can yet be tweaked to improve the speed of the suggested multiplier design. New engineering solutions and architectures may be developed in the future to address the demand for technological size as a result of the rapid growth of technology. Furthermore, as technology advances, researchers may concentrate on novel IC architectural techniques for low-power wireless medical and health-care applications. On the basis of current technological breakthroughs, there may be potential for establishing design methods at both the device and circuit block levels.

REFERENCES

[1]. As part of the production system for Journal of Physics: Conference Series, online versions of all Parhi, K. (2007). VLSI digital

signal processing systems: Design and implementation. New Delhi: Wiley.

[2]. Proakis, J. G. (2001). Digital signal processing: Principles algorithms and applications. New Delhi: Pearson Education India.

[3]. Chen, K.-H., &Chiueh, T.-D. (2006). A low-power digit-based reconfigurable FIR filter. IEEE Transactions on Circuits and Systems II, 53(8), 617–621.

[4]. Hartley, R. (1996). Sub extapion sharing in filters using canonic signed digit multipliers. IEEE Transactions on Circuits and Systems II, 43(10), 677–688.

[5]. Khoo, K.-Y., et al. (1996). A programmable FIR digital filter using CSD coefficients. IEEE Journal of Solid-State Circuits, 3(1), 869–874.

[6]. Maher, P. K., Chandrasekaran, S., & Amira, A. (2008). FPGA realization of FIR filters by efficient and flexible systolizationusing distributed arithmetic. IEEE Transaction on Signal Processing, 56(7), 3009–3017.

[7]. Shanthala, S., & Kulkarni, S. Y. (2009). VLSI design and implementation of low power MAC unit with block enabling technique.European Journal of Scientific Research, 30(4),620– 630.

[8]. Rahul Narasimhan, A. (2015). High speed multiply-accumulator coprocessor realized for digital filters. IEEE international conference on electrical, computer and communication technologies (ICECCT).

[9]. Fayed, A., &Bayoumi, M. A. (2002). A merged multiplier accumulator for high speed signal processing applications. In 2002 IEEE international conference on acoustics, speech, and signal processing.

[10]. Ghamkhari, S. F., &Ghaznavi-Ghoushchi, M. B. (2012). In 20th Iranian conference on electrical engineering (ICEE2012).

[11]. Mohanty, B. K., &Meher, P. K. (2016). A high-performance energy efficient architecture for FIR adaptive filter based on new distributed arithmetic formulation of block LMS algorithm. IEEE Transactions on VLSI Systems, 24(5).

[12]. Naga Jyothi, G., & Sri Devi, S. (2017). Distributed arithmetic architectures for FIR filters: A comparative review. IEEE International Conference on Wireless Communications, Signal Processing and Networking (Wisp NET).

[13]. Naga Jyothi, G., &Sridevi, S. (2019). High speed and low area decision feed-back equalizer with novel memory less distributed arithmetic filter. Multimedia Tools and Applications, 78(23), 32679–32693.

[14]. [14] Alam, SA, &Gustafsson, O, 2016, 'On the Implementation of Time Multiplexed Frequency Response Masking Filters', IEEE Transactions on Signal Processing, vol. 64, no.15, pp. 3933- 3944.

[15]. Sailaja, SN, 2016 'Comparative Analysis of FIR Filter for High Speed and Power Critical Applications', Comparative International Journal of VLSI System Design and Communication Systems, vol. 04, no. 09, pp. 735-740.

[16]. Jagadguru Swami Sri Bharati Krishna TirthajiMaharaj 2009 Vedic mathematics. Delhi: Agarwala V S Delhi Motilal Banarasidass Publishers Pvt. Ltd

[17]. Deepa A and Marimuthu C N 2018 High speed VLSI architecture for squaring binary numbers using Yavadunam Sutra and bit reduction technique. Int. J. Appl. Eng. Res. 13(6): 4471–4474.

[18]. Deepa A and Marimuthu C N 2017 A modified multiplier architecture using Yavadunam sutra: a squaring algorithm of Vedic mathematics. Perspectivasemciencia da Informacao 22(3): 322– 332.

[19]. Kyritsis, E &Pekmestzi, K, 2016, 'Hardware Efficient Fast FIR Filter Based on Karatsuba Algorithm', IEEE 5th International Conference on Modern Circuits and Systems Technologies (MOCAST), pp. 1-4.

[20]. Akila, M, Gowribala, C &Shaby, SM, 2016, 'Implementation of high speed Vedic multiplier using modified adder', IEEE Conference on Communication and signal processing (ICCSP), pp. 2244 -2248.