2581-4575

International Journal For Recent Developments in Science & Technology

Crossref

A Peer Reviewed Research Journal

FIR FILTER DESIGN USING HYBRID APPROXIMATE MULTIPLIER USING CPA AND BOOTH ALGORITHM

[1]BATCHU SAI SINDHU, [2]NGN PRASAD M.Tech,(Ph.D.)

[1]M.TECH VLSI, DEPT OF E.C.E, KAKINADA INSTITUTE OF ENGINEERING AND TECHNOLOGY, KORANGI,ANDHRAPRADESH, INDIA, 533461

[2]ASSOCIATE PROFESSOR, KAKINADA INSTITUTE OF ENGINEERING AND TECHNOLOGY, KORANGI,ANDHRAPRADESH, INDIA, 533461

ABSTRACT:

Digital signal processing (DSP) is one of the most powerful technologies which will shape the science, engineering and technology of the twenty-first century. Since 1970, revolutionary changes took place in the broad area of DSP which has made it an essential tool in many engineering applications. Digital filter is considered to be one of the most important components of almost every DSP sub-systems and therefore a number of extensive works had been carried out by researchers on the design of such filter. In order to meet the stringent requirements of filter specification, order of the designed filter is generally assumed to be very large and this leads to high power and area consumption during their implementation. As a matter of fact, design of hardware efficient digital filter has drawn enormous attention which needs to be addressed by various useful means. Finite Impulse Response filter based on hybrid Booth approx. multiplier is designed and compared with conventional filter, in which former reduces both area and delay. The design of proposed filter has been carried out using Radix-8 encoding scheme. The direct form structure of FIR filter has been used to design the proposed filter as this approach gives a better performance than common structures in terms of speed of operation. Here, the carry propagation adder is used in the FIR filter which further reduces the delay.

INTRODUCTION:

FINITE impulse response (FIR) digital filter is widely used in several digital signal processing applications such as speech processing, loud speaker equalization, echo cancellation, adaptive noise-cancellation, and various communication applications including software defined radio (SDR) etc. [1]. Many of these applications require FIR filters of large order to meet the stringent frequency specifications [2]– [4]. Very often these filters need to support high sampling rate for high-speed digital communication [5]. The number of multiplications and additions required for each filter output, however, increases linearly with the filter order.

Since, there is no redundant computation available in the FIR filter algorithm, real-time implementation of a large order FIR

International Journal For Recent Developments in Science & Technology
2581-4575
Crossref
A Peer Reviewed Research Journal

filter in a resource constrained environment is a challenging task. Filter coefficients very often remain constant and known a priori in signal processing applications. This feature has been utilized to reduce the complexity of realization of multiplications. Several designs have been suggested by various researchers for efficient realization of FIR filters (having fixed coefficients) using distributed arithmetic (DA) [22] and multiple constant multiplication (MCM) methods [10], [13]–[16].

DA-based designs use look-up-tables (LUTs) to store pre-computed results to reduce the computational complexity. The MCM method on the other hand reduces the number of additions required for the realization of multiplications by common subexpression sharing, when a given input is multiplied with a set of constants. The MCM scheme is more effective when a common operand is multiplied with more number of constants. Therefore, MCM scheme is suitable for the implementation of large order FIR filters with fixed coefficients.

But, MCM blocks can be formed only in the transpose form configuration of FIR filters. Block-processing method is popularly used to derive highthroughput hardware structures. Not only does it provide throughput-scalable design but also improves the area-delay efficiency. The derivation of block-based FIR structure is straight-forward when direct-from configuration is used [21] whereas the transpose-form configuration does not directly support block processing.

But, to take the computational advantage of the MCM, FIR filter is required to be realized by transpose form configuration. Apart from that, transpose form structures are inherently pipelined and suppose to offer higher operating frequency to support higher sampling rate. There are some applications such as SDR channelize where FIR filters need to be implemented in a reconfigurable hardware to support multi-standard wireless communication [6]. Several designs have been suggested during the last decade for efficient realization of reconfigurable FIR (RFIR) using general multipliers, and constant multiplication schemes [7]– [12], [17], [18].

A programmable multiply-accumulator based processor is proposed in [7] for FIR filtering. The area and power requirement of these architectures are significantly large and, therefore, they are not suitable of SDR channelizer. The structure of [9] is multiplier-based and uses poly-phase decomposition scheme. In [10], a reconfigurable FIR filter architecture using computation sharing vector-scaling technique of [8] has been proposed.

In [11], a programmable canonical signed digit (CSD) based architecture was proposed using Booth encoding to generate partial products and Wallace tree adder for addition of partial products. Chen et al [12] have proposed a CSD-based reconfigurable FIR filter where the non-zero CSD values are modified to reduce the precision of filter coefficients without significant impact on filter behavior. But, the reconfiguration

International Journal For Recent Developments in Science & Technology
2581-4575
Crossref
A Peer Reviewed Research Journal

overhead is significantly large and does not provide an area-delay efficient structure.

The architectures of [8]–[12] are more appropriate for lower order filters and they are not suitable for channel filters due to their large area complexity. Constant shift method (CSM) and programmable shift method (PSM) have been proposed in [16], [17] for RFIR filters specifically for SDR channelizer. Recently, Park et al. [18] have proposed an interesting distributed arithmetic (DA) based architecture for RFIR filter.

The existing multiplier-based structures use either direct-form configuration or transposeform configuration. But, the multiplier-less structures of [16], [17] use transpose-form configuration whereas the DA-based structure of [18] uses direct-form configuration. But, we do not find any specific block-based design for RFIR filter in the literature. A block-based RFIR structure can easily be derived using the scheme proposed in [20], [21]. But, we find that the block structure obtained from [20], [21] is not efficient for large filter lengths and variable filter coefficients such as SDR channelizer.

Therefore, the design methods proposed in [20], [21] are more suitable for 2-D FIR and BLMS adaptive filters. We explore the possibility of realization of block FIR filter in transpose-from configuration in order to take advantage of the MCM schemes and the inherent pipelining for area-delay efficient realization of large order FIR filters for both fixed and reconfigurable applications.

EXISITING DESIGN:

RECONFIGURABLE ARCHITECTURE:

Reconfigurable FIR filter architecture The architecture of block FIR filter for Reconfigurable applications is shown in the Fig.1 for block size L=4. The main blocks are one Register Unit (RU), one Coefficient Storage Unit (CSU), one Pipeline Adder Unit (PAU), and M number of Inner Product Units (IPUs).

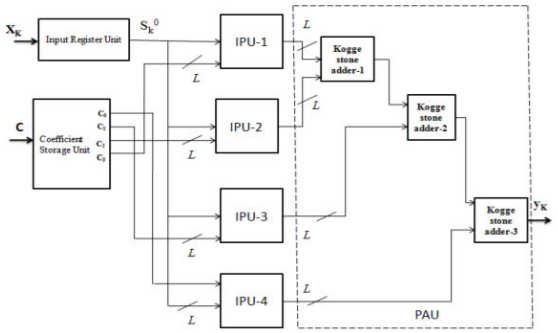

Fig.1. Block FIR filter for reconfigurable applications.

The Coefficient Storage Unit (CSU) is used to store the coefficients of all the filters. These coefficients are used in the reconfigurable applications. It has N Read Only Memory (ROM) Lookup Tables (LUTs) where N is the length of the filter (N=ML). The Register Unit (RU) is used for storing the input samples is shown in Fig.2. It contains (L-1) registers. During the K th cycle, the register unit accepts input sample XK and computes L rows of SK 0 in parallel. The outputs from the RU are given as inputs to M Inner Product Units.

International Journal For Recent Developments in Science & Technology
2581-4575
Crossref
A Peer Reviewed Research Journal

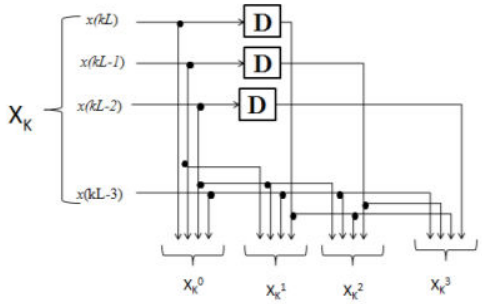

Fig.2. Internal structure of Register Unit (RU) for block size L=4

The Inner Product Unit (IPU) is used to perform a multiplication operation of SK 0 with the small weight vector cm is shown in Fig.3. The M Inner Products Units accepts L rows of SK 0 from the RU and M small weight vectors from the CSU. Each Inner Product Unit contains L number of Inner Product Cells (IPCs) which performs L inner product computations of L rows of SK 0 with coefficient vector cm and produces a block of L number of partial inner products. All the four IPUs work simultaneously and M blocks of a result are obtained.

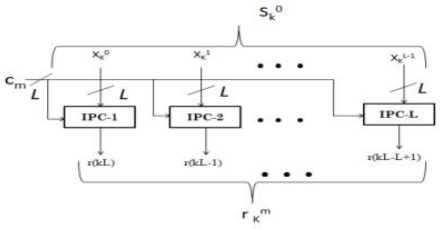

Fig.3. Internal structure of (m+1)th IPU.

The internal structure of (l +1) th IPC is shown in Fig.4. The Inner Product Cell (IPC) accepts (l+1)th row of Sk 0 and small weight vector cm and produce a partial result of inner product r(kL −l), for $0 \leq l \leq L - 1$. Each IPC consists of L multipliers and (L-1) number of adders.

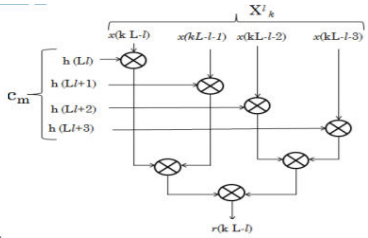

Fig.4. Internal structure of (l + 1)th IPC.

The Pipelined Adder Unit (PAU) receives partial products from all the M IPUs. Array of Kogge Stone Adder is used in PAU to add all the partial products is shown in Fig.5. KSA is one of the Carry Tree Adders or Parallel Prefix Adders. Kogge Stone Adders gains more importance among all the adders because of its high performance.

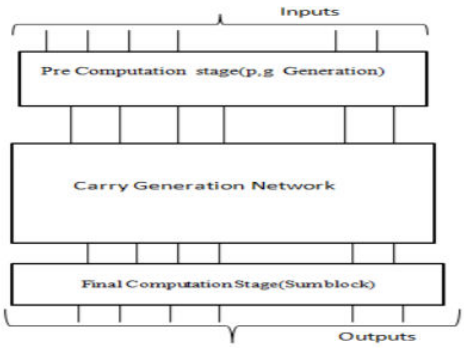

Fig.5. Different Stages in Kogge Stone Adder

KSA can be implemented in 3 stages, namely Pre-Computation Stage, Carry generation network and final computation stage. Generate and Propagate signals are computed in Pre-Computation Stage, corresponding to each pair of input bits A and B. The second stage compute carries corresponding to each bit. Execution of these operations is performed in parallel form, and they are partitioned into smaller pieces. Group generate and propagate bits which are computed in the first stage are

International Journal For Recent Developments in Science & Technology
2581-4575
Crossref
A Peer Reviewed Research Journal

used as intermediate signals in carry generation network. The final computation stage is common for all the adders of this family which gives the summation of input bits.

PROPOSED OBJECTIVES:

1. Optimize Error tolerant algorithm for approx. adder
2. FIR filter design using Approx. adder
3. Improved scenario for image Thresholding algorithm for image segmentation.

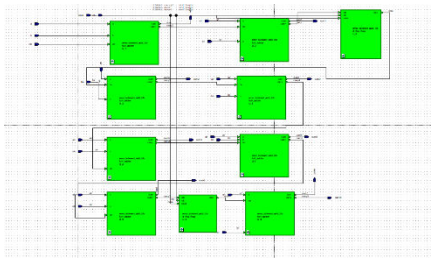

Figure 6: Block diagram for approx. adder.

ADVANTAGES:

1. Power, delay and area reduction analysis would improvise the existing FIR filter with Approx. Adder scenario to ensure the image processing applications.

DISADVANTAGES:

1. Unsigned and signed multiplication would provide difficult scenario for each set of Accurate and inaccurate

ADDER DESIGN AND ITS CIRCUIT DIAGRAM:

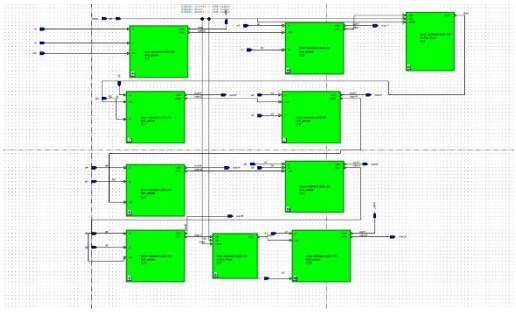

Figure 7: Hybrid approx. adder 8 bit design.

The current design approaches with case of the D-FF with a scenario of the approx. carry additions and saving at one of the logics in 8 bit design. This incorporates a leaser view of the current design utilizing the different adder model and other application scenario. As per the abstract point of view we have acknowledged the design for the FIR filter using proposed approx. adder for the additions and input scenario. As the adder design we have utilized a formulation on the current design estimating the power area and delay at each filter stage using hybrid approach of the kogge stone tree and Wallace tree on carry propagation adder.

Block Diagram for the proposed Architecture:

The current design model for the FIR filter based on the approx. adder would suggest a filter with 16 bit design and 14 tap filter. Here the tap representing the stage of the filter and for each such filter stages we have the input bit of length 16 bits.

International Journal For Recent Developments in Science & Technology
2581-4575
Crossref
A Peer Reviewed Research Journal

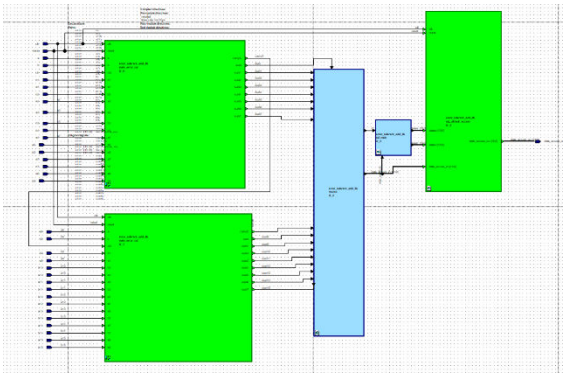

Figure 8: FIR FILTER using CPA and Booth algorithm.

The error tolerance of the approx. adder is controlled using FF at different position of the carry at each stage of 8bit addition resulting carry save and approximation using conditional approach for output observed at each addition. Design and its analysis of the proposed circuit consists of 16 bits of data from a0-a15 and b0-b15 and after addition of these data which are stored with 16 bit data one and data 2 resulting in threshold values for each data input. These values are utilized for the FIR filter for coefficients of 14 tap improving the delay and other performance characteristics.

RESULTS AND DISCUSSION

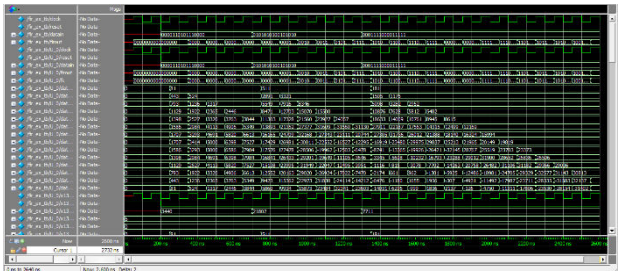

Figure 9: Correct Outputs for delay free condition:

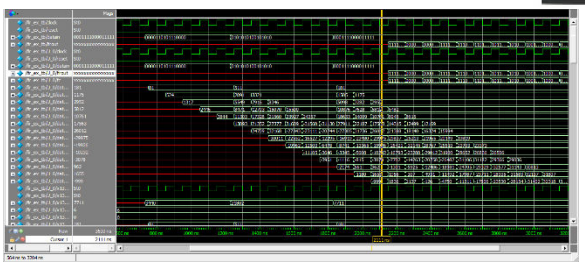

Figure 10a: Data1 consideration for the FIR filter design for 14 tap filter with 16 bit width

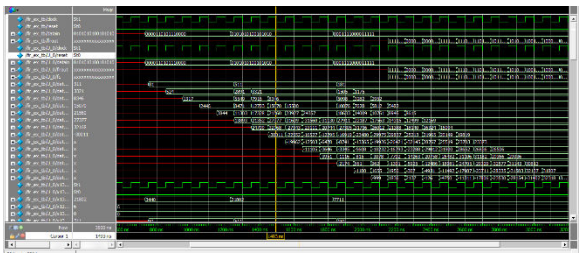

Figure 10b: Data2 consideration for the FIR filter design for 14 tap filter with 16 bit width

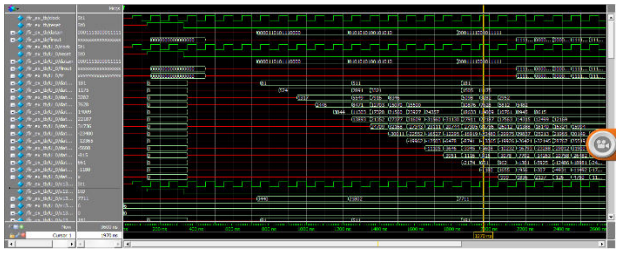

Figure 10c: Data 3 consideration for the FIR filter design for 14 tap filter with 16 bit width

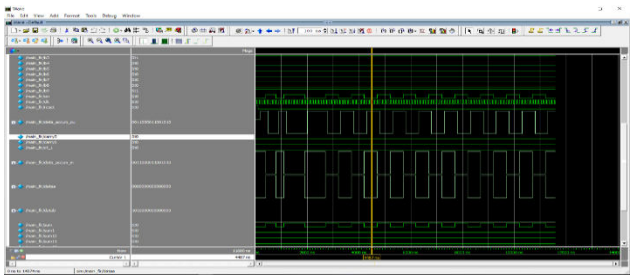

Figure 11: Representing the FIR FILTER ANALOG OUTPUT

International Journal For Recent Developments in Science & Technology
2581-4575
Crossref
A Peer Reviewed Research Journal

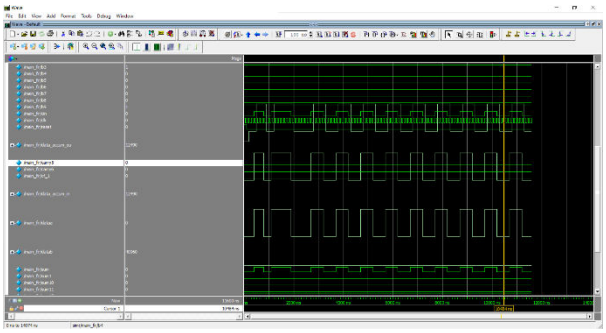

Figure 11a) Representing the threshold value at 45066.

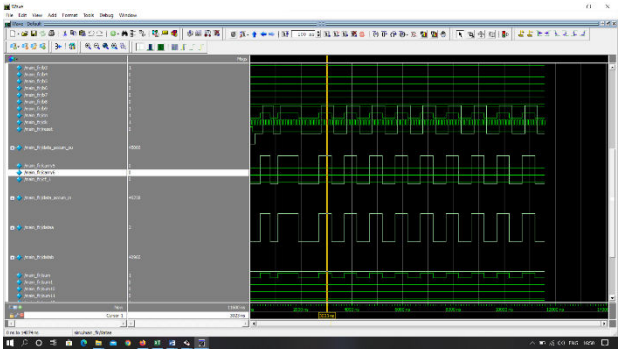

Figure 11b) Representing the threshold value at 12094.

Results discussion

1. From the above figure we have estimated and for different data streams with different simulated delay condition which would results in inappropriate data representation while operation.
2. Such delays can be overcome by representing the correct way of understanding rst condition and its impact with other stages so that each stage will be able to display the correct results as shown figure1.
3. Apart from the delays observed in FIR filter the proposed approx. adder with FIR filter design would improvise a constant square output with values ranging from 45066 to 12490. Hence resulting Thresholding approach for the FIR filter.

SYNTHESIS REPORT:

Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slices	130	89088	0%
Number of Slice Flip Flops	240	178176	0%
Number of 4 input LUTs	223	178176	0%
Number of bonded IOBs	34	960	3%
Number of GCLKs	1	32	3%
Number of DSP48s	14	96	14%

On-Chip Power Summary				
On-Chip	Power (mW)	Used	Available	Utilization (%)
Clocks	16.72	1	---	---
Logic	0.00	224	178176	0
Signals	0.00	606	---	---
IOs	0.00	34	960	4
DSPs	0.00	14	96	15
Quiescent	1344.95			
Total	1361.67			

Results Discussion:

Each such parameters such area power, and delay values are estimated according to the simulated results based on the correct analysis for the output verification where these parameter would play important role in shaping the design and its structures.

The below figure shows the tabulated results for existing and proposed design.

Note:

The current design is far superior from the existing design hence results in better performance as shown in the tabulated results.

CONCLUSION

FIR filters are extensively used in wired, wireless communications, video, audio processing and handheld devices are preferred because of their stability and linear phase properties. This paper presents a novel design methodology for an optimized FIR

International Journal For Recent Developments in Science & Technology
2581-4575
Crossref
A Peer Reviewed Research Journal

digital filters from software level to the hardware level.

The main goal is to encompass all the fields that are used in the efficient hardware realization of filters i.e. design method, selection of structure and the algorithm to reduce the arithmetic complexity of FIR filtering.

Theoretical and experimental result suggests that the power and area analysis for the current design would results in better and optimized latency for the structure implemented using direct-form structure approach is simpler, more robust to withstand the quantization errors, low cost and offers better performance than other common structures.

Proposed optimized filter implementation using an appropriate quantization scheme results in reducing arithmetic complexity, area and hardware resources. Comparison revealed that the optimized filter implementation is requiring 28% less hardware resources than the normal filter implementation.

REFERENCES

[1] J. G. Proakis and D. G. Manolakis, Digital Signal Processing: Principles, Algorithms and Applications. Upper Saddle River, NJ, USA:Prentice-Hall, 1996.

[2] J. Mitola, Software Radio Architecture: Object-Oriented Approaches to Wireless Systems Engineering. New York, NY, USA: Wiley, 2000.

[3] P. K. Meher, S. Chandrasekaran, and A. Amira, "FPGA realization of FIR filters by efficient and flexible systolization using distributed arithmetic," IEEE Trans. Signal Process., vol. 56, no. 7, pp. 3009–3017,Jul. 2008.

[4] P. K. Meher, "New approach to look-up-table design and memory based realization of FIR digital filter," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 57, no. 3, pp. 592–603, Mar. 2010.

[5] J. Park, W. Jeong, H. Mahmoodi-Meimand, Y. Wang, H. Choo, and K. Roy, "Computation sharing programmable FIR filter for low- power and high-performance applications," IEEE J. Solid State Circuits, vol. 39, no. 2, pp. 348–357, Feb. 2004.

[6] K.-H. Chen and T.-D. Chiueh, "A low-power digit-based reconfigurable FIR filter," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 53, no. 8, pp. 617–621, Aug. 2006.

[7] R. Mahesh and A. P. Vinod, "New reconfigurable architectures for implementing FIR filters with low complexity," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 29, no. 2, pp. 275–288, Feb. 2010.

[8] B. K. Mohanty and P. K. Meher, "A high-performance energy-efficient architecture for FIR adaptive filter based on new distributed arithmetic formulation of block LMS algorithm," IEEE Trans. Signal Process.,vol. 61, no. 4, pp. 921–932, Feb. 2013.

[9] B. K. Mohanty, P. K. Meher, S. Al-Maadeed, and A. Amira, "Memory footprint reduction for power-efficient realization of 2-D finiteimpulse response filters," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 61, no. 1,pp. 120–133, Jan. 2014.

[10] S. Y. Park and P. K. Meher, "Efficient FPGA and ASIC realizations of a DA-based

2581-4575
International Journal For Recent Developments in Science & Technology
Crossref
A Peer Reviewed Research Journal


reconfigurable FIR digital filter," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 61, no. 7, pp. 511–515, Jul. 2014.

[11] Ning Chi Huang,Szu Ying chen and kai Chaing Wu, "Sensor based Approximate Adder Design for Accelerating Error Tolerant and Deep learning applications"Design, Automation And Test in Europe (DATE 2019)978-3-9819263-2-3/DATE19/c 2019 EDAA