

Investigations on Low Power FIR Filter Design Using Arithmetic Strength Reduction Technique

¹R. Mahalakshmi and ²T. Sasilatha

¹Associate Professor, Saveetha Engineering College, Chennai-602105, India

²Head/ECE & Vice Principal Sri Sastha Institute of Engineering & Technology, Chennai-600123, India

Abstract: This paper presents an arithmetic strength reduction technique for the design of Low power FIR filter used for speech processing application. Arithmetic strength reduction is achieved by reducing the number of multipliers at the cost of adders. Fast FIR algorithm with symmetric coefficients arrangement and novel Carry save adder in accelerating the speed of final additions operations in parallel multipliers, are employed to reduce power consumption. The final chip of proposed 3-parallel 27-tap FIR filter with effective floor planning, placement and routing methodology has been presented. This paper results in two reports namely chip comparison report for 3- parallel 27- tap FIR filter and arithmetic efficiency report for various higher level tap length. The chip comparison report indicates that the proposed FIR filter with novel CSA structure consumes less dynamic and cell leakage power and also occupies reduced number of critical paths, nets and logical cells comparatively. The arithmetic efficiency report exhibits the reduction in multipliers and the improvement in arithmetic efficiency when the order of filter is increased. The experimental results show that proposed FIR filter achieves significant amount of hardware and power savings without compromising the filter performance.

Key words: Finite impulse response (FIR) filter • Novel carry save adder • Field-programmable gate array (FPGA) • Arithmetic efficiency

INTRODUCTION

The research on low power VLSI design becomes wide spread area in modern world due to the need of portable, less weight, optimum speed and low power electronic devices. In our research, arithmetic optimization is performed in digital filters by reducing the required number of multipliers using fast FIR algorithm and novel carry save adder to speed up the addition operation. Many researches focused on the optimization of filter coefficients to reduce the number of multiplications which leads to reduced power consumption [1-3].

Reconfigurable FIR filters have been proposed in the literature to achieve low power consumption [4-6]. Though the frequency based fast Fourier transform (FFT) algorithm reduces the number of computations in FIR filter, the computational complexity is always high due to signal processing operations. In fast linear convolution method, computational complexity is reduced by strength reduction technique [7]. In the fast convolution algorithms, the long convolution is decomposed into

small convolutions iteratively and Lagrange interpolation concept is applied to reduce the number of multiplications [8, 9]. In algorithmic strength reduction, the stronger multiplications are replaced by adders. Since optimizing the speed and area of the filter is the major design issue. We utilize the novel carry save adder to speed up the final addition operations in parallel multipliers.

Fast FIR Filter: The digital filters are employed in signal processing applications like noise removal, feature extraction and spectral analysis in recent times instead of their analog counterpart. FIR filters are preferred in wide variety of signal processing applications due to the stable and linear phase response characteristics. The working principle of FIR filter is similar to the convolution operation. The filter produces output by convolving the finite impulse response of the filter with the given input. The filter output is expressed as;

$$y(n) = \sum_{k=0}^{M-1} h(k).x(n-k) \quad (1)$$

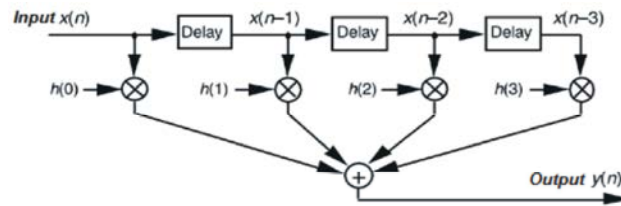


Fig. 1: Third order FIR filter

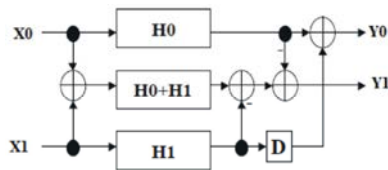


Fig. 2: Two parallel fast FIR structure

where $y(n)$, $x(n)$ and $h(k)$ represent the output, input and impulse response of the filter respectively. M is the length of the filter and $M-1$ is the order of the filter. FIR filter with order 3 can be implemented using three delay elements, which is depicted in Figure 1.

The polyphase decomposition technique divides the filter into number of small size parallel FIR filters by rearranging the filter coefficients for complexity reduction and better performance, but the number of computations is not reduced [10]. To improve the performance of the parallel FIR filter arrangement, fast FIR algorithms have been introduced [11]. The filter output is expressed in the form of polynomial multiplication of input and impulse response as;

$$Y_0 + z^{-1}Y_1 = (H_0 + H_1z^{-1})(X_0 + X_1z^{-1}) \tag{2}$$

$$= H_0X_0 + z^{-1}(H_0X_1 + H_1X_0) + z^{-2}H_1X_1$$

Equation (2) is modified using fast FIR algorithm (FFA) as;

$$Y_0 = H_0X_0 + z^{-2}H_1X_1 \tag{3}$$

$$Y_1 = (H_0 + H_1)(X_0 + X_1) - H_0X_0 - H_1X_1 \tag{4}$$

From the equation (3) and (4), fast FIR structure can be implemented using 3 multipliers, 4 adders and a delay element. The two parallel fast FIR structure is shown in Figure 2. It is significant to note that the fast FIR structure requires 3 multiplications which is one less compared to the conventional parallel structure.

Signed Digit Multiplier with Novel CSA: Figure 3 depicts the signed digit multiplier which consists of partial

product selection unit, partial product generator and signed digit adder. The generated partial products are added using signed digit adder. This adder circuit is designed using the novel carry save addition.

The novel 16 bit CSA shown in Fig. 4. consists of 21 half adders and 11 full adders in order to reduce the power consumption, however the full adder having default zero input can be replaced by half adder, so that considerable amount of power is saved in the novel CSA [16].

Proposed FIR Filter Design: This work focuses on the FIR filter design using fast FIR algorithm discussed in the section 2 and novel carry save addition discussed in section 3. The Figure 5 depicts the main steps involved in the proposed fast FIR filter design. The input samples are applied into the filter which is arranged using fast FIR algorithm with symmetric coefficient combination. Delayed input samples are produced from the given input sample using array of delay elements. Number of multipliers are reduced at expense of adders using filter coefficients with symmetry arrangement. Once the number of multipliers and adders are finalized, input samples and respective filter coefficients are multiplied using booth multiplier based on the partial product selection and generation. Partial products are combined using novel carry save adder to reduce the required elements. The multiplied values are combined using adders to produce the final output of the filter.

Fast FIR algorithms reduce the number of multiplications based on the rearrangement of input samples and impulse response coefficients. In our work, the number of multiplication is further reduced by combining symmetric impulse response coefficients. The filter coefficients need to be symmetric to satisfy the linear phase characteristics. Polyphase decomposition is applied to separate the whole block into sub filter blocks to utilize the coefficient symmetry. Half the number of multiplications is reduced in this arrangement because the sub filter can be reused. Sub filter blocks H_0 and H_1 are combined with input samples to produce the output of the filter Y_0 and Y_1 using equations (5) and (6). Input, delay, adder and sub filter blocks are arranged as shown in Figure 6.

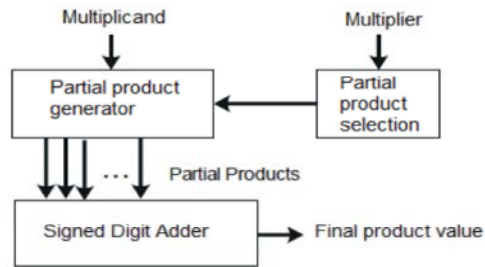


Fig. 3: Signed digit multiplier

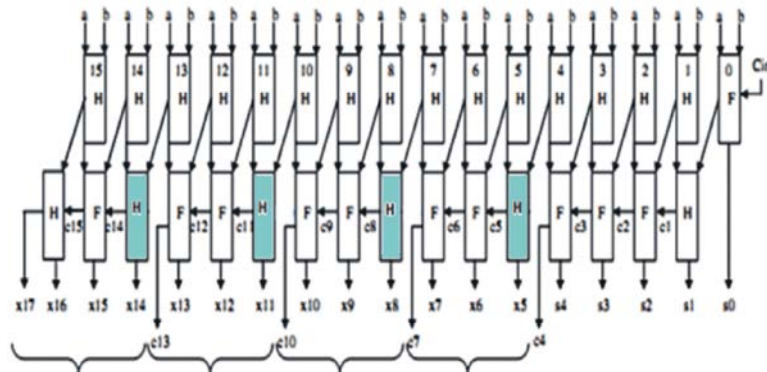


Fig. 4: Block Diagram of 16 bit novel carry save adder

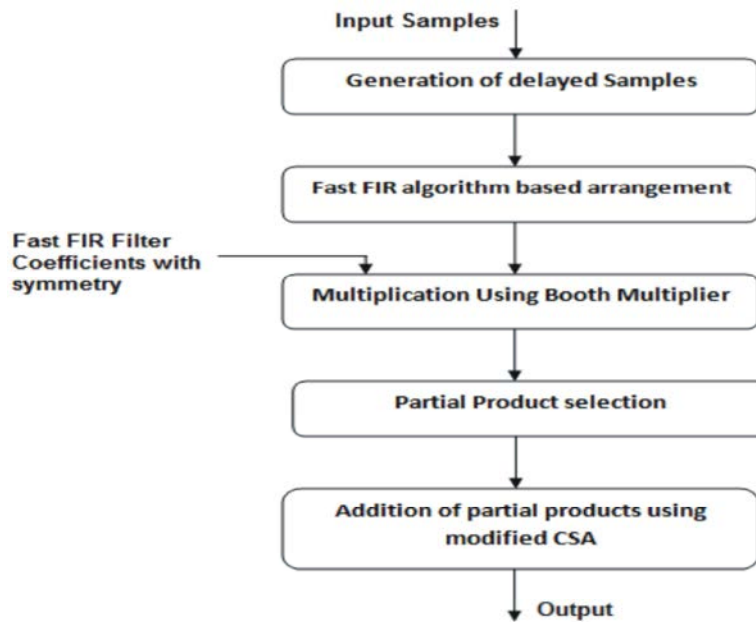


Fig. 5: Proposed fast FIR filter design flow

$$Y_0 = \frac{1}{2} \{ (H_0 + H_1)(X_0 + X_1) + (H_0 - H_1)(X_0 - X_1) \} - H_1 X_1 + z^{-2} H_1 X_1 \quad (5)$$

$$Y_1 = \frac{1}{2} \{ [(H_0 + H_1)(X_0 + X_1)] - [(H_0 - H_1)(X_0 - X_1)] \} \quad (6)$$

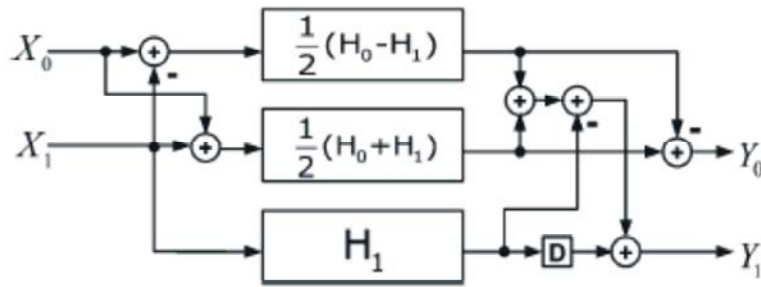


Fig. 6: Proposed fast FIR algorithm

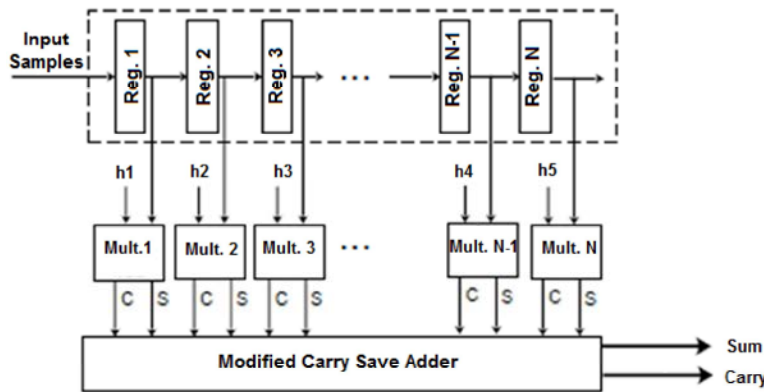


Fig. 7: Filter coefficient multiplication and modified carry save addition

Modified Booth algorithm is used to construct multiplier circuits and the partial products are added using novel CSA. Input samples $x(n), x(n-1), x(n-2), \dots, x(n-N+1)$ are combined with the respective filter coefficients $x(n), x(n-1), x(n-2), \dots, x(n-N+1)$ to perform multiplication using modified Booth algorithm. Figure 7 depicts the above procedure using register, multiplier and modified carry save adder elements.

Comparative Analysis and Discussion: The proposed design has been modeled using Verilog HDL and verified using test benches with various input combinations. The Hardware Description Language (HDL) code is synthesized using Synopsys Design Compiler targeting 65-nanometer Taiwan Semiconductor Manufacturing Company library and target technology (TSMC). Before implementing the design, it is simulated using modelsim for functionality verification using test inputs which is shown in Figure 8.

The design is synthesized using Xilinx ISE for visualizing register transfer level (RTL) schematic and to get the information about the LUT utilization and logic cells used. Synthesis is a three-phase process where it starts with translating the RTL code to the gate level net list. Figure 9 shows the RTL schematic obtained using synthesis process.

The synthesized net list with the constraints file is taken into the back end design flow or physical design flow. During this phase, floor planning, placement, clock tree synthesis and routing is done on the design to obtain the GDSII file, which can be sent for fabrication [14]. Figure 10(a) and 10(b) shows the floor-planned view of the design. 130 I/O cells are placed on the perimeter, the cell utilization is considered to be 80%. The power supply for I/O cells and core area are separated, as both require different power supplies. Five metal layers are used for routing the entire design, power supply and ground connects are on the top layer. Floor planning is carried out using Jupiter XT tool from Synopsys.

Routing is two step process, first global routing is carried out and then detailed routing is performed. Figure 11 shows the routed design. Routing ensures that all the cells are interconnected as per the net list obtained during synthesis. Besides interconnection, it checks for whether the timing is met or not.

Figure 12 gives an idea about the final chip which can be fabricated using CMOS process technologies like twin tub and silicon on insulator. The number of cells, nets, I/O ports and area are also visualized in the final chip. The power savings are high compared to the conventional approach.

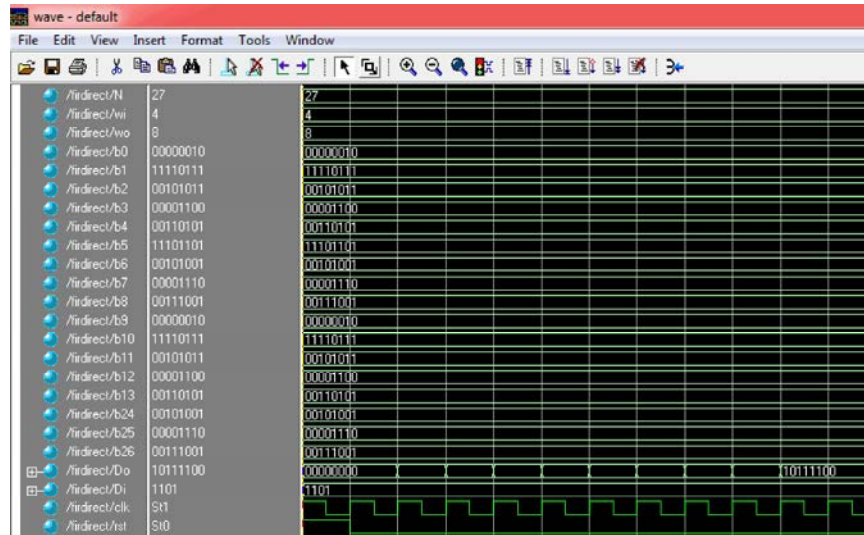


Fig. 8: Simulation of fast FIR filter with length $M=27$

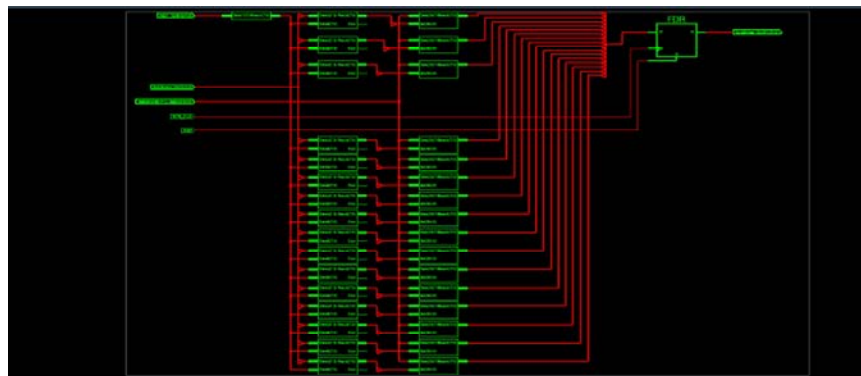


Fig. 9: RTL schematic obtained using synthesis

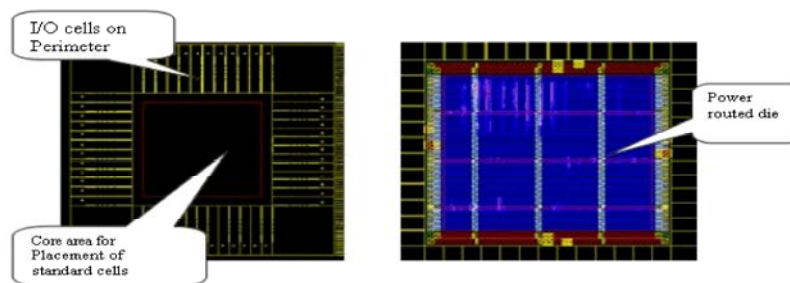


Fig. 10: Floor planned die (a) I/O cell placed (b) With power routing

The reduction in multipliers and the efficiency of the proposed 3-parallel FIR filter design is listed in Table 1. From the Table, we observe that while increasing the order of the filter, number of multipliers is reduced and the arithmetic efficiency is improved to the significant level.

The final chip of proposed fast FIR filter has 50 I/O ports, 180 nets and 325364 logic cells. The filter

performance is improved and total dynamic power and leakage power are reduced while comparing with the existing fast FIR filter design.

In general, fast FIR algorithms reduce the number of multiplication in accordance with the filter parallel level organization. For example, existing fast FIR filter of order 81 with parallel level $L=5$ requires 94 multiplication while the same filter with level $L=3$ requires 106 multiplications.

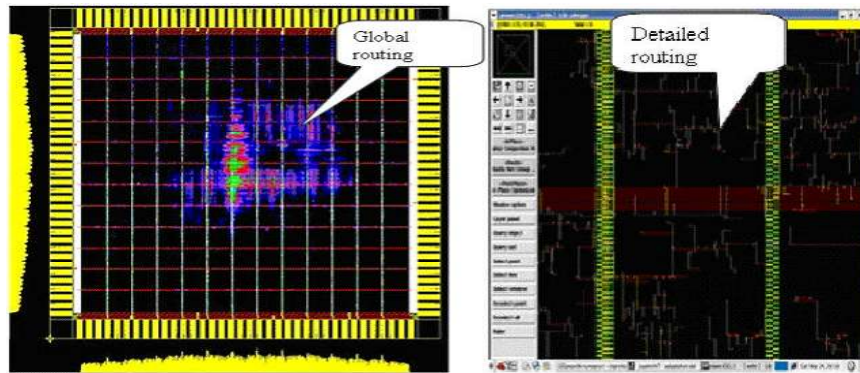


Fig. 11: Routed design (a) After global routing (b) Detailed routing

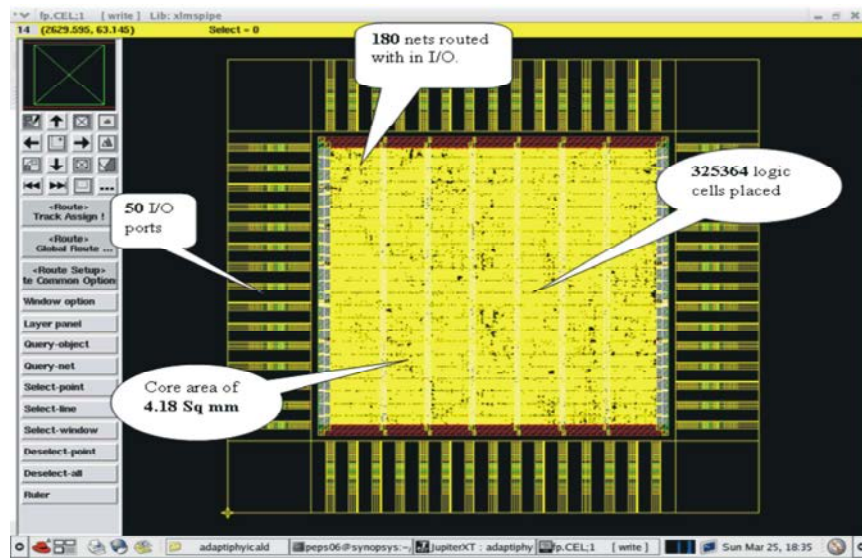


Fig. 12: Final chip of the proposed FIR filter

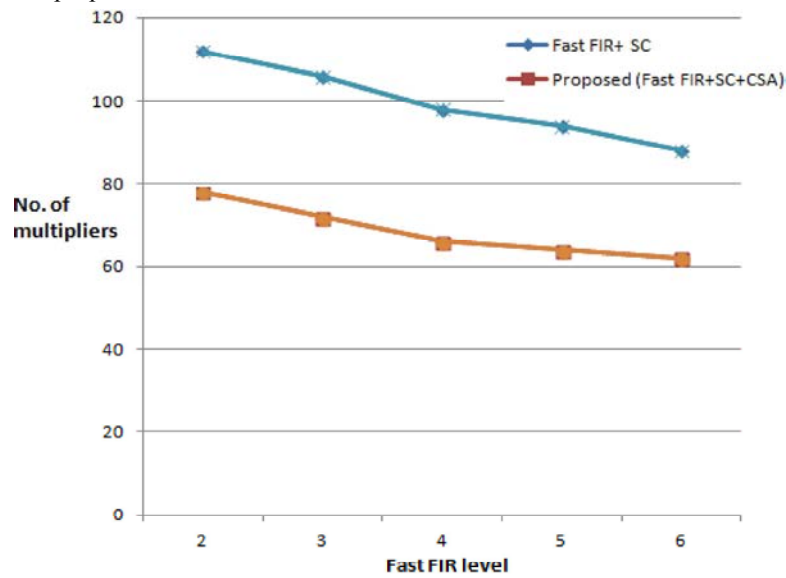


Fig. 13: Comparison graph based on parallel level (or) block size

Table 1: Multipliers and Adders Used for 'L=3' Parallel Fir Filter

Length	Method	Number of Multipliers	Reduced multipliers	Number of Adders	Increased Adders
27-tap	Fast FIR+ symmetric convolution	34	10 (29.4%)	17	13
	Proposed Fast FIR+modified CSA	24		30	
81-tap	Fast FIR+ symmetric convolution	106	34 (32.1%)	28	14
	Proposed Fast FIR+modified CSA	72		42	
147-tap	Fast FIR+ symmetric convolution	164	62 (37.8%)	42	22
	Proposed Fast FIR+modified CSA	102		64	

Table 2: Chip Report Comparison for 3-parallel 27-tap Filter

Parameters	Fast FIR+ symmetric convolution	Proposed Fast FIR+modified CSA
Slack	0.29	0.17
Maximum critical paths	456	424
Number of ports	50	50
Number of nets	180	180
Total logic cells	348394	325364
Total Dynamic Power (mW)	64	45
Cell Leakage Power (μ W)	164	114

CONCLUSION

This work proposed a new fast FIR filter based on fast FIR algorithm with slightly modified coefficient arrangement and modified carry save adder. It mainly reduced the number of multiplications at the cost of extra adders. The efficiency of the arithmetic strength reduction is high while dealing with increased order FIR filters. The multiplier reduction for filter order 27, 81 and 147 are 29.4%, 32.1% and 37.8% respectively. The addition operation and partial product addition are performed using modified carry save adders to reduce the power consumption. The total dynamic power and cell leakage power are reduced by 29.8% and 31.5% respectively.

REFERENCES

- Samueli, H., 1989. An improved search algorithm for the design of multiplier less FIR filter with powers-of-two coefficients, *IEEE Trans. Circuits Syst.*, 36(7): 1044–1047.
- Hartley, R.I., 1996. Sub expression sharing in filters using canonical signed digit multipliers, *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, 43(10): 677-688.
- Gustafsson, O., 2007. A difference based adder graph heuristic for multiple constant multiplication problems, in *Proc. IEEE Int. Symp. Circuits Syst.*, pp: 1097-1100.
- Chen, K.H. and T.D. Chiueh, 2006. A low-power digit-based reconfigurable FIR filter, *IEEE Trans. Circuits Syst. II, Exp. Briefs*, 53(8): 617-621.
- Mahesh, R. and A.P. Vinod, 2010. New reconfigurable architectures for implementing filters with low complexity, *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, 29(2): 275-288.
- Yu, Z., M.L. Yu, K. Azadet and A.N. Wilson, Jr, 2001. A low power FIR filter design technique using dynamic reduced signal representation, in *Proc. Int. Symp. VLSI Tech., Syst., Appl.*, pp: 113-116.
- Cheng, C. and K.K. Parhi, 2004. Hardware efficient fast parallel FIR filter structures based on iterated short convolution, *IEEE Trans. Circuits Syst. I, Reg. Papers*, 51(8): 1492-1500.
- Cheng, C. and K.K. Parhi, 2005. Further complexity reduction of parallel FIR filters, in *Proc. IEEE ISCAS*, May 2005, 2: 1835-1838.
- Cheng, C. and K.K. Parhi, 2007. Low-cost parallel FIR structures with 2-stage parallelism, *IEEE Trans. Circuits Syst. I, Reg. Papers*, 54(2): 280-290.
- Chung, J.G. and K.K. Parhi, 2002. Frequency-spectrum-based low-area low power parallel FIR filter design, *EURASIP J. Appl. Signal Process.*, 9: 444-453.
- Tsao, Y.C. and K. Choi, 2010. Area-efficient parallel FIR digital filter structures for symmetric convolutions based on fast FIR algorithm, *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, 20(2): 366-371.
- Ramkumar, B., Harish M. Kittur, P. Mahesh Kannan, 2010. ASIC Implementation of Modified Faster Carry Save Adder, *European Journal of Scientific Research* ISSN 1450-216X, 42(1): 53-58.

13. Geoff Bostock 1996. 'FPGA's and Programmable LSI: A Designer's Handbook, Butterworth-Heinemann, 1st Edition.
14. Khosrow Golshan, 2007. Physical Design Essentials: An ASIC Design Implementation Perspective, Springer Publishers.
15. Mahalakshmi, R. and T. Sasilatha, 2013. A Power efficient carry save adder and modified carry save adder using CMOS Technology, 978-1-4799-1597-2/13/\$31.00 ©2013 IEEE.
16. Mahalakshmi., R. and T. Sasilatha, 2014. Low Power Modified Carry Save Adder For Reconfigurable Digital Filters, International Journal of Applied Engineering Research, ISSN 0973-4562, 9(24): 24601-24609.