

## DESIGN OF AREA EFFICIENT PARALLEL FIR DIGITAL FILTERS USING FFA ALGORITHM

S.Priya<sup>\*</sup>

V.Gopi<sup>\*\*</sup>

### ABSTRACT-

In recent days, filters with large lengths are started to use. So parallel FIR filter is essential, especially when the length of the filter is very large. Parallel FIR digital filters can be used either for high-speed or low-power applications whereas traditional parallel filter implementations cause linear increase in the hardware cost with respect to block size. Multipliers are the major portions in hardware consumption for the parallel FIR filter implementation. The proposed new structures exploit the nature of symmetric coefficient and further reduce the number of multipliers required at the expense of additional adders. Replacing adders with multipliers is advantageous only because adders weigh less than multipliers in terms of silicon area. In addition, the increase in adders stay fixed, not increasing along with the length of the filter, whereas reduced multipliers increase along with the length of the filter. Overall, the proposed parallel FIR structures can lead to significant hardware savings for symmetric convolutions from the existing FFA FIR filter.

*Keywords*— Digital signal processing (DSP), fast FIR algorithms (FFAs), parallel FIR, symmetric convolution, very large scale integration (VLSI).

<sup>\*</sup> PG Student, Applied Electronics, Department of Electronics and Communication Engineering,

<sup>\*\*</sup> HOD, Department of Electronics and Communication Engineering,

<sup>\* & \*\*</sup> PSN College of Engineering and Technology, Melathediyoor

## I. INTRODUCTION

Nowadays FIR filters are the most popular type of filters implemented in software and are widely used in various DSP applications ranging from wireless communications to video and image processing. Some applications need the FIR filter to operate at high frequencies such as video processing, whereas some other applications request high throughput with a low-power circuit such as multiple input multiple output (MIMO) systems used in cellular wireless communication. Furthermore when narrow transition-band characteristics are required, the much higher order in the FIR filter is unavoidable. There are two techniques used in Digital Signal Processing. They are parallel and pipelining processing. Both of the techniques are used to reduce the power consumption. But we prefer parallel processing over pipelining processing. Pipelining shortens the critical path by interleaving pipelining latches along the data path, at the price of increasing the number of latches and the system latency, whereas parallel processing increase the sampling rate by replicating hardware so that multiple inputs can be processed in parallel and multiple outputs are generated at the same time. In both ht techniques, sampling speed does not increase. In this paper, parallel processing in the digital FIR filter will be discussed. Many algorithms are known to reduce the arithmetic complexity of FIR filtering. In this paper we have presented a new class of algorithms for FIR filtering in which multiplications are replaced by adders. We provide a new parallel FIR filter structures based on Fast FIR algorithm (FFA). It can reduce the amount of multiplications in the sub-filter section. This algorithm is very efficient in reducing the hardware cost.

## II. PREVIOUS RESEARCH

There have been a few papers proposing ways to reduce the complexity of parallel FIR filter. One important research study that demonstrates this was conducted by Chung and Keshab.K.Parhi, who found that parallel FIR digital filters can be used either for high speed or low power(with reduced supply voltage) applications. Traditional parallel filter implementations cause linear increase in hardware cost with respect to block size. The paper makes two contributions. First, the filter spectrum characteristics are exploited to select the best fast filter structures. Second, a novel block filter quantization algorithm is introduced using filter benchmarks. It is shown that the use of appropriate fast FIR filter structures and the proposed quantization scheme can result in reduction in the number of binary adders up to 20%. In a research article by Chao Cheng and Keshab.K.Parhi, when it comes to symmetric convolution, the symmetry of coefficients has not been taken into consideration for the design of structures yet, which can lead to significant saving in hardware cost. This paper presents an Iterated Short Convolution (ISC) algorithm based on mixed radix algorithm and fast convolution algorithm. A similar study was conducted by Chao Cheng and Keshab.K.Parhi, who argued low cost parallel FIR filter structures with 2-stage parallelism. This paper proposes a new parallel FIR filter structure with less hardware complexity. The sub filters in the previous parallel FIR structures are replaced by a second stage parallel FIR filter. The proposed 2-stage parallel FIR filter structures can efficiently reduce the number of required multiplications and additions at the expense of delay elements. The proposed structure can save 67% multiplications, 30% additions; compared to previous parallel FIR filters. In a research article by Yan Sun and Min Sik Kim they present an approach to implement a high performance 8-tap

digital FIR filter using the Logarithmic Number System. In the past, FIR filters were implemented by a conventional number system; their speed was limited because of the multiply accumulate operations. LNS algorithm allows a simple implementation of multiplication using a fixed point adder. But the serious demerit of LNS algorithm, conversions to and from conventional number representations, is effectively overcome by pipelining to reduce the delay and complexity of filter.

### III. FAST FIR ALGORITHM (FFA)

Consider an N-tap FIR filter which can be expressed in the general form as

$$y(n) = \sum_{i=0}^{N-1} h(i)x(n-i), \quad n=0,1,2,\dots,\infty \quad (1)$$

where  $\{x(n)\}$  is an infinite-length input sequence and  $h(i)$  are the length- N FIR filter coefficients. Then, the traditional parallel FIR filter can be derived using polyphase decomposition

$$\begin{aligned} \sum_{p=0}^{L-1} Y_p(Z^L)Z^{-p} &= \sum_{q=0}^{L-1} X_q(Z^L)Z^{-q} \\ &= \sum_{p=0}^{L-1} H_r(Z^L)Z^{-r} \end{aligned} \quad (2)$$

Where  $X_q = \sum_{k=0}^{\infty} Z^{-k} x(Lk+q),$

$$H_r = \sum_{k=0}^{\infty} Z^{-k} x(Lk+r),$$

$$Y_p = \sum_{k=0}^{\infty} Z^{-k} x(Lk+p), \text{ for } p, q, r = 0,1,2,\dots,L-1. \text{ From this FIR}$$

filtering equation, it shows that the traditional FIR filter will require  $L^2$  -FIR subfilter blocks of length for implementation.

#### 3.1. 2X2 FFA (L=2)

According to (2), a two-parallel FIR filter can be expressed as ,

$$\begin{aligned}
 Y_0 + Z^{-1}Y_1 &= (H_0 + Z^{-1}H_1)(X_0 + Z^{-1}X_1) \\
 &= H_0X_0 + Z^{-1}(H_0X_1 + H_1X_0) + Z^{-2}H_1X_1 \quad (3)
 \end{aligned}$$

Implying that

$$Y_0 = H_0X_0 + Z^{-2}H_1X_1 \quad (4)$$

$$Y_1 = H_0X_1 + H_1X_0.$$

Equation (4) shows the traditional two-parallel filter structure, which will require four length-  $N/2$  FIR subfilter blocks, two post processing adders, and totally  $2N$  multipliers and  $(2N-2)$  adders. However, (4) can be written as

$$Y_0 = H_0X_0 + Z^{-1}H_1X_1$$

$$Y_0 = (H_0 + H_1)(X_0 + X_1) - H_0X_0 - H_1X_1 \quad (5)$$

The implementation of (5) computes three FIR sub filter blocks of length  $N/2$ , one pre-processing and three post processing adders, and  $3N/2$  multipliers and  $3(N/2-1)+4$  adders. This structure reduces approximately one fourth over the traditional 2-parallel filter hardware cost from

(4). Direct implementation of (5) is shown in Fig.1

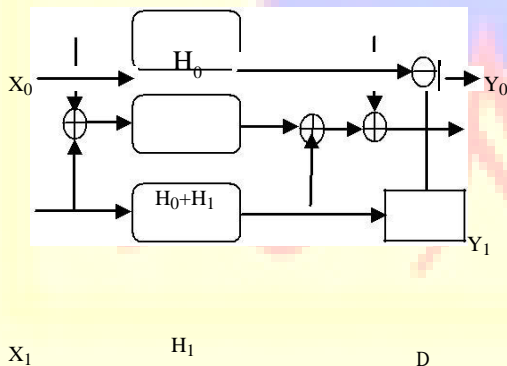


Fig. 1 Two-parallel FIR filter implementation using FFA.

The implementation of (5) will require three FIR sub filter blocks of length  $N/2$ , one pre-processing and three post processing adders, and  $3N/2$  multipliers and  $3(N/2-1)+4$  adders, which reduces approximately one fourth over the traditional two-parallel filter hardware cost from

(4). The two-parallel (L=2) FIR filter implementation using FFA obtained from (5) is shown in Fig. 1.

3.2. 3x3 FFA (L=3)

The (3X3) FFA produces a parallel filtering structure of block size 3. From (2), with L=3 we have,

$$Y_0 = H_0X_0 - Z^{-2}H_2X_2 + Z^{-3} \times [(H_0+H_1) (X_1+X_2)-H_1X_1]$$

$$Y_1 = [(H_0+H_1) (X_0+X_1)-H_1X_1] - (H_0X_0 - Z^{-3}H_2X_2)$$

$$Y_2 = [(H_0+H_1+H_2) (X_0+X_1+X_2)] - [(H_0+H_1) (X_0+X_1) - H_1X_1] - [(H_1+H_2) (X_1+X_2) - H_1X_1] \quad (6)$$

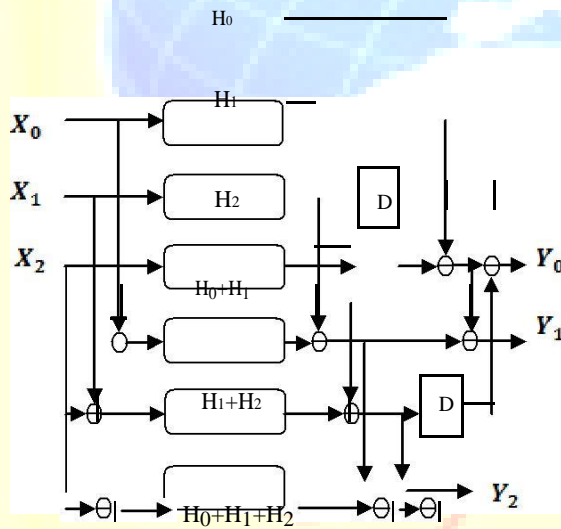


Fig. 1 3-parallel FIR filter Implementation using FFA

Direct implementation of (6) computes a block of 3 outputs using six length-N/3 FIR sub filter blocks, three pre-processing and seven post processing adders, and three multipliers and 2N+4 adders. This structure provides a saving of approximately one third over the traditional 3-parallel filter hardware cost.

IV. PROPOSED FFA STRUCTURE

The main objective of the proposed structure is to earn as many sub filter blocks as possible so that half the number of multiplications in the single sub filter block can be reused for the multiplications of the whole taps, which is similar to the fact that a set of both odd and even symmetric coefficients would only require half the filter length of multiplications in a single FIR filter. Therefore for an N-tap L-parallel FIR filter the total amount of saved multipliers would be the number of sub filter blocks.

4.1 Symmetric convolution based FFA (L=2)

From (4), a two-parallel FIR filter can also be written as

$$\begin{aligned}
 Y_0 &= \left\{ \frac{1}{2} [(H_0 + H_1)(X_0 + X_1)] + (H_0 - H_1) \right. \\
 &\quad \left. (X_0 - X_1) - H_1 X_1 \right\} + Z^{-2} H_1 X_1 \\
 Y_1 &= \frac{1}{2} [(H_0 + H_1)(X_0 + X_1) \\
 &\quad - (H_0 - H_1)(X_0 - X_1)]
 \end{aligned}
 \tag{7}$$

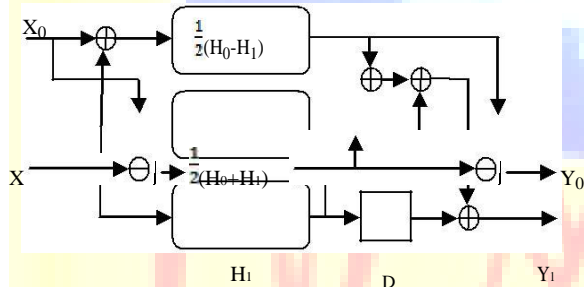


Fig. 3 PROPOSED 2-PARALLEL FIR FILTER IMPLEMENTATION

When it comes to a set of even symmetric coefficients, (7) can earn one more sub filter block containing symmetric coefficients than (5), the existing FFA parallel FIR filter. Fig. 3 shows implementation of the symmetric convolution two-parallel FIR filter based on (7).

4.2 Symmetric convolution based FFA (L=3)

When the number of symmetric coefficients N is a multiple of 3, the proposed 3-parallel FIR structure enables four

subfilter blocks with symmetric coefficients in total, whereas the existing FFA parallel FIR structure has only two ones out of six subfilter blocks. The proposed structure saves N/3 multipliers from the existing FFA structure. However, again the proposed 3-parallel FFA structure also brings an overhead of seven additional adders in preprocessing and postprocessing blocks.

$$Y_0 = [(H_0+H_1)(X_0+X_1) + (H_0-H_1)(X_0-X_1)] - H_1X_1$$

$$+ Z^{-3} \{ (H_0+H_1+H_2)(X_0+X_1+X_2) - (H_0+H_2)(X_0+X_2) - \frac{1}{2} [(H_0+H_1)(X_0+X_1) - (H_0-H_1)(X_0-X_1)] - H_1X_1 \}$$

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

$$+ Z^3 \{ \frac{1}{2} [(H_0+H_2)(X_0+X_2) + (H_0-H_2)(X_0-X_2)]$$

$$- \frac{1}{2} (H_0+H_1)(X_0+X_1) + (H_0-H_1)(X_0-X_1) \} + H_1X_1 \}$$

$$Y_2 = \frac{1}{2} [(H_0+H_2)(X_0+X_2) - (H_0-H_2)(X_0-X_2)] + H_1X_1 \quad (8)$$

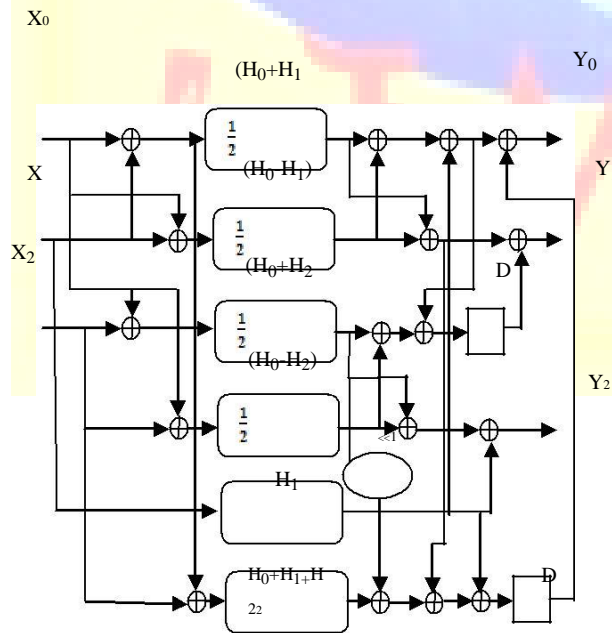


Fig. 4 Proposed 3-parallel FIR filter Implementation

#### 4.3. Proposed Cascading FFA

A Monthly Double-Blind Peer Reviewed Refereed Open Access International e-Journal - Included in the International Serial Directories Indexed & Listed at: Ulrich's Periodicals Directory ©, U.S.A., Open J-Gate as well as in Cabell's Directories of Publishing Opportunities, U.S.A.

The proposed cascading process for the larger block-sized proposed parallel FIR filter is similar. When cascading the proposed FFA parallel FIR structures for larger parallel block factor  $L$ , the increase of adders can become larger. Therefore, other than applying the proposed FFA FIR filter structure to all the decomposed sub filter blocks, the existing FFA structures which have more compact operations in pre-processing and post processing blocks are employed for those sub filter blocks that contain no symmetric coefficients, whereas the proposed FIR filter structures are still applied to the rest of sub filter blocks with symmetric coefficients. An illustration of the proposed cascading process for a six-parallel FIR filter ( $L=6$ ) as an example and the realization is shown in Fig. 5. From Fig. 7, The proposed six-parallel FIR filter will result in 6 more symmetric sub filter blocks, equivalently  $N/2$  multipliers saved for an  $N$ -tap FIR filter, than the existing FFA, at the expense of an additional 32 adders. The 6 parallel FIR filter is generated by cascading a (2-by-2) FFA with a (3-by-3) FFA. Beginning with the (2-by-2) FFA is applied resulting in 3 filtering operations. The (3-by-3) FFA is then applied to. It should be noted that when (2-by-2) and (3-by-3) FFAs are cascaded, the (2-by-2) FFA is always applied first as this will lead to the lowest implementation cost. The resulting  $\&$ parallel filtering structure requires  $18N/6$ , or  $3N$ , multipliers and  $42+18(N/6 - 1)$  adders. This reduced-complexity  $G$ -parallel filtering structure provides an area savings of approximately 50% compared to the traditional  $\&$ parallel filtering structure.

#### V. EXPERIMENTAL RESULTS

The proposed FFA structures and the existing FFA structures are implemented in Verilog with filter length of 24, word length 16-bit and 32-bit, respectively. The subfilter is based on canonical signed digit (CSD) structure and Carry-Select Adders are used.

Filter Structure	AREA( based on carry save adder)	AREA( based on Carry select adder)
Traditional FIR	3558	3543
FFA based FIR	1470	1380
Symmetric convolution based FFA FIR	1720	1840

Table 4.1 Performance analysis of three parallel FIR filter structure for area.



Logic Utilization	Structure	Utilization
No of 4 input LUTs	Existing	2%
	Proposed	1%
No of occupied slices	Existing	2%
	Proposed	1%
Total no of input LUTs	Existing	2%
	Proposed	1%
No of bonded IOBs	Existing	48%
	Proposed	13%

Table 4. 2. Area Comparison for 2-parallel FIR filter

Logic Utilization	Structure	Utilization
No of 4 input LUTs	Existing	4%
	Proposed	2%
No of occupied slices	Existing	5%
	Proposed	2%
Total no of input LUTs	Existing	4%
	Proposed	2%
No of bonded IOBs	Existing	66%
	Proposed	19%

Table 4. 3. Area Comparison for 3-parallel FIR filter

Logic Utilization	Structure	Utilization
No of 4 input LUTs	Existing	11%
	Proposed	5%
No of occupied slices	Existing	11%
	Proposed	5%
Total no of input LUTs	Existing	11%
	Proposed	5%
No of bonded IOBs	Existing	69%
	Proposed	26%

Table 4. 4 Area Comparison for 4-parallel FIR filter

L	Structure	Net Delay(ns)
2-parallel	Existing	54.348
	Proposed	36.893
3-parallel	Existing	42.554
	Proposed	41.651
4-parallel	Existing	49.527
	Proposed	43.507

Table 4. 5 Delay Comparison

From the analysis of Area and delay for parallel FIR filter structures, carry SELECT adder based FIR design is hardware efficient for Traditional and Symmetric convolution based FFA FIR structures. From the analysis of power for parallel FIR filter structures, carry SELECT adder based FIR design is low power design for all these three structure.

## VI. CONCLUSION

In this paper, we have presented new FFA-based parallel FIR filter structures for  $L= 2, 3,4$ , which save significant amounts of multipliers at the expense of increase of additional adders by exploiting the nature of symmetric coefficients with use of the proposed poly phase decompositions. Multipliers are the major portions in hardware consumption for the parallel FIR filter implementation. It has been shown that the hardware cost and power consumption of parallel FIR filters can be reduced significantly by exploiting the nature of symmetric coefficients. This proposed structure saves a significant amount of multipliers at the expense of additional adders. Since multipliers outweigh adders in hardware cost, exchanging multipliers with adders is advantageous and profitable. In addition, the overhead from the additional adders in preprocessing and postprocessing blocks stay fixed and do not increase along with the length of the filter, whereas the number of reduced multipliers increases along with the length of the FIR filter portions. Overall, in this paper, we have provided new parallel FIR structures consisting of advantageous polyphase decompositions dealing with symmetric convolutions comparatively better than the existing FFA structures in terms of hardware consumption.

## REFERENCES

- [1] Z.-J. Mou and P. Duhamel, "Short-length FIR filters and their use in fast nonrecursive filtering," *IEEE Trans. Signal Process.*, vol. 39, no.6, pp. 1322–1332, Jun. 1991.
- [2] J. G. Chung and K. K. Parhi, "Frequency-spectrum-based low-area low-power parallel FIR filter design," *EURASIP J. Appl. Signal Process*, vol. 2002, no. 9, pp. 444–453, 2002.
- [3] C. Cheng and K. K. Parhi, "Hardware efficient fast parallel FIR filter structures based on iterated short convolution," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 51, no. 8, pp. 1492–1500, Aug. 2004..
- [4] Yongtao Wang and Roy.K, "A novel low-complexity method for parallel multiplierless implementation of digital FIR filters," *IEEE Int. Symp. Circuits Syst. (ISCAS 2005)*, vol. 3, no. 8, pp. 2020–2023
- [5] Shiann-Shiun and Hsing-Chen Lin, "FPGA implementation of FIR filter using M-bit parallel Distributed Arithmetic," *IEEE Int. Symp. Circuits Syst. (ISCAS 2006)*, 2006.
- [6] C. Cheng and K. K. Parhi, "Low-cost parallel FIR structures with 2-stage parallelism," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol.54, no. 2, pp. 280–290, Feb. 2007.
- [7] R.Mahesh and P.Vinod, "A new common sub expression elimination algorithm for realizing low-complexity higher order digital filters," *J.Computer-Aided design of Integrated Circuits and Systems*, vol. 29, issue 5, pp. 844-848, May.2010.
- [8] Yu-Chi Tsao and Ken Choi "VLSI implementation for parallel linear phase FIR digital filters of odd length based on FFA," *IEEE transactions on large Circuits and Systems*, 2012.
- [9] I.-S. Lin and S. K. Mitra, "Overlapped block digital filtering," *IEEETrans. Circuits Syst. II, Analog Digit.*