



## **Fast Parallel Linear Phase Fir Filter Implementation Based On Fast Fir Algorithm**

**V.Logeswari**

P.G Student/VLSI Design, Nandha Engineering College, Erode, Tamilnadu, India

**T.Manickam**

Associate Professor, Dept. of ECE, Nandha Engineering College, Erode, Tamilnadu,  
India

**Dr.C.N.Marimuthu**

DEAN, Dept. of ECE, Nandha Engineering College, Erode, Tamilnadu, India

### ***Abstract:***

*In recent days filters with large lengths are started to use. So parallel processing is essential at any cost. In this paper proposes new parallel FIR filter structures, which are beneficial to symmetric coefficients in terms of the hardware cost, under the condition that the number of taps is a multiple of 2 or 3. The proposed parallel FIR structures use symmetric property to reducing half the number of multipliers in sub filter section at the expense of additional adders in preprocessing and post processing blocks. Exchanging multipliers with adders is advantageous because adders weigh less than multipliers in terms of silicon area; in addition, the overhead from the additional adders in preprocessing and post processing blocks stay fixed and do not increase along with the length of the FIR filter, whereas the number of reduced multipliers increases along with the length of the FIR filter. Parallel FIR filter is essential, especially when the length of the filter is large. Overall, the proposed parallel FIR structures can lead to significant hardware savings for symmetric convolution in odd length from the existing FFA parallel FIR filter, particularly when the length of the filter is large.*

**Key words:** Digital Signal Processing (DSP), fast FIR algorithm (FFAs), parallel FIR, VLSI.

## 1.Introduction

Digital signal processing (DSP) has many advantages over analog signal processing. Digital signals are more robust than analog signals with respect to temperature and process variations. The accuracy in digital representations can be controlled better by changing the word length of the signal. Furthermore, DSP techniques can cancel the noise and the interference while amplifying the signal. In contrast, both signal and noise are amplified in analog signal processing. Digital signals can be stored and recovered, Transmitted and received, processed and manipulated, all virtually without error. While analog signal processing is indispensable for systems that require extremely high frequencies such as the radio frequency transceiver in wireless communications, or extremely low area and low power such as micro machine sensors used to detect cracks and other stress-related material defects, many complex systems are realized digitally with high precision, high signal to noise ratio (SNR), repeatability, and flexibility. The finite-impulse response (FIR) filter has been and continues to be one of the fundamental processing elements in any digital signal processing (DSP) system. FIR filters are used in DSP applications that range from video and image processing to must be a low-power circuit, capable of operating at moderate frequencies. Parallel, or block, processing can be applied to digital FIR filters to either increase the effective throughput or reduce the power consumption of the original filter. Traditionally, the application of parallel processing to an FIR filter involves the replication of the hardware units that exist in the original filter.

Parallel processing in the digital FIR filter will be discussed. Due to its linear increase in the hardware implementation cost brought by the increase in the block size  $L$ , the parallel processing technique loses its advantage to be employed in practice. There have been a few papers proposing ways to reduce the complexity of the parallel FIR filter in the past [1]–[10]. In [1]–[4], polyphase decomposition is mainly manipulated, where the small-sized parallel FIR filter structures are derived first and then the larger block-sized ones can be constructed by cascading or by iterating small-sized parallel FIR filtering blocks. Fast FIR algorithms (FFAs) introduced in [1]–[3] show that they can implement an  $L$ -parallel filter using approximately  $(2L - 1)$  sub filter blocks, each of which is of length  $N/L$ . It reduces the required number of multipliers to  $(2N - N/L)$  from  $L \times N$ . In [5]–[9], the fast linear convolution is utilized to develop the small-sized filtering

structures, and then a long convolution is decomposed into several short convolutions, i.e., larger block-sized filtering structures can be constructed through iterations of the small-sized filtering structures.

## 2. Techniques For Implementing Parallel FIR Filter Using Fast FIR Algorithm

### 2.1. Two Parallel Proposed FFA ( $L=2$ )

Based on Fast FIR Algorithm, the two-parallel FIR filter can also be written as

$$\begin{aligned} Y_0 &= \{ \frac{1}{2} [(H_0 + H_1) (X_0 + X_1) + (H_0 - H_1) (X_0 - X_1)] - H_1 X_1 \} + z^{-2} H_1 X_1 \\ Y_1 &= \frac{1}{2} [(H_0 + H_1) (X_0 - X_1) - (H_0 - H_1) (X_0 - X_1)] \end{aligned} \quad (3.2)$$

When it comes to a set of even symmetric coefficients, proposed method can earn one more sub filter block containing symmetric coefficients than the existing FFA parallel FIR filter. The proposed two- parallel FIR filter implementation is given as

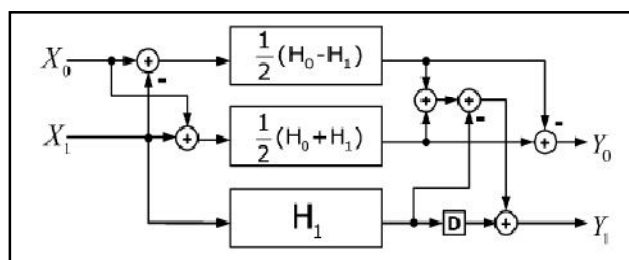


Figure 1: Proposed Two- Parallel FIR Filter Implementation

### 2.2. Three Parallel Proposed FFA ( $L=3$ )

With the similar approach, from existing method, a three-parallel FIR filter can also be written as,

$$\begin{aligned} Y_0 &= \frac{1}{2} [(H_0 + H_1) (X_0 + X_1) + (H_0 - H_1) (X_0 - X_1)] - H_1 X_1 \\ &\quad + z^{-3} \{ (H_0 + H_1 + H_2) (X_0 + X_1 + X_2) - (H_0 + H_2) (X_0 + X_2) \\ &\quad - \frac{1}{2} [(H_0 + H_1) (X_0 + X_1) - (H_0 - H_1) (X_0 - X_1)] - H_1 X_1 \} \end{aligned}$$

$$\begin{aligned} Y_1 &= \frac{1}{2} [(H_0 + H_1) (X_0 + X_1) - (H_0 - H_1) (X_0 - X_1)] + \\ &\quad Z^{-3} \{ \frac{1}{2} [(H_0 + H_2) (X_0 + X_2) + \\ &\quad (H_0 - H_2) (X_0 - X_2)] - \frac{1}{2} [(H_0 + H_1) (X_0 + X_1) + (H_0 - H_1) (X_0 - X_1)] + \\ &\quad H_1 X_1 \} \end{aligned}$$

$$Y_2 = \frac{1}{2} [(H_0 + H_2) (X_0 + X_2) - (H_0 - H_2) (X_0 - X_2)] + H_1 X_1$$

When the number of symmetric coefficients  $N$  is the multiple of 3, the proposed three-parallel FIR filter structure presented above enables four sub filter blocks with

symmetric coefficients in total, whereas the existing FFA parallel FIR filter structure has only two ones out of six sub filter blocks.

The proposed three-parallel FIR structure also brings an overhead of seven additional adders in preprocessing and post processing blocks. The proposed cascading process for the larger block-sized proposed parallel FIR filter is similar to that introduced. However, a small modification is adopted here for lower hardware consumption. As we can see, the proposed parallel FIR structure enables the reuse of multipliers in parts of the sub filter blocks but it also brings more adder cost in preprocessing and post processing blocks. 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 preprocessing 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.

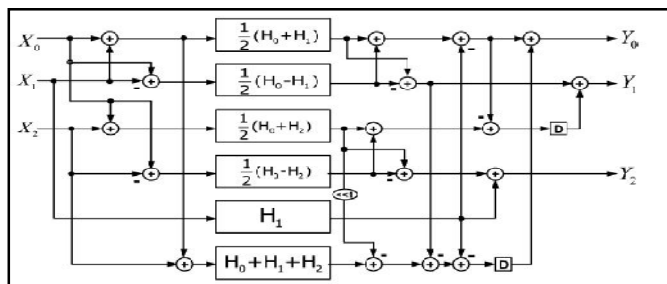


Figure 2: Proposed Three-Parallel FIR Filter Implementation

### 3.Structures For Symmetric Convolution Of Odd Length

The main idea is to manipulate the polyphase decomposition to earn as many sub- filter blocks as possible, which contain symmetric coefficients the existing two-parallel FFA structure naturally has benefits to symmetric convolutions in odd length. When it comes to a set of odd-length symmetric coefficients, two out of three sub filters contain symmetric coefficients, i.e.,  $H_0$  and  $H_1$ , shown in Fig. 1. However, the existing three-parallel FFA structure is not as advantageous. In this section, new three-parallel FIR filter can save more hardware cost over the existing FFA.

Proposed Structure 3A,  $((N \bmod 3) = 0)$ : From (5), it can also be presented as (7). For a set of symmetric coefficients in odd length  $N$ , when  $(N \bmod 3)$  equals zero, (7) can

earn two more sub filter blocks containing symmetric coefficients than (5). The implementation of the proposed three-parallel FIR filter based on (7) is shown in Fig. 3.

Consider a 27-tap FIR filter with a set of symmetric coefficients as follows:

$h(0) = h(26), h(1) = h(25), h(2) = h(24), h(3) = h(23), h(4) = h(22), h(5) = h(21), \dots, h(12) = h(14)$ , applying to the proposed structure 3A, and then, we gain two more sub filter blocks with symmetric coefficients as

$$H_0 \pm H_2 = \{h(0) \pm h(2), h(3) \pm h(5), h(6) \pm h(8), \dots, h(18) \pm h(20), h(21) \pm h(23), h(24) \pm h(26)\}$$

where  $h(0) \pm h(2) = \pm (h(24) \pm h(26))$   $h(3) \pm h(5) = \pm (h(21) \pm h(23))$   
 $h(6) \pm h(8) = \pm (h(18) \pm h(20))$

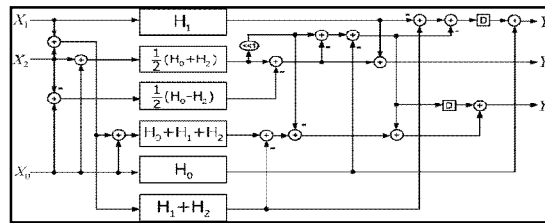


Figure 3: Implementation of the proposed Structure 3A

So that half the number of multipliers within a single sub filter block can be utilized for the multiplications of whole taps.

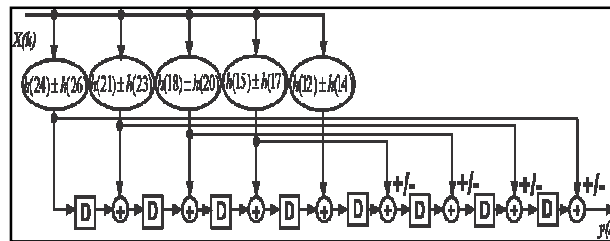


Figure 4: Implementation of sub filter blocks

Existing FFA	Proposed 3A
$H_0$	$H_1$
$H_1$	$\frac{1}{2}(H_0+H_2)$
$H_2$	$\frac{1}{2}(H_0-H_2)$
$H_0+H_1$	$H_0+H_1+H_2$
$H_1+H_2$	$H_0$
$H_0+H_1+H_2$	$H_1+H_2$

Figure 5: Comparison of sub filter blocks between the existing FFA and the proposed structure 3A.

after applying the proposed structure 3A, in Fig. 3, four out of six sub filter blocks, i.e.,  $H_1$ ,  $H_0 \pm H_2$ ,  $H_0 + H_1 + H_2$ , are with symmetric coefficients now, which means a single sub filter block can be realized in Fig. 4, with only half the amount of multipliers required. Each output of multipliers responds to two taps, except the middle one. Note that the transposed direct-form FIR filter is employed. Compared with the existing FFA three-parallel FIR filter structure, the proposed structure leads to two more sub filter blocks, which contains symmetric coefficients. Therefore, for an  $N$ -tap three-parallel FIR filter, the proposed structure can save  $N/3$  multipliers from the existing FFA structure. However, it comes with the price of the increase in amount of adders, i.e., five additional adders, in preprocessing and post processing blocks.

### 3.1. Proposed Cascaded FFA

The proposed cascading process for the larger block-sized proposed parallel FIR filter is similar to that introduced in past. However, a small modification is adopted here for lower hardware consumption. As we can see, the proposed parallel FIR structure enables the reuse of multipliers in parts of the sub filter blocks but it also brings more adder cost in pre-processing and post processing blocks. When cascading the proposed FFA parallel FIR structures for larger parallel block factor  $L$ , the increase of adders can become larger.

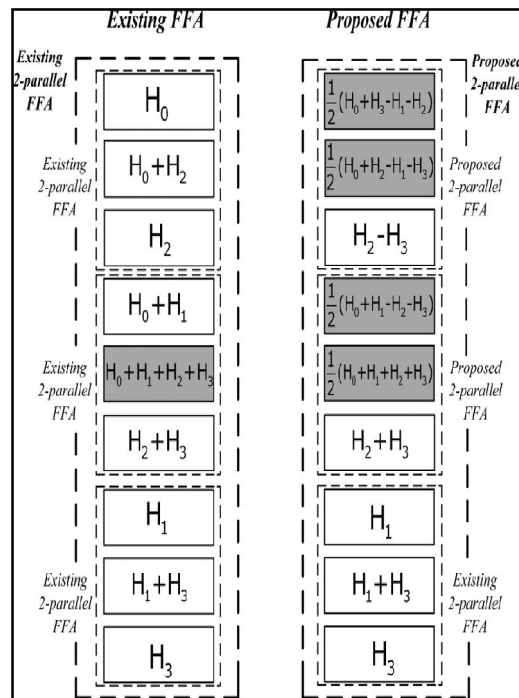


Figure 6: Proposed four parallel FIR filter implementation using cascaded FFA

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 four-parallel FIR filter ( $L=4$ ) as an example is shown in Fig.6, it is clear to see that the proposed four-parallel FIR structure earns three more sub filter blocks containing symmetric coefficients than the existing FFA one, which means  $3N/8$  multipliers can be saved for an  $N$ -tap FIR filter, at the price of 11 additional adders in pre-processing and post processing blocks.

By this cascading approach, parallel FIR filter structures with larger block factor  $L$  can be realized. 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. Also, the proposed eight-parallel FIR filter will lead to seven more symmetric sub filter blocks, equivalently  $7N/16$  multipliers saved for an  $N$ -tap filter, than the existing FFA, with the overhead of additional 54 adders.

$L$	Length	Structure	M.	R.M.	Required Adders		I.A.
					Sub.	Pre/Post.	
3	27-tap	FFA	46	8	48	10	5
		Proposed	38	(17.4%)		15	
	81-tap	FFA	136	26	156	10	
		Proposed	110	(19.1%)		15	
	147-tap	FFA	246	48	288	10	
		Proposed	198	(19.5%)		15	
	591-tap	FFA	986	196	1176	10	
		Proposed	790	(19.9%)		15	
4	27-tap	FFA	54	3	63	20	4
		Proposed	51	(5.6%)		24	
	81-tap	FFA	159	10	180	20	
		Proposed	149	(6.3%)		24	
	147-tap	FFA	279	18	324	20	
		Proposed	261	(6.5%)		24	
	591-tap	FFA	1110	74	1323	20	
		Proposed	1036	(6.7%)		24	
6	27-tap	FFA	82	6	72	42	11
		Proposed	76	(7.3%)		53	
	81-tap	FFA	224	21	234	42	
		Proposed	203	(9.4%)		53	
	147-tap	FFA	402	36	432	42	
		Proposed	366	(9.0%)		53	
	591-tap	FFA	1586	147	1764	42	
		Proposed	1439	(9.3%)		53	
8	27-tap	FFA	100	8	81	80	16
		Proposed	92	(8.0%)		96	
	81-tap	FFA	277	20	270	80	
		Proposed	257	(7.2%)		96	
	147-tap	FFA	477	36	486	80	
		Proposed	441	(7.6%)		96	
	591-tap	FFA	1850	148	1971	80	
		Proposed	1702	(8.0%)		96	

Whereas,

L - Level of parallelism,

N - Length of the filter (Tap length),

M - Required number of Multipliers,

R.M - Reduced Multipliers,

SS - Number of Required Adders in Sub filter section,

I.A - Number of Increased Adders.

#### 4.Simulation Results

The Simulation Results Were Obtained By Xilinx 10.1 Version Supported With Modelsim Simulator. For 2x2 Parallel Fir Filter Structure Using Fast FIR algorithm and it is simulated by the Xilinx ISE 10.1 version or later version. The output is shown by the Modelsim 5.7 version.  $X_0$ ,  $X_1$ -inputs of FIR filter.  $H_0$ ,  $H_1$ -filter coefficients,  $Y_0$ ,  $Y_1$ -output of 2x2 parallel fir filters.

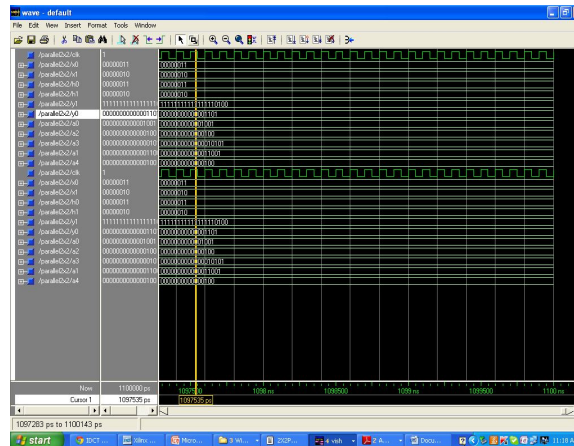


Figure 7: Simulated output of 2x2 parallel FIR filter

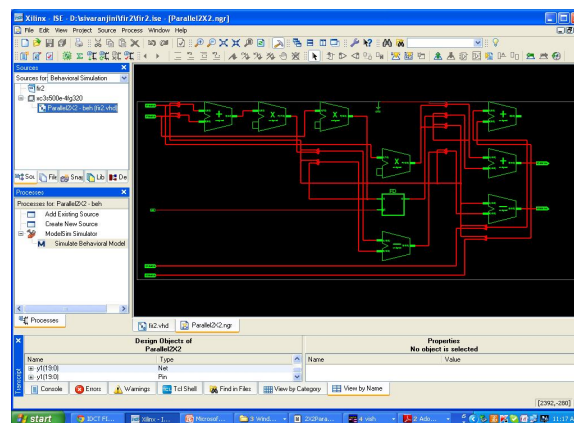


Figure 8: Schematic of 2x2 parallel FIR filter



Area	L=3	L=4	L=6
FFA	305605	361312	532441
Proposed	280861	338728	522198

Table 1: Comparison of Area with FFA and proposed structure

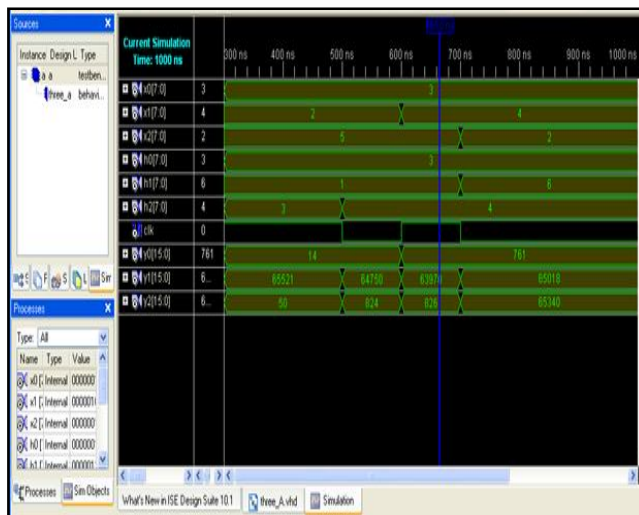


Figure 9: Simulated output of proposed structure 3A

Power	L=3	L=4	L=6
FFA	166.15	193.18	276.05
Proposed	129.76	174.48	266.44

Table 2: Comparison of power with FFA and Proposed structure

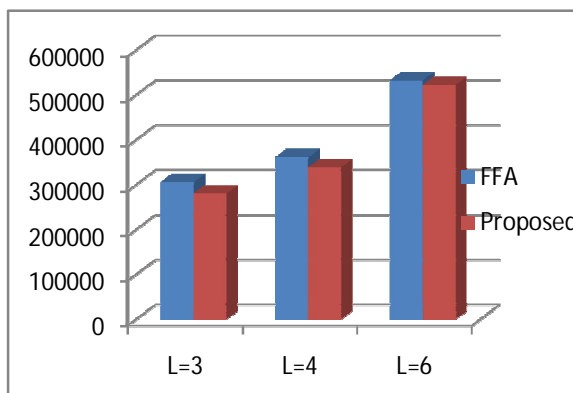
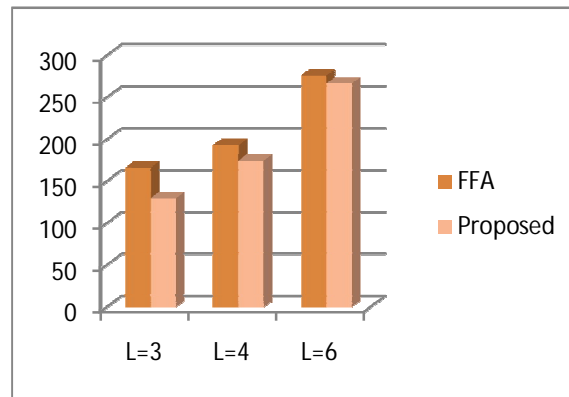


Figure 10: Comparison chart of Area



*Figure 11: Comparison Chart of Power*

The Fig.10 and Fig.11 shows the area, power

### **5.Conclusion**

In this paper, we have presented new parallel FIR filter structures, which are beneficial to symmetric convolutions of odd length. Multipliers are the major portions in hardware consumption for the parallel FIR filter implementation. The proposed new structures exploit the nature of symmetric co-efficient of odd length and further reduce the amount of multipliers required at the expense of additional adders. Since multipliers outweigh adders in hardware cost, it is profitable to exchange multipliers with adders. Moreover, the number of increased adders stays still when the length of FIR filter becomes large, whereas the number of reduced multipliers increases along with the length of the FIR filter.

**6.Reference**

1. D. A. Parker and K. K. Parhi, "Low-area/power parallel FIR digital filter implementations," *J. VLSI Signal Process. Syst.*, vol. 17, no. 1, pp. 75–92, Sep. 1997.
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, Jan. 2002
3. K. K. Parhi, *VLSI Digital Signal Processing systems: Design and Implementation*. New York: Wiley, 1999.
4. 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.
5. J.I.Acha, "Computational structures for fast implementation of L path and L-block digital filters," *IEEE Trans. Circuits Syst.*, vol. 36, no. 6, pp. 805–812, Jun. 1989.
6. 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.
7. C. Cheng and K. K. Parhi, "Further complexity reduction of parallel FIR filters," in *Proc. IEEE ISCAS*, May 2005, vol. 2, pp. 1835–1838.
8. 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.
9. I.-S. Lin and S. K. Mitra, "Overlapped block digital filtering," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 43, no. 8, pp. 586–596, Aug. 1996.
10. Y.-C. Tsao and K. Choi, "Area-efficient parallel FIR digital filter structures for symmetric convolutions based on fast FIR algorithm," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 20, no. 2, pp. 366–371, Feb. 2010.