Advanced Production and Industrial Engineering
R.M. Singari and P.K. Kankar (Eds.)

doi:10.3233/ATDE220715

Finite Impulse Response Filter Growth and Applications

[1] R. Prameela Jyothi, [2]K. Sangeet Kumar and [3*]Mahesh K. Singh

[1,2]Department of ECE, Aditya College of Engineering, Surampalem, India
[3]Department of ECE, Aditya Engineering College, Surampalem, India
[*3]mahesh.singh@accendere.co.in

Abstract. A "Finite Impulse Response (FIR)" filter's impulse response has a finite period. Higher order FIR filter is used in numerous "Digital Signal Processing (DSP)" applications to attain accurate frequency specifications. With increasing filter length, there is a linear increase in the number of additions and multiplications, increasing computing complexity. This paper discusses a variety of FIR filter implementation techniques. The FIR filters are used to reduce the number of arithmetic operations required for inner product calculations is a predetermined number, whilst the "look up table (LUT)" design stores the pre-computed result to keep things simple. Filters are commonly used in a variety of applications; the end goal of using a filter is to create a form of frequency selectivity on the spectrum of the incoming signal. Any DSP subsystem's FIR filter is regarded as one of the most important components. The major purpose of this project is to briefly examine numerous design strategies in order to aid future development.

Keywords: DSP, FIR, IIR, LUT, LMS

1. Introduction

Modern electronic systems, such as mobile communication and healthcare applications, rely heavily on FIR filters for their design [1]. Signal separation and restoration are the primary functions of the filters. With higher order fir filters, signal separation is required to remove noise and other signal pollutant. The complexity of calculations rises as the order of arithmetic operations rises. To simplify resource-intensive tasks, the multiple constant multiplication technique is used. One of the furthermost important elements in DSP processing is the digital filter [2]. FIR filter having mainly multipliers, adder and delay unit. So, optimizing FIR filters its necessary to used optimized adder and multipliers [3, 4]. Its primary purpose is to reduce noise and suppress undesirable signals. SDR channels use it as well [18]. Multipliers must be fast enough in this case, and an adder is combined with multipliers and a memory sampling with a single sample clock cycle delay.

FIR filters have the following advantages over IIR filters:

• FIR filters can be constructed with linear phase, and they are simple to implement.

• On finite-precision arithmetic, FIR filters are simple to implement (many microcontrollers can function with 16-bit words, but for IIR filter correct operation, 32 bits are required in some circumstances to hold "Y" coefficients). As a result, IIR filter

implementation on 16-bit MCUs has significantly more difficulties than FIR filter implementation) [5].

The following are the disadvantages of FIR filters over IIR filters:

• To acquire the appropriate response, FIR requires additional memory and calculations [6].

The main purpose of this project is to look at a few different design ideas in order to make future advancements easier [7].

2. Literature Review

In this section, we attempt to summaries the growth of FIR filters year by year. L-path and L-block digital filters were implemented using multi-DSP (digital signal processor) implementations in 1998, and a difficult technique termed variable multiplication by numerous constants was employed to lessen the complexity of the implementation. Lagrange's Multiplier, Bellman's principle and Pontyagrin's principle were some of the most extensively utilized nonlinear optimization methodologies of the last century [1, 8]. There should be enough speed in the multiplier so that the large number of signals multiplied by constants and combined together does not affect overall throughput [2, 9]. According to this work, which was published in 1997, a new method for producing area-efficient parallel (block) FIR filters was developed, which required less hardware than conventional block FIR filter implementations. Due to the fact that it eventually reaches zero, a FIR filter's impulse response is limited in length. The output of a FIR filter is just the total of its previous, current, and perhaps future outputs [3, 10].

Originally established in 2002, parallel (or block) FIR digital filters provide for high-speed and low-power (low supply voltage) applications depending on the filter architecture. As the size of the block increases in traditional parallel filter implementations, hardware costs rise linearly; however, shifting to a group-based approach encourages smaller groups and produces better results in terms of area and power [11, 5]. Contact was made in 2004 with researchers from prior decades. There are a variety of excellent filter building algorithms that may be tested. MCM can't be applied directly to direct form, and block filtering takes a lengthy time because partial results must be obtained. This approach provides a method for implementing block filtering while utilising direct form [12, 7].

The filter designs described in this 2005 paper are capable of managing massive volumes of data, such as the one detailed in this paper. Consequently, the hardware design is re-evaluated at the bit level, resulting in a decrease in complexity. The pipelined multiplier array's intrinsic delays are also fully utilized [13]. Previously, parallel FIR filter topologies were thought to be simple to implement, however new research from 2005 demonstrates that this is not the case. To keep things simple and take up less space, a filter's adder count should be kept to a minimum. It is possible to maintain previously computed results by using this method [14, 20-21].

A novel parallel FIR filter structure is introduced for the first time in this 2007 study, which makes use of recently published low-complexity parallel FIR filter designs in order to reduce hardware complexity. By providing frequency selectivity in the input signal spectrum, filters are widely used to improve performance [15]. Basant Mohanty and P. K. Meher et al. [16] presented FIR filtering with the BLMS algorithm [16]. Using the LUT sharing technique, both convolution and correlation can be done on the same set of LUTs. Ganasekaran. K and Dr. M. Manikandam et al. [17] proposed a Shift and

Add technique for RFIR implementation. A multiplier consists of three blocks: a shifter, an adder, and a control unit. By 2020, numerous studies predict that the entire potential of FIR filters will still be underutilized [18, 19].

3. Methodology: Common Sub-Expression Sharing

This method of creating a block FIR filter is directly derived from the multiple constant multiplication method (MCM). Using block filtering results in a long delay because it collects partial results, which are then transposed and can't be converted to direct form right once. As a result, it provides a simple way to use MCM to achieve block filtering.

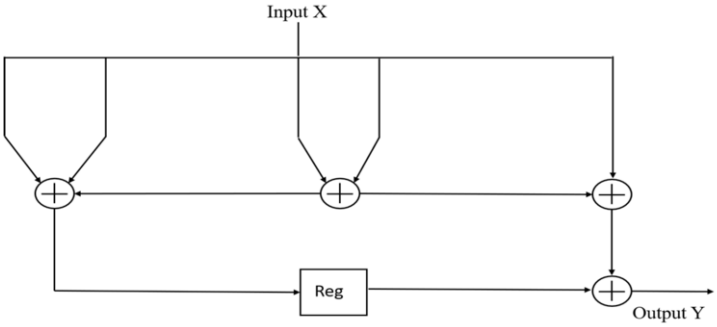

Figure 1. Common subexpressions can be eliminated by using CSD representation.

Figure 1 The MCM's FIR filter MCM demonstrates how to remove subexpressions that appear frequently. After finding and sharing similar coefficient-related subexpressions, the number of operations required to add or shift it can be minimized. It is expected that the complexity of MCM's computations will be lowered by this change.

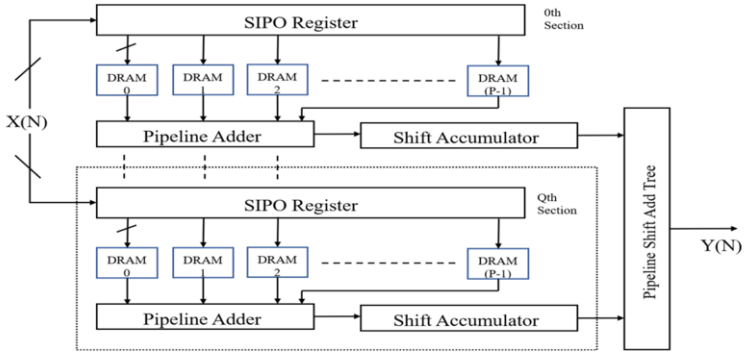

Figure 2. DA based design of FIR filter

Figure 2 this sample demonstrates shift operations and distributed arithmetic. Using distributed arithmetic, this method demonstrates how an effective RFIR filter shared LUT design can be achieved. Filter coefficients can be modified while the application is running. Because each LUT has only two bits of registers, a substantial number of resources are saved by realising each bit slice in DRAM in an FPGA device. The inner

product generator consists of Q parallel modules, each with R time-multiplexed operations that match the RFIIR.

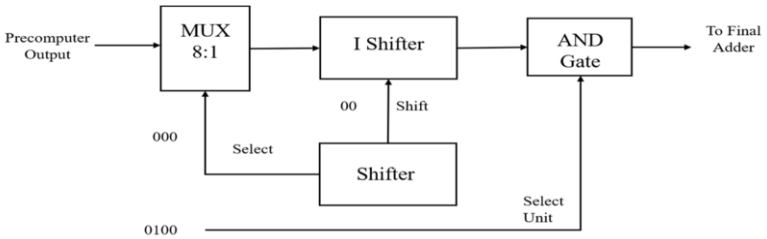

Figure 3. CSHM design select unit

FIR filters are often developed using a transposed form implementation. "Vector scaling" refers to the practice of multiplying a coefficient vector C by X at a specific time (n). The vector scaling algorithm selects tiny bit patterns in order to acquire the same multiplication results as add and shift operations. The bit patterns used to represent the coefficients of vector C are known as alphabets as shown in Figure 3.

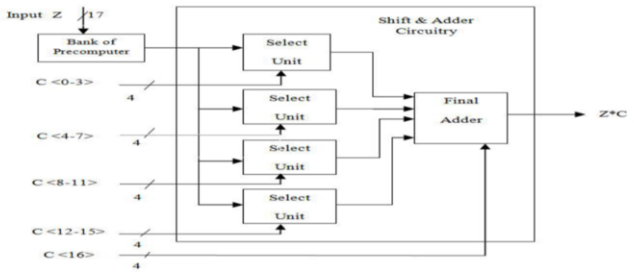

Figure 4. CSHM Design for FIR filter [30]

Figure 4 shows how FIR filter multipliers are turned into sum and adder circuits using a pre-computer as an input. Because it minimizes the number of computations, block filtering is faster in terms of area delay. The technique excludes the use of higher-order filters. This strategy, on the other hand, is more effective when dealing with area delays. However, in terms of power and performance, this design still has a big edge.

4. Results and Discussion

The length of time it takes depends on how good the filter is CSM contains a greater amount of adder and FFs than other's structure, increasing the overall complexity of the hardware. It computes filter output using the LUT sharing methodology, which allows it to make use of the shared LUTs for both convolution and correlation operations on the data. The LUT is refreshed with each output, and the adder saves a significant amount of money as a result of this strategy. Due to the fact that adders are not affected by block size, the number of adders has no effect on the size of the block. When compared to other methods, the number of flip flops required to achieve more order is less in this case. RFIR implementation was proposed by K. Gunasekaran and Manikandan, with the shifter consisting of a multiplexer to choose the input signal and the adder consisting of a multiplier to multiply the output signal in table 1.

Table 1: Assessment of dissimilar proposal method of FIR filter

S. No	author	Method	N (filter Length)	Area	MCP (ns)	ADP	Power (mw)
1	Maher and Yu pan [2]	Sub expression Sharing (MCM)	8	16955	0.33	1398.92	7.32
2	Mohanty and Maher [4]	Block LMS algorithms (DA)	32	112724	1.36	40254.09	56.3
3	Park and Maher [6]	LUT based technique (DA)	32	56326	0.99	56558.14	25.9
4	Mahesh and Vinod [3]	Constant shift technique	32	67948	1.28	21638.80	29.7
5	J. Park, H. Meimand [7]	Computation Sharing Multiplier (CSHM)	10	5×10^6	7	35.71×10^6	238.8
6	Gunasegaram [8]	Shift Add method	8	_	_	_	124

The size of the input block, rather than the length of the filter, determines the cycle period in this method. For larger filter lengths, the saving in-cycle period is noticeable when compared to other direct form structures. It also gives lower ADP for longer filter lengths due to the reduction in cycle time. Hardware requirement from different structures shown in table 2.

Table 2: Hardware requirement from different structures

Structures	Parameters		
	Filter length	Adder	Flip-flo
Block LMS algorithm	16	60	120
Block LMS algorithm	32	124	248
LUT based method	16	71	120
LUT based method	32	143	432
Constant shift method	16	114	360
Constant shift method	32	226	744

As a result, higher order filters can be considered with this planning. It reduces the amount of mathematics required, but it does not provide a design for higher order filters or an appropriate block filtering design. Because block filtering improves the efficiency of the area delay. At low order, however, this design beats others in terms of power and conductivity.

5. Conclusion

FIR filters will be discussed in detail in this paper. In MCM design, a direct type of FIR Filter is employed for the purpose of sharing common subexpressions. However, the MCM technique is better suited for use with the transpose form of a FIR filter, as previously stated. As a result, the MCM approach cannot be used by the subexpression sharing method. It is possible to reduce filter coefficients' exactness by changing non-zero CSD values without affecting the filter's behavior or reconfiguration overhead with the CSD-based RFIR filter, on the other hand is the best choice for fixed filter coefficient constant sharing multiplier because it saves money on hardware by allowing DA units to share the same register. This methodology is used to choose the most appropriate FIR filter implementation approach for both fixed and reconfigurable applications.

References

[1] Nandan, Durgesh, et al. "Speaker Identification Based on Physical Variation of Speech Signal." *Traitement du Signal* 39.2 (2022).

[2] Brundana, M. S. S., et al. "Successive Approximation Compressor for Efficient FIR Filters in C-MOS VLSI Design." *2021 6th International Conference for Convergence in Technology (I2CT)*. IEEE, 2021.

[3] Shively, R. "On multistage finite impulse response (FIR) filters with decimation." *IEEE Transactions on Acoustics, Speech, and Signal Processing* 23.4 (1975): 353-357.

[4] Padma, Uppalapati, Samudrala Jagadish, and Mahesh K. Singh. "Recognition of plant's leaf infection by image processing approach." *Materials Today: Proceedings* 51 (2022): 914-917.

[5] Lee, Kevin, et al. "Classification of microarray datasets using finite impulse response extreme learning machine for cancer diagnosis." *IECON 2011-37th Annual Conference of the IEEE Industrial Electronics Society*. IEEE, 2011.

[6] Punyavathi, G., M. Neeladri, and Mahesh K. Singh. "Vehicle tracking and detection techniques using IoT." *Materials Today: Proceedings* 51 (2022): 909-913.

[7] Mouli, D. Chandra, et al. "Video Retrieval Queries of Large Scale Images: An Efficient Approach." *2021 6th International Conference on Signal Processing, Computing and Control (ISPCC)*. IEEE, 2021.

[8] Raju, G. Siva Naga, G. Jyothirmai, and Sanjeev Kumar. "Design and analysis simple microstrip low-pass filter for wireless application." *Materials Today: Proceedings* 51 (2022): 1017-1020.

[9] Priya, B. Jyothi, Parvateesam Kunda, and Sanjeev Kumar. "Design and Implementation of Smart Real-Time Billing, GSM, and GPS-Based Theft Monitoring and Accident Notification Systems." *Proceedings of International Conference on Recent Trends in Machine Learning, IoT, Smart Cities and Applications*. Springer, Singapore, 2021.

[10] Balaji, V. Nithin, P. Bala Srinivas, and Mahesh K. Singh. "Neuromorphic advancements architecture design and its implementations technique." *Materials Today: Proceedings* 51 (2022): 850-853.

[11] Díaz, Daniel Suescún, Aquilino Senra Martinez, and Fernando Carvalho Da Silva. "Calculation of reactivity using a finite impulse response filter." *Annals of Nuclear Energy* 35.3 (2008): 472-477.

[12] Kanchana, V., Surendra Nath, and Mahesh K. Singh. "A study of internet of things oriented smart medical systems." *Materials Today: Proceedings* 51 (2022): 961-964.

[13] Prasanna, GV Sai, K. Pavani, and Mahesh Kumar Singh. "Spliced images detection by using Viola-Jones algorithms method." *Materials Today: Proceedings* 51 (2022): 924-927.

[14] Santhoshi, M. Siva, et al. "An investigation on rolling element bearing fault and real-time spectrum analysis by using short-time fourier transform." *Proceedings of International Conference on Recent Trends in Machine Learning, IoT, Smart Cities and Applications*. Springer, Singapore, 2021.

[15] Singh, M., Nandan, D., & Kumar, S. (2019). Statistical Analysis of Lower and Raised Pitch Voice Signal and Its Efficiency Calculation. *Traitement du Signal*, *36*(5), 455-461.

[16] Acha, Jose I. "Computational structures for fast implementation of L-path and L-block digital filters." *IEEE transactions on circuits and systems* 36.6 (1989): 805-812.

[17] Mou, Z-J., and Pierre Duhamel. "Short-length FIR filters and their use in fast nonrecursive filtering." *IEEE Transactions on Signal Processing* 39.6 (1991): 1322-1332.

[18] Singh, M. K., Singh, A. K., & Singh, N. (2019). Multimedia analysis for disguised voice and classification efficiency. *Multimedia Tools and Applications*, *78*(20), 29395-29411.

[19] Chung, Jin-Gyun, and Keshab K. Parhi. "Frequency spectrum based low-area low-power parallel FIR filter design." *EURASIP Journal on Advances in Signal Processing* 2002.9 (2002): 1-10.

[20] Simpson, James J., Jason I. Gobat, and Robert Frouin. "Improved destriping of GOES images using finite impulse response filters." *Remote Sensing of Environment* 52.1 (1995): 15-35.

[21] Siddiqa, S. K., et al. "Documentation on smart home monitoring using internet of things." *ICCCE 2020*. Springer, Singapore, 2021. 1115-1124.