ISSN (Online) 2249 - 054 X
International Journal of Computing and Corporate Research
Specialized and Refereed Journal for Research Scholars, Academicians, Engineers and Scientists

http://www.ijccr.com
International Manuscript ID : ISSN2249054X-V2I4M3-072012
VOLUME 2 ISSUE 4 July 2012

Review of Architectures for Low Power and Reconfigurable FIR Filters

Ankita Mandal[1], Er. Vikas Mittal[2]

1. M-Tech Student, 2. Assistant Professor in MMEC

Electronics and Communication Department

M.M. University, Mullana (Ambala), Haryana, INDIA

Abstract— Reconfigurability, low complexity and low power consumption are the key requirements in a FIR filter. Many works have been done on the reduction of area and power requirements but Reconfigurability is less addressed in the papers. In this paper, a survey has been conducted on the works done in the past on the Reconfigurability of the FIR filters and various algorithms being used for making the FIR filters less complex.

Keywords—Canonical signed digit (CSD), CSHM, finite-impulse response (FIR) digital filters, Reconfigurability, subexpression elimination.

I. INTRODUCTION

FIR digital filters find extensive applications in mobile communication systems for applications. A software-defined radio system, or SDR, is a radio communication system where components that have been typically implemented in hardware like mixers, filters, amplifiers, modulators /demodulators, detectors, etc. are instead implemented by means of software on a personal computer or embedded computing devices. A basic **SDR system** is made up of a personal computer having a sound card, or other analog-to-digital converter, preceded by some form of RF front end. It makes the whole system to receive and transmit different radio protocols.SDR systems are very versatile. Wireless systems employ protocols that vary from one service to another. Even in the same type of service the protocol usually differs from one country to another. A single SDR set capable of working with any configuration can be used in any mode, anywhere in the world. Changing the service type, the mode, and/or the modulation protocol involves simply selecting and launching the requisite computer program, and making sure the batteries are adequately charged if portable operation is concerned. **Reconfigurability** of the receiver to work with different wireless communication standards is another key requirement in an SDR. **The binary common subexpression elimination method (BCSE) method** provides improved adder reductions and thus low complexity FIR filters. A method based on pseudo floating point method was used to encode the coefficients but it was limited to filter lengths less than 40 [1]. As is well known if the multiplier is

ISSN (Online) 2249 - 054 X
International Journal of Computing and Corporate Research
Specialized and Refereed Journal for Research Scholars, Academicians, Engineers and Scientists

http://www.ijccr.com
International Manuscript ID : ISSN2249054X-V2I4M3-072012
VOLUME 2 ISSUE 4 July 2012

represented in Canonical Signed Digit (CSD) form, then the number of additions (or subtractions) used will be a minimum. But it is not suited for reconfigurable filters. Several implementation approaches for reconfigurable FIR filters have been proposed in literature. These designs include either a fully programmable multiply-accumulate (MAC) based filter processor or dedicated architectures where the filter coefficients can be stored in registers. The multiply–accumulate operation is computes the product of two numbers and then adds that product to an accumulator. The hardware unit that performs the operation is known as a multiplier–accumulator (MAC, or MAC unit). The MAC operation modifies an accumulator 'a' according as:

$$a \leftarrow a + (b \times c).$$

The application of the multiplier blocks to the time-multiplexed digital filter designs is also being studied. It is shown that, the redundancy can be reduced and the resulting specialized multiplier design can be much more efficient in terms of area and computational complexity compared to the general-purpose multiplier with its associated coefficient store. This methodology was named **Reconfigurable Multiplier Blocks (ReMB)**. (ReMB) offer significant area, delay and possibly power reduction in time multiplexed implementation of multiple constant multiplications. Many algorithms have been proposed to make the multiplier block as simple as possible: Bull–Horrocks (BH) algorithm [3], n-dimensional reduced adder graph (RAGn) algorithm [2], recursive bipartite matching algorithm [4] etc. [5]

II. CANONIC SIGNED DIGIT MULTIPLICATION

A general method of carrying out multiplication by a constant value is by using a sequence of shifts and ads.

If the hardware components are to be more efficiently used then the existing method is modified by taking into account the subtractions also. Such a modification helps not only to decrease the number of additions as well as subtractions but also handles the negative multipliers. These results are best accomplished by using Canonic Signed Digit (CSD) form.

Figure-1 Unoptimized expression trees [7].

III. CHOICE OF SUBEXPRESSION

The choice of which subexpressions to choose at each iteration is not dependent only on the number of times an expression occurs, but rather on the expected reduction in number of operators. There are two types of operators in the circuit, adders and delay latches.

A. Estimation of the number of adders: The number of adders that will be eliminated by a common subexpression occurring *N* times is *N* - 1. The only trick in making sure that the count is accurate is to take care that overlapping subexpressions are not counted twice.

ISSN (Online) 2249 - 054 X
International Journal of Computing and Corporate Research
Specialized and Refereed Journal for Research Scholars, Academicians, Engineers and Scientists

http://www.ijccr.com
International Manuscript ID : ISSN2249054X-V2I4M3-072012
VOLUME 2 ISSUE 4 July 2012

B. Estimation of Latch Count: To get a good estimate of the cost of such a realization, it is necessary to make an estimate of latch count. The number of latches need will in general depend on the maximum number of adders that may be cascaded between two latches. In order to estimate accurately the number of latches needed, it is necessary to carry out a complete scheduling of the circuit, followed by latch insertion [6].

IV. SUBEXPRESSION ELIMINATION

Performing subexpression elimination can sometimes increase the number of registers substantially, and the overall area could possibly increase. Consider the two expressions F1 and F2 which could be part of the multiplier block.

F1 = A+B+C+D
F2 = A+B+C+E

Figure-2 Extracting common expression (A + B + C) [7]

Both the expressions have a minimum critical path of two addition cycles. These expressions require a total of six registered adders for the fastest implementation, and no extra registers are required. From the expressions we can see that the computation A + B + C is common to both the expressions. If we extract this subexpression, we get the structure shown in Figure 2. Since both D and E need to wait for two addition cycles to be added to (A + B + C), we need to use two registers each for D and E, such that new values for A,B,C,D and E can be read in at each clock cycle. Assuming that the cost of an adder and a register with the same bit width are the same, the structure shown in Figure 2 occupies more area than the one shown in Figure 1. A more careful subexpression elimination algorithm would only extract the common subexpression A + B (or A+C or B + C). The number of adders is decreased by one from the original, and no additional registers are added [7].

V. CONSTANT SHIFTS METHOD

The steps involved in CSM are as follows:

Step 1: Get the input x.
Step 2: Get the coefficients from the LUT and use as the select signal for the multiplexers.
Step 3: Perform the final shifting function on the output of the multiplexer.
Step 4: Perform the addition of intermediate sums using the final adder unit.
Step 5: Store the final result, h*x, in the delay unit ‘D’.
Step 6: Go to step 2 if the coefficients in the LUT are not finished, else go to 1.

The three most significant bits of the coefficient will be given as the select signal to the Mux1, the next 3-

ISSN (Online) 2249 - 054 X
International Journal of Computing and Corporate Research
Specialized and Refereed Journal for Research Scholars, Academicians, Engineers and Scientists
http://www.ijccr.com
International Manuscript ID : ISSN2249054X-V2I4M3-072012
VOLUME 2 ISSUE 4 July 2012

bits to Mux2 and so on till the least significant bits to the last multiplexer [8].

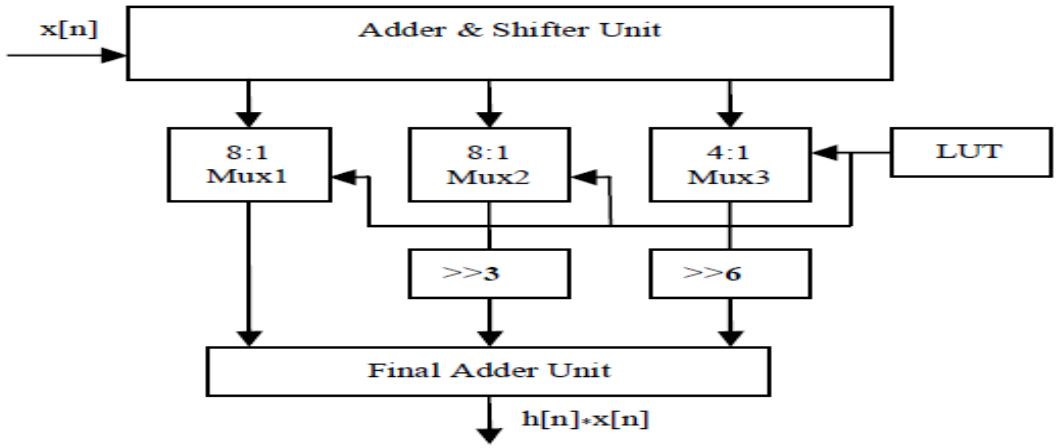

Figure-3 Constant Shifts Method [8]

VI. PROGRAMMABLE SHIFTS METHOD

The steps involved in PSM are as follows:

Step 1: Obtain the BCSs from filter coefficients using BCSE algorithm.
Step 2: Store the resultant coefficients in the prescribed format in the LUT.
Step 3: Get the input x.
Step 4: Get the coefficients from the LUT and use as the select signal for the multiplexers and the PS.
Step 5: Perform the final shifting function on the output of the multiplexer using PS.
Step 6: Perform the addition of intermediate sums using the final adder unit.
Step 7: Store the final result, h*x, in the delay unit 'D'.
Step 8: Go to step 4 if the coefficients in the LUT are not finished, else go to 3 [8].

ISSN (Online) 2249 - 054 X
International Journal of Computing and Corporate Research
Specialized and Refereed Journal for Research Scholars, Academicians, Engineers and Scientists
http://www.ijccr.com
International Manuscript ID : ISSN2249054X-V2I4M3-072012
VOLUME 2 ISSUE 4 July 2012

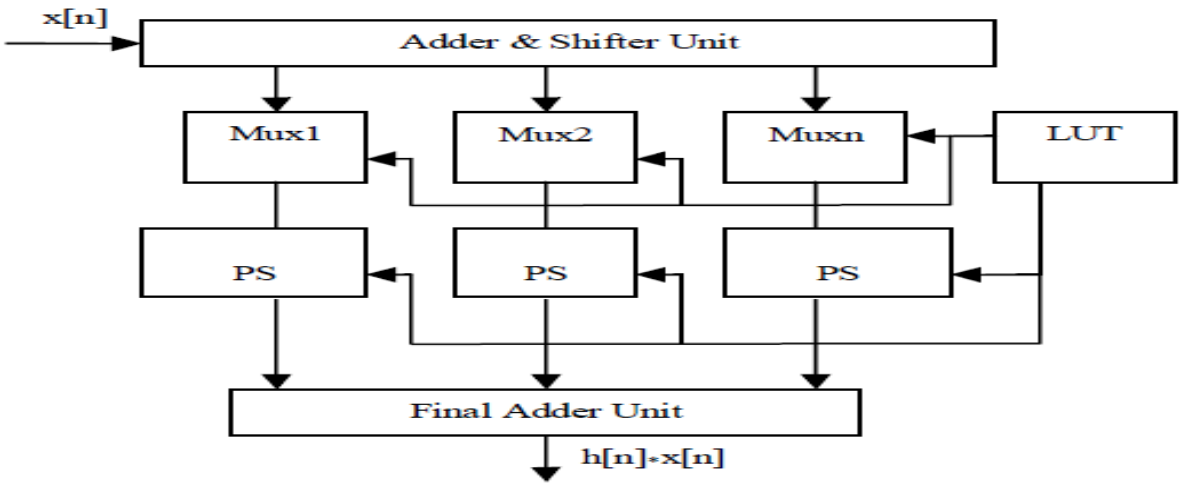

Figure-4 Programmable Shifts Method [8]

VII. COMPUTATION SHARING MULTIPLIER METHOD (CSHM)

A linear time invariant (LTI) FIR filter can be described by the following equation:

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

The ith coefficient c, can be obtained in the same manner. It can be represented as

$$C_i=\sum_{k=0}^{L} 2^{m_k} a_{k,j}$$

ISSN (Online) 2249 - 054 X
International Journal of Computing and Corporate Research
Specialized and Refereed Journal for Research Scholars, Academicians, Engineers and Scientists
http://www.ijccr.com
International Manuscript ID : ISSN2249054X-V2I4M3-072012
VOLUME 2 ISSUE 4 July 2012

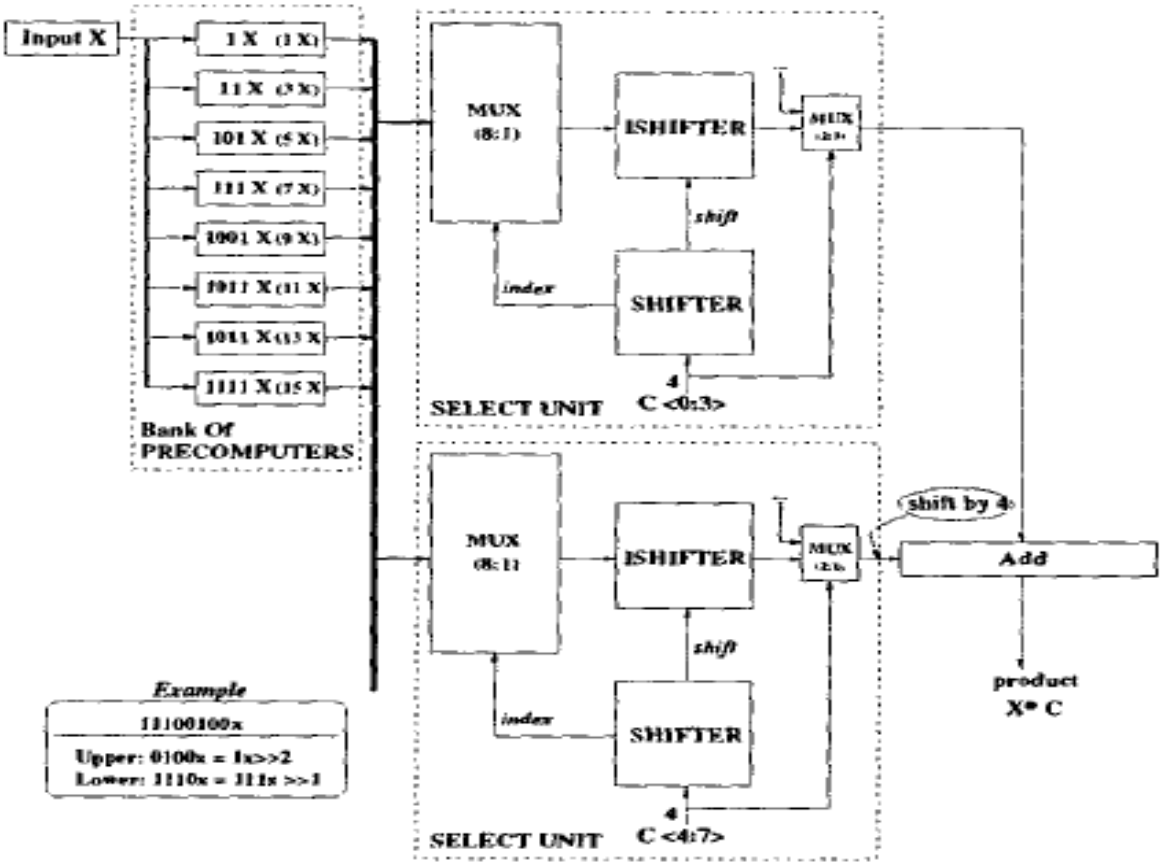

Figure-5 Architecture of CSHM [5]

The bank of precomputer performs the computations $a_k.x$, k=0, 1, 2. . . 8. As a result, the outputs of the precomputer bank are Ix, 3x, 5x, 7x, 9x, 11x, 13x, 15x. Shifters perform the right shift operation to find the correct alphabet and send an appropriate index signal to 8-to-1 MUXes. They also send the exact shifted values to I-Shifters. The MUXes select the correct answer among the 8 values received from Shifters, $a_k.x$, k=0, 1, 2, . . . , 8.

I-Shifters simply inverse the operation performed by Shifters. A 2-to-1 MUX is used to deal with the zero (0000) coefficient input. We refer to Shifter-MUX-I shifter- MUX as select unit. The upper select unit generates the multiplication of 4 LSB's of the coefficient with the input x. The lower select unit produces the product, of next 4 bits with input x. A shift of the upper 4 bits is performed when those two values are fed to the adder. A simple adder produces the final result [9].

ISSN (Online) 2249 - 054 X
International Journal of Computing and Corporate Research
Specialized and Refereed Journal for Research Scholars, Academicians, Engineers and Scientists

http://www.ijccr.com
International Manuscript ID : ISSN2249054X-V2I4M3-072012
VOLUME 2 ISSUE 4 July 2012

VIII. WALLACE TREE MULTIPLIER METHOD (WTMM)

A Wallace tree is an efficient hardwires implementation of a digital circuit that multiplies two integers.

The Wallace tree has three steps:

1. Multiply (that is - AND) each bit of one of the arguments, by each bit of the other, yielding n^2 results. Depending on position of the multiplied bits, the wires carry different weights, for example wire of bit carrying result of a_2b_3 is 32.
2. Reduce the number of partial products to two by layers of full and half adders.
3. Group the wires in two numbers, and add them with a conventional adder.

The demerits of this method are that Wallace trees do not provide any advantage over ripple adder trees in many FPGAs. Also due to the irregular routing, they may actually be slower and are certainly more difficult to route. Adder structure increases for increased bit multiplication. Wallace Tree multiplier can also be implemented using Carry Save Adders.

IX. CONCLUSION

In this paper, various methods related to the reduction of complexity and area of a FIR filter is proposed. The merits and demerits of the methods are also discussed. In the past few years a lot of work has been done on reduction of area and complexity but much less is being done on the aspect of Reconfigurability of the filter. The recent works are based on the Constant Shifts Method (CSM) and the Programmable Shifts Method (PSM). Reconfigurable filters are very much needed so as to suit themselves in all the different communication protocols without the change in the hardware of the FIR filter.

REFERENCES

[1]. R.Mahesh, A.P. Vinod, “New reconfigurable architectures for implementing FIR filters with low complexity”, IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS, VOL. 29, NO. 2, pp 275-288, FEBRUARY 2010.

[2] A. G. Dempster and M. D. Macleod, “Use of minimum adder multiplier blocks in FIR digital filters,” *IEEE Trans. Circuits Syst. II*, vol. 42, pp. 569–577, Sept. 1995.

[3] D. R. Bull and D. H. Horrocks, “Primitive operator digital filters,” *Proc. Inst. Elect. Eng.*, pt. G, vol. 138, pp. 401–412, Mar. 1991.

ISSN (Online) 2249 - 054 X
International Journal of Computing and Corporate Research
Specialized and Refereed Journal for Research Scholars, Academicians, Engineers and Scientists
http://www.ijccr.com
International Manuscript ID : ISSN2249054X-V2I4M3-072012
VOLUME 2 ISSUE 4 July 2012

[4] M. Potkonjak, M. B. Srivastava, and A. Chandrakasan, "Efficient substitution of multiple constant multiplications by shifts and additions using iterative pairwise matching" in *Proc. 31st ACM/IEEE Design Automation Conf.*, 1994, pp. 189–194.

[5] I. C. Park and H. J. Kang, "FIR filter synthesis algorithms for minimizing the delay and the number of adders," *IEEE Trans. Circuits Syst. II*, vol. 48, no. 8, pp. 770–777, Aug. 2001.

[6]. R. I. Hartley, "Subexpression sharing in filters using canonic signed digit multipliers," *IEEE Trans. Circuits Syst. II*, vol. 43, no. 10, pp. 677–688, Oct. 1996.

[7]. Shahnam Mirzaei, Anup Hosangadi, Ryan Kastner, "FPGA Implementation of High Speed FIR Filters Using Add and Shift Method", International Conference on Computer Design, (ICCD), pp 308-313, 2006.

[8]. R. Mahesh and A. P. Vinod, "Reconfigurable low cmplexity FIR filters for software radio receivers," in Proc. 17th IEEE Int. Symp. Personal Indoor Mobile Radio Commun. (PIMRC), Helsinki, Finland, Sep. 2006, pp. 1–5.

[9]. J. Park, H. Choo, K. Muhammad, and K. Roy, "Non-adaptive and adaptive filter implementation based on sharing multiplication," in Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing, June 2000, pp. 460–463.