International Journal of Emerging Engineering Research and Technology
Volume 3, Issue 11, November 2015, PP 15-22
ISSN 2349-4395 (Print) & ISSN 2349-4409 (Online)

Design and Implementation of Digit Serial Fir Filter

Mohammed Arif[1], K.S.Indrani[2]

[1]*Department of ECE, MREC (Autonomous), JNTUH, Hyderabad, India (PG Scholar)*
[2]*Department of ECE, MREC (Autonomous), JNTUH, Hyderabad, India (Associate Professor)*

ABSTRACT

In many digital signal processing systems, the multiple constant multiplication (MCM) operation has significant impact on the complexity and performance of the design because a large number of constant multiplications are required and is also a performance bottleneck in many other DSP systems. For this purpose many efficient algorithms and architectures such as shift-add and common sub-expression elimination (CSE) have been introduced for the design of low complexity MCM operations. These algorithms are very accurate but it contains complexity if suppose we increase the bit length. Therefore to overcome this disadvantage and for low complexity MCM operations we introduce an digit serial fir filter whose operators occupy less area and are independent of data word length. Thus the proposed method reduces the delay, area and power consumption. Hence the proposed method simulation results are shown as less delay and area utilization compared to existing common sub-expression elimination (CSE) method.

Keywords: Digit serial arithmetic, multiple constant multiplication (MCM), common sub-expression elimination, finite impulse response (FIR) filters.

INTRODUCTION

In digital signal processing (DSP) systems such as fast Fourier transforms, discrete cosine transforms (DCT's), and error correcting codes, Finite impulse response (FIR) filters have great importance because of their linear- phase characteristics and feed forward implementations. The two forms of FIR filters i.e, direct and transposed form FIR filter implementations are illustrated in fig 1(a) and (b).

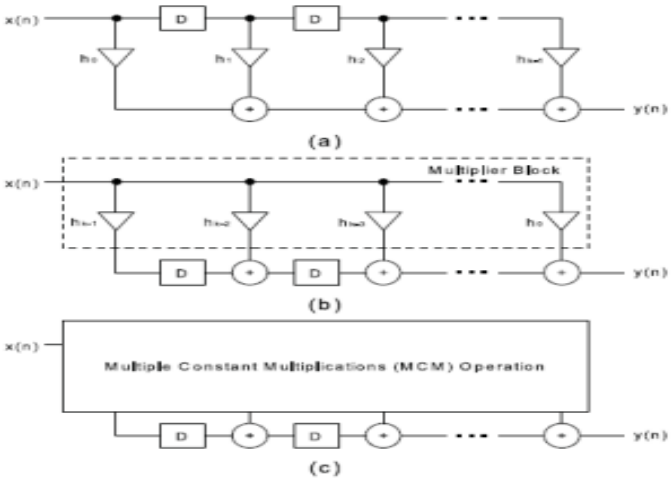

Figure1. *FIR filters implementations. (a) Direct form. (b) Transposed form with generic multipliers. (c) Transposed form with an MCM block.*

Although both architecture have similar complexity in hardware, the transposed form is generally preferred because of its higher performance and power efficiency [1]. The multiplier block of the digital FIR filter in its transposed form, where the multiplication of filter coefficients with the filter

Address for correspondence:

arif.mohd29691@gmail.com

input is realized, has significant impact on the complexity and performance of the design because a large number of constant multiplications are required. This is generally known as multiple constant multiplications (MCM) operation. Although area, delay, and power- efficient multiplier architectures, such as Wallace [2] and modified booth [3] multipliers, have been proposed, the full flexibility of a multiplier is not necessary for the constant multiplications, since filter coefficients are fixed and determined beforehand by the DSP algorithms [4].

Shift- Add Architecture

In this method the multiplication of filter coefficients with the input data is generally implemented under a shift adds architecture [5], where each constant multiplication is realized using addition/subtraction and shift operations in an MCM operation [Fig 1(c)]. For the shift adds Implementation of constant multiplications, a straightforward method, generally known as digit based recoding [6], initially defines the constant in binary. Then, for each "1" in the binary representation of the constant, according to its bit position, it shifts the variable and adds up the shifted variables to obtain the result.

As a simple example, consider the constant multiplications 29x and 43x.Their decompositions in binary are listed as follows: Which requires six addition operations as illustrated in Fig. 2(a).

$$29x = (11101)\, b_{in}{}^{x} = x \ll 4 + x \ll 3 + x \ll 2 + x \quad (1)$$

$$43x = (101011)\, b_{in}{}^{x} = x \ll 4 + x \ll 3 + x \ll 2 + x \quad (2)$$

EXISTING SYSTEM

MCM is involved to produce constant multiplication in Digital Signal Processing (DSP) systems, MIMO (Multiple Input Multiple Output) systems, Error correcting codes, Frequency multiplication, Graphics and Control applications. In such applications full fledge of multipliers are not needed. Since coefficients are constant to produce constant multiplication. Once the MCM architecture is constructed, it can be called as many times it required. Constant multiplication either can be done by digit parallel design or digit serial design. Digit parallel design of constant multiplier needs external wire for shifting. It requires more area while implementation takes place in FPGA or any other ASIC. Hence digit serial design overcomes area constrain with acceptable delay timing. Multiplication with constant is called constant multiplication.

Common Sub-Expression Elimination Method

In existing, in order to design MCM architecture without partial product sharing algorithm (Digit based recoding), Common Sub-expression Elimination (CSE) algorithm [7]-[9] is used in existing methods. According to MCM principle constant multiplication is performed by number of shifting and addition operation. For the pair, 29x and 43x without partial product sharing algorithm requires six addition and six shifting operations and CSE algorithm requires four additions and four shifting operations. As shown in below figure 2(a) and 2(b).

However, the digit-based recoding technique does not exploit the sharing of common partial products, which allows great reductions in the number of operations and consequently and consequently, in area and power dissipation of the MCM design at the gate level. Hence, the fundamental optimization problem , called the MCM problem, is defined as finding the minimum number of addition and subtraction operations that implement the constant multiplications. Note that, in parallel design of constant multiplications, shifts can be realized using only wires in hardware without representing any area cost.

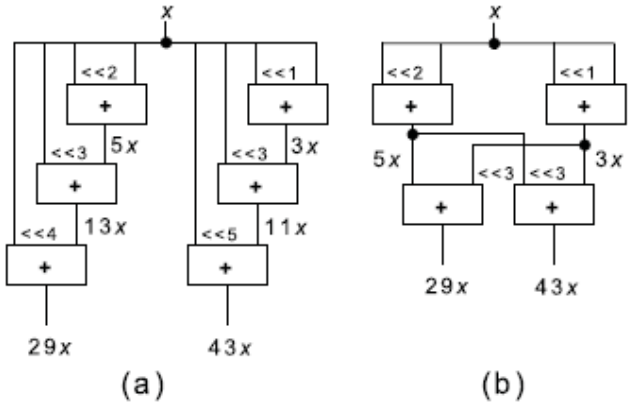

Figure2. *Shift-adds implementations of 29x and 43x. (a) Without partial product sharing [7] and with partial product sharing. (b) Exact CSE algorithm [10]*

The algorithm design for the MCM problem is common sub expression elimination(CSE) algorithm initially extract all possible sub expressions from the representation of the constants when they are defined under binary ,canonical signed digit(CSD) [7], or minimal signed digit(MSD) [8]. Then they find the "best" sub expression, generally the most common, to be shared among the constant multiplications. Returning to our example in fig.2 the exact CSE algorithm gives a solution with four operations by finding the most common partial products 3x=(11)binary x and 5x=(101)binary x when constants are defined under binary, as illustrated in fig.2. Compare to the digit based recoding technique CSE MCM algorithm is very accurate but it contains complexity if suppose we increase the bit-length. We know that whenever we increase the bit length the architecture of multiplier will gets changes so that changing the bit length will increases the architecture in MCM. So, because of this disadvantage what happens was the efficiency of the system will gets reduced because with the increased architecture delay will be more so that area, power will gets increases. In order to overcome these constraints we proposed a new method which is a Digit-series FIR Filter

PROPOSED SYSTEM

Here in proposed method which is a Digit-Series FIR Filter, here we are increasing the bit length so that what happens means even though we are increasing the bit length , architecture will not gets increases it just divides the bit length [13] so that there is no such a complexity exists in this new method. We implemented for both 29x,43x,59,89x, so in this we are going to compare the delay area, efficiency and power between the existing method and proposed method. So compare to existing method proposed method must gives the less delay compare to previous method.

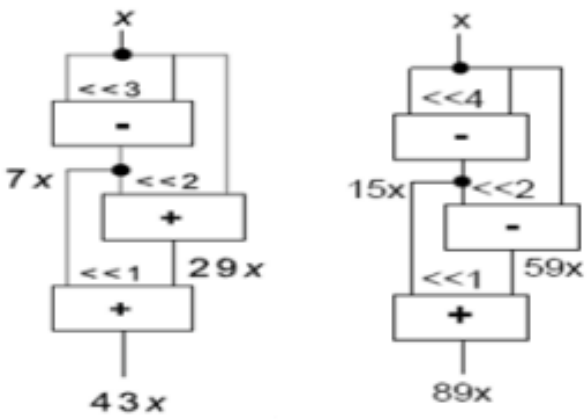

Figure3. *29,43,59,89 coefficient for Digit-series FIR Filters*

However, in existed method the algorithms assume that the input data x is processed in parallel. On the other hand, in digit serial arithmetic, the data words are divided into digit sets, consisting of d bits, that are processed one at a time. Since digit-serial architectures offer alternative low complexity designs when compared to the bit-parallel MCM design.

29x = x<<3+x<<2+x.

43x = x<<2+x<<1+x.

59x = x<<4 - x<<2 – x.

89 = x<<2 – x<<1 – x.

In existing method the output for each and every coefficients will needs only one clock cycle. In proposed method the output which are coefficients will needs atleast two or more than two clock cycles i.e, the output will comes after some clock cycles but the complexity will be less so that even though it needs more clock cycles to retrieve the output delay will be very less and efficiency will be more.

RESULTS AND DISSCUSION

The proposed design has been simulated using Xilinx and Modelsim, the wave form obtained after simulating is as shown in below figs.

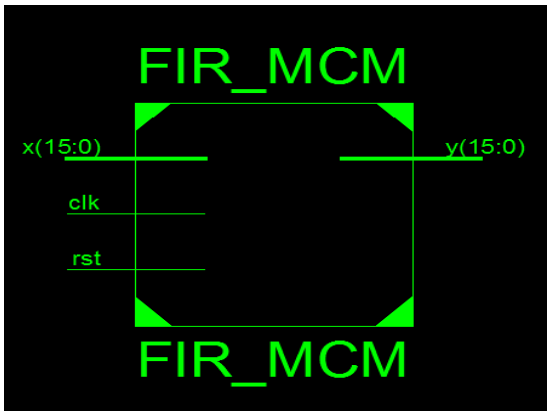

Figure4. *Block diagram of FIR_MCM*

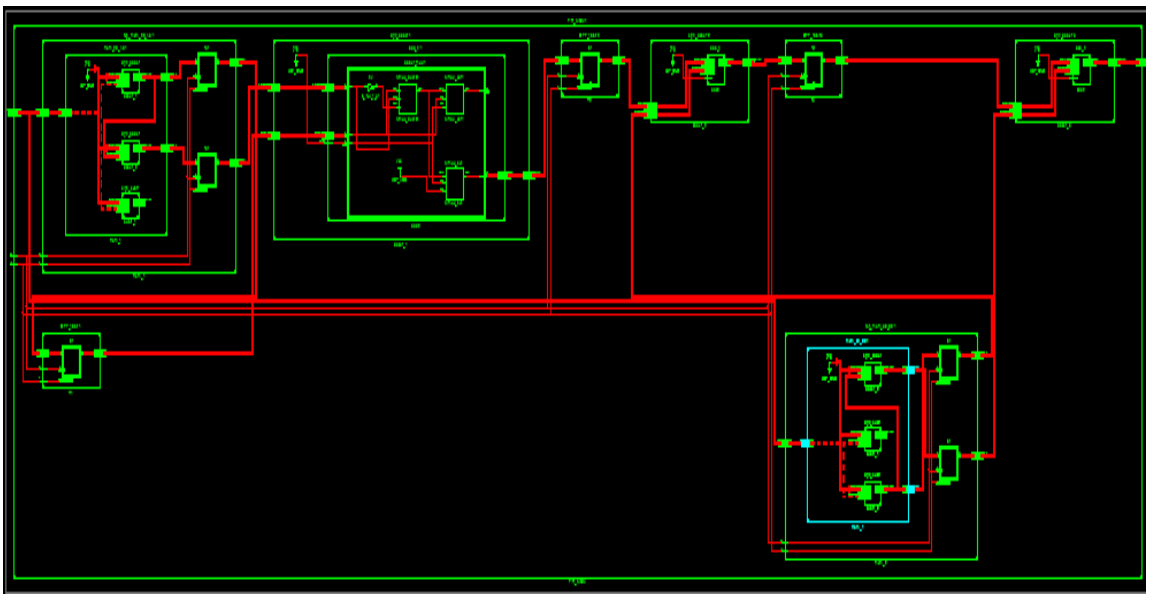

Figure5. *RTL ARCHITECTURE of FIR MCM*

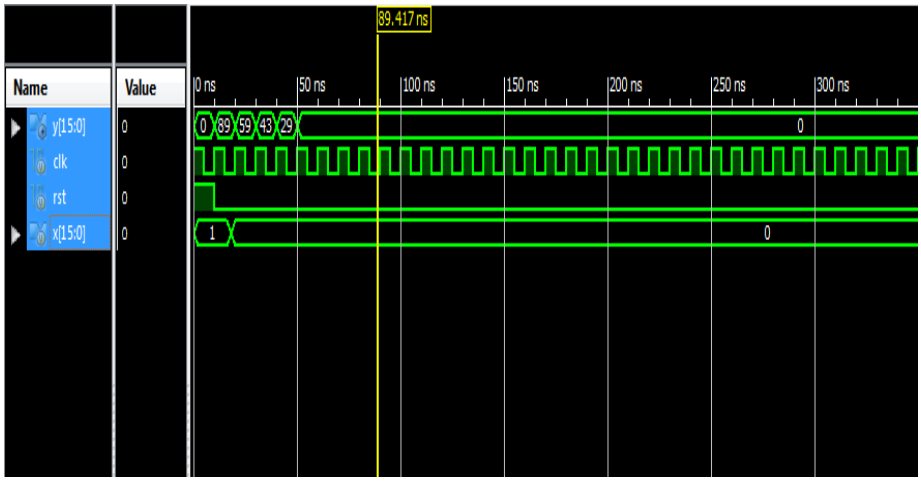

Figure6. *Output Waveform of FIR MCM*

Table1. *Delay for Existing FIR MCM*

```
Timing constraint: Default OFFSET OUT AFTER for Clock 'clk'
  Total number of paths / destination ports: 272 / 16
--------------------------------------------------------------------------------
Offset:              10.889ns (Levels of Logic = 9)
  Source:            mcm_1/m_43_0 (FF)
  Destination:       y<14> (PAD)
  Source Clock:      clk rising
```

Table2. *Area of Existing FIR MCM*

Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slice Registers	105	18224	0%
Number of Slice LUTs	271	9112	2%
Number of fully used LUT-FF pairs	84	292	28%
Number of bonded IOBs	34	232	14%
Number of BUFG/BUFGCTRLs	1	16	6%

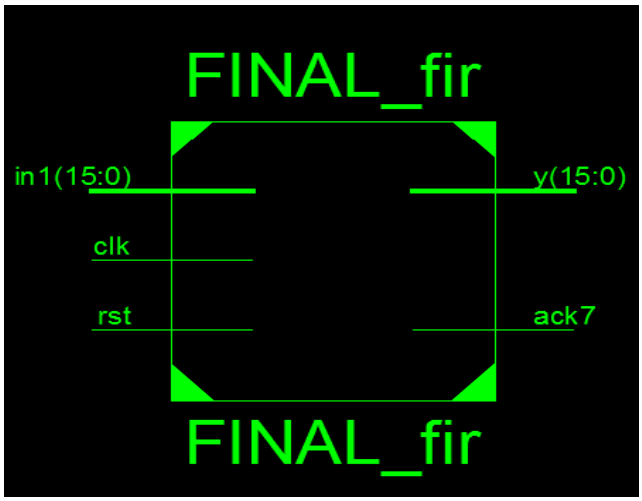

Figure7. *Block Diagram for Digit Serial FIR FILTER*

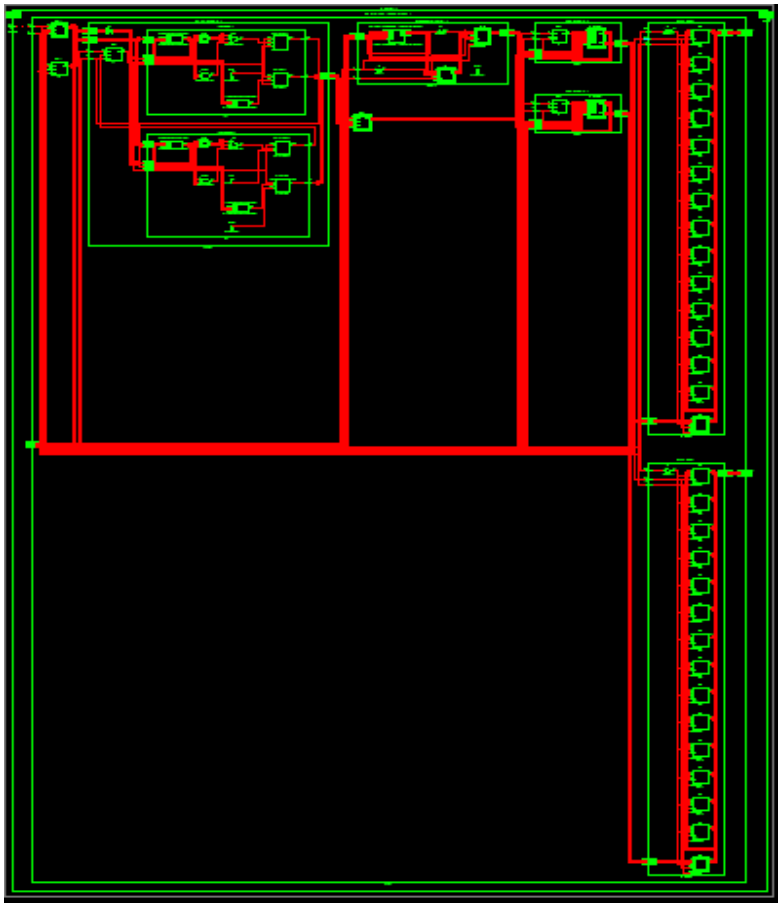

Figure8. *RTL schematic architecture for DIGIT SERIAL FIR FILTER*

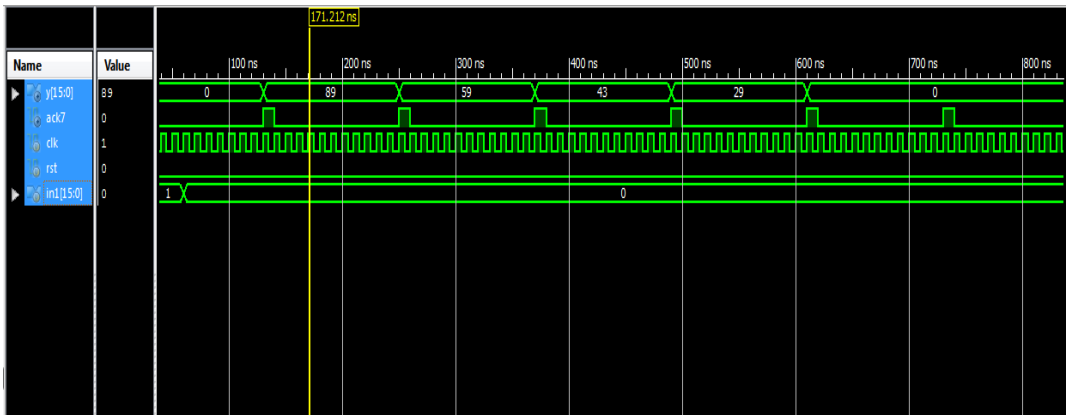

Figure9. *Output Waveform for Digit Serial FIR FILTER*

Table3. *Delay for DIGIR SERIAL FIR FILTER*

```
Timing constraint: Default OFFSET OUT AFTER for Clock 'clk'
  Total number of paths / destination ports: 17 / 17
--------------------------------------------------------------------------------
Offset:              4.323ns (Levels of Logic = 1)
  Source:            adder_3/ack (FF)
  Destination:       ack7 (PAD)
  Source Clock:      clk rising

  Data Path: adder_3/ack to ack7
                                Gate     Net
    Cell:in->out      fanout   Delay   Delay  Logical Name (Net Name)
    ----------------------------------------  ------------
     FDR:C->Q             33   0.447   1.305  adder_3/ack (adder_3/ack)
     OBUF:I->O                 2.571          ack7_OBUF (ack7)
    ----------------------------------------
    Total                      4.323ns (3.018ns logic, 1.305ns route)
                                       (69.8% logic, 30.2% route)
```

Table4. *Area of SERIAL FIR FILTER*

Device Utilization Summary (estimated values)			[-]
Logic Utilization	Used	Available	Utilization
Number of Slice Registers	264	18224	1%
Number of Slice LUTs	182	9112	1%
Number of fully used LUT-FF pairs	90	356	25%
Number of bonded IOBs	35	232	15%
Number of BUFG/BUFGCTRLs	1	16	6%

CONCLUSION

In this paper, we introduce the Digit Serial FIR Filter it divides the input bit-length and then by using shifting, addition, subtraction it process its operation and gives the output at different clock cycles. But in previous method it was designed by using MCM, in this method it has high complexity but it gives the output at single clock cycle only. The main difference between the existed and proposed method was about complexity, area, delay and power .In, existing the output need only single clock cycle but in proposed the output achieves at different clock cycle which is one of the constraint in this proposed method. Here, we are comparing Delay, Area, Power and Efficiency with the previous method which is a MCM method. So, based on Experimental results the proposed method got less delay and area and efficiency got increased when compared to the previous method.

ACKNOWLEGEMENT

I am very much thankful to my project guide Mrs. K.S.INDRANI, Associate Professor, for his extensive patience and guidance throughout my project work.

I am grateful to our Project Coordinator, Mr.Kesavan Gopal, Associate Professor, for extending his support and cooperation throughout my project work.

I would like to thank Dr. M. Ch. P. Jagadesh, Head of the ECE Department, for providing the freedom to use all the facilities available in the department, for successful completion of project.

I whole-heartedly express my deep sense of gratitude to our beloved Principal, Dr. S.SUDHAKAR REDDY, Principal, MALLA REDDY ENGINEERING COLLEGE (Autonomous) for providing me adequate facilities for my successful completion of both course work and project work.

I sincerely thank all the staff of the Department, for their timely suggestions, healthy criticism and motivation during the course of my project work.

I would also like to thank all my friends for providing help and moral support at the right timing. With great affection and love, I thank my parents who were the backbone behind my deeds.

REFERENCES

[1] L. Wanhammar, DSP Integrated Circuits. New York: Academic, 1999.

[2] C. Wallace, "A suggestion for a fast multiplier," IEEE Trans. Electron. Comput., vol. 13, no. 1, pp. 14–17, Feb. 1964.

[3] W. Gallagher and E. Swartzlander, "High radix booth multipliers using reduced area adder trees," in Proc. Asilomar Conf. Signals, Syst. Comput., vol. 1. Pacific Grove, CA, Oct.–Nov. 1994, pp. 545–549.

[4] J. McClellan, T. Parks, and L. Rabiner, "A computer program for designing optimum FIR linear phase digital filters, "IEEE Trans. Audio Electroacoust., vol. 21, no. 6, pp. 506–526, Dec. 1973.

[5] H. Nguyen and A. Chatterjee, "Number-splitting with shift-and-add decomposition for power and hardware optimization in linear DSP synthesis," IEEE Trans. Very Large Scale Integer. (VLSI) Syst., vol.8, no. 4, pp. 419–424, Aug. 2000.

[6] M. Ercegovac and T. Lang, Digital Arithmetic. San Mateo, CA: Morgan Kaufmann, 2003.

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

[8] I.-C. Park and H.-J. Kang, "Digital filter synthesis based on minimal signed digit representation," inProc. DAC, 2001, pp. 468–473

[9] L. Aksoy, E. Costa, P. Flores, and J. Monteiro, "Exact and approximate algorithms for the optimization of area and delay in multiple constant multiplications, "IEEE Trans. Comput.-Aided Design Integer. Circuits Syst., vol. 27, no. 6, pp. 1013–1026, Jun. 2008.

[10] A.Dempster and M. Macleod, "Use of minimum-adder multiplier blocks in FIR digital filters," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 42, no. 9, pp. 569–577, Sep. 1995.

[11] Y. Voronenko and M. Püschel, "Multiplierless multiple constant multiplication,"ACM Trans. Algor., vol. 3, no. 2, pp. 1–39, May 2007.

[12] L. Aksoy, E. Gunes, and P. Flores, "Search algorithms for the multiple constant multiplications problem: Exact and approximate," J. Microprocess. Microsyst., vol. 34, no. 5, pp. 151–162, Aug. 2010.

[13] R. Hartley and k. Parhi, Digit-serial computation. Norwell, ma: kluwer,1995.

AUTHOR'S BIOGRAPHY

MOHAMMED ARIF

B.Tech Passed in 2013 in ECE branch from Shadan College of engineering and Technology, JNTUH, Ranga Reddy.

M.Tech from Malla Reddy Engineering College (AUTONOMOUS)

Branch: Digital Systems and Computer Electronics, Dept of ECE