IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

SPEED MULTIPLIERS USING FUZZY TECHNIQUE COMPLEXITY METHOD FOR IIR FILTERS

M.Vinoth
Assistant Professor, Dept., of ECE
Sri Chandrasekharendra Saraswathi Viswa Mahavidyalaya,
vinothmurali@kanchiuniv.ac.in Kancheepuram, Tamilnadu – 631561

R D S Kashyap, Vinay Chand P, Manohar, Vinay Vithesh
Students, Sri Chandrasekharendra Saraswathi Viswa Mahavidyalaya, Kancheepuram, Tamilnadu – 631561.

***Abstract*- FIR filter plays an major role in the emerging wireless communication and Digital Signal Processing (DSP) applications. Designing multiplier less FIR filter for low complexity reduces the overall performance. Hence the optimization technique is proposed to optimize filter coefficients of finite-impulse response (FIR) filter to share common sub expressions within and among coefficients. The Binary Common Sub expression (BCS) algorithm is done to prove its efficiency over all other existing methods. The BCS technique uses partial product generator which forms the partial products for the input signal and coefficients. The complexity and speed can be improved and it is compared with all other existing methods. The simulation was done using Modelsim and the hardware synthesis was done using EDA tools.**

***Keywords*-Finite Impulse Response (FIR) filter, Digital Signal Processing (DSP), Binary Common Sub (BCS) expression algorithm, Electronic Design Automation (EDA).**

I. Introduction

The rapid growth in mobile computations and multimedia related applications are the main cause of low power Digital Signal Processing (DSP) systems. The finite impulse response filtering is the most widely used operation performed in digital signal processing[1]-[4]. In digital signal processing, filters are classified into various types based on its parameters. The most important types of digital filters are FIR filters and IIR filters. Both types have some advantages and disadvantages that should be carefully considered when designing a filter[5]. Besides, it is necessary to take into account all fundamental characteristics of a signal to be filtered as these are very important when deciding which filter to use. In most cases, it is only one characteristic that really matters and it is whether it is necessary that filter has linear phase characteristic or not[6]. The FIR filter has linear phase characteristic and higher order stability. The multipliers are designed in such a way to reduce the area and complexity by various methods.

Initially the multiplier based method is used to perform the basic multiplication with the multiplicand and multiplier. This method has various steps to perform the basic multiplier operation and consumes area. Since it has various multipliers its complexity is higher and in turn reduces the speed of the circuit[7]. To reduce the number of multipliers the shift/add method is used and it can perform the multiplier operation by shifting and adding the digits by left or right. This method adds the multiplicand X to itself Y times, where Y denotes the multiplier[8]. To multiply two numbers, the algorithm is to take the digits of the multiplier one at a time from right to left, multiplying the multiplicand by a single digit of the multiplier and placing the intermediate product in the appropriate positions to the left of the earlier results This can perform well for low order bits and fails to show its performance on higher order bits.

The CSD based technique well suits for the multiplier operation between constant coefficients and inputs by add and shift operation[9][10]. Though it performs shift and add operation it uses redundant adder for multiplier block and hence low hardware footprints can be achieved for higher order filters.

II. Related Works

Shyh-Feng Lin proposed a low-power architecture for dedicated linear phase FIR filter Four schemes are suggested, including retimed structure, balanced modular architecture, separated signed processing data flow and modification of the CSD representation. It compared the four structures with their area, power, vdd and frequency. The proposed signal processing schemes reduce circuits' transition in the accumulation path to achieve the maximum efficiency of hardware components. The proposed schemes not only address the linear-phase FIR filter, but also can improve the non linear-phase FIR filer.

Pramod Kumar Meher designed a one dimensional and two dimensional pipelined architecture computing area, delay, power relation of FIR filter by systolic array decomposition using distributed arithmetic. The hardware-efficient memory-based realization of FIR filters can be obtained by appropriate choice of address length of the LUTs used for the DA-based computation. The performance measures like number of slices, maximum usable frequency, dynamic power consumption, energy density, and energy throughput are estimated for different filter orders and address lengths. The FPGA realization is fully parameterized, modular and scalable, so that it can be readily used as an IP core in a number of environments.

Jiafeng Xie used fully pipelined architectures for area, time, power- efficient implementation of finite impulse response (FIR) filter. This architecture helps to obtain a suitable tradeoff and analysis the performance of different filter order for various address lengths. This power efficient FIR filter is compared with the LUT less DA architecture of FIR filter. The parameters like number of LEs, maximum frequency, power consumption are compared for different filter orders and address lengths. This pipelined architecture not only reduces area but also has great significance on the speed of the FIR filter.

Dong Shi proposed an algorithm based on Mixed Linear Integer Programming (MILP) traverses the discrete coefficients for the optimum result and in turn reduces the number of adders in the circuit. In addition to the proposed algorithm a monitoring mechanism is introduced for the awareness of optimality. This can be done with design examples and traversed with sub expression process to achieve maximum adder constraint in the FIR filter design.

Shen-Fu Hsiao designed low cost FIR filter by using the rounded truncated multipliers. The non uniform coefficient quantization with proper FIR filter is proposed to reduce the total area cost. Multiple constant multiplication/accumulation in direct FIR filter structure is implemented using an improved truncated multipliers. The optimization technique and low cost FIR filter design is implemented by considering the it width and hardware resources. This leads to small area cost and reduced power consumption in the system.

III. Proposed Work

The Distributed Arithmetic (DA) representation is used for the processing of any digital filters and the example is shown below for the inner product design.

$$y=\sum_{n=0}^{N-1} A_n * X_n$$

If A_n is constant and X_n is variable then X_n can be represented in DA as

$$X_n=\sum_{b=0}^{B-1} 2^b x_{n,b}\text{, where b is the bit number}$$

The proposed block diagram for the Binary Common Sub expression (BCS) method is shown in Figure 1. The blocks such as pre shifter, multiplexer, accumulator and two's complement unit are present. The input signal and the coefficients are taken as inputs and applied those inputs for BCS technique. These BCS method is proposed for linear phase FIR filter to increase its efficiency by optimizing various parameters.

A. Coded Coefficients

The coefficients are the inputs which are used to sample the input signal based on the interpolation factor of the filters. In this paper, 10 tap filter is used for the design example and this means there are ten sub filter is used in the filter design. The sub filters are used to generate the appropriate full FIR filter response.

B. Pre shifting

The input signals are taken and it is applied to the pre shifter. The pre shifter converts the input signal in binary form and performs some shifting process by which it reduces the work of multiplexer unit. For example, if 8 bit value is taken as input it changes the continues 1's as 0's and ends up with one's complement.

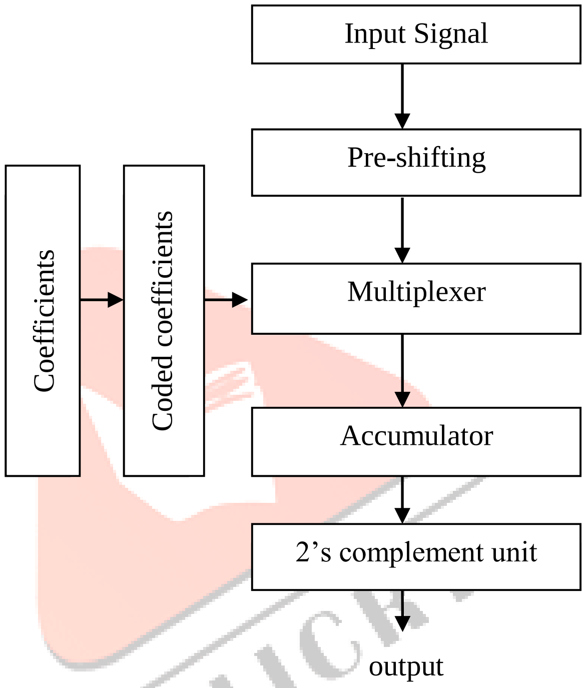


Fig 1: Block Diagram for BCS method

C. Multiplexer

The multiplexer will select the appropriate data from the partial product generator based on coded coefficients and the input signal. The 4:1 multiplexer is used to produce the partial product and this partial product generated will be added to the other coefficients to perform the multiplication process.

The partial product generator is used to generate the partial products among the filter coefficients and the input samples during multiplication.

The partial product for N-tap FIR filter is given by the following equation as

$$y[n]= \sum_{i=0}^{N-1} h.x[n-i]$$

In this BCS method, the partial product generator computes the value for all the input combinations and it is redundant for the same value. Hence, it can greatly reduce the operation time and hardware complexity.

D. Accumulator

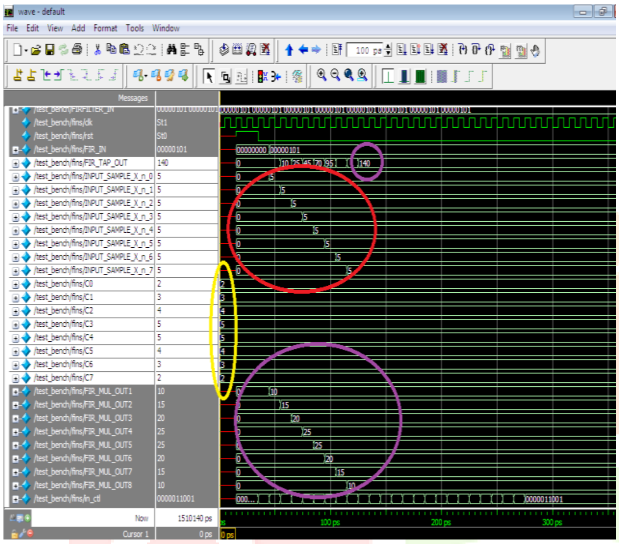
The accumulator basic operation is adding all the sub filter outputs. The output from various multiplexer unit is taken and it is summed to form the output. The value from the overall sum of the sub filter acts passes to the two's complement unit. This two's complement unit changes the magnitude of the output bits according to its coded coefficients.

The BCS representation of signal for n-bit binary number forms $2^{(n-1)}$-1 among themselves. The 4 bit binary values can form 7 BCS representations are [11], [101], [111], [1111], [1101], [1011] and [1001].

The common sub expressions [0011], [1100] and [0110] are implemented using [11], and the common sub expressions [0101],[1010] are implemented using [101].By this way the common sub expressions are formed and multiplied with the coded coefficients to get the multiplied value.

IV. Result

The simulation result for Binary Common Sub expression method was done for 10-tap FIR filters.



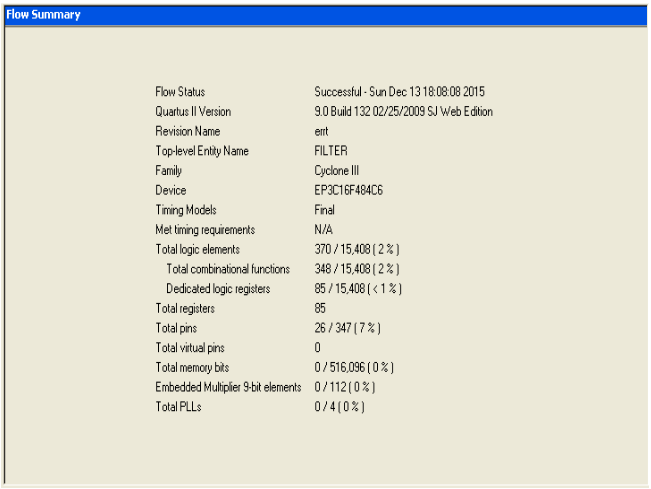
Messages	
/test_bench/FIRFILTER_IN	00000101 00000101
/test_bench/fins/clk	St1
/test_bench/fins/rst	St0
/test_bench/fins/FIR_IN	00000101
/test_bench/fins/FIR_TAP_OUT	140
/test_bench/fins/INPUT_SAMPLE_X_n_0	5
/test_bench/fins/INPUT_SAMPLE_X_n_1	5
/test_bench/fins/INPUT_SAMPLE_X_n_2	5
/test_bench/fins/INPUT_SAMPLE_X_n_3	5
/test_bench/fins/INPUT_SAMPLE_X_n_4	5
/test_bench/fins/INPUT_SAMPLE_X_n_5	5
/test_bench/fins/INPUT_SAMPLE_X_n_6	5
/test_bench/fins/INPUT_SAMPLE_X_n_7	5
/test_bench/fins/C0	2
/test_bench/fins/C1	3
/test_bench/fins/C2	4
/test_bench/fins/C3	5
/test_bench/fins/C4	5
/test_bench/fins/C5	4
/test_bench/fins/C6	3
/test_bench/fins/C7	2
/test_bench/fins/FIR_MUL_OUT1	10
/test_bench/fins/FIR_MUL_OUT2	15
/test_bench/fins/FIR_MUL_OUT3	20
/test_bench/fins/FIR_MUL_OUT4	25
/test_bench/fins/FIR_MUL_OUT5	25
/test_bench/fins/FIR_MUL_OUT6	20
/test_bench/fins/FIR_MUL_OUT7	15
/test_bench/fins/FIR_MUL_OUT8	10
/test_bench/fins/in_ctl	0000011001

Fig 2: Simulation of BCS method

The output is the FIR convolution of input signals and the coded coefficients. The 10-tap FIR filter shows that there are ten sub filters used for the simulation.

The simulated result given in figure 1 is the multiplied value of the coded coefficients and the input signal. The simulation of BCS was done using Modelsim software and the analysis was made using Altera Quartus software. The area utilization report gives the total logic elements used by the circuit.

This area utilization report is shown in Fig 3. This can be used for the comparison over other existing methods.



Flow Status	Successful - Sun Dec 13 18:08:08 2015
Quartus II Version	9.0 Build 132 02/25/2009 SJ Web Edition
Revision Name	errt
Top-level Entity Name	FILTER
Family	Cyclone III
Device	EP3C16F484C6
Timing Models	Final
Met timing requirements	N/A
Total logic elements	370 / 15,408 (2 %)
Total combinational functions	348 / 15,408 (2 %)
Dedicated logic registers	85 / 15,408 (< 1 %)
Total registers	85
Total pins	26 / 347 (7 %)
Total virtual pins	0
Total memory bits	0 / 516,096 (0 %)
Embedded Multiplier 9-bit elements	0 / 112 (0 %)
Total PLLs	0 / 4 (0 %)

Fig 3: Area utilization result

V. Comparison and Discussion

To know the performance of proposed method, the comparison has to be made for proposed method over all other methods.

FIR type used	Multiplier used	LE's used
MUL based	10	591
Shift/add method	NIL	666
CSD	NIL	602
BCS	NIL	370

Table 1: Trade off analyzes of various methods with QUARTUS II hardware synthesis using CYCLONE III family

The data given in table 1 shows the comparison between proposed system with other existing methods for the parameters logic elements and multipliers used.

The other comparison is shown in table 2 which is given below. It shows the comparison between logic elements used and the maximum frequency.

FIR type used	LE's used	Fmax
Mul based	591	150.31MHz
Shift/add method	666	81.67 MHz
CSD	602	137.01 MHz
BCS	370	312.79 MHz

Table 2: Frequency comparison of various methods with QUARTUS II hardware synthesis using CYCLONE III family

The fmax frequency given in table 2 is used to find the operating speed of the given circuit as the frequency is inversely proportional to time. The number of multipliers and logic elements used actually increases the complexity of the circuit and hence it can be reduced for the optimum result. The BCS method uses 370 logic elements and this value is minimum when compared with the other existing methods.

VI. Conclusion

In this paper, the BCS method is proposed to increase the overall performance of FIR filters. The BCS method is done in the multiplier section to reduce the complexity of the circuit by generating the partial product generators. The number of multiplexers are used based on the input signals and coded coefficients. The BCS method is simulated in Modelsim software and obtained the analysis of power, logic elements used, area using Quartus. The obtained result is compared with the existing methods and the comparison table is presented. The comparison table shows that BCS has increased speed and less complexity.

The work may be extended using booth multiplier and vedic multiplier. The booth algorithm used radix-4 and radix-8 according to the number of inputs or number of multiplications. The vedic multiplier uses some conventional methods for the multiplier in FIR filters. These methods can also be compared and the performance is evaluated.

References

[1]. S.-F. Lin, S.-C. Huang, F.-S. Yang, C.-W. Ku, and L.-G. Chen, "Power-efficient FIR filter architecture design for wireless embedded system," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 51, no. 1, pp. 21–25, Jan. 2004.

[2]. Y. Son, K. Ryoo, and Y. Kim, "1:4 interpolation FIR filter," IEEE Electron. Lett., vol. 40, no. 25, pp. 1570–1572, Dec. 2004.

[3]. J. Xie, J. He, and G. Tan, "FPGA realization of FIR filters for high-speed and medium-speed by using modified distributed arithmetic architectures," Microelectron. J., vol. 41, no. 6, pp. 365–371, Jun. 2010.

[4]. P. K. Meher, S. Chandrasekaran, and A. Amira, "FPGA realization of FIR filters by efficient and flexible systolization using distributed arithmetic," IEEE Trans. Signal Process., vol. 56, no. 7, pp. 3009–3017, Jul. 2008.

[5]. K.-H. Chen and T.-D. Chieueh, "A low-power digit-based reconfigurable FIR filter," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 53, no. 8, pp. 617–621, Aug. 2006.

[6]. Xilinx Inc., San Jose, CA, USA. (2014). Xilinx Logicore Multiplier IP version v9.0, [Online]. Available: http://www.xilinx.com/support/documentation/white_papers/wp277.pdf

[7]. J. Chandran, R. Kaluri, J. Singh, V. Owall, and R. Velijanovski, "Xilinx Virtex II Pro implementation of a reconfigurable UMTS digital channel filter," in Proc. IEEE Workshop Electron. Des., Test and Appl., Jan. 2004, pp. 77–82.

[8]. F. Sheikh, M. Miller, B. Richards, D. Markovic, and B. Nikolic, "A 1–190 MSample/s 8–64 tap energy-efficient reconfigurable FIR filter for multi-mode wireless communication," in Proc. IEEE Symp. VLSI Circuits, Jun. 2010, pp. 207–208.

[9]. S.-J. Lee, J.-W. Choi, S. W. Kim, and J. Park, "A reconfigurable FIR filter architecture to trade off filter performance for dynamic power consumption," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 19, no. 12, pp. 2221–2228, Dec. 2011.

[10]. S.-F. Hsiao, J.-H. Zhang Jian, and M.-C. Chen, "Low-cost FIR filter designs based on faithfully rounded truncated multiple constant multiplication/accumulation," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 60, no. 5, pp. 287–291, May 2013.