IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)
e-ISSN: 2278-2834,p- ISSN: 2278-8735.
PP 162-166
www.iosrjournals.org

Design Of Multiplierless Fir Filter Using Graph Based Optimization

Hariprasad[1] ,Suresh Krishna S[2]
[1](Student,Arulmigu Meenakshi Amman College Of Engineering,Kanchipuram)

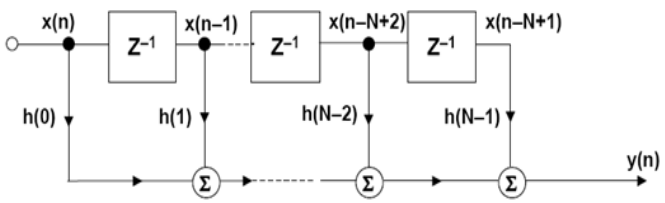
ABSTRACT: *The FIR filter is play major role on DSP applications. But the design complexity and the different structure of the FIR filter is complex to design which is limiting the uses of FIR filter. Mostly direct and transposed structures are used to design a FIR filter based on its input and co-efficient flow. And the multiplication is the major problem of the FIR filter design. In this paper we present the Multiplierless multiplication method. Instead of multiplication we use shift and add process only. And another advantage of our paper is we can use instead of direct and transposed structure to proposed structure of this paper. So we can avoid the conflict to use whether direct and transposed structure.*

I. INTRODUCTION

FIR filter is commonly used in DSP system and it is generally implemented system for high speed and low power design requirements. The concern of DSP application is focusing on low power, efficient area and high speed arithmetic. The main complexity of the FIR filter is multiplication of the large number of the filter coefficients by the filter input or its time shifted elements. Constant multiplication is a common operation in many digital filters. A variable can be multiplied by a given set of fixed point constants using a multiplier block. This paper reviews how the constant multiplication can be designed using shifts and adder/ subtractors. By knowing the properties of binary multiplication, it is easy to design the constant multiplication by using the shifters and the adder/ subtractors. In this paper we present the two different form of filter namely direct and transposed. And here we show that how the multiplication process can realize for the each structure of the filter. In our paper the filter design optimization (FDO) problem that is to find a set of filter coefficients which yields the least design complexity while meeting the required filter constraints. Existing algorithms use efficient search methods, but none of them can guarantee the minimum design complexity. We introduce an approximate algorithm that can handle filters with a large number of coefficients using less computational resources than the other FDO algorithm and find better solutions than existing FDO algorithm. Hence, we propose an exact algorithm called SIREN and NAIAD, which finds an optimum solution of the FDO problem under the minimum quantization value.

II. FIR FILTER DESIGN

A digital filter uses a digital processor to perform numerical calculations on sampled values of the signal. The processor may be a general-purpose computer such as a PC, or a specialized DSP (Digital Signal Processor) chip. As the terminology suggests, these classifications refer to the filter's impulse response. By varying the weight of the coefficients and the number of filter taps, virtually any frequency response characteristic can be realized with an FIR filter. FIR filters can achieve performance levels which are not possible with analog filter techniques (such as perfect linear phase response). However, high performances FIR filters generally require a large number of multiply-accumulates and therefore require fast and efficient DSPs. If the impulse response becomes zero after a finite number of samples it is a finite-length impulse response (FIR) filter. For a given specification the filter order, *N*, is usually much higher for an FIR filter than for an IIR filter. However, FIR filters can be guaranteed to be stable and to have a linear phase response, which corresponds to constant group delay.



- **Y(n) = h(n) * x(n) = $\sum_{k=0}^{N-1} \boldsymbol{h}(\mathbf{n})\ \boldsymbol{x}(\boldsymbol{n}-\boldsymbol{k})$**
- *** = symbol for convolution**
- **Requires N multiply-accumulate for each output**

Where,

x(n) and y(n) are the inputs and output sequences. If the input sequence, x(n), is an impulse the impulse response, h(n), is obtained as output sequence.

III. EXISTING SYSTEM

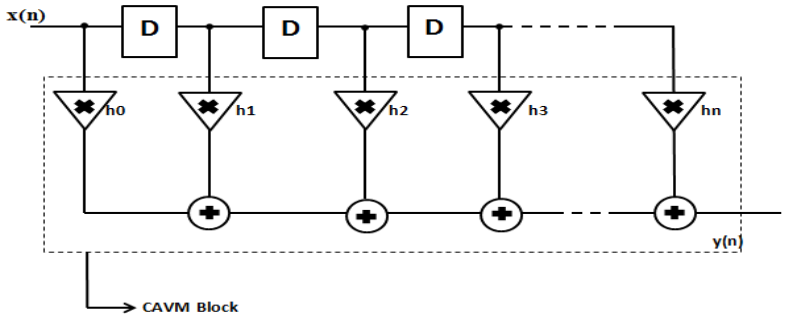
The FIR filters design based on its structure. It contain two major structures are,

Direct form

Transposed forms.

Direct Form Of The Filter

In the direct form structure, input is fed into the multiplier after the time shifter.



The I/O equation for FIR filter of order M is,

$y(n) = h_0\, x(n) + h_1\, x(n-1) + h_2\, x(n-2) + \ldots + h_M\, x(n-M)$

with impulse response $h= [h_0,h_1,h_2, \ldots, h_M]$.

It also written as,

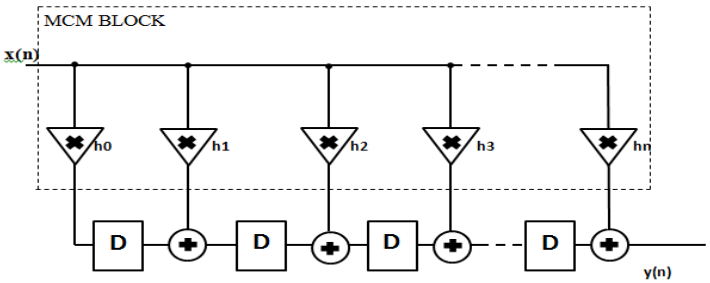
$y[n] = \sum_{k=0}^{M} b[k]x[n-k] - \sum_{k=1}^{N} a[k]y[n-k]$

In order to mechanize this equation, we need to use an adder to accumulate the sum of products in the right-hand side. We need multipliers to implement the multiplication by the filter weights. And we need delays to implement the delayed terms x(n-1), x(n-2), x(n-3),….

This above equation and the structure is known as direct form of the FIR filter. Because, it is directly realizes all the terms in the right-hand side. This delays are depends on the M value.

Transposed Form

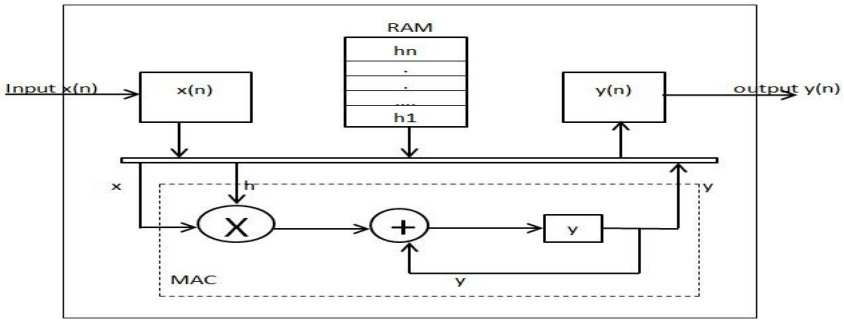
In transposed form the first the inputs are multiple with the set co-efficient, then every multiplied values are added after time shifts.



This realization diagram is obtained by transposing the canonical realization following the transposition rules of replacing adders by nodes, nodes by adders, reversing all flows, and exchanging input with output. This is known as transposed realization.We assign an internal state variable w1(n) to hold the contents of the delay register.

IV. PROPOSED SYSTEM

In proposed system, the two structures of the FIR filter are represented in one structure.



Where the input is x(n) and the co-efficient is h(n). Output is represented as y(n).

Where the h(n) is stored in the RAM memory from the FDO algorithm. This co-efficient is obtained from the SIREN and NAIAD algorithm. These algorithms are makes the co-efficient to easy handled by the filter. Especially it is convenient to multiplied. The proposed system is explained by two major parts are,

Co-efficient optimization

Co-efficient multiplication

Co-Efficient Optimization

To find the co-efficient, SIREN and NAIAD algorithms are used.

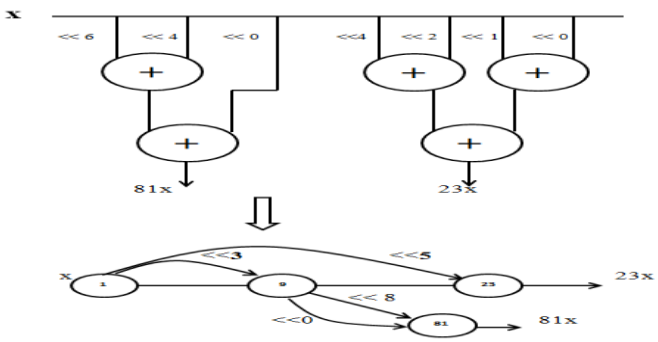
Siren was developed to find a set of filter coefficients yielding a minimum number of adders/subtractors in the filter design and satisfying the filter constraints. It takes the five-tuple *fspec* denoting the filter specifications as input and returns a set of fixed-point coefficients *sol*. In Algorithm stands for the quantization value used to convert floating-point numbers to integers. SIREN will be described in detail using a symmetric FIR filter with *fspec* (8,0.2π ,0.7π , 0.01, 0.01)

NAIAD was developed based on two observations: i) given filter specifications, finding a set of floating-point coefficients, that satisfies the filter constraints, takes a polynomialtime; ii) given a set of coefficients, finding a Multiplierless design of coefficient multiplications including a number of adders/subtractors very close to the minimum can be done in a reasonable time. Hence, NAIAD consists of two main parts: i) exploring sets of coefficients that satisfy the filter constraints and finding the ones with the smallest EWL value; ii) exploring the search area in the neighborhood of each solution obtained in the first part and finding the one that leads to the minimum design complexity. In following, these two parts are described in detail using a symmetric FIR filter with *fspec* as an example.

Co-Efficient Multiplication

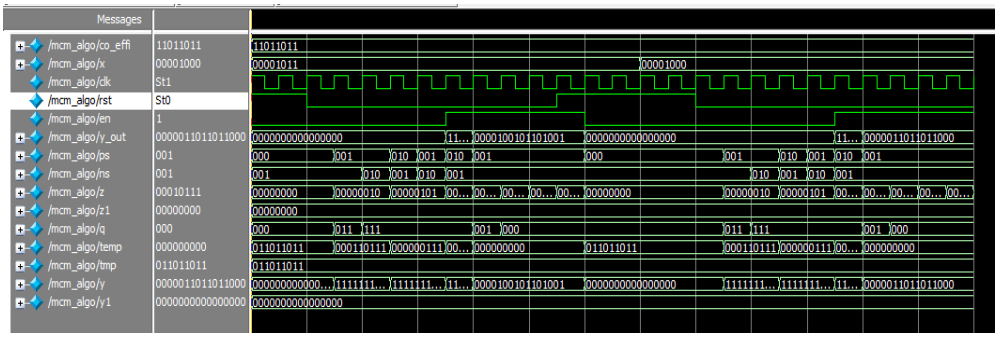
MAC BLOCK

In MAC block contain both the multiplication and the addition operation. It contains the MCM algorithm for the multiplication process. In proposed system, for the multiplication process the MCM algorithm is used. The graph based algorithm which is introduced to reduce the addition in shift adds implementations of constant multiplications. These methods are not limited to any particular number representation and consider a large number of alternative implementations of a constant yielding better solution than the shift add implementation of constant multiplication.



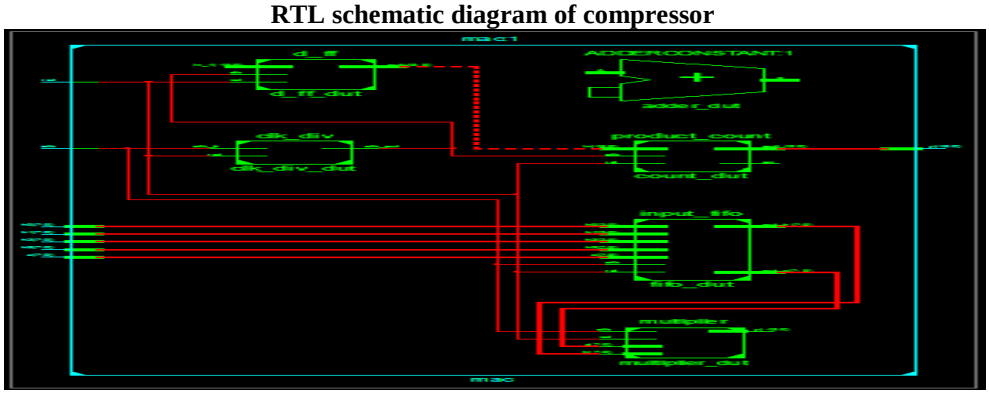
The graph based algorithm contain the set of modules for each co-efficient. These basic graphs are classified based on the cost of the co-efficient. This is the basic structure for the co-efficient. Based on this graph, the co-efficient is separated as a graph. And the input x(n) is involved into the graph. Then we got the final output which is equal to multiplied value.content of w3 is no longer needed and can be updated to the next time instant. Similarly, once h2w2 has been accumulated into y, w2 may be updated, and so on. In general, for a filter of order M

V. SIMULATION RESULT

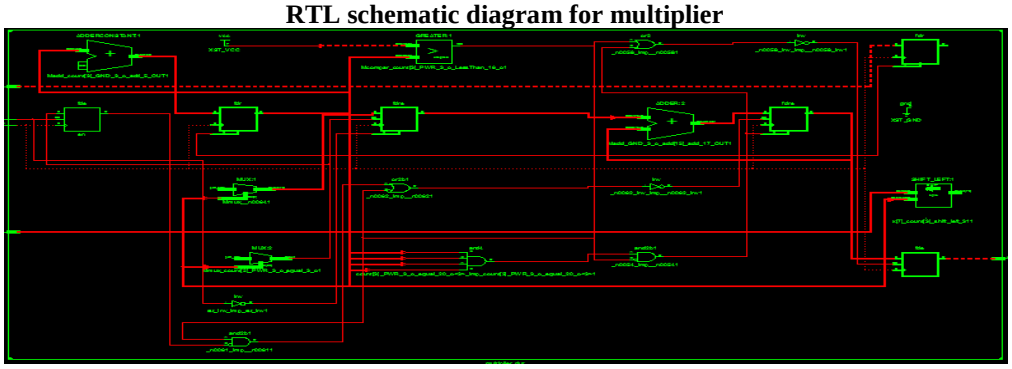


VI. IMPLEMENTATION

RTL schematic diagram of compressor



RTL schematic diagram for multiplier



VII. CONCLUSION

It concludes the FIR filter design with effective methods of FDO problem and the Multiplierless multiplications. In this FIR filter design is easy to construct. Area and the power are very low compared to the previous FIR filter design. This article addressed the problem of optimizing the number of operations in the FIR filter design while satisfying the filter constraints, generally known as the FDO problem. It presented exact and approximate FDO algorithms, all of which are equipped with efficient methods to find the fewest operations in the shift-adds design of the coefficient multiplications. These algorithms can be modified to target different filter constraints and filter forms and to handle a delay constraint in the multiplier blocks of filters. It was indicated that the total number of operations, EWL value, filter length, quantization value, and filter form have a significant Impact on the gate-level area, delay, and power dissipation results of filter designs.

REFERENCES

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

[2]. K. Johansson, O. Gustafsson, and L. Wanhammar, "A detailed complexity model for multiple constant multiplication and an algorithm to minimize the complexity," in Proc. IEEE Eur. Conf. Circuit Theory Design, 2005, pp. 465–468.

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

[4]. L. Aksoy, E. Costa, P. Flores, and J. Monteiro, "Finding the optima tradeoff between area and delay in multiple constant multiplications,"Elsevier J. Microprocessors Microsyst., vol. 35, no. 8, pp. 729–741, 2011

[5]. M.Kumm, P. Zipf,M. Faust, and C.-H. Chang, "Pipelined adder graphoptimization for high speed multiple constant multiplication," in Proc.IEEE Int. Symp. Circuit Syst., 2012, pp. 49–52.

[6]. K. Johansson, O. Gustafsson, L. DeBrunner, and L. Wanhammar,"Minimum adder depth multiple constant multiplication algorithm forlow power FIR