International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 03 Issue: 05 | May-2016 www.irjet.net p-ISSN: 2395-0072

An Efficient FIR Filter Design Using Reversible Adder and Multiplier

Rafiya Banu Nadaf[1], Dr.U Eranna[2]

[1]*PG Scholar ,VLSI Design and Embedded systems, BITM Ballari, VTU Belgaum, Karnataka, India*

[2]*HOD, Dept of ECE, BITM Ballari, VTU Belgaum, Karnataka, India*

Abstract - *The Finite Impulse Response (FIR) filter are a class of digital filter that have finite impulse response and are extensively used in signal processing and communication system in applications like noise reduction, echo cancellation, image enhancement, speech and waveform synthesis etc. As the complexity of implementation grows with the filter order and the precision of computation, real-time realization of these filters with desired level of accuracy becomes a challenging task. In this paper we are going to optimize the area and increase the speed of FIR filter by the use of efficient adders and multipliers.*

Key Words: **FIR , Filter , digital filter, Multiplier, Adder**

1. INTRODUCTION

A filter is a device or process that removes some unwanted component or feature from a signal. Filtering is a class of signal processing, the defining feature of filter being the complete or partial suppression of some aspect of the signal. Impulse Response: A filter can be described in terms of its response to an impulse input.
Impulse response is useful because:
(i) Any signal can be viewed as the sum of a number of shifted and scaled impulses, hence the response a linear filter to a signal is the sum of the responses to all the impulses that constitute the signal.
(ii) An impulse input contains all frequencies with equal energy, and hence it excites a filter at all frequencies
(iii) Impulse response and frequency response are Fourier transform pairs.
A digital filter uses a digital processor to perform numerical calculations on sampled values of the signal. Digital filters eliminate a number of problems associated with their classical analog counterparts and thus are preferably used in place of analog filters.

1.1 FIR Filter

FIR filters are said to be finite because they do not have any feedback. Therefore, if we send an impulse through the system (a single spike) then the output will invariably become zero as soon as the impulse runs through the filter. A non-recursive filter has no feedback and its input-output relation is given by

$$Y(n) = \Sigma\, H(k)*X(n-k) \quad \text{where } k=0, 1\ldots.N-1$$

The output of a non-recursive filter is a function only of the input signal . The response of such a filter to an impulse consists of a finite sequence of *M+*1 samples, where *M* is the filter order. Hence, the filter is known as a *Finite-Duration Impulse Response* (FIR) filter. Other names for a non-recursive filter include all-zero filter, feed-forward filter or moving average (MA) filter a term usually used in statistical signal processing literature.

The basic characteristics of Finite Impulse Response (FIR) filters are:

a) Linear phase characteristic

b) High filter order (more complex circuits)

c) Stability

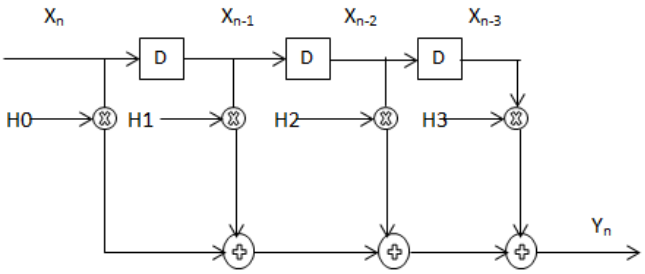

Fig 1: 4 Tap FIR Filter

Where, H0, H1, H2 & H3 are 8 bit impulse response & X_n is the input signal.

In Fig 1 . D is the D register and H (n) and X(n) are convolved using reversible multiplier. And addition of convolved result is performed using QCA adder. The result obtained after 3 additions is the filter output Y (n).

1.2 8 bit Reversible multiplier

Multiplication includes two basic steps. First step is the generation of partial products and the second one is the addition of the generated partial products. The partial products can be generated using Fredkin gate. When one of the input to the Fredkin gate is assigned a constant value of zero, then the Fredkin gate behaves like an AND gate and partial product can be generated. The generated partial products can now be added using Parallel ripple carry adder that is designed by cascading the HNG gate. For a 4x4

multiplier, 16 Fredkin gates are required for generating the entire partial products. Three stages of reversible ripple carry adder using HNG gate is required for adding the above generated partial products and obtaining the final 8bit product.

In this n x n reversible multiplier is designed using HNG and Fredkin gates. The basic cell for such a multiplier is the full adder block. It is evident from the figure that the TSG gate can act as full adder when one of the inputs is assigned a constant value of zero. C is assigned as zero the HNG gate acts as a full adder and the sum and carry output is obtained at S and R respectively. The HNG gate implemented as a full adder. Half Adder is implemented using Feyman Gate and Fredkin Gate.

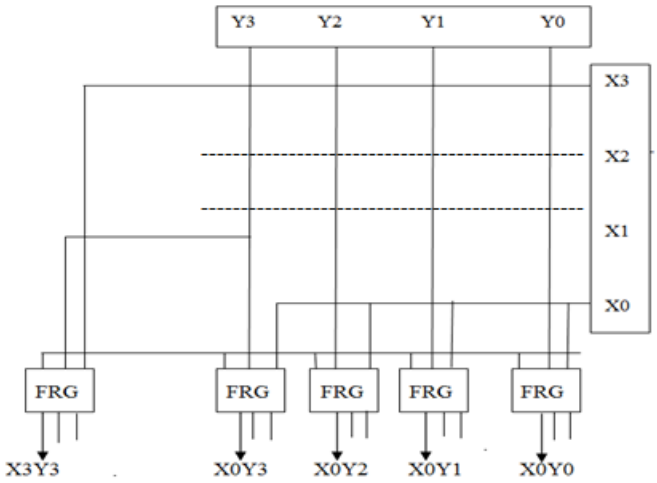

Fig 2: 4*4 Reversible multiplier

1.3 QCA adder

The logic elements of QCA are an inverter and majority gate. An inverter is designed by positioning cells diagonally from each other to achieve the inversion functionality. A majority gate consists of five QCA cells that realize the function of *M* (*a; b; c*) = *ab* + *bc* + *ac*.

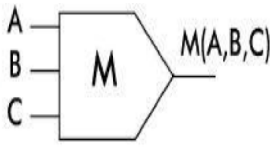

Fig 3: Majority cell

Carry-look ahead is arguably the most important technique in the design of fast adders, especially large ones.

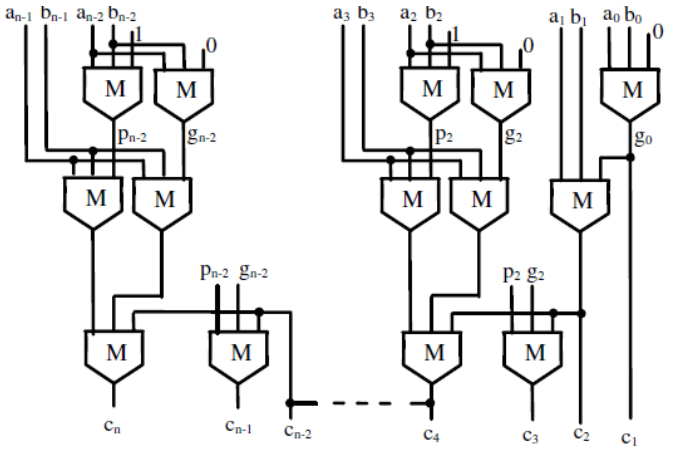

Fig 4: Carry chain

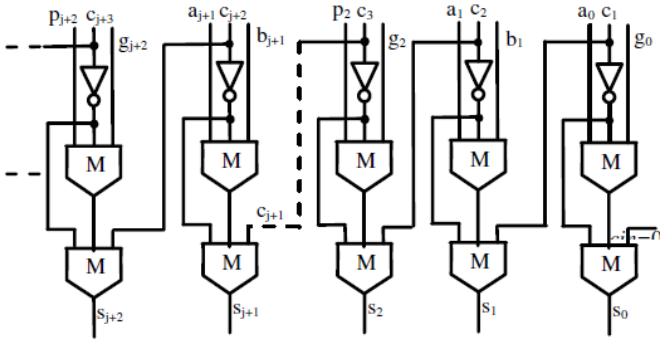

Fig 5: Sum block

In straightforward addition, e.g. in a ripple adder, the operational time is limited by the (worst-case) time allowed for the propagation of carries and is proportional to the number of bits added. So faster adders can be obtained by devising a way to determine carries before they are required to form the sum bits.

A carry, C_i, is produced at bit-stage 'i' if either one is *generated* at that stage or if one is *propagated* from the preceding stage. So a carry is generated if both operand bits are 1, and an incoming carry is propagated if one of the operand bits is 1 and the other is 0. Let P_i and G_i denote the generation and propagation, respectively, of a carry at stage i, A_i and B_i denote the two operands bits at that stage and C_{i-1} denote the carry into the stage. Then we have

$G_i=A_iB_i$

$P_i = A_i \hat{} B_i$

$C_i = G_i + P_iC_{i-1}$ and

The sum can be written as,

$S_i == P_i \hat{} C_{i-1}$ which allows the use of shared logic to produce S_i and P_i.

This carry block is cascaded with the propagate and generate block. So, that carry is obtained. And the sum block is cascaded with carry block. So, that the sum is obtained.

2. RESULT

When, X(n)=00110001(49), H0=00101000(40), H1=00110110(54), H2=01000110(70), H3=10000001(129)

Y (n) =49*(40) + 49*(54) + 49*(70) + 49*(129)

=0011100000010101(14375)

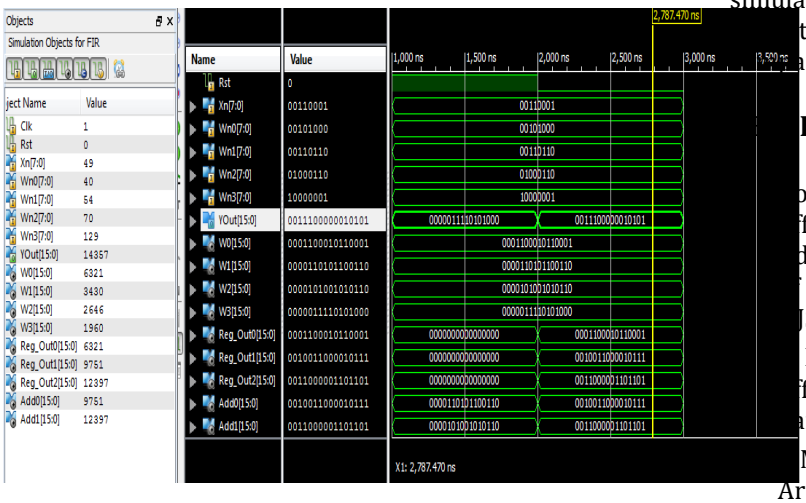

Fig 6: Simulation results of FIR filter.

Device Utilization Summary			
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slice Flip Flops	48	12,288	1%
Number of 4 input LUTs	678	12,288	5%
Number of occupied Slices	355	6,144	5%
Number of Slices containing only related logic	355	355	100%
Number of Slices containing unrelated logic	0	355	0%
Total Number of 4 input LUTs	678	12,288	5%
Number of bonded IOBs	58	240	24%
Number of BUFG/BUFGCTRLs	1	32	3%
Number used as BUFGs	1		
Average Fanout of Non-Clock Nets	3.46		

Fig 7: Device Utilization summary of FIR

Timing Summary	Delay
Input arrival time before clock	12.381ns
Output arrival time after clock	11.168ns
Maximum combinational path delay	14.866ns

Fig 8: Timing Summary

3. CONCLUSIONS

In this paper, a FIR filter using reversible multiplier and QCA adder for area efficient design is implemented. . The above designs is modeled using Verilog. Xilinx Project Navigator 14.7 is used as a synthesis tool and ISE simulator is used for simulation. The proposed architecture achieves better tages in terms of area, power and delay when ared to other architectures.

RENCES

owrishankar V, Manoranjitham D and Jagadeesh P ," fficient FIR Filter Design Using Modified Carry Select dder & Wallace Tree Multiplier," International Journal Science, Engineering and Technology Research JSETR) ,Volume 2, Issue 3, March 2013.

Ramkumar and Harish M Kittur," Low Power and Area fficient Carry Select Adder", IEEE Transaction on Very arge Scale Integration Systems, February 2012.

M. Vijaya Prakash, K.S. Gurumurthy,"A Novel VLSI Architecture for Low Power FIR Filter", Published in International Journal of Advanced Engineering & Application, Jan 2011.

[4] Stefania Perri, Pasquale Corsonello, and Giuseppe Cocorullo,"Area-Delay Efficient Binary Adders in QCA," Ieee Transactions on Very Large Scale Integration (Vlsi) Systems., April 21, 2013.

[5] V. Pudi and K. Sridharan, "Efficient design of a hybrid adder in quantumdot cellular automata," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 19, no. 9, pp. 1535–1548, Sep. 2011.

[6] "Design of a Reversible ALU Based on Novel Reversible Logic Structures", Matthew Arthur Morrison, *University of South Florida*, January 2012.