ISSN: 0976-3104
SPECIAL ISSUE
Srividya.

ARTICLE OPEN ACCESS

IMPLEMENTATION OF AREA EFFICIENT MULTIPLIER AND ADDER ARCHITECTURE IN DIGITAL FIR FILTER

Srividya

Sahyadri College of Engineering & Management, ECE Dept, Mangalore, INDIA

ABSTRACT

The title of this paper include an implementation of low area efficient multiplier and adder architecture in digital FIR filter design.FIR filter are mainly used in digital signal processing application in which multiplier and adder are the basic fundamental blocks. Efficiency of filter design depends on the architecture used for multiplier and adder block which differ in the delay, area and power. In this paper two architectures are used for multiplier design, they are modified booth algorithm and Vedic algorithm, in which Vedic algorithm is used for implementation as it consumes less area than modified booth algorithm.And for design of adder block efficient architecture is used which include carry save adder. Hence the output responses of FIR filter is obtained using Vedic multiplier and carry save adder. The multiplier and adder architecture are simulated and synthesized using Xilinx ISE tool. The above two multiplier architecture are also simulated, synthesized and physical design of it is done using Cadence tool in order to obtain the GDSII file

Received on: 29th-May-2015
Revised on: 30th-July-2015
Accepted on: 01st-Aug-2015
Published on: 4th-Jan-2016

KEY WORDS

FIR filter; Modified booth algorithm; Vedic algorithm; Carry save adder; GDSII

***Corresponding author:** Email: svsvidya82@yahoo.co.in **Tel:** +91-9480503355

INTRODUCTION

In the recent trends, VLSI technology has brought a significant development in the area of chip design which mainly depends on the factors like area, speed and power. Considering all these factors, much architecture are developed to implement Digital FIR filter which is used in digital signal processing applications. Filter is implemented using multipliers and adder blocks. Selection of efficient architecture for multiplier and adder is a major factor to be considered which in turn improves the efficiency of FIR filter. Digital FIR filter is the linear time invariant filter. A linear time invariant is the filter which does not vary with time and it interacts with the input signal and filter coefficients through a process called linear convolution and hence produces the output response. The synthesized architectures are shown in the following sections. In this paper two architecture for multiplier are simulated and synthesized and the physical design of it is performed using cadence tool and the adder is simulated and synthesized using Xilinx tool.

MATERIALS AND METHODS

Existing Algorithm

Array multiplier is the existing algorithm and one of the most widely used multiplier technique and it is a long multiplication process. It has a regular structure and used for unsigned multiplication. The computation cost and time taken by array multiplier is more. From the simulation and synthesis result obtained for array multiplier using Xilinx tool, the delay of array multiplier is found to be 17.522ns.So booth multiplier is used for multiplication process. Booth multiplier multiplies two signed numbers and it is faster than array multiplier. The operation of booth algorithm includes arithmetic shift operation after addition and subtraction operation. The drawback of this type of multiplier is that this uses a large number of add and subtraction operation which becomes inconvenient to design parallel multiplier. Another drawback is that when there is isolated ones the algorithm becomes inefficient. From the simulation and synthesis result for booth multiplier, the delay is found to be 6.712ns. These drawbacks are overcome by modified booth algorithm and Vedic algorithm.

ISSN: 0976-3104

Modified booth algorithm

Modified booth algorithm is used to multiply signed as well as unsigned numbers and the partial product is reduced to N/2 from N, if N is the multiplier. Modified booth algorithms are so called as radix-2, radix-4, and radix-8 depending on the number which is taken as a base. Modified booth algorithm is faster than array multiplier and booth multiplier. Advantage of modified booth algorithm is that its computation is faster than conventional multiplier as it halves the partial products and it is used for longer operands whereas booth algorithm has partial products same as that of the multiplier bits and can be used only for smaller operands. Disadvantage of this is that it requires and encoder circuit to encode the multiplier bits which consumes more area. This encoding of multiplier is done by grouping the bits in blocks of two(radix-2), blocks of three (radix-4), blocks of four(radix-8),such that one bit overlapping of each block is performed. And the encoding of the bits is performed based on the booth recording table which is shown.**[Table–1, Table–2, Table–3].**The halving of partial product is obtained by shifting and adding for every second column [1].

Table: 1. Radix-2 booth recoding table

Block A(Multiplier)	Re-coded digits	Operations on B(Multiplicand)
00	0	0*B
01	+1	+1*B
10	-1	-1*B
11	0	0*B

Table: 2. Radix-4 booth recoding table

Block A(Multiplier)	Re-coded digits	Operations on B(Multiplicand)
000	0	0*B
001	+1	+1*B
010	+1	+1*B
011	+2	+2*B
100	-2	-2*B
101	-1	-1*B
110	-1	-1*B
111	0	0*B

Table: 3. Radix-8 booth recoding table

Block A(Multiplier)	Re-coded digits	Operations on B(Multiplicand)
0000	0	0*B
0001	+1	+1*B
0010	+1	+1*B
0011	+2	+2*B
0100	+2	+2*B
0101	+3	+3*B
0110	+3	+3*B
0111	+4	+4*B
1000	-4	-4*B
1001	-3	-3*B
1010	-3	-3*B
1011	-2	-2*B
1100	-2	-2*B
1101	-1	-1*B
1110	-1	-1*B
1111	0	0*B

Vedic algorithm

Vedic algorithm is one of the ancient algorithms used for fast multiplication process. The Vedic algorithm is implemented based on Urdhva Tiryakbhyam (vertical and crosswise) Sutra. Out of 16 sutras; this is one such sutra which is used for all multiplication cases. Here partial product can be generated by concurrent addition of partial product.The multiplier using Vedic algorithm is independent of clock frequency of the processor because of parallel calculation of partial product and addition. One such feature of Vedic algorithm is reducing multi-bit multiplication into single bit multiplication. The carry propagation is reduced from LSB to MSB, as the partial product is generated in single step. This is well known algorithm which consumes less area compared to modified booth algorithm. The 2x2 Vedic multiplier blocks multiplies two bit binary numbers and this type of multiplication uses Urdhva Triyagbhyam algorithm. Consider two numbers 'a' and 'b',each of two bits then a[0]&b[0] represents LSB and a[1]&b[1] represents MSB.The

ISSN: 0976-3104

output is 4 bit with y[0] forms the LSB,y[1] forms the second bit from LSB,y[2] forms the third bit from LSB and y [3] forms the carry bit.

Carry save adder

Carry save adder is one such adder with lesser delay in comparison with different types of adders like ripple carry adder, carry look ahead adder, carry select adder and so on. Operation of Carry save adder is same as that of full adder with three inputs. In carry save adder, sum and carry is calculated separately and then both are added to give the final sum. Carry save adder also consumes less area compared to other types of adder and it has a delay of 9.043ns.

RESULTS

Modified booth algorithm

Figure–1 shows the RTL view of modified booth algorithm using cadence tool.The RTL view represents the encoder circuit which encodes the multiplier and then multiplies with the multiplicand to generate the partial products and these partial products are then added using carry save adder to get the final result. **Figure–2** shows physical design of modified booth algorithm using cadence tool. The physical design shows the layout of the design which are obtained by process such as partitioning,floorplanning and placement, clock tree synthesis, routing,compaction and finally the physical verification process.

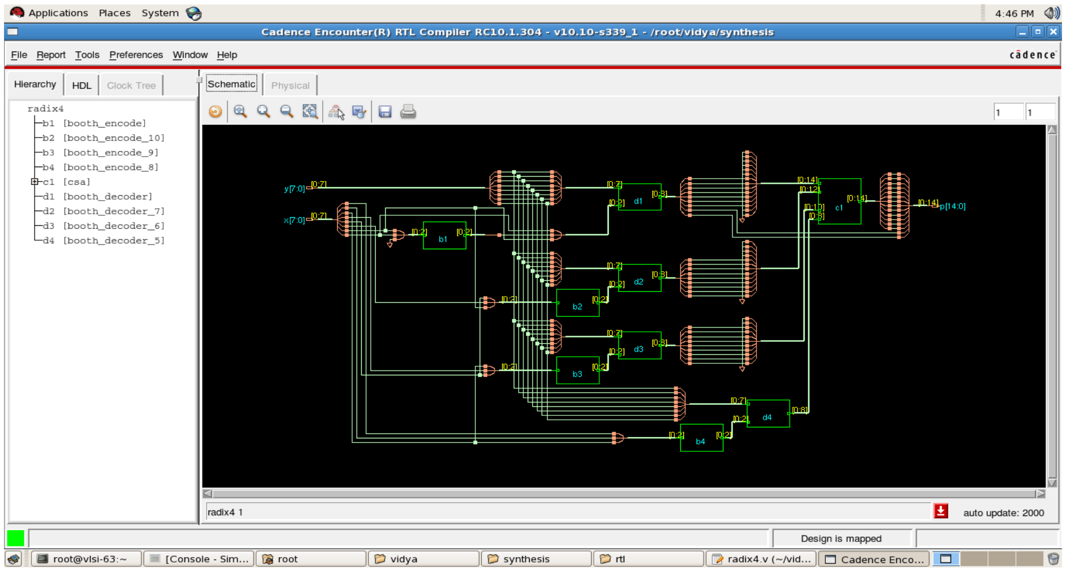


Fig: 1. RTL view of modified booth algorithm using cadence tool

Vedic algorithm

The 4x4 multiplier block can be easily constructed using four 2x2 bit multiplier blocks. And the partial product generated is fed to the addition tree and lower 2 bit of partial product are taken as least two bits to the result. Again 8x8 multiplier block is constructed using four 4x4 multiplier block and the partial product are then added to get the final result.8x8 multiplier blocks are better than 4x4 multiplier block because computations are performed which are of 8 bit in many kind of processors. Here addition is performed using carry save adder which has less delay compared to all other adder. Carry save adder also consumes less area compared to other adders such as carry look ahead adder, carry select adder. **Figure– 3** shows the RTL view of 8x8 multiplier using cadence tool.This figure indicates that using the parallel techniques the computation can be easily performed. **Figure– 4** represents the physical design of 8x8 Vedic multiplier which include the process of generating the layout of the design. The final step of physical design process is the GDSII file generation which are further used for fabrication process. **Figure–5** shows the simulation results of 8x8 Vedic multiplier which explains that multiplication of two 8 bit number obtains a 16 bit result.

THE IIOAB JOURNAL
ISSN: 0976-3104

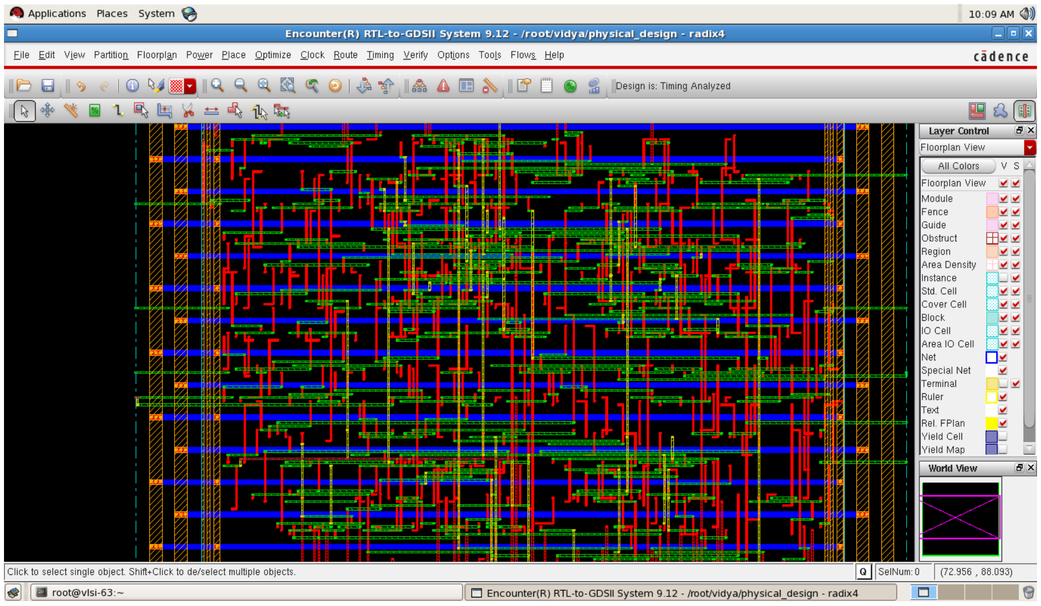

Applications Places System
10:09 AM
Encounter(R) RTL-to-GDSII System 9.12 - /root/vidya/physical_design - radix4
File Edit View Partition Floorplan Power Place Optimize Clock Route Timing Verify Options Tools Flows Help
cādence
Design is: Timing Analyzed
Layer Control
Floorplan View
All Colors V S
Floorplan View
Module
Fence
Guide
Obstruct
Region
Area Density
Instance
Std. Cell
Cover Cell
Block
IO Cell
Area IO Cell
Net
Special Net
Terminal
Ruler
Text
Rel. FPlan
Yield Cell
Yield Map
World View
Click to select single object. Shift+Click to de/select multiple objects.
SelNum:0
(72.956 , 88.093)
root@vlsi-63:~
Encounter(R) RTL-to-GDSII System 9.12 - /root/vidya/physical_design - radix4

Fig: 2. Physical design of modified booth algorithm using cadence tool

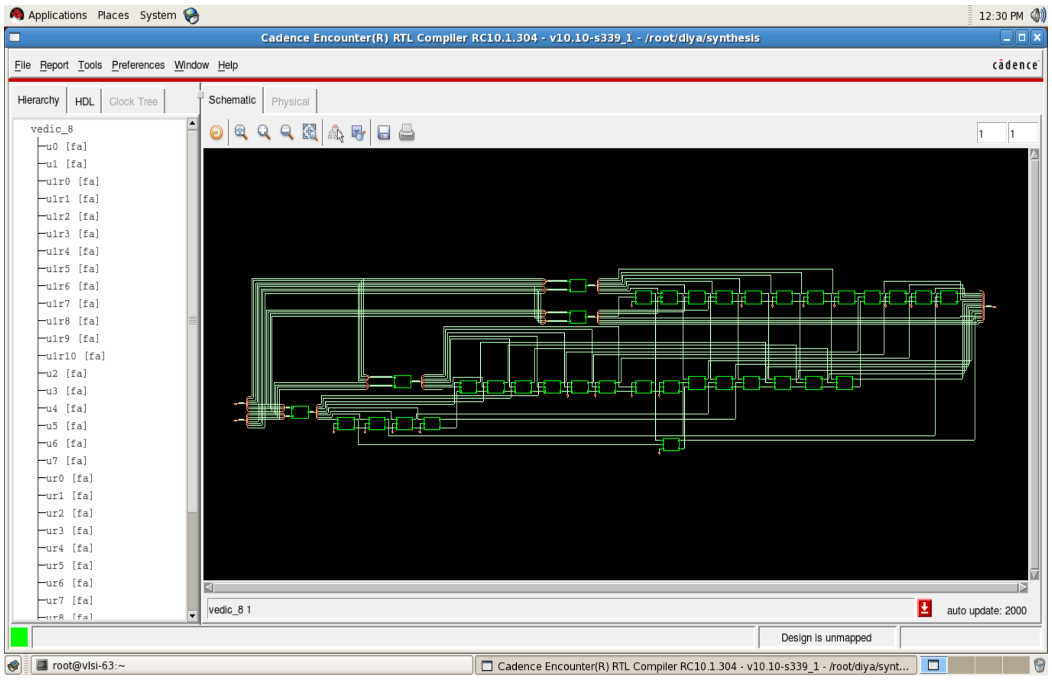


Fig: 3. RTL view of 8x8 Vedic multiplier using cadence tool

THE IIOAB JOURNAL
ISSN: 0976-3104

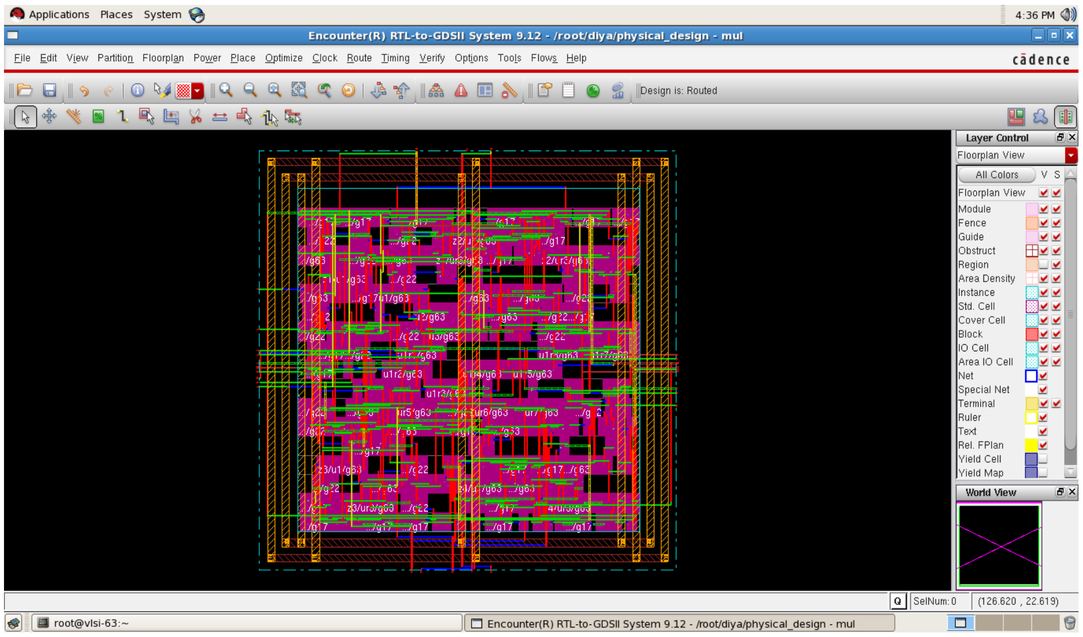
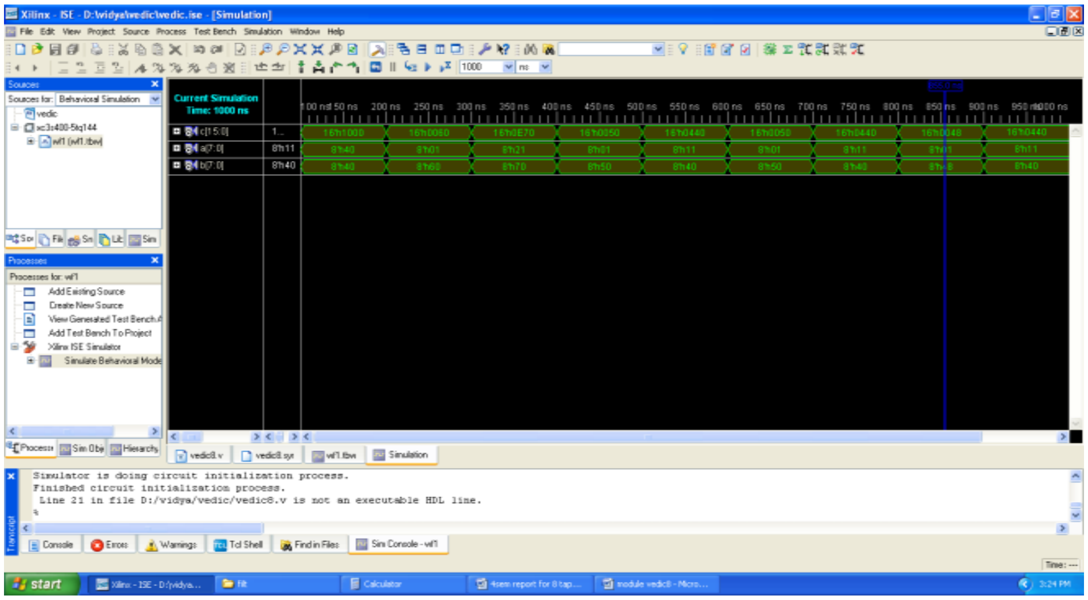


Fig: 4. Physical design of 8x8 Vedic multiplier using cadence tool


Xilinx - ISE - D:\vidya\vedic\vedic.ise - [Simulation]
File Edit View Project Source Process Test Bench Simulation Window Help
Sources
Sources for: Behavioral Simulation
vedic
xc3s400-5tq144
wf1 (wf1.tbw)
Processes
Processes for: wf1
Add Existing Source
Create New Source
View Generated Test Bench As
Add Test Bench To Project
Xilinx ISE Simulator
Simulate Behavioral Model
Current Simulation Time: 1000 ns
c[15:0]
a[7:0]
b[7:0]
8'h11
8'h40
vedic8.v
vedic8.syr
wf1.tbw
Simulation
Console
Errors
Warnings
Tcl Shell
Find in Files
Sim Console - wf1
start
Calculator

```
Simulator is doing circuit initialization process.
Finished circuit initialization process.
 Line 21 in file D:/vidya/vedic/vedic8.v is not an executable HDL line.
%
```

Fig: 5. Simulation result of 8x8 Vedic multiplier using Xilinx tool

ISSN: 0976-3104

DISCUSSION

From the point of implementation of efficient multiplier and adder architecture, It was found that modified booth algorithm is better since it has lesser delay and consumes low power than Vedic multiplier. But in terms of cell area usage, it has seen that Vedic algorithm is far more efficient multiplier. While considering the efficient adder structure, carry save adder was found to be one such adder with lesser delay and low power consumption. **Table– 4** shows the Comparison of modified booth algorithm and Vedic algorithm used for multiplication operation.

Table: 4. Comparison table of multiplier

Multiplier	Cell area	Leakage power(nW)	Dynamic power(nW)	Total power(nW)
Radix4 Modified booth algorithm (8x8)	6912	210.760	136531.5	136742.348
Vedic algorithm (8x8)	6044	280.756	357930.524	358211.280

CONCLUSION

This paper presents an efficient architecture for multiplier and adder and from the analysis result, modified booth algorithm is found to be efficient in terms of power consumption and Vedic algorithm is efficient in terms of area consumption. So it is better to consider a hybrid architecture which involves the combination of modified booth algorithm and Vedic algorithm for design of FIR filter.

ACKNOWLEDGEMENT

I would like to thank Mr. Praveen Kumar, Assistant Professor of Sahyadri college of Engineering & Management for helping me in completion of this project.
.

CONFLICT OF INTEREST
Author declares no conflict of interest.

FINANCIAL DISCLOSURE
No financial support was received to carry out this project.

REFERENCES

[1] Narasimha AR, Rajasekhar K, Rani S. [2012] Implementation of low area and power efficient architecture for Digital FIR filter. *IJARCSSE*, 2(8): 238-244.
[2] Chandel D, Kumawat G, Lahoty P, Chandrodaya VV, Sharma S. [2013] Booth Multiplier: Ease of multiplication. *IJETAE*, 3(3): 326-330.
[3] Shukla P, Gahlan NK, Kaur J. [2012] Techniques on FPGA Implementation of 8-bit Multiplier. *IJCST*, 3(2): 455-463.
[4] Kumar KSG, Prasannam JD, Christy MA. [2014] Analysis of Low Power, Area and High Speed Multipliers for DSP Applications. *IJETAE*, 4(3): 278-282.
[5] Choubey R, Arif M. [2014] Low Power and Area Optimized Using Modified Booth Algorithm Radix-2 and Ra-dix-4. *IJETAE*, 4(4): 641-646.
[6] Chaudhary M, Narula MS. [2013] FPGA Implementations of Booth's and Baugh-Wooley Multipliers. *IJITEE*, 3(1): 221-224.
[7] Ganeshkumar G, Charisma V. [2012] Design of high speed Vedic multiplier with Vedic mathematics tech-niques. IJSRP, 2(3): 1-5.
[8] Babulu K, Parasuram G. [2011] High speed Arithmetic Logic using booth multiplication. *IJCSIT*, 2 (3): 4-9.
[9] Ashenden JP. [2008] Digital design-An embedded system approach using verilog. *Elsevier, USA*
[10] Warren HS. [2012] Montgomery Multiplication. http://www.hackersdelight.org/MontgomeryMultiplication.pdf
[11] Kumar H. [2013] Implementation and analysis of power, area and delay of array, Urdhva, Nikilam Vedic multiplier. IJSRP, 3(1): 1-5.
[12] Lamberti F, Andrikos N, Antelo E, Montuschi P. [2011] Reducing the Computation Time in (Short Bit-Width) Two's Complement Multipliers. *IEEE Transactions on Computers*. 60 (2):148-156.
[13] Seo YH, Kim DW. [2010] A New VLSI Architecture of Parallel Multiplier-Accumulator Based on Radix-2 Modified Booth Algorithm. *Very Large Scale Integration*, 18(2):201-208.