Study of Combinational and Booth Multiplier

Neha Goyal *, Khushboo Gupta, Renu Singla****

*,** Department Of Computer Science & Engineering
Shri Ram College of Engineering & Management, Palwal, Haryana, INDIA

Abstract- Low power consumption and smaller area are some of the most important criteria for the fabrication of DSP systems and high performance systems. Optimizing the speed and area of the multiplier is a major design issue. However, area and speed are usually conflicting constraints so that improving speed results mostly in larger areas. In our project we try to determine the best solution to this problem by comparing a few multipliers.
The area and speed of the multiplier is an important issue, increment in speed results in large area consumption and vice versa. Multipliers play vital role in most of the high performance systems. Performance of a system depends to a great extent on the performance of multiplier thus multipliers should be fast and consume less area and hardware. This idea forced us to study and review about the Booth's Algorithm.

Index Terms- Partial Products, Booth Algorithm, 4-bit Multiplier, Multiplicand, Multiplier, and simulator.

I. INTRODUCTION

Multiplication in hardware can be implemented in two ways either by using more hardware for achieving fast execution or by using less hardware and end up with slow execution [3]. Traditional hardware multiplication is performed in the same way multiplication is done by hand: partial products are computed, shifted appropriately, and summed.
This algorithm can be slow if there are many partial products (i.e. many bits) because the output must wait until each sum is performed. Booth's algorithm cuts the number of required partial products in half. This increases the speed by reducing the total number of partial product sums that must take place.

II. OBJECTIVE AND PERFORMANCE CITERIA

Our objective is to study about the performance of Combinational Multiplier and Booth multiplier using Simulator

The performance of the any processor will depend upon its power and delay. The power and delay should be less in order to get a effective processor. In processors the most commonly used architecture is multiplier. If the power and delay of the multiplier is reduced then the effective processor can be generated [2].
Low power consumption and smaller area are some of the most important criteria for the high performance systems. Area of the circuit depends on the number of logic gates used in the circuit.

III. COMBINATIONAL MULTIPLIER

Combinational Multipliers do multiplication of two unsigned binary numbers. Each bit of the multiplier is multiplied against the multiplicand, the product is aligned according to the position of the bit within the multiplier, and the resulting products are then summed to form the final result. Main advantage of binary multiplication is that the generation of intermediate products is simple.

If the multiplier bit is a 1, the product is an appropriately shifted copy of the multiplicand;

If the multiplier bit is a 0, the product is simply 0[8].

The design of a combinational multiplier to multiply two 4-bit binary number is illustrated below:

			A_3	A_2	A_1	A_0
			B_3	B_2	B_1	B_0
			$A_3 .B_0$	$A_2 .B_0$	$A_1 .B_0$	$A_0 .B_0$
		$A_3 .B_1$	$A_2 .B_1$	$A_1 .B_1$	$A_0 .B_1$	
	$A_3 .B_2$	$A_2 .B_2$	$A_1 .B_2$	$A_0 .B_2$		
$A_3 . B_3$	$A_2 .B_3$	$A_1 .B_3$	$A_0 .B_3$			
S_6	S_5	S_4	S_3	S_2	S_1	S_0

If two n-bit numbers are multiplied then the output will be less than or equals to 2n bits [8].

A .Design of Combinational Multiplier

A combinational circuit for implementing the 4-bit multiplier is shown in figure:
Each of the ANDed terms is called a *partial product*. The resulting product is formed by accumulating down the columns of partial products, propagating the carries from the rightmost columns to the left. The multiplier will multiply two 4 bit number

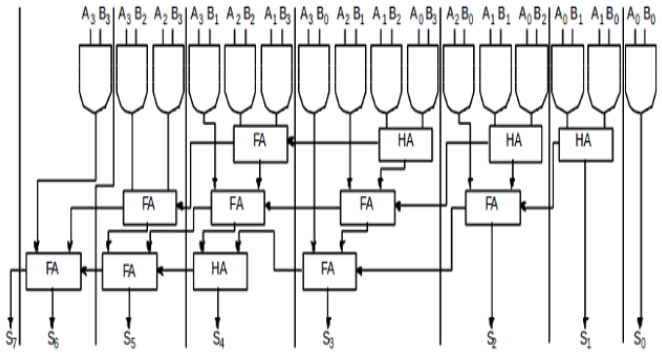

Figure-1 4x4 Combinational Multiplier

The first level of 16 AND gates computes the individual partial products. The second- and third-level logic blocks form the accumulation of the products on a column-by-column basis. The column sums are formed by a mixture of cascaded half adders and full adders. In the figure, inputs from the top are the bits to be added and the input from the right is the carry-in. The output from the bottom is the sum and to the left is the carry-out.

B. Performance

Combinational Multiplier gives more power consumption as well as optimum number of components required, but delay for this multiplier is larger. It also requires larger number of gates because of which area is also increased; due to this, combinational multiplier is less economical [6].Thus, it is a fast multiplier but hardware complexity is high.[5]

IV. BOOTH'S ALGORITHM

The algorithm was invented by Andrew Donald Booth in 1950 while doing research on crystallography at Birkbeck College in Bloomsbury, London[3].
Traditional hardware multiplication is performed in the same way multiplication is done by hand:

- partial products are computed,
- shifted appropriately, and
- Summed.

This algorithm can be slow if there are many partial products (i.e. many bits) because the output must wait until each sum is performed.
Booth's algorithm cuts the number of required partial products in half. This increases the speed by reducing the total number of partial product sums that must take place.

A. Algorithm

The multiplicand and multiplier are placed in the m and Q registers respectively.
A 1 bit register is placed logically to the right of the LSB (least significant bit) Q0 of Q register. This is denoted by Q-1.
A and Q-1 are initially set to 0.
Control logic checks the two bits Q0 and Q- 1.

If the two bits are same (00 or 11) then all of the bits of A, Q, Q-1 are shifted 1 bit to the right.

If they are not the same and if the combination is 10 then the multiplicand is subtracted from A and

if the combination is 01 then the multiplicand is added with A.

In both the cases results are stored in A, and after the addition or subtraction operation, A, Q, Q-1 are right shifted.

The shifting is the arithmetic right shift operation where the leftmost bit namely; An-1 is not only shifted into An-2 but also remains in An-1. This is to preserve the sign of the number in A and Q. The result of the multiplication will appear in the A and Q[7].

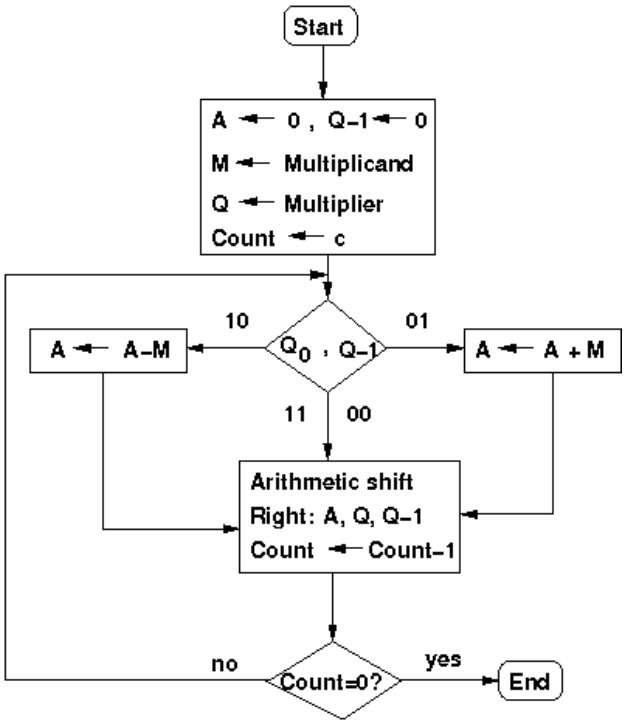

Figure-2 Design flow of Booth algorithm

B. Design Of Booth Multiplier

Booth's algorithm can be implemented in many ways. This experiment is designed using a controller and a data path. The operations on the data in the data path are controlled by the control signal received from the controller. The data path contains registers to hold multiplier, multiplicand, intermediate results, and data processing units like ALU, adder/subtractor etc., counter and other combinational units

Following is the schematic diagram of the Booth's multiplier which multiplies two 4-bit numbers in 2's complement of this experiment.
Here the adder/subtractor unit is used as data processing unit M, Q, A are 4-bit and Q-1 is a 1- bit register.

M holds the multiplicand,
Q holds the multiplier,
A holds the results of adder/subtractor unit.

The counter is a down counter which counts the number of operations needed for the multiplication. The data flow in the data path is controlled by the five control signals generated from the controller. These signals are *load* (to load data in registers), *add* (to initiate addition operation), *sub* (to initiate subtraction operation), *shift* (to initiate arithmetic right shift operation), *dc* (this is to decrement counter).
The controller generates the control signals according to the input received from the data path. Here the inputs are the least significant Q0 bit of Q register, Q-1 bit and count bit from the down counter.[7]

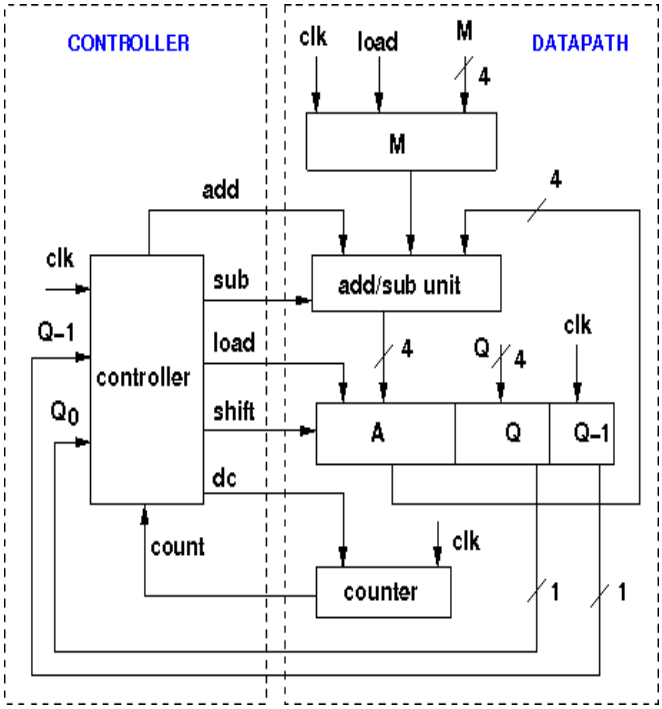

Figure-3 Schematic diagram of the Booth's multiplier which multiplies two 4-bit numbers in 2's complement

C. *Performance*

The method of Booth multiplication reduces the numbers of adders and hence the delay required to produce the partial sums . The high performance of booth multiplier comes with the drawback of power consumption. The reason is large number of adder cells required that consumes large power [5].

V. SIMULATION RESULTS

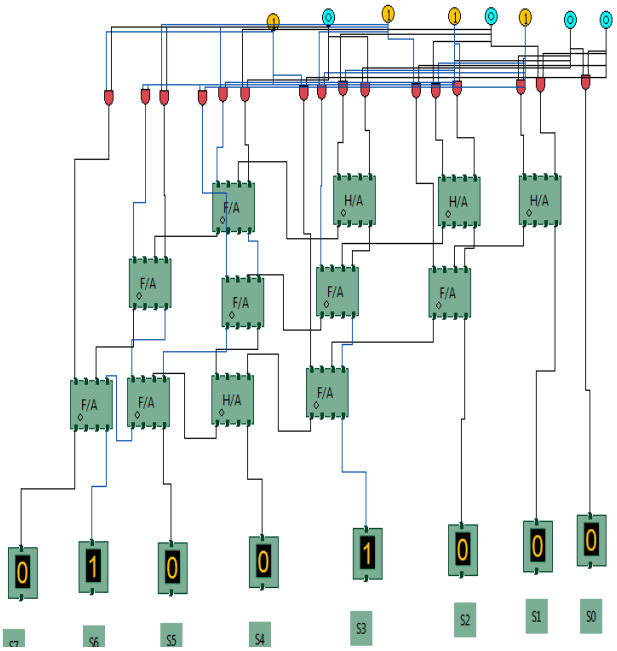

Figure-4 combinational multiplier implementation on simulator

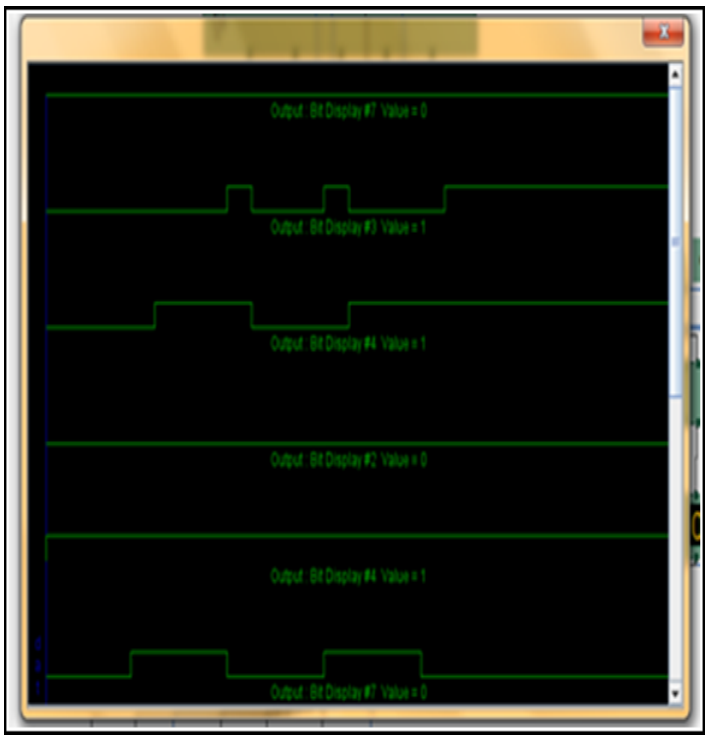

FIGURE-5 GENERATED WAVEFORM OF COMBINATIONAL MULTIPLIER.

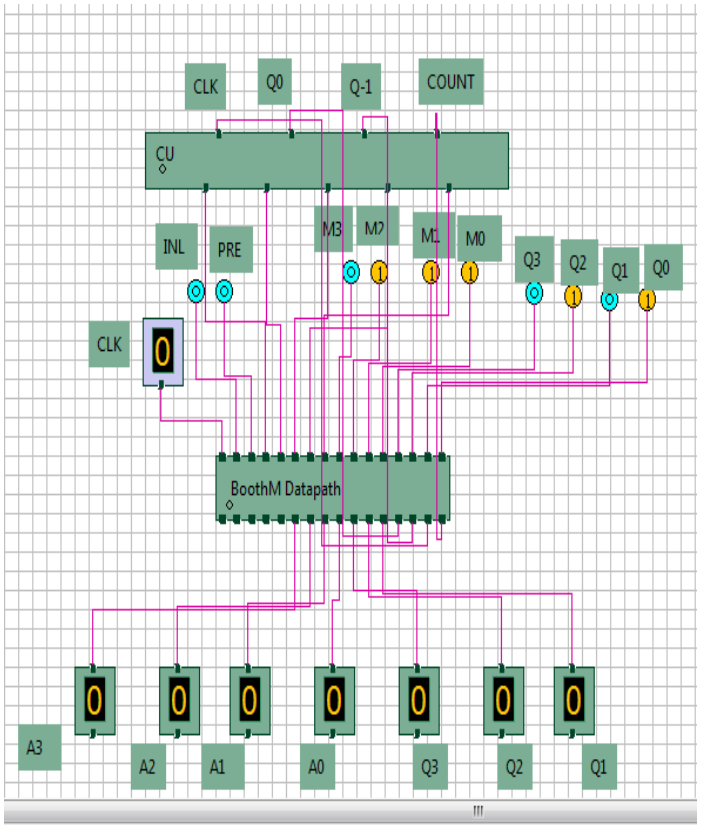

FIGURE-6 IMPLEMENTATION OF BOOTH MULTIPLIER ON SIMULATOR

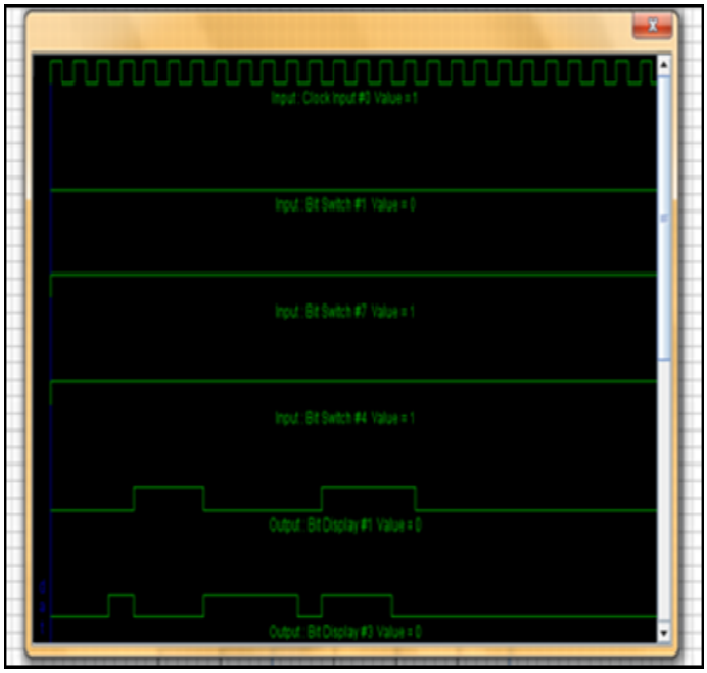

FIGURE-7 GENERATED WAVEFORM OF BOOTH MULTIPLIER

VI. CONCLUSION

It can be concluded that Booth Multiplier is superior in respect like area, Complexity. In booth multiplier number of gates is reduced and hence area of booth multiplier is less than combinational multiplier. However Combinational Multiplier gives optimum number of components required. Hence for less delay requirement Booth's multiplier is suggested.

Further work can be carried out on this project in the power estimation section and to improve the speed or to minimize the delay and power of multipliers.

Booth Multiplier can be further extended to-

- Modified booth multiplier
- Radix -2 booth multiplier
- Radix- 4 booth multiplier

ACKNOWLEDGMENT

We are thankful to our Principal **Dr. S.K Gupta** for providing necessary facilities to wards carrying out this work. We acknowledge the diligent efforts of our head of department Mr**. Dinesh Sorout in** assisting us towards implementations of this idea.

REFERENCES

[1] Dr. Ravi Shankar Mishra,Prof. Puran Gour,Braj Bihari Soni / International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com "Design and Implements of Booth and Robertson's multipliers algorithm on FPGA"

[2] Anju, "Performance Comparison of Vedic Multiplier and Booth Multiplier" in International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-2, Issue-5, June 2013.

[3] Deepali Chandel1, Gagan Kumawat, Pranay Lahoty, Vidhi Vart Chandrodaya , Shailendra Sharma "Booth Multiplier: Ease of multiplication" in International Journal of Emerging Technology and Advanced Engineering Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 3, Issue 3, March 2013).

[4] Schaum's Outline of Theory and Problems of Computer Architecture Copyright © The McGraw-Hill Companies Inc. Indian Special Edition 2009

[5] Sumit Vaidya and Deepak Dandekar "Delay-power performance Comparison of multipliers in VLSI Circuit design" in International Journal of Computer Networks & Communications (IJCNC), Vol.2, No.4, July 2010..

[6] Morris Mano, "Computer System Architecture" ,PP. 346-347, 3rd edition, PHI. 1993.

[8] Aditya Deshpandey, Jeff Draper "Squaring Units and a Comparison with Multipliers".

[9] virtual-labs.ac.in/labs/cse10/booth.html 1/

[10] virtual-labs.ac.in/labs/cse10/cmult.html

[11] http://www.parl.clemson.edu/~dstanzi/201lab/lab7.pdf

[12] http://www.engineering.uiowa.edu/~carch/lectures10/55035-100303-prn-pdf

[13] http://users.encs.concordia.ca/~asim/COEN_6501/Lecture_Notes/L3Notes.Pdf.

[14] http://en.wikipedia.org/wiki/Booth's_multiplication_algorithm

AUTHORS

Neha Goyal is a M.Tech student in Computer Science and Engineering at Shri Ram College Of Engineering & Management, Palwal. (goyalgirl19@gmail.com)

Khushboo Gupta is a M.Tech student in Computer Science and Engineering at Shri Ram College Of Engineering & Management, Palwal. (khushboo9024@gmail.com)

Renu Singla is working as a lecturer in department of computer science at Shri Ram College of Engineering & Management, Palwal . She has published research paper in the area of indexing in oracle (ranu1987singla@gmail.com).