iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 9 **Issue:** XII **Month of publication:** December 2021

DOI: https://doi.org/10.22214/ijraset.2021.39398

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue XII Dec 2021- Available at www.ijraset.com

Design and Implementation of Radix 8 Booth Encoding Multiplier for Low Area and High-Speed Applications

Mutyala Sri Anantha Lakshmi[1], P. Pushpalatha[2]
[1]*PG Student, Department of Electronics and Communication Engineering, VLSI and Embedded Systems, University College of Engineering Kakinada (Autonomous) - 533003, Andhra Pradesh, India*
[2]*Assistant Professor, Department of Electronics and Communication Engineering, University College of Engineering Kakinada (Autonomous) - 533003, Andhra Pradesh, India*

***Abstract:** In this paper, we present the design and implementation of the Radix 8 Booth Encoding Multiplier. There are many multipliers in existence in which Radix 8 Booth Encoding Multiplier offers a decrease in area and provides high speed due to its diminution in the number of partial products. This project is designed and simulated on Xilinx ISE 14.7 version software using VHDL (Very High Speed Integrated Circuit Hardware Description Language). Simulation results show area reduction by 33.4% and delay reduction by 45.9% as compared to the conventional method.*
***Keywords:** Booth Multiplier, Radix 8, Partial Product*

I. INTRODUCTION

In a digital signal processing systems, multipliers play an important role. Multiplication is one of the most fundamental and crucial mathematical operations in programming. In reality, multiplication operations account for 9.72 percent of all instructions in typical scientific applications. Further, these multipliers are employed in the design of various filter architectures. In the past, several multipliers were proposed with compact space, low power, and great performance in mind. Multiplication is achieved by the addition of a certain number of partial products rows. Each partial product row is created by multiplying the multiplier bit to multiplicand one by one. In general to multiply two 8 bit inputs it produces 8 partial products which are complex and require more number of half adders and full adders to get the final multiplied result. This increases the design in terms of both area and complexity. The incorporation of efficient multipliers in the design of architectures enhances the overall efficiency of the design in terms of different parameters like area and delay. So a Radix 8 Booth Encoding Multiplier is proposed which offers diminution in these parameters.

II. RADIX 8 BOOTH ENCODING MULTIPLIER

The Booth multiplier is the most commonly used multiplier to increase the multiplier performance. By using this Radix 8 booth encoding multiplication, for N bit multiplication it generates (N/2 - 1) partial products that is for 8-bit multiplication it generates (8/2 - 1) which is equal to 3 partial products.

Thus by Booth encoding, the number of partial products rows that must be combined to provide the multiplication result can be minimized, further the number of half adders and full adders[1] required to add these partial products decreases which is a major advantage [2]-[4].

A. Booth Algorithm

The booth algorithm is a multiplication algorithm that allows us to multiply the two signed binary integers in 2's complement, respectively. It is very efficient and speeds up the multiplication process performance.

Andrew Donald Booth devised the algorithm in 1950 while working on crystallographic research at Birkbeck College in Bloomsbury, London. Booth devised a method to improve the speed of desk calculators that were faster at shifting than adding. In the field of computer architecture, Booth's algorithm is fascinating.

It works on the string bits 0's in the multiplier that requires no additional bit only shift the right-most string bits and a string of 1's in a multiplier bit weight 2k to weight 2m that can be considered as 2k+ 1 - 2m.

B. Radix 8 Modified Booth Algorithm

Radix 8 represents that the number of bits (multiplier and multiplicand bits) that are used in the multiplication is eight-bit binary.

Here the multiplication process completes in 3 steps.

1) *Step 1:* Every multiplication contains a multiplier (P) and a multiplicand (C). Consider an 8bit multiplication, initially, the multiplier bits are segregated into groups i.e. 4 bits are grouped with one overlap bit. Later formed groups are passed to a decoder, where it will decide which operation has to be performed on the multiplicand. *(The operations to be performed is shown in the Radix 8 booth encoding table, Table 1)*
2) *Step 2:* It will generate the partial products by performing a corresponding operation on the multiplicand.
3) *Step 3:* Finally, the obtained partial products in the above step are to be added using adders.

TABLE 1: Radix 8 Booth Encoding Table

Segregated Multiplier 4-bits	Corresponding Operation
0000	0
0001	1 X C
0010	1 X C
0011	2 X C
0100	2 X C
0101	3 X C
0110	3 X C
0111	4 X C
1000	-4 X C
1001	-3 X C
1010	-3 X C
1011	-2 X C
1100	-2 X C
1101	-1 X C
1110	-1 X C
1111	0

C. Sign Extension Corrector

The Sign Extension Corrector is meant to expand the capacity of the Booth multiplier by multiplying both the unsigned and signed numbers.

The following is the principle of sign extension, which converts a signed multiplier to a non signed multiplier:

When unsign is signalled, that is unsigned bit value = 0 indicates that the unsigned number is being multiplied, while unsigned bit value = 1 indicates that the signed number is being multiplied.

Table 2: Sign Extension Corrector

Sign	Corresponding Operation
0	Unsigned Multiplication
1	Signed Multiplication

III. RESULTS AND DISCUSSION

This project is implemented on Xilinx software 14.7 version using VHDL language, Spartan 6 device family, XC6SLX100T Device and the package used is FGG900 with a speed grade of "-3".

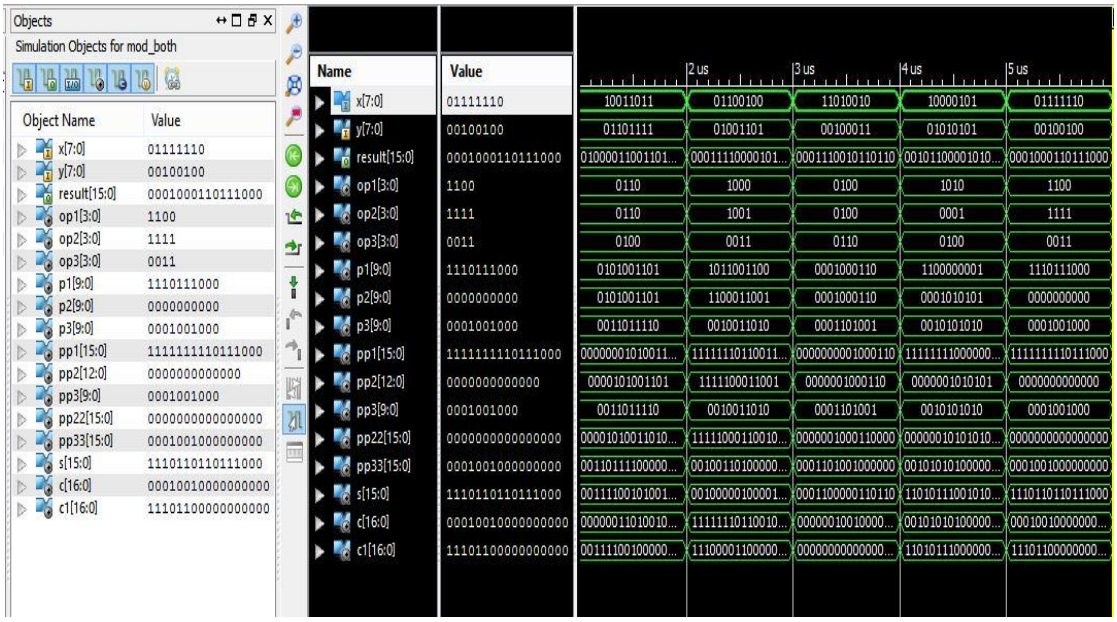


Figure 1: Simulation result of Radix 8 Booth Encoding Multiplier using Weinberger Adder

The comparison of specific parameters such as area and time between the existing and proposed methods is shown in Table 3. By these values, we can say that the proposed method requires less area and provides high speed compared to the existing conventional multiplier.

Table 3: Comparison of the existing and proposed method

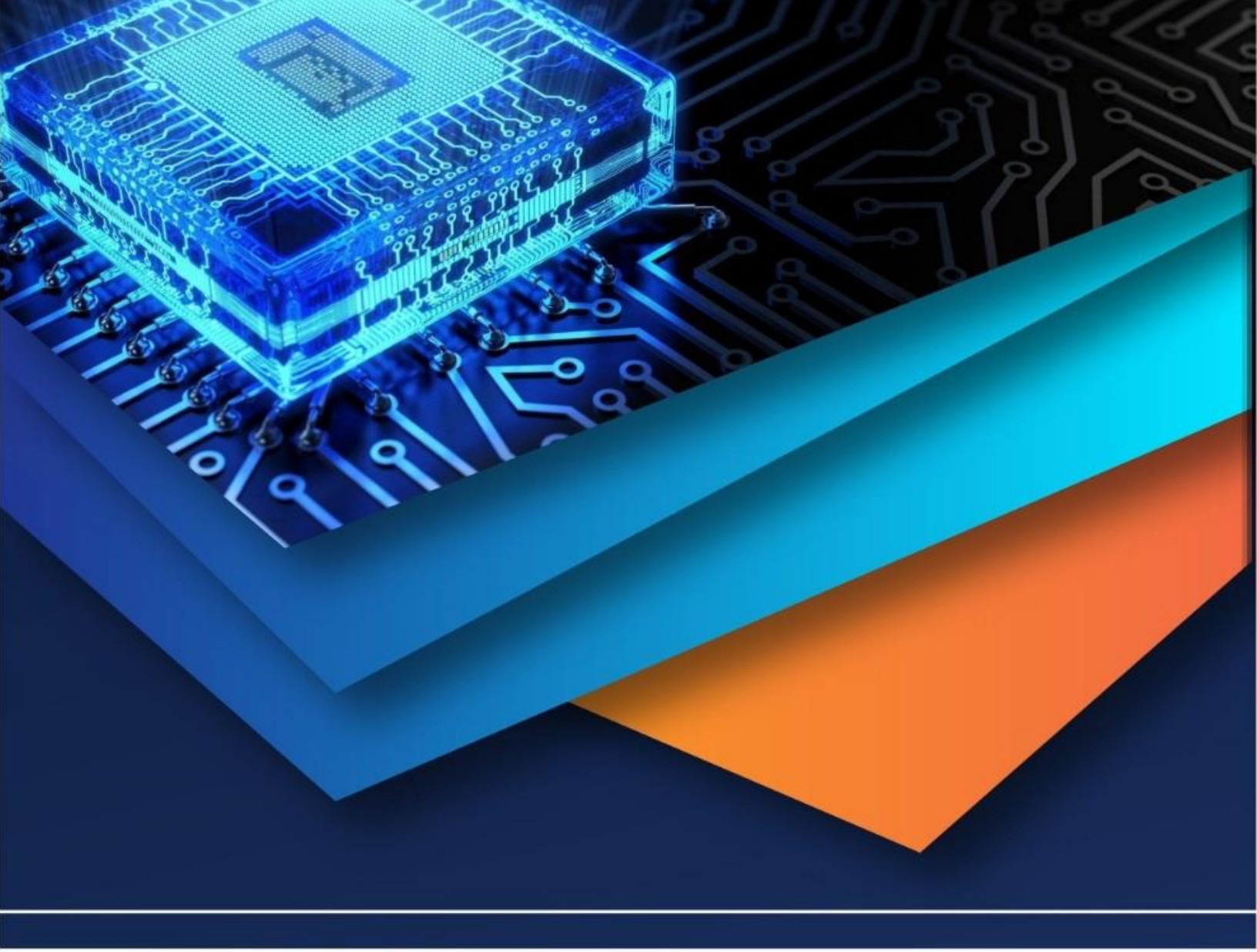
Parameter	Existing Method (General Multiplier)	Proposed Method (Radix 8 Booth Encoding Multiplier)
Area	871	580
Time(ns)	29.024	15.704

IV. CONCLUSION

In this brief, by the design and implementation of Radix 8 Booth Encoding Multiplier, it has been proved that the speed of multiplication can be increased and also it requires less area compared to the conventional method. Thus this proposed method can be used for less area and high-speed digital signal processing applications.

REFERENCES

[1] Comparison of High-Speed VLSI Adders, Jayanthi A.N. (2013) [22] High-performance VLSI adders, R. Suganya, D. Meganathan, (2015) D. Kornack and P. Rakic, "Cell Proliferation without Neurogenesis in Adult Primate Neocortex," Science, vol. 294, Dec. 2001, pp. 2127-2130, doi:10.1126/science.1065467.
[2] Design of Low Power Approximate Radix-8 Booth Multiplier, K. Sindhuja, C. Thiruvenkatesan, RTICCT – 2017 (Volume 5 – Issue 17)
[3] Approximate Radix-8 Booth Multipliers for Low-Power and High-Performance Operation, (2015), IEEE Transactions on Computers
[4] High Speed-Low Power Radix-8 Booth Decoded Multiplier, International Journal of Computer Applications (0975 – 8887) Volume 73– No.14, July 2013



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY