A Novel Power Efficient Pre Encoded Modified Booth Multiplier Encoder

Saumya Sharma
M Tech Scholar
Department of Electronics and Communication Engineering
LNCT, Bhopal

Bharti Gupta, PhD
Reasearch Guide
Department of Electronics and Communication Engineering
LNCT, Bhopal

ABSTRACT

The pre-encoded multipliers for encoding is quite useful for the digital signal processing in various applications of communication and data processing devices.The modified booth encoder proposed in this work is the technique to simplify the products implementation with the improvements in power consumption.The proposed architecture has less power than previous architecture. The synthesis results show the power required for proposed architecture is 14mW only. This will work longer on the same power as given to previous design.

Keywords

Pre-encoded, Booth Multiplier, Encoder

1. INTRODUCTION

For implementing a digital multiplier a large variety of computer arithmetic algorithms could be used. Most techniques take into consideration generating a set of partial products, and then adding the partial products together once they have been shifted. In a multiplier to increase its speed, the number of partial product to be generated should be reduced. A higher representation radix effectively indicates to fewer digits. Thus, a single-digit multiplication algorithm necessitates fewer cycles as to start moving to much higher radices, which automatically leads to a lesser number of partial products. Several algorithms have been developed for this purpose like Booth's Algorithm, Wallace Tree method etc. For the summation process several adder architectures are available viz. Ripple Carry Addition, Carry Look-ahead Addition, Carry Save Addition etc. But to reduce the power consumption the summation architecture of the multiplier should be carefully chosen.

A general multiplier is needed if one performs multiplication between two arbitrary variables. However, when multiplying by a known constant, it can exploit the properties of binary multiplication in order to obtain a less expensive logic circuit that is functionally equivalent to simply asserting the constant on one input of a general multiplier. In many cases, using a cheaper implementation for only multiplication still results in significant savings when considering the entire logic circuit because multiplication is relatively expensive. Furthermore, multiplication could be the dominant operation, depending on the application.

The giant strides humanity has taken in terms of technological progress, though unfathomable, has been punctuated by significant achievements. The hallmark among them being in the fields of VLSI Design. The greatest advantage in this field lies in the immense scope for future developments. Improved system organization and efficient computing algorithms have added to the reliability of high-speed processing. That too at a low and affordable price. As a result computers have circumvaletted human progress with a touch of finesse and efficiency and encroached into spheres like animation, electronic design, telecommunications, space research and innumerable other ones.

To write program for the implementation of any digital circuit there are various languages available, called as Hardware Description Language e.g. Verilog, VHDL. For designing Verilog (standardized as IEEE 1364 HDL) language used for programming. Verilog is one of the common techniques used in digital system emergent process. The technique is implemented in program using certain software which carries out simulation and examination of the designed system. The designer only needs to describe the digital circuit design in textual form which can remove without the effort to alter the hardware. XILINX 13.1 platform has used to write proposed design programs. All the RTL simulations has been done using this software only. Also for delay report the synthesis tool embedded in Xilinx was used.

The elementary purpose of this research work is to instrument the Booth's Algorithm for the design of binary multiplier using different adder architectures and carry out power analysis at various levels. Also the delay, area and power optimization is to be taken care of. To implement Booth's algorithm for multiplier design because it reduces the number of partial products generated in a multiplication process and reduction in number of partial products results in a faster multiplication.

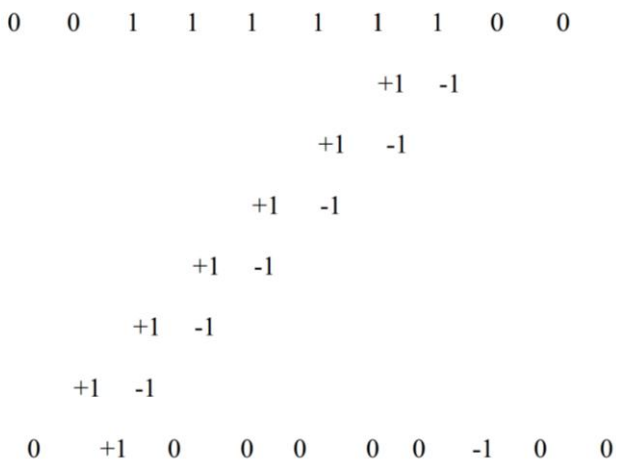
2. BOOTH'S ENCODING

Booth's encoding or Booth's multiplication algorithm [1] is a multiplication algorithm which can multiply two signed binary numbers in a two's complement notation. Booth's algorithm has the ability to perform fewer additions and subtractions in comparison to normal multiplication algorithm.

It is an encoding process which can be used to minimize the no of partial products in a multiplication process. It is based upon the expression.

$$2^n = 2^{n+1} - 2^n \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots.(1)$$

Example



```
0   0   1   1   1   1   1   1   0   0
                        +1  -1
                    +1  -1
                +1  -1
            +1  -1
        +1  -1
    +1  -1
0  +1   0   0   0   0   0  -1   0   0
```

Figure 2.1 Example of Booth encoding.

Booth's algorithm examines consecutive bits of the N-bit multiplier Y in signed two's complement representation, which includes an implicit bit below the least significant bit, y_{-1}= 0. For each bit y_i as i runs from 0 to N-1, the bits y_i and yj are considered. When these two bits are equal, the product accumulator P stays unchanged. Where y_i = 0 and y_{i-1} = 1, the multiplicand times 2i is added to P; and where yi = 1 and y_{i-1} = 0, the multiplicand times 2i gets subtracted from P. The final value of P will be the signed product.

The representation of the multiplicand and product are not specified; typically, these are also in two's complement representation, like a multiplier, but any number system that supports addition and subtraction will work as well. The order of the steps is not determined. Generally, it proceeds from LSB to MSB, starting at i = 0; the multiplication by 2i is then replaced by incremental shifting of the P accumulator to the right between steps; low bits will be shifted out, and subsequent additions or subtractions can then be done just on the highest N bits of P. There are many variations and optimizations on these details.

The algorithm is often described as converting strings of 1's in the multiplier to a high- order +1 and a low-order -1 at the ends of the string. When the string runs through the MSB, there is no high-order +1, and the net effect is interpretation as a negative of the appropriate value.

3. PROPOSED ARCHITECTURE

Proposed multiplier architecture is based on modified booth encoder provide power efficiency. Figure 3.1 demonstrated conventional, modifiedBooth encoder.

One of the many solutions of realizing high speed multipliers is enhancing parallelism which helps in decreasing the number of subsequent calculation levels. The original version of Booth algorithm (Radix-2) had two particular drawbacks. They were:

- The number of add-subtract operations and shift operations become variable and causes inconvenience in designing parallel multipliers.
- The algorithm becomes inefficient when there are isolated 1' s.

These problems are overwhelmed by using modified Radix4 Booth algorithm which scans strings of three bits using the algorithm given below:

The indispensable figure-of-merit of a digital circuit are speed and power consumption with the spped being measured in terms of a (reciprocal) delay time or a maximum clock frequency. Efficiency of power could be defined as the total power or also in terms of the switching energy, i.e., the average energy spent for one switching transition of the digital device.

1) Lengthen the sign bit 1 position if necessary to ensure that n is even.
2) Add a 0 to the right of the LSB of the multiplier.
3) Corresponding to the value of each vector, each Partial Product will be 0, +M, -M, +2M or -2M.

The negative values of M are made by taking its 2's complement. The multiplication of M is done by shifting M by one bit to the left (in case it's multiplied with 2). Thus, in any case, in designing an n-bit parallel multiplier, only n/2 partial products are generated. Figure 3.2 demonstrated the modified booth encoder multiplier.

Total power dissipated in a design can be broadly divided in two categories: static and dynamic.

A. Static Power

Static power is the power dissipated by a gate when it's not switching. It is caused by the current that flows through the transistors even when they are turned off.

B. Dynamic Power

Dynamic power is the power dissipated when the circuit is active i.e. while performing some function.

- Switching Power

Switching power can be defined as the power which is dissipated while charging and discharging the output load capacitance.

- Internal Power

Internal power is consumed within a cell while charging and discharging internal cell capacitances. Short-circuit power is also included in the Internal power.

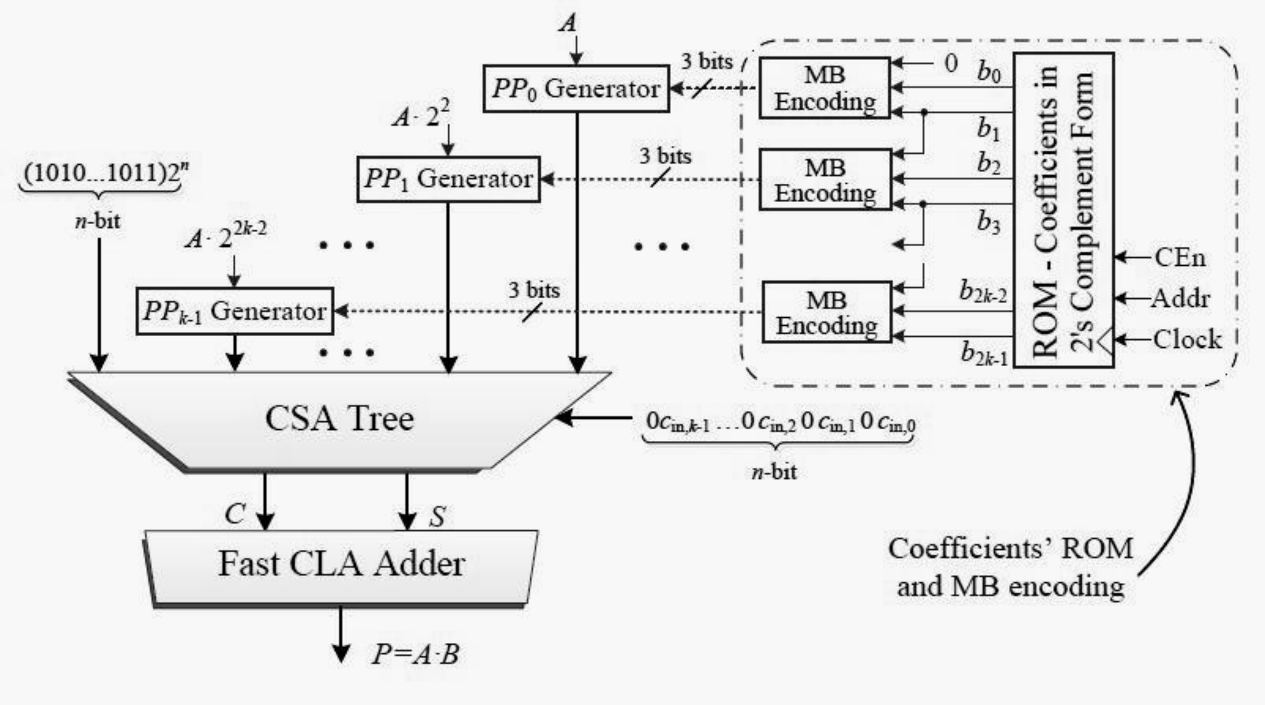


Fig 3.1 System Architecture of Conventional Modified Booth Encoder

As noted above the dynamic power consumed by a circuit depends on the logic transitions which occur within the design while operating. Therefore, for power estimation and optimization it needs to accurately specify this information (called switching activity) to the tool performing these tasks. Dynamic power represents the majority of total power.

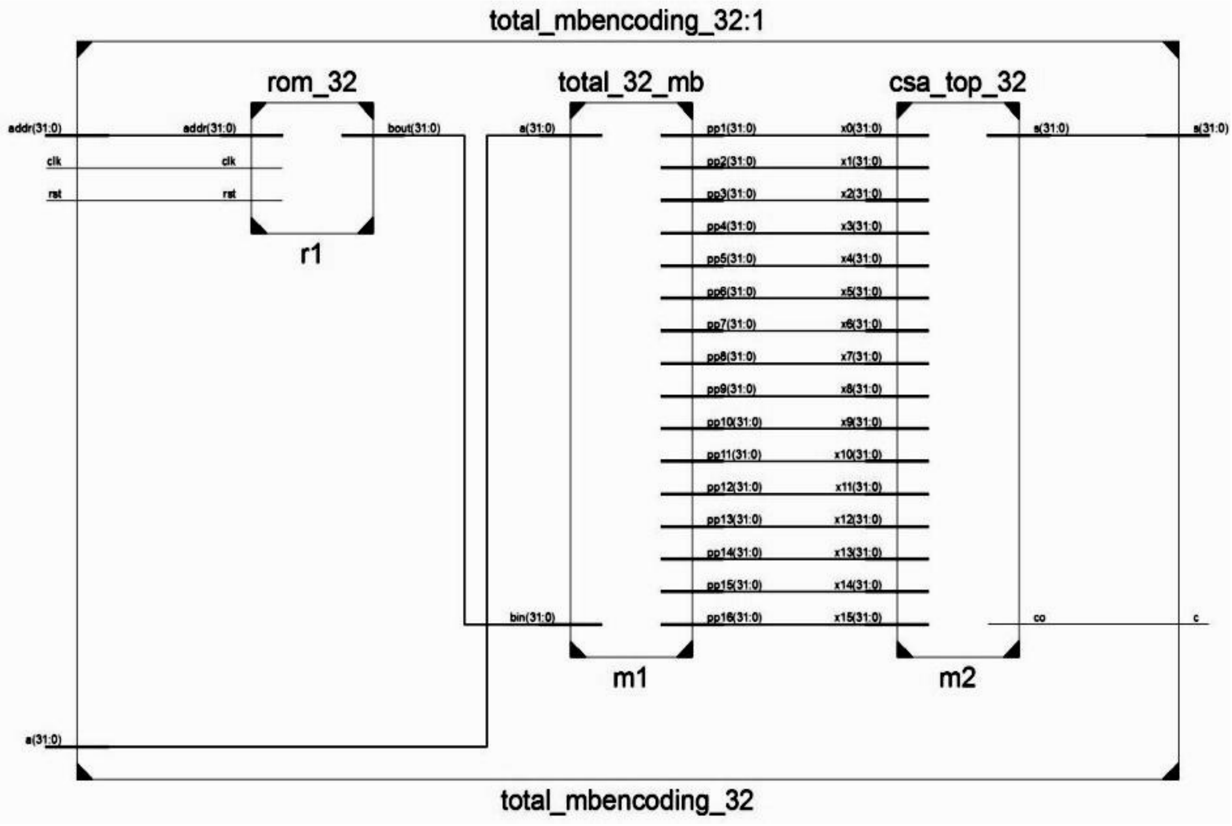
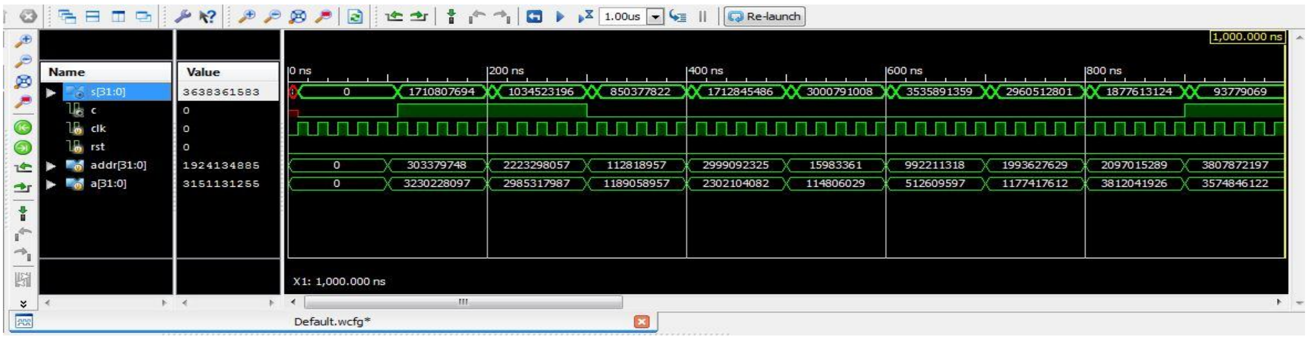


Fig. 3.2 System Architecture of Proposed Pre-Encoded Modified Booth Multiplier Encoder.

4. SYNTHESIS RESULTS

Implementation of proposed work has done on Xilinx 13.1. ISE and simulation of proposed work has on ModelSim simulator. The come wave form of proposed work has given in figure below. Figure 4.1 illustrate the 4.1 execution of proposed based on text bench function of Xilinx. In figure 4.1 there are three parameters clk- Clock, rst-Reset. s[31:0] is a 32 bit sum output, and two input addr and a are 32 bit input signal.Figure 4.3 demonstrated the power analysis report of proposed work. Comparison of performance of proposed Booth encoder with existing encoder is given in table 1 where proposed work is better than the existing work with respect to area and power.



Name	Value
s[31:0]	3638361583
c	0
clk	0
rst	0
addr[31:0]	1924134885
a[31:0]	3151131255

Figure 4.1 Waveform screen of outcome.

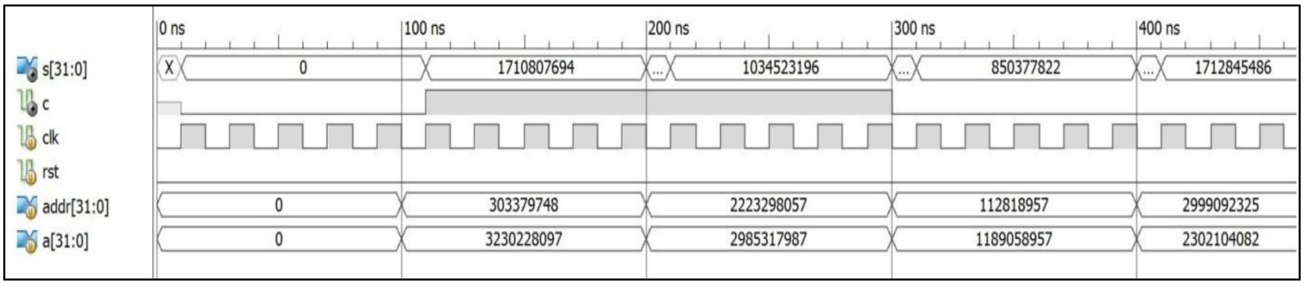
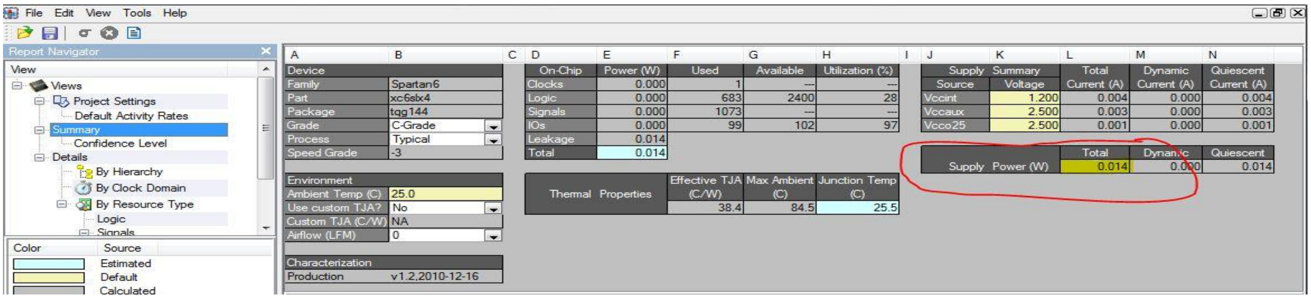


Figure 4.2 Outcome waveform of proposed work.



Color	Source
🟦	Estimated
🟨	Default
▨	Calculated

Device	
Family	Spartan6
Part	xc6slx4
Package	tqg144
Grade	C-Grade
Process	Typical
Speed Grade	-3

Environment	
Ambient Temp (C)	25.0
Use custom TJA?	No
Custom TJA (C/W)	NA
Airflow (LFM)	0

Characterization	
Production	v1.2,2010-12-16

On-Chip	Power (W)	Used	Available	Utilization (%)
Clocks	0.000	1	---	---
Logic	0.000	683	2400	28
Signals	0.000	1073	---	---
IOs	0.000	99	102	97
Leakage	0.014			
Total	0.014			

Thermal Properties	Effective TJA (C/W)	Max Ambient (C)	Junction Temp (C)
	38.4	84.5	25.5

Supply Summary		Total	Dynamic	Quiescent
Source	Voltage	Current (A)	Current (A)	Current (A)
Vccint	1.200	0.004	0.000	0.004
Vccaux	2.500	0.003	0.000	0.003
Vcco25	2.500	0.001	0.000	0.001

	Total	Dynamic	Quiescent
Supply Power (W)	0.014	0.000	0.014

Figure 4.3 Power analysis chart of propped work.

Table 1 Result comparison table

System Design	Power Requirements	Area Utilization
Proposed 32 Bit	14 mW	Slice Registers: **63** Slice LUTs: **4058** Fully used LUT-FF pairs: **50** Bonded IOBs: **195**
Previous 32-Bit [1]	27.3 mW	Slice Registers: **126** Slice LUTs: **8116** Fully used LUT-FF pairs: **100** Bonded IOBs: **3900**

5. CONCLUSION

Some of the application demand for the multiplication to be achieved at a faster rate, while others tend to exploit less hardware and therefore more modest speed. The various design strategies have been exploited, in order to meet the demand of these applications. In this work implementation and synthesis of 32 bit modified booth encoder has done for the improvements in power consumption. The new design provides a better time measured compared with similar structure available in literature in terms of power, area usage and area-time complexity. Modified Booth's algorithm is used to reduce the number of the partial products and improve the total hardware area of the similar algorithm.

In future more work can be done for negative numbers, and for the application floating point operation, pipelining approach can be applied and any other proper enhancement techniques.

6. REFERENCES

[1] Small area modified booth multiplier design for predetermined coeffi¬cients," IEICE Trans. Fundam. Electron. Commun. Comput. Sci., vol. E90-A, no. 3, pp. 694-697, Mar. 2007.

[2] K. K. Parhi, VLSI Digital Signal Processing Systems: Design and Implementation. John Wiley & Sons, 2007

[3] B. C. Paul, S. Fujita and M. Okajima, "ROM-Based Logic (RBL) Design: A Low-Power 16 Bit Multiplier," in IEEE Journal of Solid-State Circuits, vol. 44, no. 11, pp. 2935-2942, Nov. 2009.

[4] R. P. Rajput and M. N. S. Swamy, "High Speed Modified Booth Encoder Multiplier for Signed and Unsigned Numbers," 2012 UKSim 14th International Conference on Computer Modelling and Simulation, Cambridge, 2012, pp. 649-654.

[5] N. G. NikDaud, F. R. Hashim, M. Mustapha and M. S. Badruddin, "Hybrid modified booth encoded algorithm-carry save adder fast multiplier," The 5th International Conference on Information and Communication Technology for The Muslim World (ICT4M), Kuching, 2014, pp. 1-6.

[6] B. Dinesh, V. Venkateshwaran, P. Kavinmalar and M. Kathirvelu, "Comparison of regular and tree based multiplier architectures with modified booth encoding for 4 bits on layout level using 45nm technology," 2014 International Conference on Green Computing Communication and Electrical Engineering (ICGCCEE), Coimbatore, 2014, pp. 1-6.

[7] K. Tsoumanis, N. Axelos, N. Moschopoulos, G. Zervakis and K. Pekmestzi, "Pre-Encoded Multipliers Based on Non-Redundant Radix-4 Signed-Digit Encoding," in IEEE Transactions on Computers, vol. 65, no. 2, pp. 670-676, Feb. 1 2016.