Comparative Analysis of Booth Multiplier Algorithms

Course Project EE593

Shri Janani Senthil*, Rupesh Yadav†, Anisha Sharma‡
School of Computing and Electrical Engineering, IIT Mandi
Himachal Pradesh, India
Email: *b20232@students.iitmandi.ac.in, †b20226@students.iitmandi.ac.in, ‡b20029@students.iitmandi.ac.in

***Abstract*—Multipliers are commonly used in various signal processing and machine learning applications alike. Discussed in this paper is a comparative analysis of the power, delay and transistor count of two circuits that generate the partial product for the traditional radix-4 Booth multiplier and a radix-4 Booth multiplier with a pre-encoded mechanism. The proposed design outperforms traditional designs in terms of power consumption, however has major shortcomings in terms of output voltage swing. The designs were made on Cadence Virtuoso using TSMC 60 nm technology and simulations were analysed on Cadence Spectre.**

***Index Terms*—Booth algorithm, low power multiplier, power efficiency, partial product, radix-4 Booth multiplier.**

I. Introduction

The array multiplier is the multiplier that follows the traditional method of generating n partial products for $n \times n$ multiplication. It was first implemented, imitating the mathematical procedure followed normally. However, it utilises too much hardware to generate the partial products, resulting in heavy power consumption.

Thus, the Booth multiplier (radix-2) utilises an algorithm that generates partial products based on encoding consecutive bits into a sequence of booth digits (-1, 0, +1). The steps involved are as follows:

1) Pre-processing: Convert the multiplicand and multiplier into their signed binary representations. Extend the sign bit of the multiplier to match the desired word length.
2) Initialization: Set the product to zero and initialize a register (Booth register) to hold the previous Booth digit.
3) Iteration: Repeat the following steps for each bit in the multiplier, starting from the least significant bit:
   a) Retrieve the current Booth digit by examining the current bit and the previous bit in the multiplier. The Booth digit can be -1, 0, or +1.
   b) Based on the Booth digit, perform one of the three operations:
      i) If the Booth digit is -1, add the multiplicand to the product.
      ii) If the Booth digit is +1, subtract the multiplicand from the product.
      iii) If the Booth digit is 0, do nothing.
   c) Shift the product and the Booth register one bit to the right.
4) After the iterations are complete, the product in the register represents the result of the multiplication.

The radix-4 Booth algorithm can improve the performance of multiplication because the radix-4 Booth multiplier can reduce the number of PP rows by half. It partitions the multiplicand into three contiguous bits and encodes them according to table 1. Then they are decoded with the multiplicand to generate partial products.

y_{2i+1}	y_{2i}	y_{2i-1}	M_i	Operation on X
0	0	0	0	0X
0	0	1	+1	+1X
0	1	0	+1	+1X
0	1	1	+2	+2X
1	0	0	-2	-2X
1	0	1	-1	-1X
1	1	0	-1	-1X
1	1	1	0	0X

TABLE I: Encoding in radix-4 Booth Multiplier

We present a pre-encoded mechanism to reduce the power consumption of the radix-4 Booth multiplier by modifying the method of generating the partial products. This is done by detecting the case that generates a partial product of value zero before every multiplication. This ensures that the decoders do not need to go through unnecessary switching to end up with the known final partial product of zero in the 0X case observed in Table 1.

y_{2i+1}	y_{2i}	y_{2i-1}	Operation on X	$Zero_i$	Neg_i	ot_i
0	0	0	0X	1	0	d
0	0	1	+1X	0	0	1
0	1	0	+1X	0	0	1
0	1	1	+2X	0	0	0
1	0	0	-2X	0	1	0
1	0	1	-1X	0	1	1
1	1	0	-1X	0	1	1
1	1	1	0X	1	1	d

TABLE II: Control signals generated during pre-encode phase

Since the decoders are not functional for generating such zero partial product, they will remain turned off through a pre-encoding mechanism and hence power is saved. The multiplicand X and multiplier Y would be set on the data

bus during the data setup time before the multiplication, called 'pre-encode' phase. To implement the pre-encoding mechanism we place a pre-encoder that instructs the encoders and decoders to be switched off when the '0X' case is detected and to function as per usual otherwise. The pre-encoded phase detection and subsequent signals to control the encoder and decoders are described in Table 2.

The overall architecture of a Booth multiplier with the proposed mechanism is shown in fig. 1.

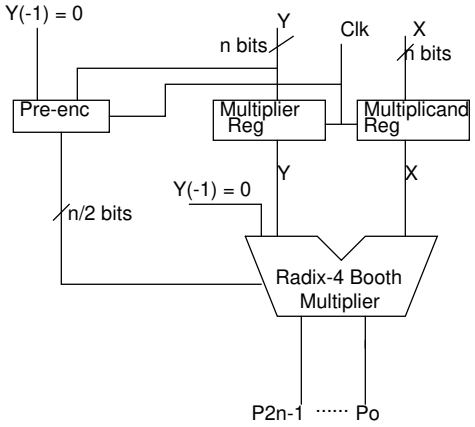

Fig. 1: Overall Architecture of proposed Booth multiplier

II. Circuits Simulated

The circuits simulated for each of the encoders and decoders used in both designs are discussed as follows:

1) Decoder in Traditional design:

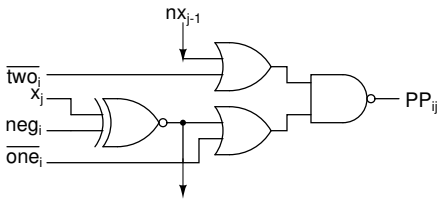

Fig. 2: Gate-level design - Decoder in Traditional design

2) Encoder in Traditional Design:

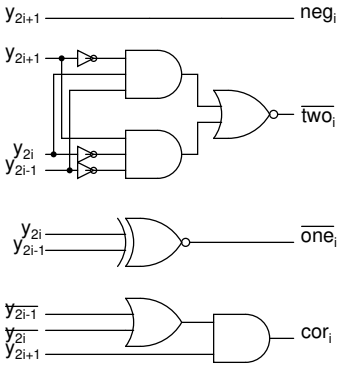

Fig. 3: Gate-level design - Encoder in Traditional design

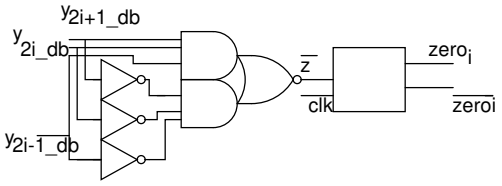

Fig. 4: Gate-level design - Pre-encoder

3) Pre-encoder in Improved Design:
4) Decoder in Improved Design:

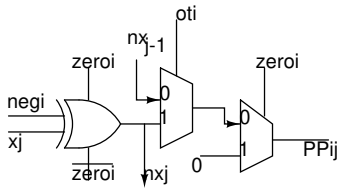

Fig. 5: Gate-level design - Improved decoder

5) Encoder in Improved Design:

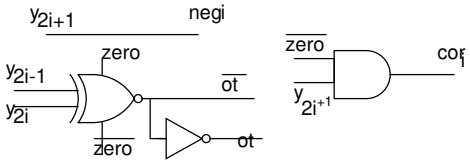

Fig. 6: Gate-level design - Improved encoder

The sizing of the transistors were done in accordance with the minimum inverter size 1, i.e., the width of PMOS to NMOS were in the ratio 2:1. The multiplexers in the improved decoder (in fig. 5) were designed as per pass-transistor logic.

III. Results

The multiplier X was considered to be of value 35. The input was given as a constant voltage from DC voltage source. The values of each bit are depicted in table 3.

X_7	X_6	X_5	X_4	X_3	X_2	X_1	X_0
0	0	1	0	0	0	1	1

TABLE III: Multiplier Bits

The multiplicand Y was considered to be of value 45. The input was given from voltage pulse source. The HIGH value was given as 1 V and the LOW value was given as 0 V. The values are clearly depicted in fig. 7 and table 4.

S. No.	y_{2i-1}	y_{2i}	y_{2i+1}
1	0	1	0
2	1	1	0
3	1	0	1
4	0	0	1

TABLE IV: Multiplier bits y_i

Expected output bits of partial products are depicted in table 5.

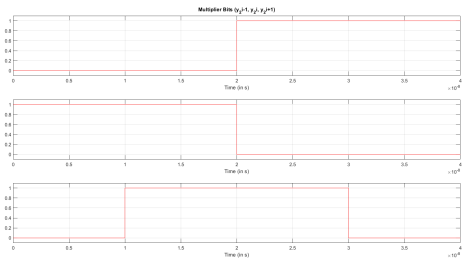

Fig. 7: Inputs y_{2i-1}, y_{2i} and y_{2i+1}

PP_{i7}	PP_{i6}	PP_{i5}	PP_{i4}	PP_{i3}	PP_{i2}	PP_{i1}	PP_{i0}
1	0	0	0	0	1	1	0
0	1	0	1	1	0	0	0
0	1	0	1	1	0	0	0
1	0	0	0	0	1	1	0

TABLE V: Caption

A. Results in Traditional design

The partial product bits were obtained as expected as shown in figs. 8 and 9. We can observe full output voltage swing.

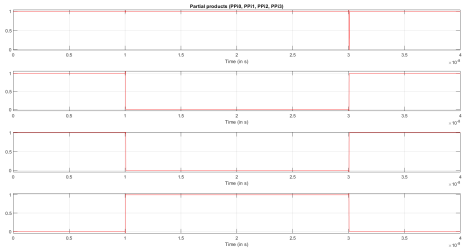

Fig. 8: Partial Products PP_{i0},PP_{i1},PP_{i2},PP_{i3}

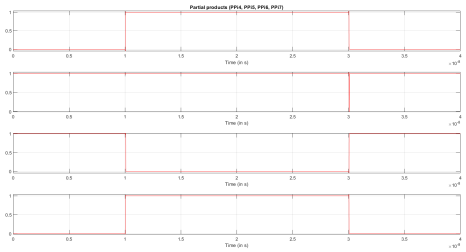

Fig. 9: Partial Products PP_{i4},PP_{i5},PP_{i6},PP_{i7}

The partial product bits were obtained as expected as shown in figs. 8 and 9. We can observe a highly reduced output voltage swing.

B. Power Consumption

The power plots obtained through spectre allowed us to observe that the average power consumed by the traditional design was higher than that consumed by the improved design. However a major shortcoming was that the traditional design had full output voltage swing and better noise margin in comparison to the improved design. The corresponding values have been tabulated in table 6.

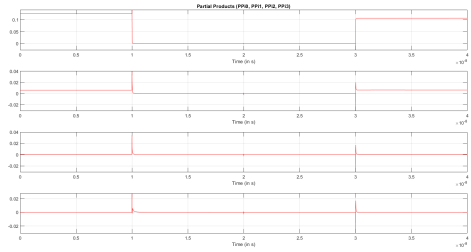

Fig. 10: Partial Products PP_{i0},PP_{i1},PP_{i2},PP_{i3}

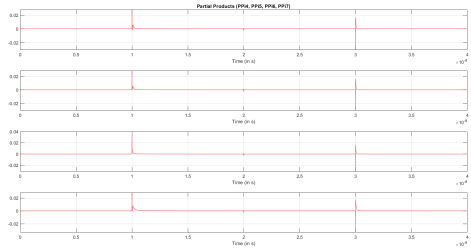

Fig. 11: Partial Products PP_{i4},PP_{i5},PP_{i6},PP_{i7}

C. Conclusions

The power analysis can easily be reasoned out on the basis of how the pre-encoding mechanism avoids unnecessary switching for the case when the partial product bit is known to be 0. The power dissipation that occurs due to this switching is by-passed by means of the pre-encoder that instructs the encoder and decoder to be defunct when this case arises among the multiplier bits in the pre-encoding phase.

On the other hand, due to the use of pass-transistor logic the improved design does not have full output voltage swing. This is due to the reason that the output of pass-transistor logic based circuits always give rise to 'weak-ones'. We observe that while the low voltage reaches to the origianl low level sufficiently, the same does not apply for the high voltage.

Design	Avg Power (in μ W)	Max Power (in mW)	V_{out} Swing (in mV)
Traditional	152	3.6	999.99
Improved	4.3	2.8	152.56

TABLE VI: Power Analysis and Voltage swing for the two designs