EVALUATION OF BOOTH ENCODING TECHNIQUES FOR PARALLEL MULTIPLIER IMPLEMENTATION

David Villeger and Vojin G. Oklobdzija

Indexing Terms: Booth Encoding, Parallel Multiplier

Although generally used in parallel multipliers, Booth encoding is shown to be obsolete due to the improvements in bit compression trees. It was found that a single row of 4:2 compressors reduces the number of partial products to one half, which is the essential function of the Booth encoding technique. With a single row of 4:2 compressors this reduction is achieved in less time and with fewer gates used.

Intorduction: Booth's algorithm [1] is widely used in the implementations of hardware or software multipliers because its application makes it possible to reduce the number of partial products. It can be used for both sign-magnitude numbers as well as 2's complement numbers with no need for a correction term or a correction step.

Booth-Mac Sorley Recoding: A modification of the Booth algorithm was proposed by Mac Sorley [2] in which a triplet of bits is scanned instead of two bits. This technique has the advantage of reducing the number of partial products by one half regardless of the inputs. The recoding is performed within two steps: encoding and selection. The purpose of the encoding is to scan the triplet of bits of the multiplier and define the operation to be performed on the multiplicand, as shown in Figure 1. This method is actually an application of a sign-digit representation in radix 4. The Booth-MacSorley algorithm, usually called the Modified Booth algorithm or simply the Booth algorithm, can be generalized to any radix. For example, a 3-bit recoding would require the following set of digits to be multiplied by the multiplicand : 0, ±1, ±2, ±3. The difficulty lies in the fact that ±3Y is computed by summing (or subtracting) 1 to ±2Y, which means that a carry propagation occurs. The delay caused by the carry propagation renders this scheme to be slower than a conventional one. Consequently, only the 2 bit Booth recoding is used and therefore considered in this paper.

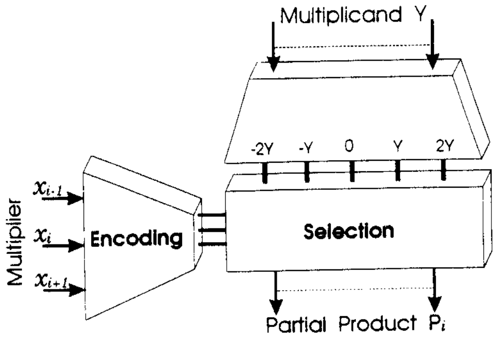

Figure 1: *Implementation of Modified Booth recoding*

Booth Recoding Versus the Use of 4:2 Compressors: Booth recoding necessitates the internal use of 2's complement representation in order to efficiently perform subtraction of the partial products as well as additions. However, floating point standard specifies sign magnitude representation which is followed by most of the non-standard floating point numbers in use today. Thus, we assume the use of sign magnitude representation and compare those multiplier implementations using Booth encoding with the ones not using it but resorting to efficient partial product addition techniques such as the use of 4:2 compressors.

The advantage of Booth recoding is that it generates only a halve of the partial products compared to the multiplier implementation which does not use Booth recoding. However, the benefit achieved comes at the expense of increased hardware complexity. Indeed, this implementation requires hardware for the *encoding* and for the *selection* of the partial products (0, ±Y, ±2Y). An optimized ***encoding*** is shown in Figure 2. The multiplexers and buffers are considered to be equivalent to an XOR gate. This implementation is then equivalent to a level of XOR gates and a level of AND gates. The ***selection*** can be implemented with a simple 5 input multiplexer, which is roughly equivalent to 3 XOR gates. However, since one input is grounded, this circuit can be designed with only a 4 input multiplexer, that is 2 XOR gates, and an AND gate.

In this case, the Booth recoding circuit is equivalent to 3 XOR plus 2 AND gates.

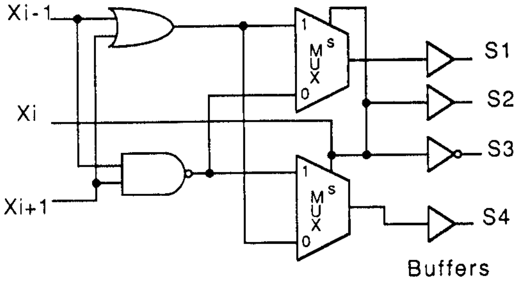

Figure 2: *An Optimized Encoding Circuit*

On the other hand, reducing the number of partial products by one half can be achieved with one level of AND gates and one row of 4:2 compressors. The 4:2 cell is designed with 3 XOR levels as shown in Figure 3 and implemented in [4,5]. The use of higher order compressors would result in even higher levels of compression.

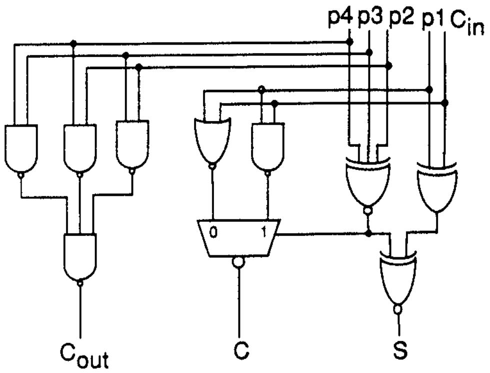

Figure 3: *Structure of 4:2 Compressor Cell*

However, main disadvantage of the Booth technique is the complexity introduced by the internal use of 2's complement representation which is necessary to compute negative partial products. Indeed, since the Booth recoding method calculates -Y and -2Y, it needs to extend the sign of negative partial products. It further needs to complement Y when -Y or -2Y are needed, that is to calculate -Y = Inv (Y) + 1 where Inv (Y) means inversion of every bit of Y. Consequently, two extra bits are necessary in the scheme: one for the sign extension and one for conversion into 2's complement. Both of the bits will be placed in the same row, therefore not increasing the number of rows. However, the correction bit (which is needed for correct sign calculation) will be placed right in the middle of the multiplier tree therefore not only increasing the number of rows by one but creating this increase in the worse possible place, i.e. in the critical path of the multiplier.

The conclusions are summarized is Table 2.

Table 2. COMPARISON OF SIGN-MAGNITUDE NUMBER MULTIPLICATION WITH AND WITHOUT BOOTH ENCODING.

Booth encoding	No Booth encoding
Internal representation : 2's complement (some partial products need to be subtracted)	*Internal representation :* sign magnitude (all the partial products are positive)
Hardware for encoding and selection	One row of 4:2 compressors
Sign extension	Only one XOR is used to compute the sign in parallel
2 extra bits (sign extension and complementation	No extra bit
The normalization requires some Leading Zero Detectors and Leading One Detectors	The normalization and even the rounding are easy [5]
The schematic and the layout are not regular.	The simplicity of the schematic allows a highly regular layout
1 XOR + 1AND (encoding), 2 XOR+1 AND (multiplexer) Total: 3 XOR + 2 AND	1 AND (partial product generation), 3 XOR (4:2 compressor) Total: 3 XOR + 1 AND

Conclusion: When Booth recoding is used the schematic and the layout of the resulting implementation are less regular leading to a more difficult design or VHDL description. In terms of speed, the Booth technique is at best equal or worse than the use of the 4:2 compressors. In the case of 2's complement representation and without Booth encoding, the last row of partial product (depending on the sign of the multiplier) is generated by using an AND gate with an inverted input. In other words, the number of gate levels is the same as in the sign magnitude case. However, the sign extension is needed with or without Booth encoding. This feature makes the two schemes comparable, although using 4:2 compressors is slightly better because of the simplicity and the fewer number of gate levels.

Acknowledgment: We thank Thierry Soulas and Simon Liu for thier input.

David Villeger *(Ecole Superieure d'Ingenieurs en Electrotechnique et Electronique 93162 Noisy le Grand CEDEX FRANCE)*

Vojin G. Oklobdzija *(Electrical and Computer Engineering Department University of California Davis, CA 95616)*

References

1 A. D. Booth, *"A Signed Binary Multiplication Technique"*, Qarterly J. Mechan. Appl. Math., Vol. IV, 1951.

2 O. L. Mac Sorley, *"High Speed Arithmetic in Binary Computers"*, Proceedings of IRE, Vol.49, No. 1, January, 1961.

3 A. Weinberger, *"4:2 Carry-Save Adder Module"*, IBM Technical Disclosure Bulletin.,Vol.23, January 1981.

4 J. Mori et al, *"A 10nS 54X54-b Parallel Structured Full Array Multiplier with 0.5-u CMOS Technology"*, IEEE Journal of Solid State Circuits, Vol. 26, No. 4, April 1991.

5 T. Soulas, D.Villeger, V. G. Oklobdzija, *"An ASIC Multiplier for Complex Numbers"*, Proceedings of EURO-ASIC-93, Paris, FRANCE, February 22-25, 1993.