NEGABINARY ADDITION AND MULTIPLICATION USING BINARY CIRCUITS

Dharma P. Agrawal

Electrical and Computer Engineering
Wayne State University
Detroit, MI 48202

Ramesh C. Joshi

Electronics and Communication Engineering
University of Roorkee
Rookee, U.P., India

Abstract

This paper deals with design techniques that allow adoption of off-the-shelf binary arithmetic circuits to perform multi-operand addition and multiplication of negabinary numbers. The multiple operands could be easily added with augmented binary adders connected in the form of a tree. The same hardware could be used to perform multiplication. But, from an LSI implementation viewpoint, the use of cellular array structures is explored and necessary changes to design a combined binary/negabinary multiplier unit, are also outlined.

Introduction

The concept of a negative base number system is not new. But only recently, growing interest has been shown by various researchers[1-2]. Negabinary or -2 base is a special case of negative radix system and because of its numerical closeness to the binary system, negabinary is the most frequently selected negative base system utilized in designing logic circuits for arithmetic operations[3-12]. Range extension in a negative radix system requires addition of leading zeros and this simplicity has been used in describing algorithms for multi-operand addition[2]. This property has also been used in designing special purpose hardware such as implementation of digital filters[13-14] and the fast-fourier transformation[15].

The addition of two negabinary numbers has been observed to require two carries[1,2]. Thus, for a general purpose application, the units for the negabinary system is considered to be more complex than the binary system. This generalized notion has been shown to be not exactly true and it has been demonstrated by Agrawal[9] that a logic circuit for a negative addition of two negabinary numbers is no more complex than the binary addition. It has also been shown that, how minor modifications allow the use of off-the-shelf binary circuits for negabinary negative addition[16-17].

The objective of this paper is to extend the usefulness of negative negabinary addition to achieve multiple-operand addition. A similar scheme could also be utilized for negabinary multiplication. From LSI suitability viewpoint, cellular-array-type structures are to be preferred. A combined binary/negabinary multiplier design necessitates that the basic arithmetic cell of the cellular array has to be more general than for the array performing only one base arithmetic.

Multiple Operand Addition

One scheme of negative negabinary addition (n.n.b.a.) using the usual binary adders, has been presented by Murugesan[16]. An alternative procedure has bee described by Agrawal[17] where, in place of odd-positioned input bits, even-positioned bits are to be inverted. Similar complementary changes are needed at the output of the added. The two possible approaches have been illustrated in Figs. 1.a and 1.b.

Either one of the two n.n.b.a. could be used to perform multiple operand addition. Let A, B, C, D ... etc. be the negabinary numbers to be added and let Z be their sum, then:

$$Z = (A + B + C + D + \ldots)$$

$$= -[-(A + B) - (C + D) \ldots] \qquad (1)$$

The relation (1) is written in such a way so that n.n.b.a. could be used as a primitive operation. This leads to realization of (1) in a tree-structured from with n.n.b.a. employed at each step. The corresponding logical scheme is shown in Fig. 2.

Negabinary Multiplication

The multiplication operation requires nothing but bit-successive addition and shift operations. In other words, once the multiplier is known, the problem of multiplication is the same as the addition of the properly shifted multiplicand. This means that a multiple-operand-addition technique could be easily utilized. Thus, the scheme of Fig. 2, with minor adjustments for shifts, could perform multiplication.

The design of Fig. 2, is not suitable for its LSI implementation. This dictates the use of iterative arrays. Several binary multiplier arrays have been considered in the literature. Before we start considering the actual array, it is worth considering one special case of n.n.b.a. when the bit b_0 of Fig. 1.a and Fig. 1.b. is zero. This leads to a simpler version of n.n.b. adders as shown in Fig. 3.a and 3.b. This form is useful in every alternative step of multiplication where a shift is required and n.n.b.a. is carried out. The two alternatives of Figs. 1 and 3 require either inversion at inputs or at the outputs of the full-adder. This requires the use of two types of arithmetic cells and the two such cells are shown in Fig. 4.a and 4.b. The function of the two cells is to work as a controlled adder and can be described as follows:

Type 1:

(i) when $b_i=1$, $S_{out} = S_{in} \oplus a_i \oplus C_{in}$ (2.a)

and $\bar{C}_{out} = C_{in}(S_{in}+a_i) + S_{in}\cdot a_i$ (2.b)

(ii) when $b_i=0$, $S_{out} = S_{in} \oplus C_{in}$ (3.a)

and $\bar{C}_{out} = C_{in}$ (3.b)

Type 2:

(i) when $b_i=1$, $\bar{S}_{out} = S_{in} \oplus \bar{a}_i \oplus C_{in}$ (4.a)

and $C_{out} = C_{in}(\bar{S}_{in} + \bar{a}_i) + \bar{S}_{in}\cdot\bar{a}_i$ (4.b)

(ii) when $b_i=0$, $S_{out} = \bar{S}_{in} \oplus C_{in}$ (5.a)

and $C_{out} = C_{in}$ (5.b)

The two cells can be used to obtain two multiplier structures as shown in Figs. 5.a and 5.b. The multiplier of Fig. 5.a is based on Figs. 1.a and 3.a while Fig. 5.b utilizes the design background indicated in Figs. 1.b and 3.b.

It may be noted that the result $Z = A(8b_3 + 4b_2 + 2b_1 + b_0)$, where Z and A both are negabinary numbers. This means $B \equiv (8b_3+4b_2+2b_1+b_0)$ represented by 4 bits $b_3b_2b_1b_0$ must be a binary number. Thus, the arrays of Fig. 5.a and Fig. 5.b will be working on negabinary multiplicant A wile the multiplier B will be in binary form. Conversion from binary to negabinary is not at all difficult and several algorithms have already been presented in the literature[17].

The advantage of negabinary system in digital filter applications has already been demonstrated[13-15]. The filter works on a mixed radix system. Thus, there is a growing need to design a multiplier that could work on both binary and negabinary numbers. This requires modification of the basic arithmetic cell and a single control line can decide the operating mode (X=0 for binary, X=1 for negabinary). Thus two basic cells (with minor differences) will be used and are shown in Figs. 5.a and 6.b. Once these cells are used as basic modules, then inverters will no longer be needed and the array structure will completely look uniform. Of course, all even-positioned columns utilize the cell of Fig. 6.a while all odd-positioned columns utilize the cell of Fig. 6.b.

Conclusion

This paper describes the design of a multiple operand adder for negabinary numbers. Two alternate cellular array structures for negabinary multiplication, have also been given. It is hoped that the arithmetic units described in this paper will find wide acceptance in the area of signal processing.

References

1. P.V. Sankar, et al., "Arithmetic Algorithms in a Negative Base," IEEE Trans. Comp., Vol. C-22, No. 2, Feb. 1973, pp. 120-125.

2. D.P. Agrawal, "Arithmetic Algorithms in a Negative Base," IEEE Trans. Comp., Vol. C-24, No. 10, Oct. 1975, pp. 998-1000.

3. D.V. Kanani and K.N. O'Keeff, "A Note on Conditional-Sum Addition for Base-2 Systems," IEEE Trans. Comp., Vol. 22, 1973, p. 626.

4. L.S. Houselander, "Cellular-array Negabinary Multiplier," Electronics Lett., Vol. 10, 1974, pp. 168-169.

5. D.P. Agrawal, "Negabinary Carry-look-ahead Adder and Fast Multiplier," Electronics Lett., Vol. 22, 1974, pp. 312-313.

6. G.S. Rao, et al., "Logic Design of a Negative Binary Adder-Substractor," Int. Jour. Elec., 1974, pp. 537-542.

7. D.P. Agrawal, "Negabinary Complex-number Multiplier," Elecronics Lett., 1974, pp. 502-503.

8. C.K. Yuen, "A Note on Base-2 Arithmetic Logic," IEEE Trans. Comp., Vol. C-24, No. 3, March 1975, pp. 325-329.

9. D.P. Agrawal, "Negabinary Parallel Counters," Digital Processes, Vol. 1, 1975, pp. 75-85.

10. D.P. Agrawal, "Negabinary Division and Square-rooting," Digital Processes, Vol. 1, 1975, pp. 267-275.

11. J.P. Chinal, "Boolean Features of Sign Flipping and Signed-Carry Logic in Base-2," IEEE Trans. Comp., Vol. C-26, No. 1, Jan. 1977, pp. 82-86.

12. D.P. Agrawal, "A Novel Technique for Computing Negabinary Squares," IEEE Trans. Comp., Vol. C-27, No. 3, March 1978, pp. 266-270.

13. S. Zohar, "New Hardware Realizations of Nonrecursive Digital Filters," IEEE Trans. Comp., Vol. C-22, No. 4, April 1973, pp. 328-338.

14. S. Zohar, "The Counting of Recursive Digital Filters," IEEE Trans. Comp., Vol. C-22, No. 4, April 1973, pp. 338-347.

15. S. Zohar, "Fast Hardware Fourier Transformation through Counting," IEEE Trans. Comp., Vol. C-22, No. 5, May 1973, pp. 433-441.

16. S. Murngesan, " Negabinary Arithmetic Circuits Using Binary Arithmetic," IEE Jour. Electro. Cir. Sys., Vol. 1, No. 2, 1977, pp. 77-78.

17. D.P. Agrawal, "On Negabinary-Binary Arithmetic Relationships and their Hardware Reciprocity," IEEE Trans. Comp., Vol. C-29, No. 11, Nov. 1980, pp. 1032-1035.

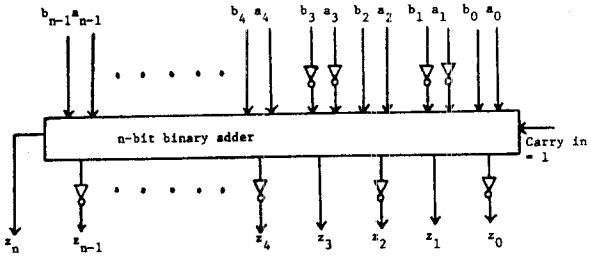

Fig. 1.a Logic diagram for negative negabinary addition (n.n.b.a.) using binary adder (for odd value of n)

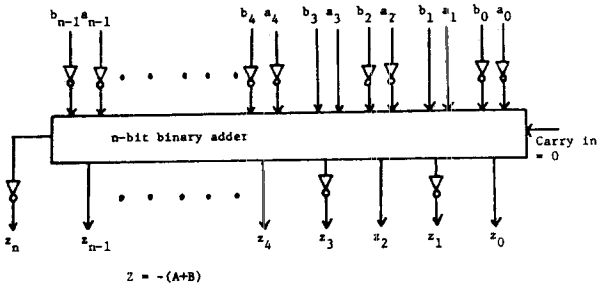

Fig. 1.b Logic diagram for n.n.b.a. using binary adder - second method (for odd n)

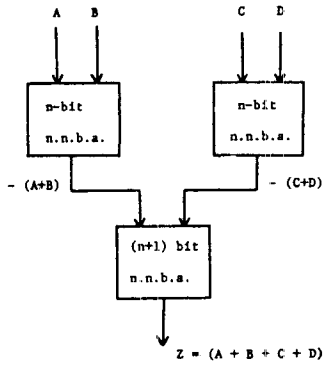

Fig. 2. Tree structure of n.n.b.a.'s for multiple operand addition.

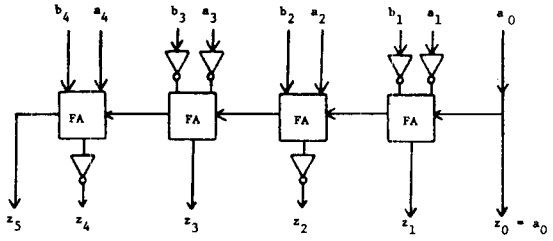

Fig. 3.a Simpler version of Fig. 1.a when $b_0 = 0$ (FA = Full Adder)

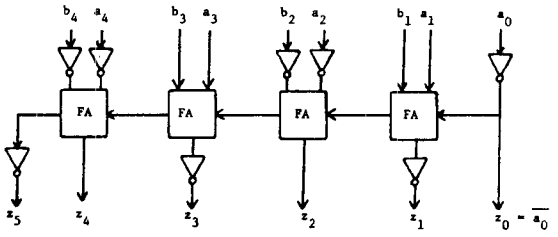

Fig. 3.b Simpler version of Fig.1.b when $b_0 = 0$.

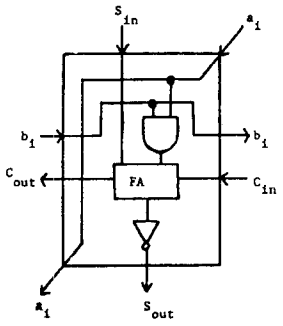

Fig. 4.a Arithmetic cell type 1 (inverter at the output)

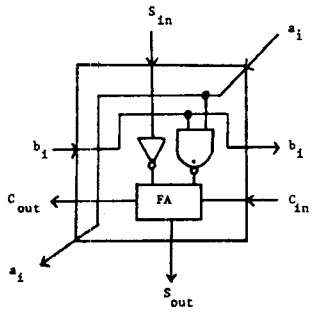

Fig. 4.b Arithmetic cell type 2 (inverters at the inputs)

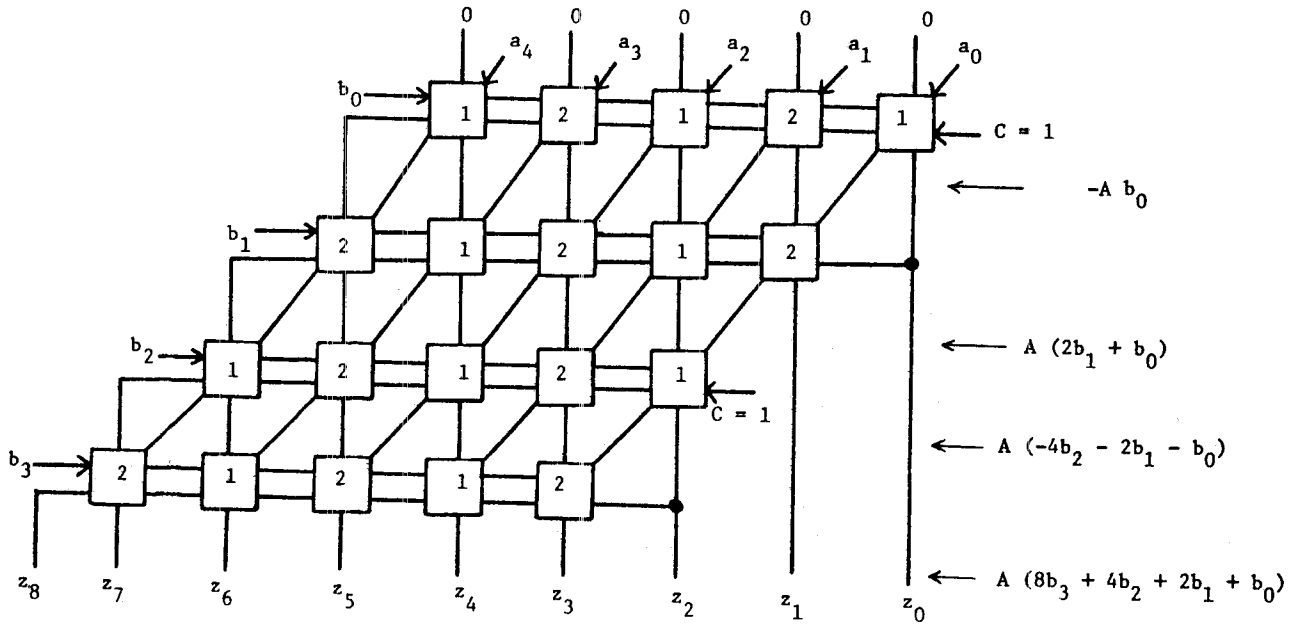

Fig. 5.a Negabinary Multiplier using Figures 1.a and 3.a
$Z = A \cdot B$ (The type of arithmetic cell is represented by 1 and 2)

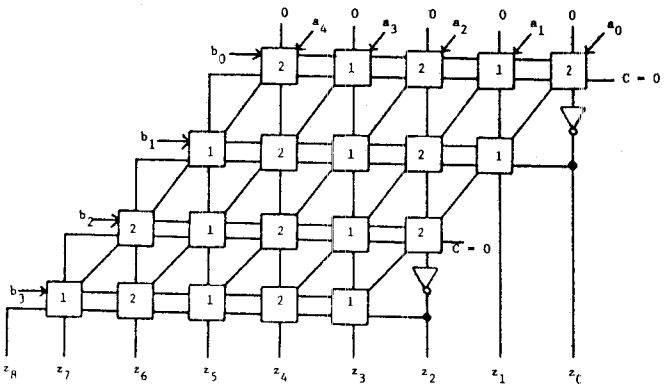

Fig. 5.b An alternate negabinary multiplier using Figs. 1.b and 3.b
$Z = A \cdot B$

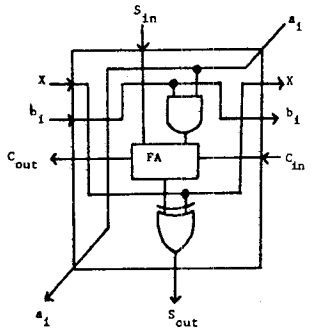

Fig.6.a Universal cell of type 1
X = 0 for binary addition
X = 1 for negabinary negative addition

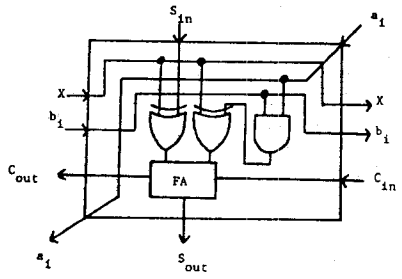

Fig. 6.b Universal cell of type 2