ISTANBUL UNIVERSITY –
JOURNAL OF ELECTRICAL & ELECTRONICS ENGINEERING

YEAR : 2004 (1025-1030)
VOLUME : 4
NUMBER : 1

A PERFORMANCE ANALYSIS OF CLASSIFIED BINARY ADDER ARCHITECTURES AND THE VHDL SIMULATIONS

Ahmet SERTBAŞ[1] **R.Selami ÖZBEY** [2]

[1,2] Department of Computer Engineering, Istanbul University
Avcilar, 34850 Istanbul,TURKEY

[1]E-mail: asertbas@istanbul.edu.tr

ABSTRACT

In this paper, the four binary adder architectures belong to a different adder class are studied and compared with each other to analyse their performances. Comparisons include the unit-gate models for area and delay. As the performance measure, the product of the area and the delay is used. By a VHDL simulator, the adder structures are simulated to verify the functional correctness and to measure delay times.

Keywords: *Binary Adders, Unit-gate model, VHDL simulations.*

1. INTRODUCTION

Addition is the most common and often used arithmetic operation on microprocessor, digital signal processor, especially digital computers. Also, it serves as a building block for synthesis all other arithmetic operations. Therefore, regarding the efficient implementation of an arithmetic unit, the binary adder structures become a very critical hardware unit.

In any book on computer arithmetic[1], someone looks that there exists a large number of different circuit architectures with different performance characteristics and widely used in the practice. Altough many researches dealing with the binary adder structures have been done, the studies based on their comparative performance analysis are only a few.

In this study, a qualitative evaluation of the classified binary adder architectures are given. Among the huge member of the adders we wrote VHDL (Hardware Description Language) code for ripple-carry, carry-select, carry-lookahead and conditional sum adder to emphasize the common performance properties belong to their classes. In the following section, we give a brief description of the studied adder architectures.

2. BINARY ADDER ARCHITECTURES

With respect to asymptotic delay time and area complexity, the binary adder architectures can be categorized into four primary classes as given in Table 1. The given results in the table are the highest exponent term of the exact formulas[2], very complex for the high bit lengths of the operands.

The first class consists of the very slow ripple-carry adder with the smallest area. In the second

Received Date : 05.06.2002
Accepted Date: 25.12.2003

class, the carry-skip, carry-select and carry-increment adders with multiple levels have small area requirements and shortened computation times. From the third class, the carry-lookahead adder and from the fourth class, the parallel prefix and conditional sum adders represent the fastest addition schemes with the largest area complexities.

In this section, the circuit structures of the binary adder architectures are given by the set of logic equations, defining single bit cell. In addition, the area and time complexities for each adder architectures based on unit-gate model are given. In this work, we only studied on the four binary adders as the typical structures belong to the different adder classes.

Table 1. Classification of the binary adder architectures

Complex. (A)	*Delay (T)*	*Product (AxT)*	*AdderClass Schemes - l*
O(n)	*O(n)*	*$O(n^2)$*	*ripple-carry* 1
O(n)	*$O(n^{1/*l+1})$*	*$O(n^{l+2/l+1})$*	carry-select 2
			carry-skip 2
			carry-inc. 2
O(n)	*O(logn)*	*O(nlogn)*	*carry -lookahead* 3
O(nlogn)	*O(logn)*	*$O(nlog^2n)$*	*cond.sum* 4
			paral.prefix 4

** l denotes the level number*

2.1. Ripple Carry Adder (RCA)

The well known adder architecture, ripple carry adder is composed of n cascaded full adders for n-bit adder, shown in fig.1.

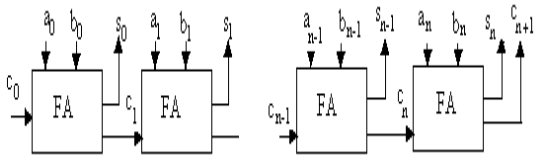

Fig. 1. Ripple Carry Adder (n-bit)

For a bit cell model, the corresponding logic equations, and the delay(T) – area(A) complexity of the ripple carry adder are given below respectively.
Logic equations:

$$g_i = a_i\, b_i\,, \qquad p_i = a_i \oplus b_i \qquad (1)$$
$$c_{i+1} = g_i + p_i c_i\,, \qquad s_i = p_i \oplus c_i$$

For a n-bit RCA structure, complexity and delay:

$$A_{RCA} = O(n) = 7n$$
$$T_{RCA} = O(n) = 2n \qquad (2)$$

2.2. Carry-Select Adder (CSLA)

In this scheme, blocks of bits are added in two ways: one assuming a carry-in of 0 and the other with a carry-in of 1. This process results two precomputed sum and carry-out signal pairs $\{s^0_{i-1:k}, c^0_i; s^1_{i-1:k}, c^1_i\}$, later as the block's true carry-in (c_k) becomes known, the corrected signal pairs are selected. Figure 2 depicts the carry-select adder stucture for n-bit added binary numbers.

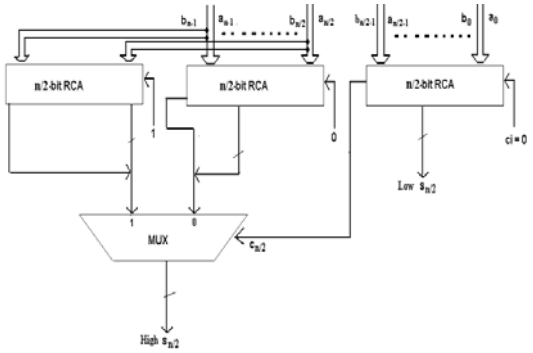

Fig.2. Carry Select Adder with one level (n-bit)

In the following, the logic expressions and complexity of the carry-select adder are given.

Logic equations:

$$s_{i-1:k} = \bar{c}_k\, s^0_{i:-1:k} + c_k\, s^1_{i-1:k} \qquad (3)$$
$$c_i = \bar{c}_k\, c^0_i + c_k\, c^1_i$$

Complexity and delay for n-bit CSLA :

$$A_{CSLA} = O(n) = 14n\,,$$
$$T_{CSLA} = O(n^{1/*l+1}) = 2.8\, n^{1/2} \qquad (4)$$

2.3. Carry-Lookahead Adder (CLA)

Carry-lookahead technique is used to speed up the carry propagation in an adder. The main idea behind it is an attempt to generate all incoming carries in parallel by additioanal logic circuitry.

Let a_i and b_i be the augend and addend inputs, c_i, the carry input, s_i and c_{i+1}, the sum and carry-out to the i^{th} bit position. If the auxiliary functions, p_i and g_i called the propagate and generate

signals, the sum output respectively are defined as follows:

$$p_i = a_i + b_i \quad (5)$$
$$g_i = a_i\, b_i$$
$$s_i = a_i \oplus b_i \oplus c_i$$

Thus, we can express the carry-out signal in terms of p_i and g_i as below:

$$c_{i+1} = g_i + p_i\, c_i \quad (6)$$

The well known expression of the carry recurrence can be yielded as follows:

$$c_{i+1} = g_i + g_{i-1}\, p_i + g_{i-2}\, p_{i-1}\, p_i + .. + c_0\, p_0\, p_1\, p_2 \ldots . p_i \quad (7)$$

Otherwise, for the modular design, the adder is divided into equal sized groups, commonly used 4-bit. Also, providing a carry lookahead over groups leads to increase the speed of the adder. Therefore, the couple group signals for a group of size 4 called the group generated carry, $g_{[i,i+3]}$ and group propagated carry, $p_{[i,i+3]}$ are defined at below.

$$g_{[i,i+3]} = g_{i+3} + g_{i+2}\, p_{i+3} + g_{i+1} p_{i+2}\, p_{i+3} + g_i\, p_{i+1}\, p_{i+2}\, p_{i+3},$$
$$p_{[i,i+3]} = p_i\, p_{i+1}\, p_{i+2}\, p_{i+3} \quad (8)$$

The complexity and the delay of a n-bit CLA:

$$A_{CLA} = O(n) = 4n\ ,$$
$$T_{CLA} = O(\log n) = 4 \log_2 n \quad (9)$$

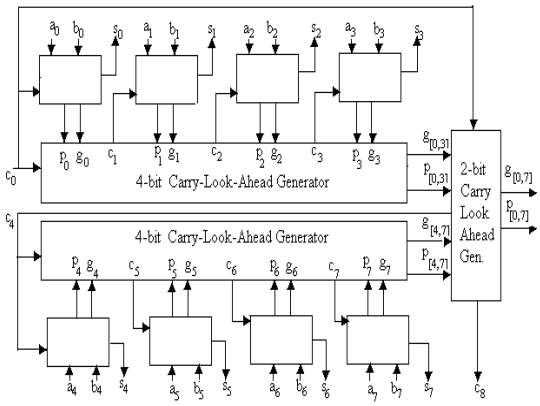

Fig.3. Carry-Look-Ahead Adder (8-bit)

2.4. Conditional Sum Adder (COSA)

Another fast addition scheme is the conditional sum adder, based on the generating two sets of output for a given group of operand bits. Each set includes the sum bits and an outgoing carry. One of sets accepts the incoming carry as zero (0), the other as one (1). Once the incoming carry is known, only the correct set of outputs is selected without waiting for the carry. In this method, the n-bits operands are divided into smaller groups and in this way, the serial carry-propagation inside the separate groups can be done in parallel, increasing the speed of the adder. In principle, the division process into the groups can be continued until a group of size 1. In this case, the all addition process is done in $\log_2 n$ steps, the level number of the used multiplexers. In Fig. 4, the application of conditional sum method is shown for the addition of two 8-bit binary numbers.

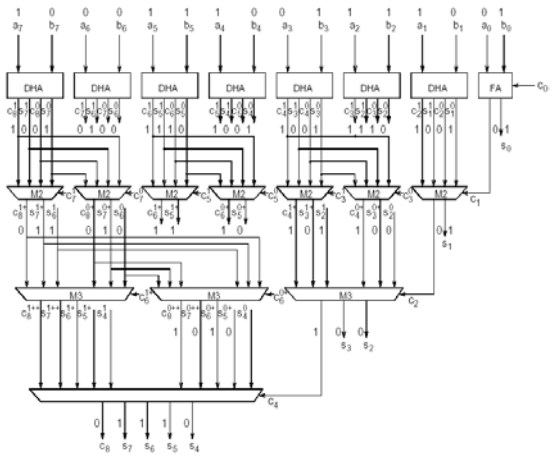

Fig. 4. Conditional Sum Adder (8-bit)

The two sets of the outputs (DHA) shown in Fig.4 are given in the logic equations as follow:

$$C^0_{i+1} = a_i\, b_i\ ,\quad S^0_i = a_i \oplus b_i\ ;$$
$$C^1_{i+1} = a_i + b_i\ ,\quad S^1_i = \acute{S}^0_i \quad (10)$$

Complexity and delay functions for n-bit COSA:

$$A_{COSA} = O(\log n) = 3n\log_2 n$$
$$T_{COSA} = O(\log n) = 2 \log_2 n \quad (11)$$

3. THE VHDL SIMULATIONS

In this section, the VHDL simulations of the adder architectures are done. In order to verify the functional correctness of the adders into the consideration, the design description files are created and then compiled by a VHDL simulator program[9]. For 8-bit operand lenghts, the needed signals, especially the outputs waveforms and the delays of the binary architectures under consideration are shown as in Fig.5a-d. Using the VHDL simulations, the delay times are measured for 8-bit additions as follows:

RCA: 20 ns, CSLA: 17.6 ns, CLSA:12.8 ns., COSA: 10.2 ns.

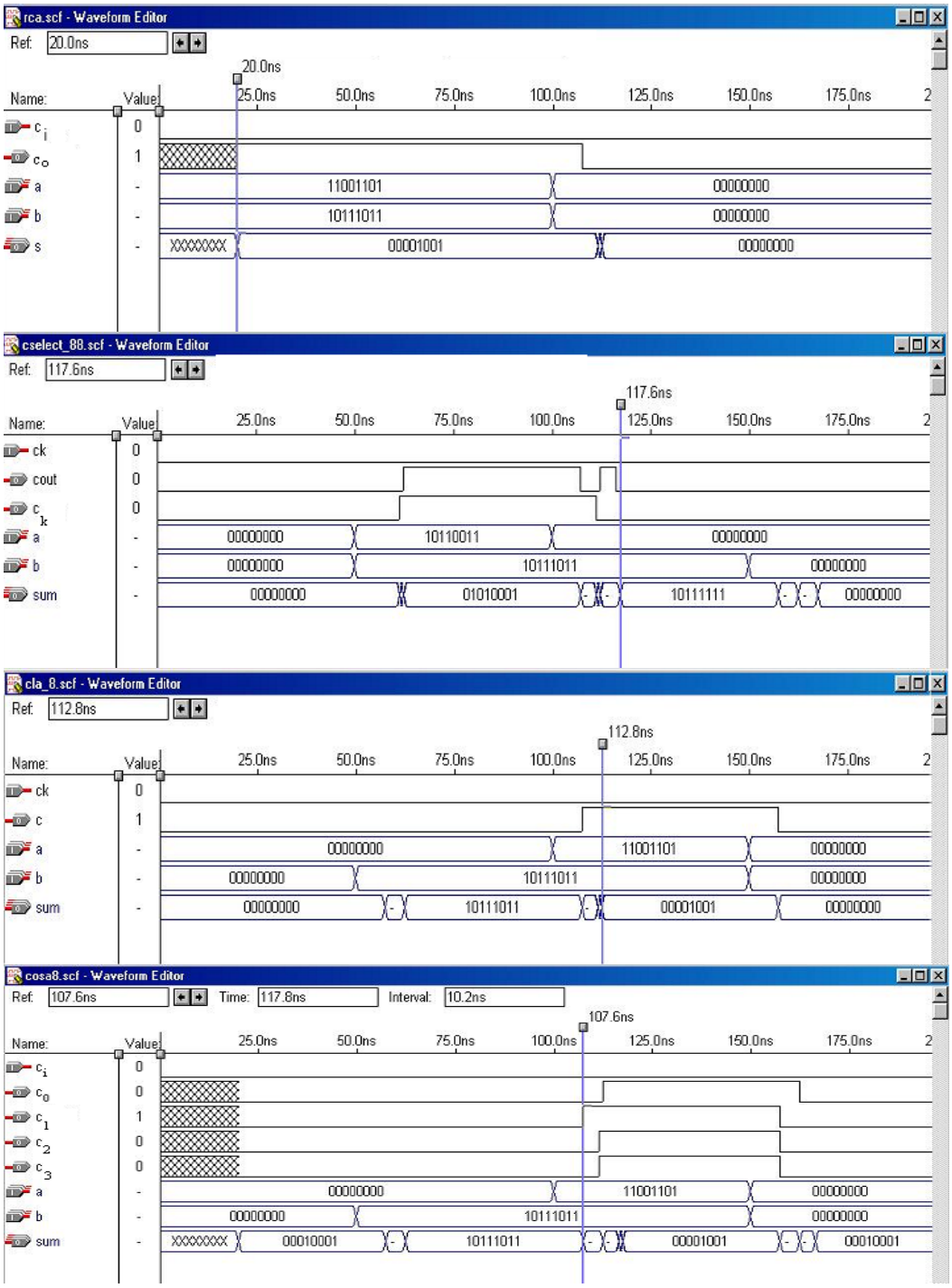

Fig. 5. By the VHDL simulator, the functional verifying and delay measurements of the binary adder architectures for : (a) RCA , (b) CSLA, (c) CLA, (d) COSA.

4. PERFORMANCE COMPARISONS

In this section, the results obtained from comparing the adder architectures are presented. Comparisons include the complexity, the number of gates used in the structures and the worst case delay, the possible longest path from inputs (a_i and b_i, or c_i) to outputs (sum, c_o) in terms of the unit gate. In this work, the performance measure is defined as the product of the complexity and the delay.

All main adders are compared for the multiples of 8-bit lengths. First comparisons are performed under a simple unit-gate model. In this model, each gate with two inputs has a gate-count and a gate-delay of one, except for the XOR/XNOR gates having gate-counts and gate-delays of two. For the gates with more inputs, the gate-counts and gate-delays can be computed in terms of the ones given for the gates with two inputs. Also, inverters and buffers are ignored. Table 1 lists the gate-count, gate-delay and the products for the studied adder architectures as a function of word lenghts. For the simplicity, only the asymptotic orders are given by the highest exponent of the bit length n and the constant factor, because of the exact formulas are very complex.

In Fig. 6, the plot diagrams show the comparisons of the adders with respect to the area complexity, A (Fig.6a), the delay T, (Fig.6b) and the product as the chosen performance measure, AT, (Fig.6c).

5. CONCLUSIONS

AxT minimization is, of course, not the only optimization criteria for adder circuits. However, AxT measures help finding the most efficient solution from a set of possible candidates. In addition, the result from the unit-gate model comparisons allow the observation of the general behaviours of the adder classes.

The results of this work can be summarized as follows:

- Regarding the circuit area complexity in the adder architectures, the ripple-carry adder (RCA)in the first class is the most efficient one, but the conditional sum adder (COSA) in the fourth class with nlogn complexity is the least efficient one.

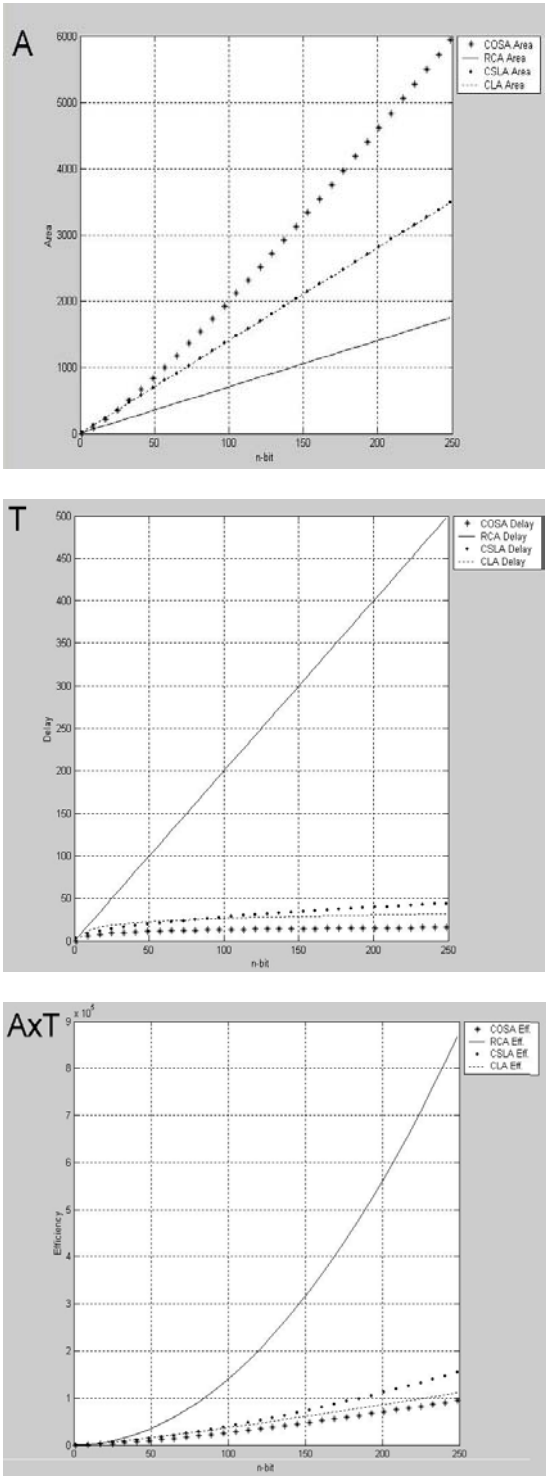

Fig. 6. (a) Area-related, (b) Delay-related, (c) AreaxDelay Product-related, comparison diagrams of the binary adders.

- Considering the circuit delay time, COSA is the fastest one for every n-bit lenght, so has the shortest delay. Otherwise, RCA is the slowest one, due to the long carry propagation.
- The other adders, the carry lookahead (CLA) and the carry select (CSLA) adders, behave similiar and, have medium area and delay requirements with respect to RCA and COSA.
- In the studied adder structures, COSA and CLA architectures have the highest performance, result in the lowest area-delay product (AxT) values, as shown in Fig. 6c.

Finally, the desirable continuation of this work is to investigate the other binary adder architectures and to extend to the performance comparisons for the all adder structures.

REFERENCES

[1] O.J. Bedrij: 'Carry Select Adders', IRE Trans. EC-11 No. 3, 1962.

[2] R.W. Doran: 'Variants of an Improved Carry Look-Ahead Adder', IEEE Trans. On Computers, 37, pp. 1110-1113, 1988.

[3] J.Sklansky.: 'Conditional Sum Addition Logic', IRE Trans. EC-9 No. 2 , 1960.

[4] R.P. Brent, and H.T. Kung: 'A Regular Layout for Parellel Adders', IEEE Trans. Comput. C-31, 1982.

[5] T.Lynch and E.jr.Swartzlander : 'A Spanning Tree Carry Lookahead Adder', IEEE Trans. Comput., vol. 41. No. 8 , 1992.

[6] H. Ling: 'High-Speed Binary Adder', IBM J. Res.Develop., vol. 25. , 1981.

[7] R. Zimmerman: 'Binary Adder Architectures for Cell-Based VLSI and their Synthesis Acknowledgments', Phd. Thesis, Swiss Federal Inst. Of Tech., Zurich, 1997.

[8] B.Parhami:'Computer Arithmetic Algorithms and Hardware Designs, Oxford University Press, 2000.

[9] Altera Corperation, MAX Plus II Data Sheets, http://www.altera.com.

Ahmet Sertbaş was born in İstanbul in 1965. He received the B.S. and M.Sc. degrees in electronic engineering from the Istanbul Technical University in 1990, the Ph.D. degree in electronic department from Istanbul University in 1997 respectively. He has worked as Research Assistant at I.T.U. during 1987-1990, research engineer at Grundıg firm during 1990-1992, an instructor at the Vocational School of Istanbul University during 1993-1999. He is currently an Assoc. Professor in the Department of Computer Engineering at the University of Istanbul. His research interests include computer arithmetic circuit design, computer architecture and computer-aided circuit design, circuit theory and applications.

R.Selami Özbey was born in Isparta. He is a master student in the Department of Computer Engineering at Istanbul University. His current research interests are computer arithmetic circuits and cyrpto algorithms.