The Fastest Carry Lookahead Adder

Yu-Ting Pai and Yu-Kumg Chen
Department of Electronic Engineering
Huafan University
fly0810@msn.com and ykchen@huafan.hfu.edu.tw

Abstract

Adder is a very basic component in a central processing unit. The speed of compute becomes the most considerable condition for a designer. The carry lookahead adder is the highest speed adder nowadays. In this paper, a new method for modifying the carry lookahead adder is proposed. Based on the analysis of gate delay and simulation, the proposed modified carry lookahead adder is faster than the carry lookahead adder.
Key words: adder, carry lookahead adder, integrated circuit, central processing unit, gate delay.

1. Introduction

Adder is widely used in the generic computer [1] because it is very important for adding data in the processor. The simplest binary adder is ripple carry adder [2]. It is easy to be understood and implemented. A more complex binary adder is carry lookahead adder (abbreviated as CLA) [3, 4]. It uses the same carry lookahead circuits to construct the higher-bit CLA recursively. It is widely used due to its superior performance over ripple carry adder.

The speed of execution is the most important factor that needs to be considered for appraising the quality of an adder. Traditional CLA is constructed by XOR, AND, and OR gates. The proposed circuit uses NAND gates to replace the AND and NOT gates in CLA, it can decrease the cost of CLA and increase the speed of CLA.

2. SpCLA and MCLA

The proposed modified carry lookahead adder (abbreviated as MCLA) is similar to CLA in basic construction. Hence, it also contains arithmetic adder circuit and carry lookahead circuit. The designed construction of carry lookahead circuit in MCLA is similar to CLA. With using the basic model of MCLA, it can use the carry lookahead circuit recursively to implement the higher-bit MCLA. In order to be analyzed in mathematics, a K^m-bit CLA model is defined in this paper. Where K is the number of bit consisted in each level of CLA and m is the level of carry lookahead circuit used in CLA.

Let K=4 and m=1, the proposed 4-bit simplified carry lookahead adder (abbreviated as SpCLA) is shown in Figure 1. It contains two parts. The one part is arithmetic adder circuit and the other one is carry lookahead circuit. In order to be used as the first level of arithmetic adder circuit in the proposed SpCLA, the part of this new full adder called metamorphosis of partial full adder (abbreviated as MPFA), as shown in Figure 2, is used in SpCLA.

In carry lookahead circuit of 4-bit SpCLA, all of the components are implemented with NAND gates except for the outputs of P and $\overline{G}$, which are implemented with AND gates. Since the output signal of MPFA is $\overline{G_i}$ implemented with NAND gates, it is faster than the G_i of PFA implemented with AND gate.

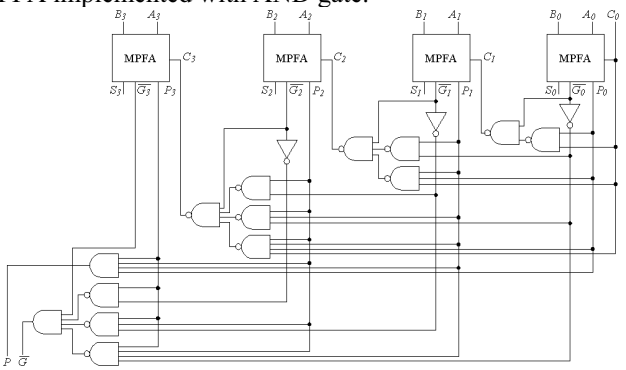

Figure 1. 4-bit Simplified carry lookahead adder.

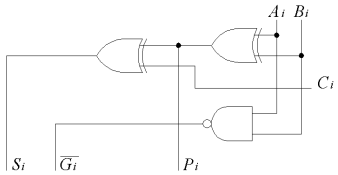

Figure 2. Metamorphosis of partial full adder.

Let i be the index of stage. There are three inputs A_i, B_i, and C_i and three outputs G_i, Pi, and S_i in MPFA. The variables A_i, B_i, C_i, and S_i are bits of augends, addend, carry, and sum at stage i, respectively. The carry of the next stage can be expressed as

$$C_{i+1} = \overline{\overline{G_i}\ \overline{P_i C_i}}. \quad (1)$$

From equation (1), the carry outputs of each stage can be listed in the following:

Proceedings of the Second IEEE International Workshop on Electronic Design, Test and Applications (DELTA'04)
0-7695-2081-2/04 $ 20.00 © 2004 IEEE

$C_0 = inupt \quad carry$,

$C_1 = \overline{\overline{G_0} \;\; \overline{P_0 C_0}}$,

$C_2 = \overline{\overline{G_1} \;\; \overline{P_1 G_0} \;\; \overline{P_1 P_0 C_0}}$,

$C_3 = \overline{\overline{G_2} \;\; \overline{P_2 G_1} \;\; \overline{P_2 P_1 G_0} \;\; \overline{P_2 P_1 P_0 C_0}}$,

$C_4 = \overline{\overline{G_3} \;\; \overline{P_3 G_2} \;\; \overline{P_3 P_2 G_1} \;\; \overline{P_3 P_2 P_1 G_0} \;\; \overline{P_3 P_2 P_1 P_0 C_0}}$.

Since the time delay grows with the number of fan-in, it is needed to consider the time delay of the number of fan-in signals [5]. If K =4. The functions of the inverse group generate and the group propagate for 4^m-bit SpCLA can be expressed as

$\overline{G} = \overline{G_3} \;\; \overline{P_3 G_2} \;\; \overline{P_3 P_2 G_1} \;\; \overline{P_3 P_2 P_1 G_0}$,

$P = P_3 P_2 P_1 P_0$.

Therefore, the C_4 of second level can be produced from G, P and C_o of the first level, which is

$C_4 = \overline{\overline{G} \;\; \overline{P C_0}}$.

By using the same method, the 16-bit SpCLA (4^2-bit SpCLA) can be implemented easily with a proposed carry lookahead adder circuit and four 4-bit SpCLAs, as shown in Figure 3. Hence the higher-bit SpCLA can be realized by using the same carry lookahead circuit as well.

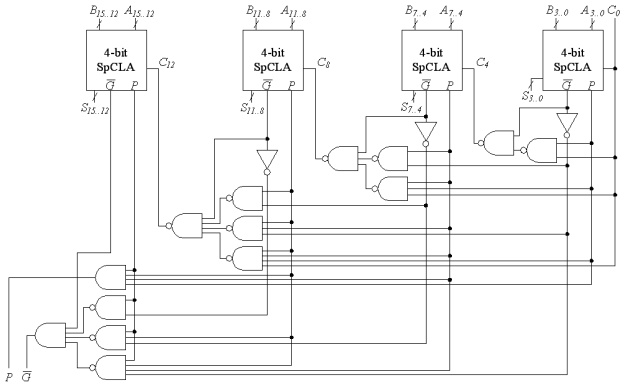

Figure 3. 16-bits SpCLA.

Although the proposed SpCLA can be implemented via the same proposed carry lookahead circuit shown in Figures 1, it is not the simplest circuit. The method to simplify this circuit is using a NAND gate and a NOT gate to replace the AND gate of the output bit G in the *4m-1*-bit (previous level) SpCLA circuit when m is greater than 1, and then cancel its NOT gate with the NOT gate in the *4m*-bit (present level) SpCLA. The new circuit is named as MCLA. Let K= 4. For example, there are three NOT gates in the carry lookahead circuit of 16-bit SpCLA. If we move back the three NOT gates into 4-bit SpCLA and simplify it with another four NOT gates in the proposed carry lookahead circuit of 16-bit SpCLA, we can derive the simplest circuits of 4-bit and 16-bit MCLA, as shown in Figures 4 and 5, respectively.

The carry lookahead circuit for the second and higher levels of MCLA is different from that of the first level of MCLA. Since their position of NOT gate is not direct after the signal G, it can reduce one gate delay time of SpCLA.

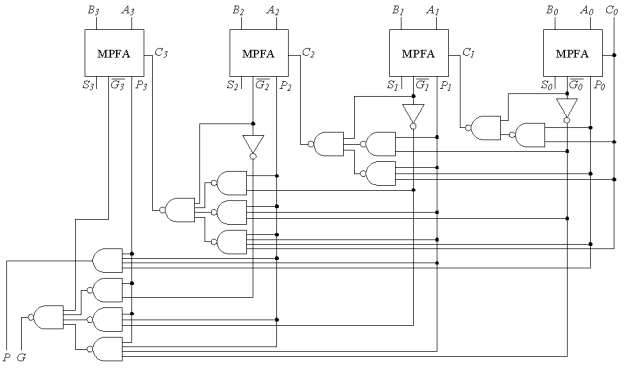

Figure 4. 4-bit MCLA.

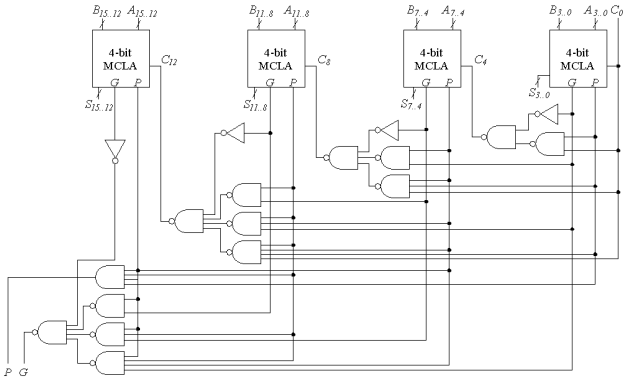

Figure 5. 16-bit MCLA.

3. Logic Implementation

In this section, the simulations of CLA and MCLA are given in different experiments. The results show that the proposed MCLA is superior to the CLA. The software used in this section is OrCAD Capture V9.0. We use the logic gates 7400, 7404 and 7408, etc. to construct the circuits of CLA and MCLA. By using the functions of SPICE, the simulations are proceeding. The hardware used for simulating is personal computer with Pentium III-MMX 450 processor, 256 MB RAM.

All bits of addend are set to 1 and all bits of augends are set to 0 except for the lowest bit with using difference frequency square periodic signal for inputs, the simulation of the time delay of CLA are shown in Figure 6, which is done in the tool of SPICE. Figure 7 shows the time delays for the sums of each bit in CLA, SpCLA, and MCLA.

Let N be the bits added in a binary adder. The time complexities of gate delay for CLA, SpCLA, and MCLA are the same, that is, $O(\log N)$. If α is the time delay of the arithmetic adder circuit (PFA or MPFA) and β is the time delay of the carry lookahead circuit for each level,

then the time needed $T(N)$ for getting the result can be precisely described by the recurrence relation

$$T(N) = T(\frac{N}{K}) + \beta \quad \text{for } N \geq 2 \text{ with } T(1) = \alpha .$$

To get an indication for the nature of the solution to this recurrence, we consider the case when N is a power of K, that is, N is equal to K^m. Iterating this recurrence gives

$$T(N) = T(1) + m\beta = \alpha + m\beta .$$

This proves that $T(N) = \alpha + \beta \log_K N$ when $N = K^m$.

Hence, the time complexity is

$$T(N) \in O(\log N) .$$

Let Y be the time delay. According to the mathematical analysis of time delay, the approximation function of time delay expressed as

$$\hat{Y} = a \log_K N + b .$$

The sum of square errors (abbreviate as SSE) [6] is

$$SSE = \sum_{i=1}^{N} (Y_i - \hat{Y}_i)^2 = \sum_{i=1}^{N} (Y_i - a \log_K i - b)^2 .$$

The approximation function, which we want to find, must have minimum value of SSE. To get a such function, we can set the partial derivatives with respect to a and b to be equal to zero, that is,

$$\frac{\partial SSE}{\partial a} = -2 \sum_{i=1}^{N} \log_K i (Y_i - a \log_K i - b) = 0 ,$$

$$\frac{\partial SSE}{\partial b} = -2 \sum_{i=1}^{N} \log_K i (Y_i - a \log_K i - b) = 0 .$$

The two equations can be further reduced as

$$a \sum_{i=1}^{N} (\log_K i)^2 + b \sum_{i=1}^{N} \log_K i = \sum_{i=1}^{N} \log_K i Y_i ,$$

$$a \sum_{i=1}^{N} (\log_K i)^2 + bN = \sum_{i=1}^{N} Y_i .$$

Now, we have two linear equations with two unknown parameters. By using the Gaussian elimination [7], we can derive the values of a and b. The curves in Figure 7 show the approximation curves for CLA, SpCLA and MCLA. We find that the proposed MCLA has the lowest value of a. Hence, the proposed MCLA is the fastest circuit than that of CLA.

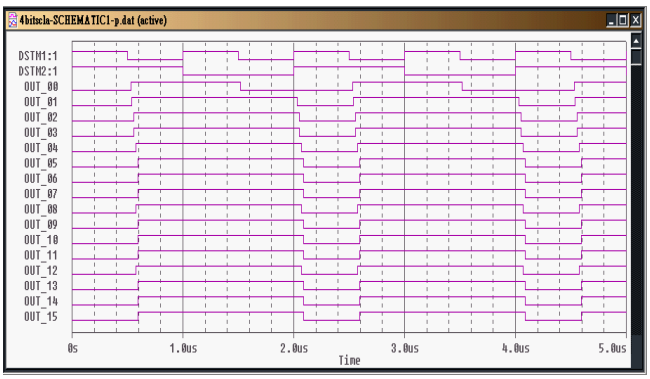

Figure 6. The simulated of time delay of CLA in the tool of SPICE.

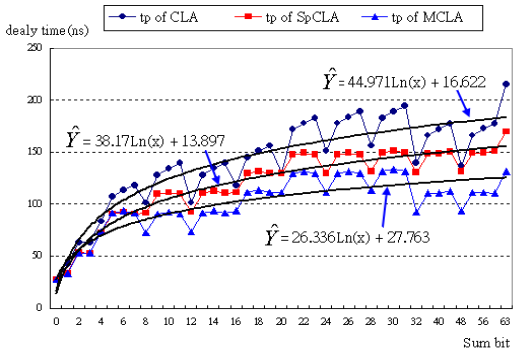

Figure 7. Diagrams for delay time of sum bit.

4. Conclusions

The adder is the basic element in CPU. All of the adder-subtractor, and multiplier, etc. are constructed with adders. Therefore, to speed up the adder efficiently is very important to CPU or processor. This paper proposed a method that uses NAND gate to simplify the carry lookahead circuit and arithmetic adder circuit of CLA. In order to make it faster, a method to modify these circuits is also proposed. Since the circuit of CLA and MCLA are kinds of recursive circuits, this recursive circuit may be useful for speeding up the other digital logic circuits. For example, the carry lookahead adder-subtractor, can be reformed from MCLA or replaced the full adder in multiplier with MCLA, it will have an excellent improvement in efficiency and cost. We also propose a recurrent relation model for analyzing the time complexity of gate delay. It will be useful for another logical circuits.

5. References

[1] F. C. Cheng, S. H. Unger, M. Theobald, and W. C. Cho, "Delay-Insensitive Carry-lookahead Adders", *VLSI Design Proceedings*, 1997, pp. 322-328.

[2] C. Nagendra, M. J. Irwin, and R. M. Owens, "Area-time-power tradeoffs in parallel adders", *IEEE Transactions on Circuits and Systems II*, 1996, vol. 43, pp. 689-702.

[3] Mano, M. M. and C. R. Kime, *Logic and computer design fundamentals*, Prentice-Hall, 2001.

[4] J. Lim, D. G. Kim, and S. I. Chae, "A 16-bit carry-lookahead adder using reversible energy recovery logic for ultra-low-energy systems", *IEEE Journal of Solid-State Circuits*, 1999, vol. 34, pp. 898-903.

[5] Weste, N. H. E. and K. Eshraghian, *Principles of CMOS VLSI Design: A Systems Perspective 2/E*, Addison-Wesley, 1998.

[6] N. R. Sharpe and R. A. Poberts, "The Relationship Among Sums of Squares", *The American Statistician*, February 1997, Vol. 51 No.1, pp. 46-48.

[7] K. Yerion, "Gaussian elimination and dynamical systems", *College Mathematic Journal*, 1997, vol. 28, pp. 89-93.