

57 Delay Optimisation in High-Performance Carry-Select Adders

¹A.N. Jayanthi and ²Dr. C.S. Ravichandran

¹Research Scholar, Department of ECE, Sri Ramakrishna Institute of Technology, Coimbatore, India
²Dean/EEE, Sri Ramakrishna Engineering College, Coimbatore, India

Abstract: This paper analyses techniques to measure the delay of 64 bit and 128-bit carry select adders. This is used for high-performance and low-power applications. It is introduced to work at a lower time delay than that required by a Ripple Carry Adder. This paper uses a very simple and efficient gate-level modification technique to significantly reduce the delay of the CSA. The proposed design has reduced delay and is compared with the 64 bit CSA. Xilinx ISE is used for simulation and synthesis.

Key words: RCA, CSA, delay, synthesis

INTRODUCTION

Incurrent VLSI technology, adders have become important. The data processed by many digital systems may have delays. Design of area-efficient and high-speed data path logic systems are one of the most important areas of research in VLSI system design. In digital adders, the speed of addition is limited by the time required to propagate a carry through the adder. While Ripple Carry Adders (RCAs) have the most compact design ($O(n)$ area) among all types of adders, they are the slowest types of adders ($O(n)$ time). On the other hand, Carry Look-ahead Adders (CLAs) are the fastest adders ($O(\log(n))$ time), but they are the worst from the area point of view ($O(n \log(n))$ area). Carry Select Adders (CSAs) have been considered as a compromise solution between RCAs and CLAs ($O(n)$ time and $O(2n)$ area) because they offer a good trade-off between the compact area of RCAs and the short delay of CLAs. As a result, some effort has been done to improve the efficiency of this CSA adder.

Design of the Carry Select Adder:

This work concentrates on static design styles since the performance advantage of both dynamic logic styles and pass-gate design is likely to decrease in deep-submicron technologies (European Semiconductor Industry Association, 2001 and Allam *et al.*, 2000). The features of lesser dynamic power consumption and higher noise margin make static CMOS particularly attractive and useful (Yung *et al.*, 2002 and De and Borkar, 2000). Moreover, the activation of the parasitic bipolar transistor is found to result in fatal erroneous states in dynamic logic and to make circuit design with pass-gates more tedious (Lu *et al.*, 1997). The renewed interest in static design styles like pseudo-NMOS (Subba, 2000) and ratioed CMOS (Tretz, 2000) shows that alternative design styles are investigated in SOI in order to reduce the power dissipation while still maintaining high-speed performance. In the branch-based logic (BBL) design style, a logic cell is made of branches that contain some transistors in series (Masgonty, 1991). The branches are connected in parallel between the power supply V_{DD} lines and the common output node. Many usual static CMOS gates have already a branch structure, as inverter and NAND / NOR gates. In the existing method, the 64-bit adder was split into four 16-bit sections (Amaury Nève *et al.*, 2004). The true sum can be chosen by a multiplexer. The control signals for the multiplexers are the carry-in and the intermediate carry signals C_{15}, C_{31} and C_{47} . Fig 1 shows the intermediate carry signals and the carry-out are generated by C_{63} the carry-select boxes (CS-boxes). The inputs of the CS-boxes are the carry-in and the conditional carry signals and (with $C_{0-15}, C_{16-31}, C_{32-47}$, or C_{48-63}), which are all generated parallel (Hwang, 1979).

The notation C_{0-15} refers to the block carry signal for bit positions 0 to 15, assuming that the carry-in is at "0." For the sake of preciseness, the indexes of the conditional carry signals have been simplified in Fig. 1. The carry-out and the intermediate carry signals are computed according to the equations presented in Table 1. To compute the carry-out, the intermediate carry is combined with and in one CS-C0 stage to avoid a complex CS-C3 stage. At their turn, the 16-bit adder blocks are implemented as carry-select adders, with 4-bit adder blocks having a carry-in either at "0" or at "1." At the 16-bit level, the same CS-boxes can be used as at the 64-bit level, sizing of the transistors being adapted to the particular load conditions. The 4-bit adder blocks can then finally be implemented as ripple carry adders, or also as carry-select adders, which was chosen here. The carry-select architecture is thus used at three different levels: in the 64-bit, the 16-bit, and the 4-bit adders (Lee *et al.*, 2001, Kim *et al.*, 2002, Bai *et al.*, 2002, Sastiak *et al.*, 2000, Garg and Katoch., 2001 and Ning, 2000).

Corresponding Author: A.N. Jayanthi, Research Scholar, Department of ECE, Sri Ramakrishna Institute of Technology, Coimbatore, India
E-mail: jayanthi_an@rediffmail.com

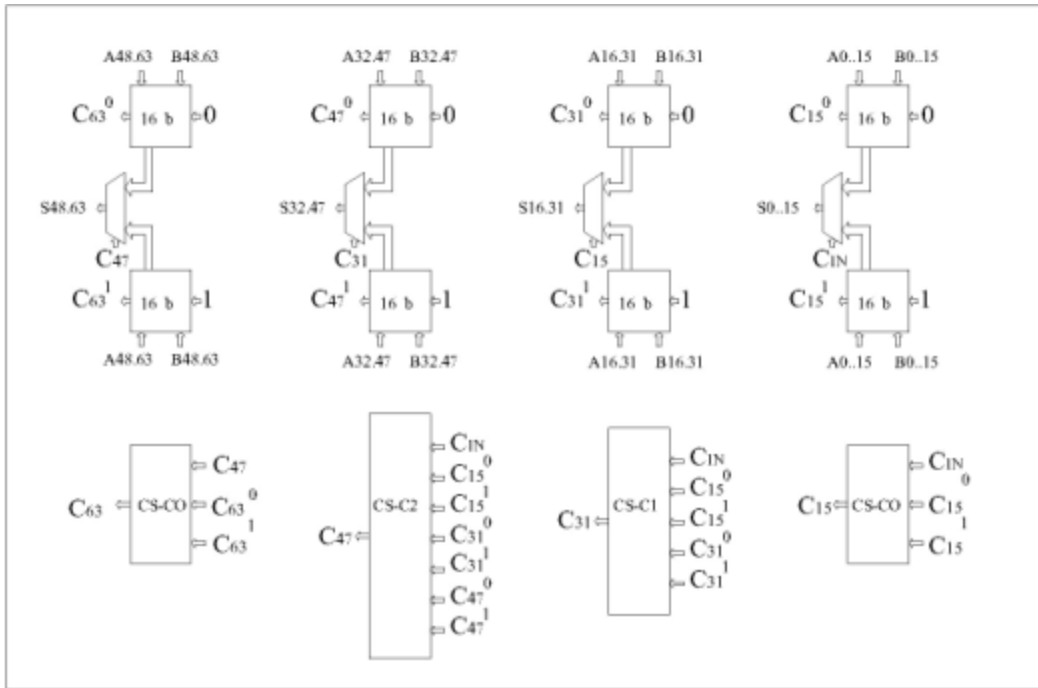


Fig. 1: Block Diagram Of 64-Bit CSA.

Table 1: Logic Equations And Circuit Blocks.

Logic Equation	Circuit Block
$C_{15} = C_{in} C_{0-15}^1 + C_{0-15}^0$	CS-C0
$C_{31} = C_{in} C_{0-15}^1 C_{16-31}^1 + C_{0-15}^0 C_{16-31}^1 + C_{16-31}^0$	CS-C1
$C_{47} = C_{in} C_{0-15}^1 C_{16-31}^1 C_{32-47}^1 + C_{0-15}^0 C_{16-31}^1 C_{32-47}^1 + C_{16-31}^0 C_{32-47}^1 + C_{32-47}^0$	CS-C2
$C_{out} = C_{63} = C_{in} C_{0-15}^1 C_{16-31}^1 C_{32-47}^1 C_{48-63}^1 + C_{0-15}^0 C_{16-31}^1 C_{32-47}^1 C_{48-63}^1 + C_{16-31}^0 C_{32-47}^1 C_{48-63}^1 + C_{32-47}^0$ $C_{48-63}^1 + C_{48-63}^0 = C_{47} C_{48-63}^1 + C_{48-63}^0$	CS-C0

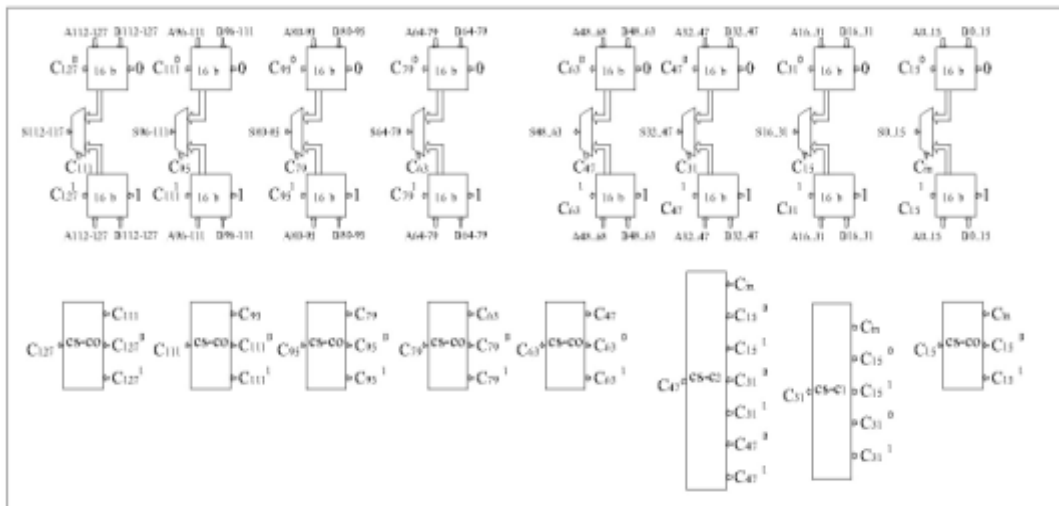


Fig. 2: Block Diagram of 128-Bit CSA.

Proposed Method:

This method analyses methods to measure the delay of 128-bit carry-select adders intended for high-performance and low-power applications. In this paper, a methodology to minimize the delay of 64 bit carry-

select adders for high performance and low-power applications, by working at three levels of abstraction: design style, cell arrangement, and adder structure is dealt(Zaker and Zahnd,1993, Katopis *et al.*,1999 and Shahidi,2002).

The approach presented in this paper can be extended towards 128 bit carry-select adders. However, the number of carry-select levels and the adder architecture may be different in order to obtain an efficient realization(Aller and Kroell,1999, Nève *et al.*,2002 , Yano *et al.*,1990 and Martin,2000).Fig 2 shows the block diagram of 128 bit CSA with 16 different sections of adders. The multiplexers shown in the Fig 2 can be used for the selection of the carry signals. Fig 3 and Fig 4 shows Simulation result and Synthesis Report of 64- Bit CSA. Fig 5 and Fig 6 shows Simulation result and Synthesis Report of 128- Bit CSA.

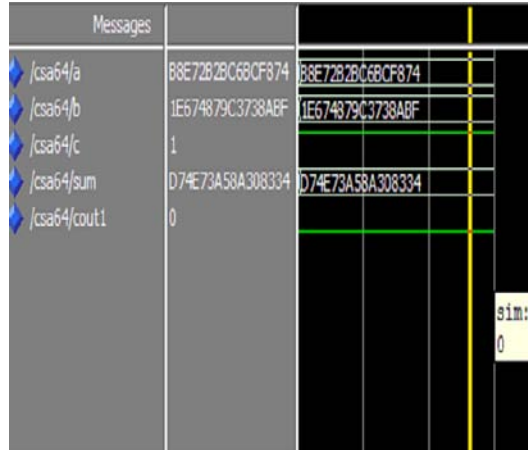


Fig. 3: Simulation Result of 64- Bit CSA.

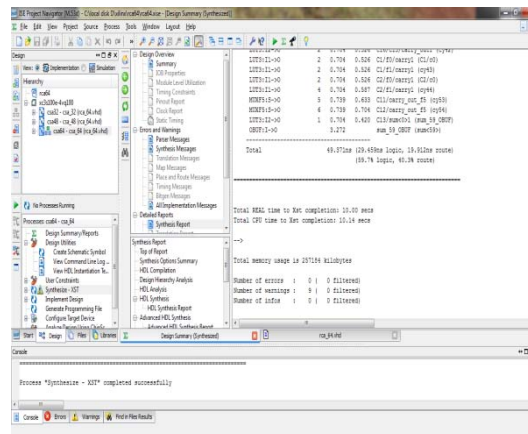


Fig. 4: Synthesis Report of 64- Bit CSA.

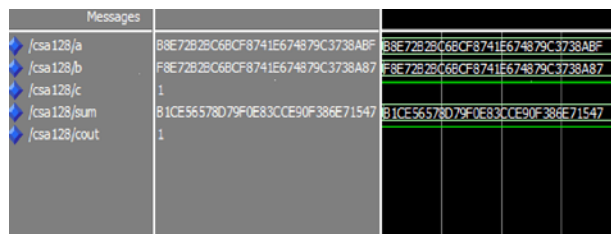


Fig. 5: Simulation result of 128 -Bit CSA.

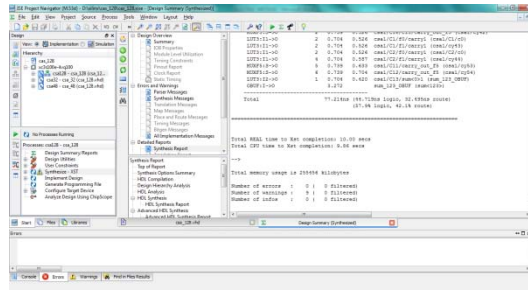


Fig. 6: Synthesis Report of 128- Bit CSA.

Results for CSA:

The delay for 128-bit CSA using Xilinx software is measured as 77.214ns. It is compared with 16,32,64 bit CSA's. Table I shows Comparison Results of various CSA's.

Table I: Comparison Results of various CSA's.

NO. OF BITS in CSA	DELAY MEASURED
16 BIT CSA	16.706ns
32 BIT CSA	23.058ns
64 BIT CSA	49.371ns
128 BIT CSA	77.214ns

Conclusion:

A carry-select adder speeds 40% to 90% faster than RCA (whose delay is more) by performing additions in parallel and reducing the maximum carry path. Thus from the above research work it is concluded that carry select adder is one of the fastest adders that can be used in many addition operations. A simple approach can be proposed to reduce the area of SQRT CSLA architecture. The reduced number of gates of this work offers the great advantage in the reduction of area. The compared results show that the 128 bit CSA has a larger delay than 64 bit CSA. This work can be further extended to n-bit Carry Select Adder. The modified CSLA architecture is therefore, low area, simple and efficient for VLSI hardware implementations of LTI (Linear Time Invariant) system.

REFERENCES

Allam, M., M. Anis and M. Elmasry, 2000. Effect of technology scaling on digital CMOS logic styles, Proc IEEE Custom Integrated Circuits Conf (CICC), 19-1-1-19-1-8.

Aller, I. and K.E. Kroell, 1999. Detailed analysis of the gate delay variability in partially depleted SOI CMOS circuits. Proc IEEE Int SOI Conf., 40-41.

Amaury Nève, E., Helmut Schettler, T. Ludwig and D. Flandre, 2004. Power-Delay Product Minimization in High-Performance 64-bit Carry-Select Adders. IEEE Trans On Very Large Scale Integration (VLSI) systems, 12(3).

Bai, X., C. Visweswariah, P. Strenski and D. Hathaway, 2002. Uncertainty-aware circuit optimization. Proc 39th Design Automation Conf., 58-63.

De, V. and S. Borkar, 2000. Low power and high performance design challenges in future technologies. Proc Great Lakes Symp VLSI, 1-6.

European Semiconductor Industry Association, Japan Electronics and Information Industries Association, Korea Semiconductor Industry Association, Taiwan Semiconductor Industry Association. International Technology Roadmap for Semiconductors and System Drivers, 2001.

Garg, M. and A. Katoch, 2001. Evaluation of skew tolerance in delayed clocking scheme for dynamic circuits. Proc ESSCIRC, 396-399.

Hennessy, J., D. Patterson and D. Goldberg, 1996. Computer Architecture, A Quantitative Approach, 2nd ed. San Mateo, CA Morgan Kaufman.

Hwang, K., 1979. Computer Arithmetic: Principles, Architecture and Design. New York: Wiley.

Katopis, G.A., W.D. Becker, T.R. Mazzawy, H.H. Smith, C.K. Vakirtzis, S.A. Kuppinger, B. Singh, P.C. Lin, J. Bartells Jr., G.V. Kihlmire, P.N. Venkatachalam, H.I. Stoller and J.L. Frankel, 1999. MCM technology and design for the S/390 G5 system. IBM J Res Develop., 43(5/6):621-650.

Kim, J.J., R. Joshi, C.T. Chuang and K. Roy, 2002. SOI-optimized 64-bit high-speed CMOS adder design. Sym. VLSI Circuits Dig Tech Papers, 122-125.

- Lee, S.J., R. Woo and H.J. Yoo, 2001. 480 ps 64-bit race logic adder. Symp. VLSI Circuits Dig Tech Papers, 27-28.
- Lu, P.F., C.T. Chuang, J. Ji, L.F. Wagner, C.M. Hsieh, J.B. Kuang, L.L.C. Hsu, M.M. Pelella, S.F. Sanford and C.J. Anderson, 1997. Floating-body effects in partially depleted SOI CMOS circuits. *IEEE J Solid-State Circuits*, 32: 1241-1253.
- Martin, K., 2000. *Digital Integrated Circuit Design*. Oxford, U.K.: Oxford Univ. Press.
- Masgonty, J.M., C. Arm and C. Piguet, 1991. Technology and power supply-independent cell library. *Proc IEEE Custom Integrated Circuits Conf (CICC)*, 25.5/1–25.5/4.
- Naffziger, S., 1996. A subnanosecond 0.5 μ m 64b adder. *Proc. IEEE Int Solid-State Circuits Conf Slide Supplement*.
- Nagendra, C., R.M. Owens and M.J. Irwin, 1994. Power-delay characteristics of CMOS adders. *IEEE Trans on VLSI Syst.*, 2: 377-381.
- Nève, A., D. Flandre, H. Schettler, T. Ludwig and G. Hellner, 2002. Design of a branch-based 64-bit carry-select adder in 0.18 μ m partially-depleted SOI CMOS. *Proc Int. Symp Low Power Electronics and Design (ISLPED)*, 108-111.
- Ning, T.H., 2000. CMOS in the new millennium. *Proc IEEE Custom Integrated Circuit Conf (CICC)*, 49-56.
- Sastiak, D., J. Tran, F. Mounes-Toussi and S. Storino, 2000. A 2nd generation 440 ps SOI 64 b adder. *IEEE Int. Solid-State Circuits Conf.*, 288-289.
- Shahidi, G.G., 2002. SOI technology for the GHz era. *IBM J Res Develop*, 46(2/3): 121-131.
- Subba, N., A. Salman, S. Mitra, D.E. Ioannou and C. Tretz, 2000. Pseudo-NMOS revisited: Impact of SOI on low power, high speed circuit design. *Proc IEEE Int SOI Conf.*, 26-27.
- Tretz, C.R., R.K. Montoye and W. Reohr, 2000. Ratioed CMOS -A low power high speed design choice in SOI technologies. *Proc IEEE Int SOI Conf.*, 28-29.
- Yano, K., T. Yamanaka, T. Nishida, M. Saito, K. Shimohigashi and A. Shimizu, 1990. A 3.8 ns CMOS 16-16-b multiplier using complementary pass-transistor logic. *IEEE J Solid State Circuits*, 25: 388-395.
- Yung, R., S. Rusu and K. Shoemaker, 2002. Future trend of microprocessor design. *Proc ESSCIRC*, 43-46.
- Zaker, S. and J. Zahnd, 1993. OPTIMOS: a branch-level digital circuit optimizer. *Proc Euro ASIC*, 563-572.