JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 22, 975-989 (2006)

Short Paper

Improved 32-bit Conditional Sum Adder for Low-Power High-Speed Applications

KUO-HSING CHENG AND SHUN-WEN CHENG +
Department of Electrical Engineering
National Central University
Chungli, 320 Taiwan
E-mail: cheng@ee.ncu.edu.tw
+*Department of Electronic Engineering*
Far East College
Tainan, 744 Taiwan
E-mail: swcheng@cc.fec.edu.tw

This paper presents an improved 32-bit conditional sum adder. Due to architectural modification, the improved adder only selects and transmits carry signals; it therefore is named conditional carry adder (CCA). This 32-bit adder focuses on reducing the numbers of internal nodes and logical gates, while maintaining high speed. The 32-bit conditional sum adder uses 186 multiplexers, and the proposed 32-bit CCA only uses 129 multiplexers. Consequently, the proposed adder is attractive for use in low-power arithmetic systems. Conventional single-end static CMOS and differential-end CPL circuits were used to implement and compare the proposed 32-bit CCA. Experimental and control chips were designed and fabricated using 0.5μm CMOS technology. Simulations and measurements under various supply voltages showed that the 32-bit CCA achieved high performance in low-voltage high-speed applications.

Keywords: 32-bit, differential-end logic, pass-transistor logic, conditional sum adder, carry select adder, low-voltage, high-speed, CMOS design

1. INTRODUCTION

High-performance digital signal processing (DSP) systems require both efficient methods for processing data as well as for building architectures that support faster operation, and low power dissipation to enable data to be processed in real-time [1]. Fast binary addition is essential to modern DSP systems. However, in an *n*-bit adder, a propagation of the carry always occurs. The ripple carry addition rule makes possible the simplest circuit structure designs and the longest delay among all adders [2]. To avoid the linear growth problem with the ripple carry delay, the carries in a carry lookahead adder (CLA) can be generated in parallel and fed to all carry generators with a carry in a signal [3]. But the number of stacked MOS transistors increases with the height of the carry bit. So the delay of the highest carry bit will limit the speed performance of the CLA. Hence,

Received June 8, 2004; revised October 5 & December 20, 2004; accepted January 5, 2005.
Communicated by Chung-Yu Wu.

the bit-length of a CLA module is limited to only four bits. This scheme is the most common one in cell-based IC design, but it is area-consuming and involves a regular layout [3, 4]. Although the propagation delay of a CLA has been reduced, it remains proportional to the bit length.

Another approach to increasing the operation speed that expends area in favor of speed is to use a carry-select adder. The carry-select adder computes two versions of the addition with different carry-ins, and then selects the right one. The construction of such pairs is repeated for a 4-bit adder module; thus, the propagation delay of this carry select adder is reduced to that of $n/4$ adder modules. The previous carry selects the appropriate sum using a multiplexer gate. The delay time of each module is only a multiplexer delay, so each module is faster than the CLA unit. The speed of the carry select adder can be improved by adding more adders and multiplexers. However, the carry select adder contains at least a pair of m-bit adders in each m-bit module. Thus, the construction is eased at the cost of increased layout area [5]. Recently, some improved conditional carry designs based on the carry select adder have been presented [6-8], but they remain restricted to carry-select propagation. These designs have complex carry-select networks and irregular structures [9]. The circuit complexities of these designs all are higher than those of the classic conditional-sum adder.

The conditional sum adder (CSA) has been shown to achieve the highest performance than other adders used in high-speed applications [10, 11]. The CSA uses a large number of multiplexer gates to accelerate the operation speed. However, this makes the network tree of the multiplexer larger and more irregular. In high-speed adder designs, the fan out limitation may seriously degrade the estimated addition speed. The CMOS process is mainly used, so this limitation causes the CSA circuits to exhibit a large layout area, a longer delay in the multiplexer tree and greater power dissipation. This problem is exacerbated as the bit-length of the addition rises.

In this paper, a modified adder scheme, the 32-bit conditional carry adder (CCA), is proposed for high-speed applications. This paper on the improved adder presents a modified conditional sum addition rule [10]. It can be used to reduce the numbers of internal nodes and multiplexers in the adder. Therefore, the proposed CCA has a smaller capacitive load, less power dissipation, and a higher operating speed. The static CMOS logic circuit and the CPL circuit [12] were used to implement this novel 32-bit adder structures. The simulated supply voltage was changed from 3.3V to 1.5V. Experimental chips were designed and fabricated using 0.5μm Double Polysilicon Double Metal (DPDM) CMOS technology. It was found that the proposed CCA has a smaller layout area and higher operation speed than the CSA design. The paper is divided into four additional sections. Section 2 describes the structure of the conditional sum adder. Section 3 explains the architecture of the proposed 32-bit CCA. Section 4 presents 32-bit circuit implementations and measurements of both the CSA and the proposed CCA. The final section gives concluding remarks.

2. 32-BIT CONDITIONAL SUM ADDER STRUCTURE

The conditional sum addition rules solve the carry propagation problem [11, 13, 14]. This adder generates all possible carriers and then uses them to simultaneously select the

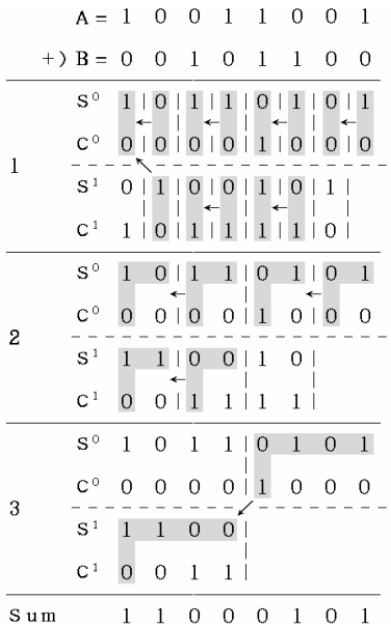

Fig. 1. Conditional sum addition rule.

true sum outputs from two provisional sums. Fig. 1 displays the 8-bit conditional sum addition rule; the arrows mark the actual carries that are generated between sections. Additions are processed simultaneously and independently on all sections. Fig. 2 shows a schematic diagram and the critical carry delay path of the 8-bit conditional sum adder. Sequences of possible sum output selections are performed to generate the true sum output. When $C_{i-1} = 0$, the sum signal can be expressed as $S_i = A_i \oplus B_i \oplus 0 = A_i \oplus B_i$. When $C_{i-1} = 1$, the sum signal can be expressed as $S_i = A_i \oplus B_i \oplus 1 = \overline{A_i \oplus B_i} = A_i \odot B_i$. Two-to-one multiplexers implement the selection procedure completely. The shading multiplexers reveal that most of the multiplexers are used in the sum selection procedure. The equation for calculating the number of MUX gates in a 2^N-bit CSA is as follows:

$$\sum_{n=1}^{N}(2^{N-n}+1)(2^n-1) = \sum_{n=1}^{N}2^N - 2^{N-n} + 2^n - 1. \tag{1}$$

For example, the number of multiplexer gates in 8-bit (2^3-bit) CSAs is 28 (= 2 × 7 + 3 × 3 + 5 × 1), respectively. The critical path of the 8-bit CSA is from A_0/B_0 to C_7, and it occurs when the input is $(A_0 \sim A_7) = (1, 0 \sim 0)$; $(B_0 \sim B_7) = (1 \sim 1)$. This involves gate delays in three stages of multiplexers. The C_3 bit must drive five 2-to-1 multiplexers; thus, the CMOS fan out limit is approached. The multiplexer network tree of the CSA is clearly large and irregular. The speed performance of the CSA is limited by the complex multiplexer network tree.

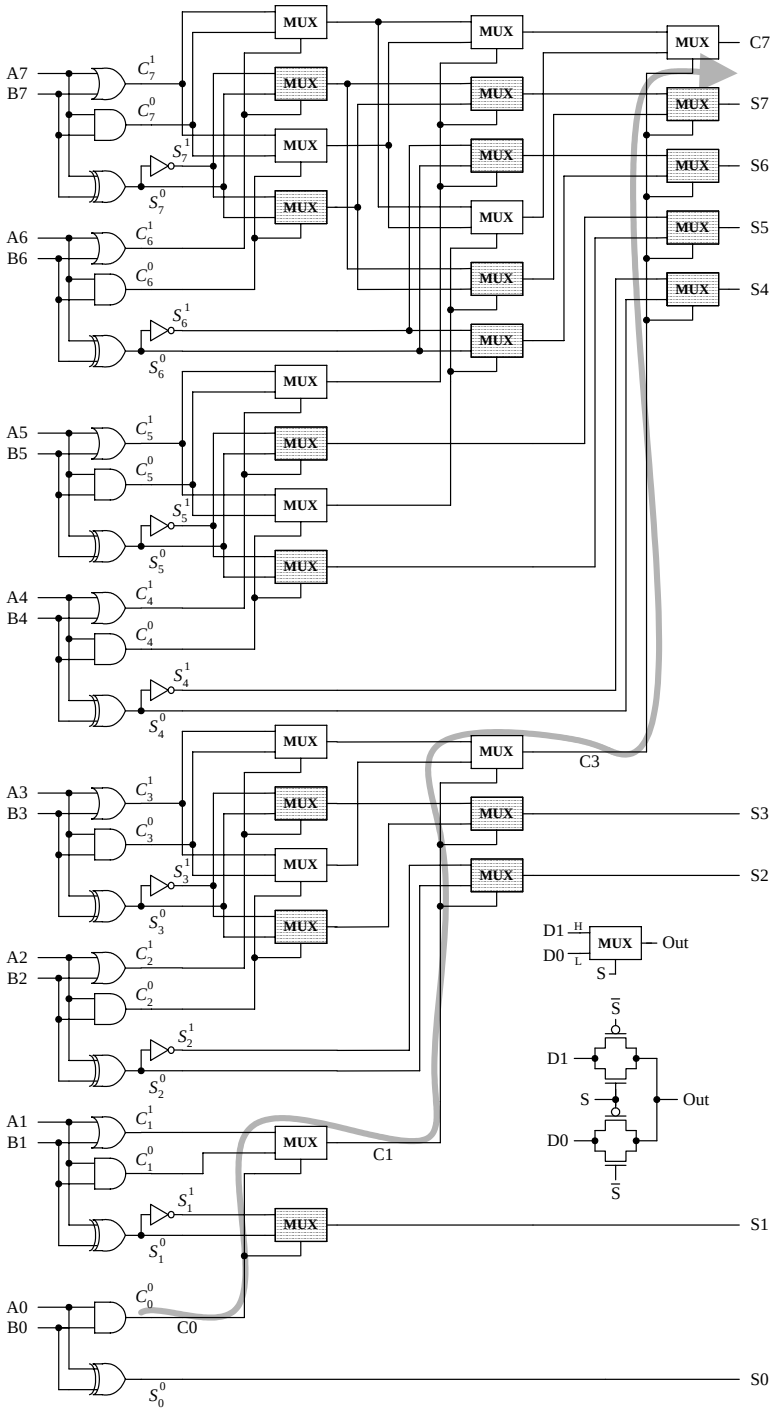

Fig. 2. Circuit and critical delay path of the 8-bit conditional sum adder (CSA).

3. PROPOSED 32-BIT CONDITIONAL CARRY ADDER SCHEME

3.1 Construction of the 32-Bit Conditional Carry Adder (CCA)

For reducing the internal capacitive load of the multiplexer network, the 2:1 multiplexers for sum signal selection are removed, as shown in Fig. 3. The number of remaining multiplexer gates is only 11. The formula for calculating the remaining multiplexer gates of the N-bit CSA is $\sum_{n=1}^{N}(2^n - 1)$, where $N = log_2 n$. Fig. 4 presents a schematic diagram and the critical delay path of the 8-bit conditional carry adder. The structure of the CCA includes three parts: a carry generation module, a conditional carry reduction unit, and XOR gates for the final sum outputs. The carry generation module generates all possible carry output signals.

The conditional carry unit is used for the carry output selections of every bit. After the number of multiplexers is reduced, the circuit shown in Fig. 3 only generates C_0, C_1, C_3, and C_7. Hence, the carry unit must supply some 2:1 multiplexers in order to select the other carry bits, C_2, C_4, C_5, and C_6. This conditional carry unit is still implemented by means of 2:1 multiplexers, as depicted in Fig. 4. When $C_{i-1} = 0$, the carry signal can be expressed as $C_i = A_iB_i + (A_i + B_i) \cdot 0 = A_iB_i$. And when $C_{i-1} = 1$, the carry signal can be expressed as $C_i = A_iB_i + (A_i + B_i) \cdot 1 = A_i + B_i$. Thus, the carry generation module of every bit only contains a 2-input AND gate and a 2-input OR gate. The total number of multiplexer gates in the 2^N-bit CCA is calculated as follows:

$$\sum_{n=1}^{N} 2^{N-n}(2^n - 1) = \sum_{n=1}^{N} 2^N - 2^{N-n}, \text{ where } N = \log_2 n. \quad (2)$$

No sum signals are selected in the unit; the proposed CCA saves 11 (= 28 – 17) multiplexers. Hence, the capacitive load of the multiplexer network of the CCA must be less than that of the CSA. The 3-input XOR gates are used to execute the final sum outputs. Fig. 4 reveals that an XOR function of the carry out and partial sum signal is required to generate the final sum outputs. The sum output signal can be expresseed as $S_{i+1} = (A_{i+1} \oplus B_{i+1}) \oplus C_i = P_{i+1} \oplus C_i$. Here, C_i is the selected signal of the final carry out.

The critical path carry of the 8-bit CCA also extends from A_0/B_0 to C_7. This occurs when the input is $(A_0 \sim A_7) = (1, 0 \sim 0)$; $(B_0 \sim B_7) = (1 \sim 1)$, which still involves gate delays of three stages of multiplexers. Fig. 5 shows the proposed conditional carry addition rule, which is an improvement of the conditional-sum addition rule. The improved addition rule can reduce the number on multiplexer gates in the conditional sum adders. The generated distant carriers are used to select the true carry outputs from two provisional carriers simultaneously under different carry input conditions. The arrows indicate the actual carriers formed between sections. The carries are generated simultaneously and independently on all sections.

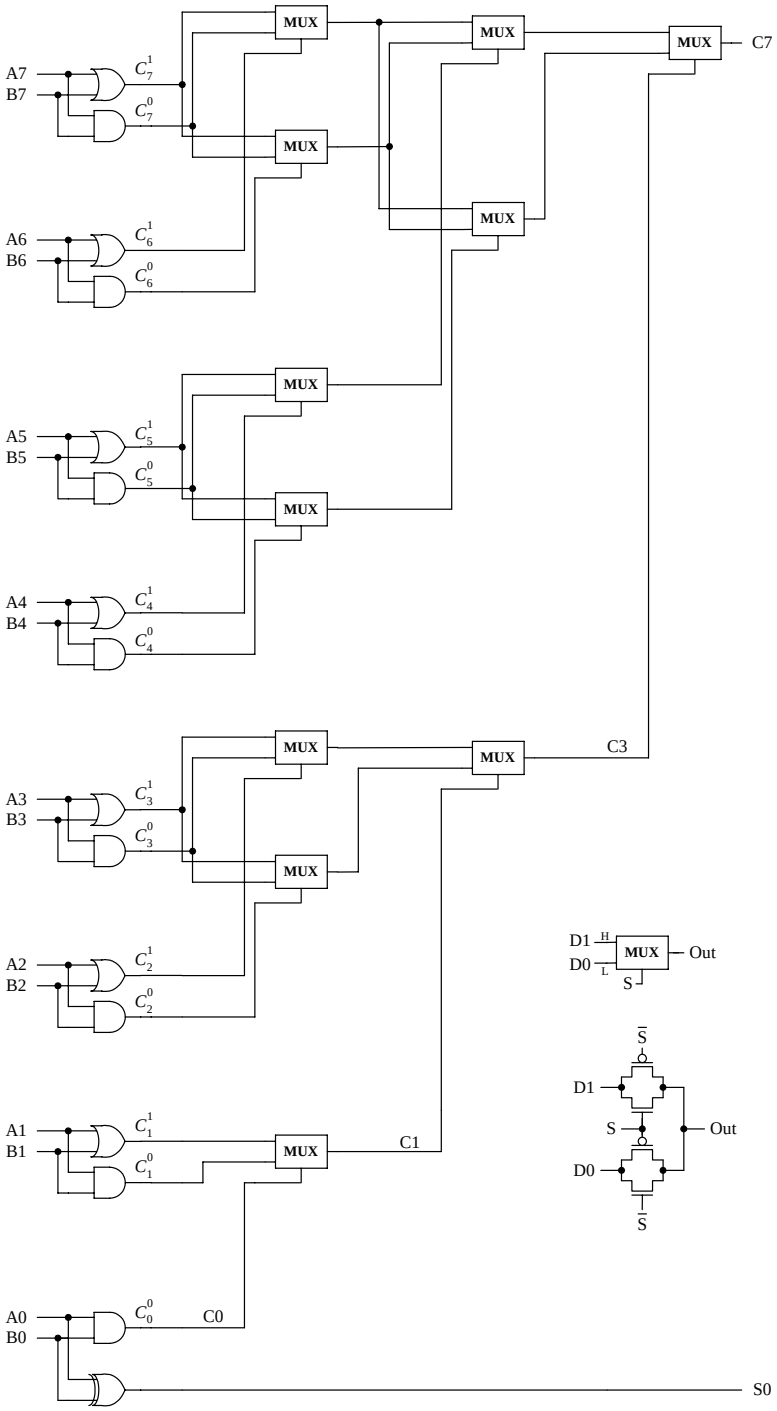

Fig. 3. Removal of multiplexers for sum output signals.

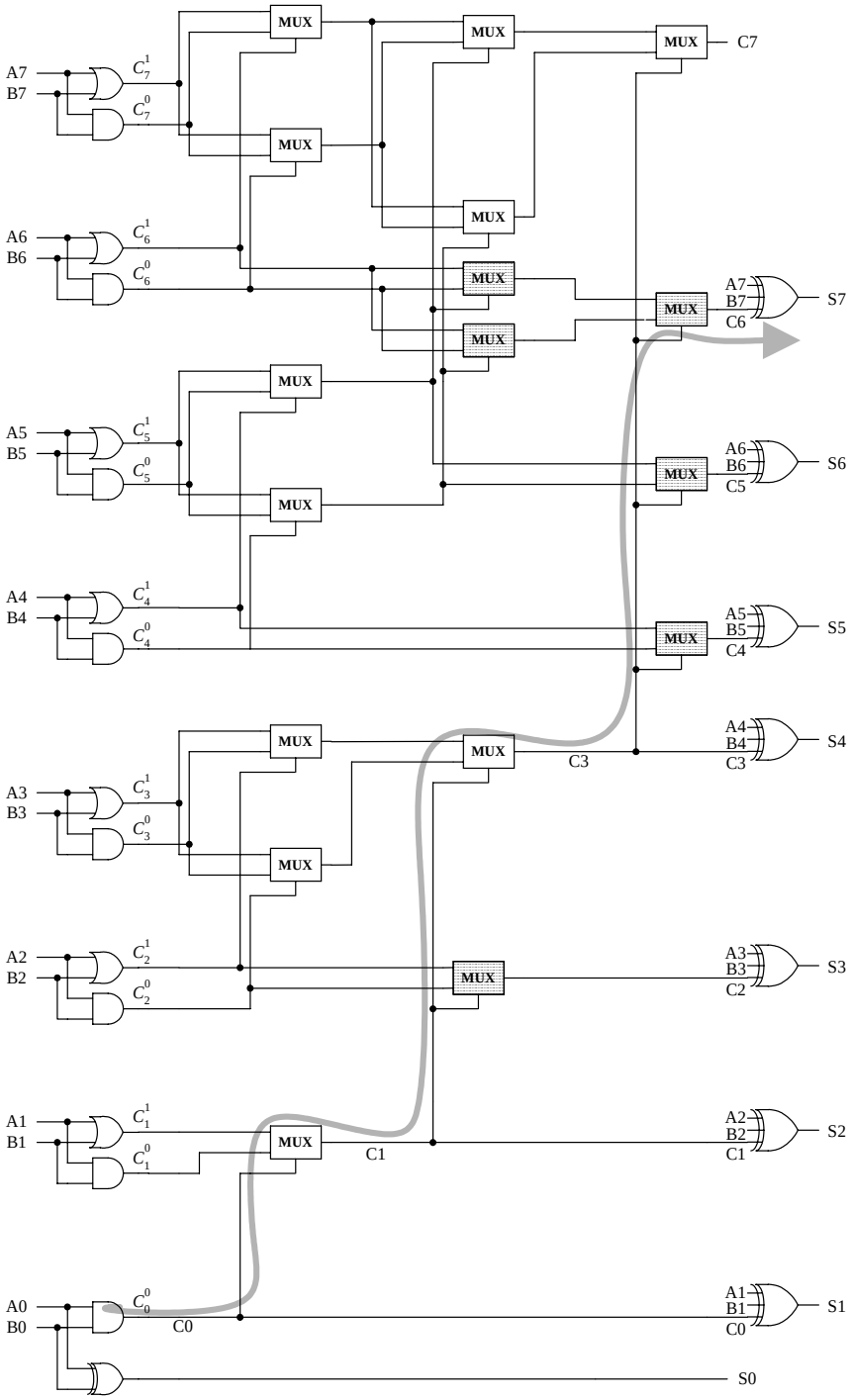

Fig. 4. Circuit and critical delay path of the 8-bit conditional carry adder (CCA).

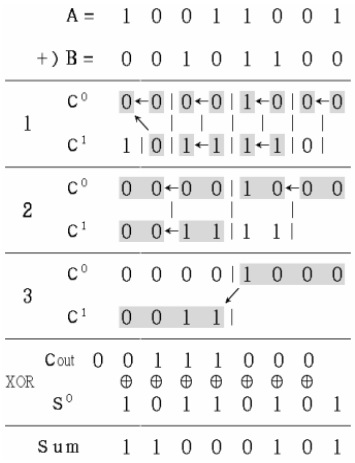

Fig. 5. Conditional carry addition rule.

3.2 Comparison of the CSA and CCA

The main difference between the proposed conditional carry addition rule and the conditional sum addition rule is that only carry signals are generated and selected in the multiplexer network tree in the CCA. Therefore, the capacitive load of the multiplexers in each network reduction step is markedly reduced, leading to high operation speed, low power dissipation and low hardware overhead. Moreover, the number of network reduction steps of the CCA is the same as that of the CSA, implying that the operation speed of the CCA is faster than that of the CSA during the multiplexer network reduction process.

The difference between the CSA and the CCA in terms of the conditional carry unit is that all shaded multiplexers are removed from the CSA in order to select sum bits, as shown in Fig. 2. Originally the sum output of the critical path of 8-bit CSA is,

$$S_7 = \overline{C_3}[\overline{(\overline{C_4^0}\cdot C_5^0 + C_4^0\cdot C_5^1)}(\overline{C_6^0}\cdot S_7^0 + C_6^0\cdot S_7^1) + (\overline{C_4^0}\cdot C_5^0 + C_4^0\cdot C_5^1)(\overline{C_6^1}\cdot S_7^0 + C_6^1\cdot S_7^1)] + C_3[\overline{(\overline{C_4^1}\cdot C_5^0 + C_4^1\cdot C_5^1)}(\overline{C_6^0}\cdot S_7^0 + C_6^0\cdot S_7^1) + (\overline{C_4^1}\cdot C_5^0 + C_4^1\cdot C_5^1)(\overline{C_6^1}\cdot S_7^0 + C_6^1\cdot S_7^1)]. \quad (3)$$

In the proposed CCA, the XOR gates are simply moved from the front-end to the back-end. Now the sum output of the critical path of the 8-bit proposed CCA is,

$$S_7 = S_7^0 \oplus \{\overline{C_3}[\overline{(\overline{C_4^0}\cdot C_5^0 + C_4^0\cdot C_5^1)}\ C_6^0 + (\overline{C_4^0}\cdot C_5^0 + C_4^0\cdot C_5^1)\ C_6^1] + C_3[\overline{(\overline{C_4^1}\cdot C_5^0 + C_4^1\cdot C_5^1)}\ C_6^0 + (\overline{C_4^1}\cdot C_5^0 + C_4^1\cdot C_5^1)\ C_6^1]\}. \quad (4)$$

The transistor count of the conditional unit of the CSA is $84p + 84n$ (= 28 × 3), and the transistor count of the conditional carry unit of the proposed CCA is $51p + 51n$ (= 17 × 3). Table 1 presents the numbers of 2-to-1 multiplexers used in the n-bit conditional

Table 1. Comparisons of multiplexer numbers.

	Num. of 2-to-1 MUX gates	
n-bit adder	CSA	**CCA**
8	28	17 (60.7%)
16	75	49 (65.3%)
32	**186**	**129 (69.4%)**
64	441	321 (72.8%)
128	1,016	769 (75.7%)

sum adder (CSA) and the n-bit proposed conditional carry adder (CCA) [10]. The 32-bit CSA needs 186 multiplexers, while the proposed 32-bit CCA needs only 129 multiplexers. The 32-bit conditional carry unit processes fewer signals than the 32-bit conditional sum adder during the same five conditional selection steps, potentially saving 57 multiplexers and reducing the number of internal nodes, and the internal parasitic capacitive load. Accordingly, the operation speed and power dissipation are both improved.

4. 32-BIT CIRCUIT IMPLEMENTATION AND MEASUREMENT RESULTS

The increased use of pass-transistor logic is a trend in low-power digital design. Based on pass-transistor logic, complementary pass-transistor logic (CPL) is very suitable for low-power low-voltage differential-end applications [10, 15, 16]. Fig. 6 shows a schematic diagram of the differential-end 2-to-1 multiplexer using CPL. CPL is based on the use of NMOS multiplexers to construct logic functions because halving the capacitance halves the level of transient power dissipation. Input signals are typically applied to both the gate and source/drain connections of MOS transistors, with the output taken from the other side. NMOS transistors are arranged in arrays that provide gate-level functions. Typically, CPL employs a PMOS cross-coupled latch to regenerate full voltage swing complementary signals from the CPL logic tree, and two static inverters are used to drive the following stages. Therefore, the need for additional buffer circuits is lower than it is for the static CMOS circuit. In addition, the complementary output signals can be obtained simultaneously, thus greatly reducing the skew problem.

Both the conventional single-end static CMOS circuit and the differential-end CPL circuit were used to implement and compare the 32-bit CSA and CCA. The proposed CCA clearly has a smaller layout area than the CSA. The implementations of the adder chip were based upon UMC 0.5μm Double-layer Polysilicon Double-layer Metal (DPDM) CMOS Process technology. Table 2 compares the layout areas of the adder designs. Compared with the 32-bit CSA, the improved 32-bit CCA architecture implemented with the static CMOS circuit reduces the number of internal nodes, the parasitic capacitive load, and the power dissipation, thus lowering the layout area by 17.8% and the delay time (by over 20% at a low voltage). The improved differential-end 32-bit CCA architecture implemented with the CPL again reduces the number of internal nodes, the parasitic capacitive load and the power dissipation, thus reducing the layout area by 23.6%, and resulting in almost equal delay times.

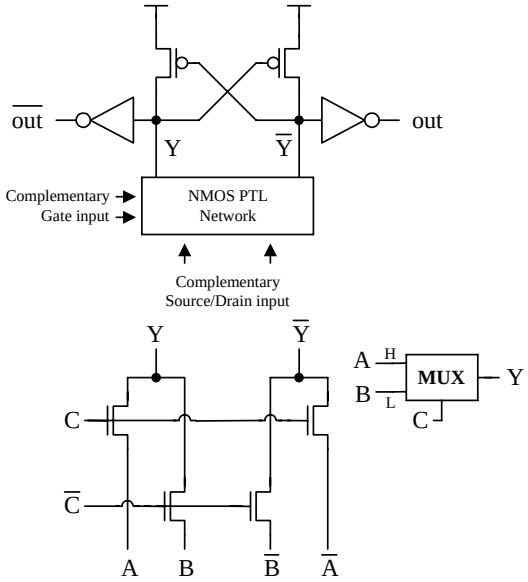

Fig. 6. Schematic diagram of the differential-end 2-to-1 multiplexer using CPL.

Table 2. Layout area comparison.

Proposed 32-bit CCA		Classic 32-bit CSA [11, 13]	
Static (single-end)	CPL (differential-end)	Static (single-end)	CPL (differential-end)
$920 \times 460\ \mu m^2$	$905 \times 550\ \mu m^2$	$920 \times 560\ \mu m^2$	$905 \times 720\ \mu m^2$

The proposed 32-bit CCA and conventional 32-bit CSA structures were completely designed and simulated using CPL and static CMOS logic. The simulated adder designs were based upon 0.5μm DPDM CMOS process technology. The simulated supply voltages varied from 1.5V to 3.3V. Both the 32-bit CSAs and 32-bit CCAs had the same critical carry path: c0 $\rightarrow$ c1 $\rightarrow$ c3 $\rightarrow$ c7 $\rightarrow$ c15 $\rightarrow$ c31 at an input of ($A_0 \sim A_{31}$) = (1, 0 ~ 0); ($B_0 \sim B_{31}$) = (1 ~ 1). Careful design and key patterns were considered in the experiment, and control chips were used to measure the critical delay and power dissipation. Fig. 7 compares of the worst delays of the 32-bit CSAs and CCAs. Fig. 8 compares their power-delay products. In the post-layout simulations, the proposed 32-bit CCA exhibited fewer delays and used less energy than the 32-bit CSA, with either the single-end CMOS or differential-end CPL circuit.

Fig. 9 compares the worst delays of the 32-bit CSAs and CCAs; Fig. 10 compares their power-delay products. When supply voltage was between 2.2V and 3.3V, the CCA implemented with the static CMOS circuit was faster than the CCA implemented with the CPL circuit. When the supply voltage was between 1.5V and 2.2V, the CPL version had better performance. When the power supply approached 1.5V, the voltage delivered from the NMOS logic tree of the CPL circuit, Vdd – Vtn, became too small. This caused a longer delay, thereby canceling out the desired benefit. Table 3 compares the worst

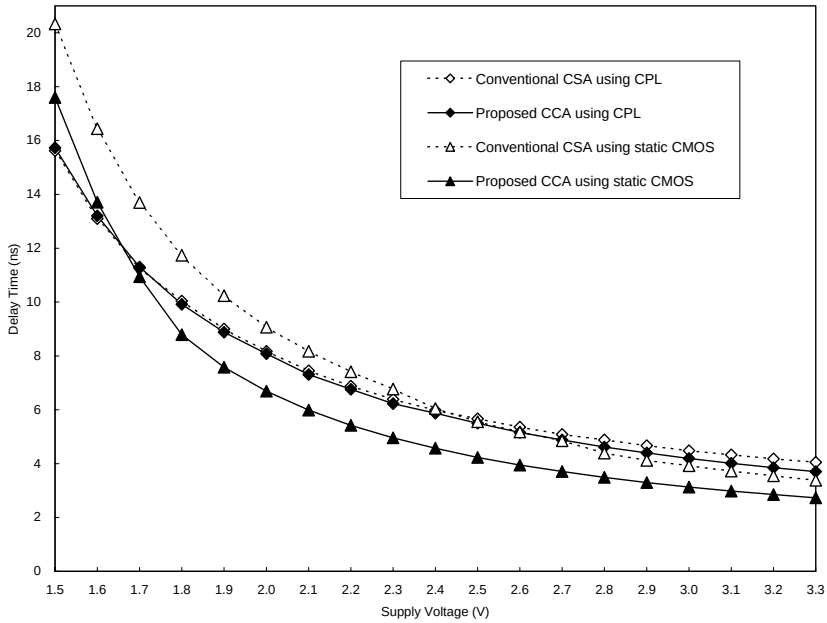

Fig. 7. Propagation delay comparison of simulated 32-bit static CMOS CSAs and CCAs.

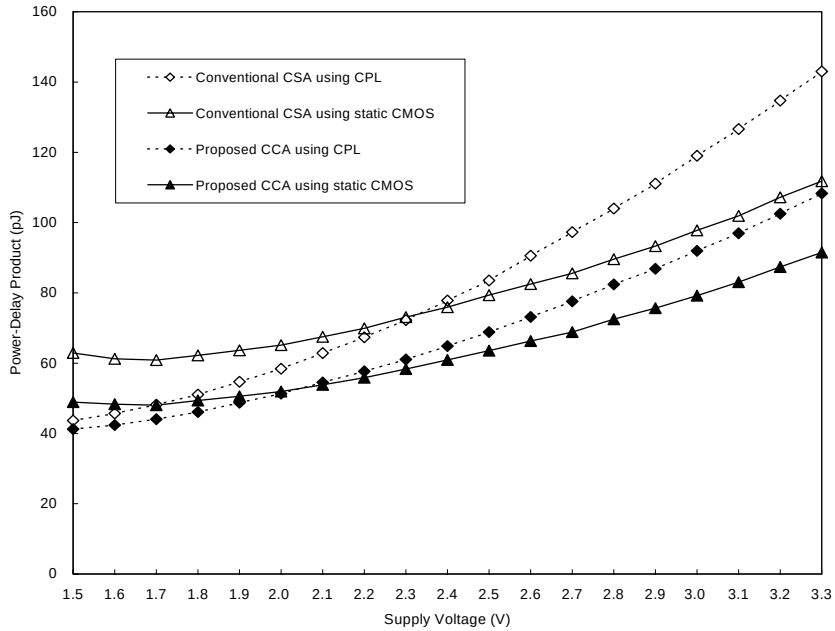

Fig. 8. Power-delay product comparison of simulated 32-bit static CMOS CSAs and CCAs.

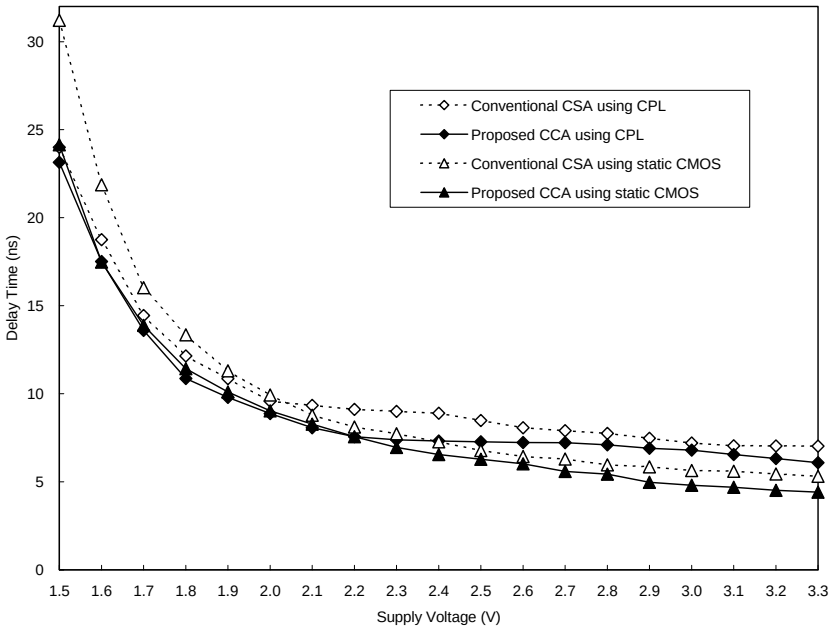

Fig. 9. Propagation delay comparison of 32-bit CSAs and CCAs.

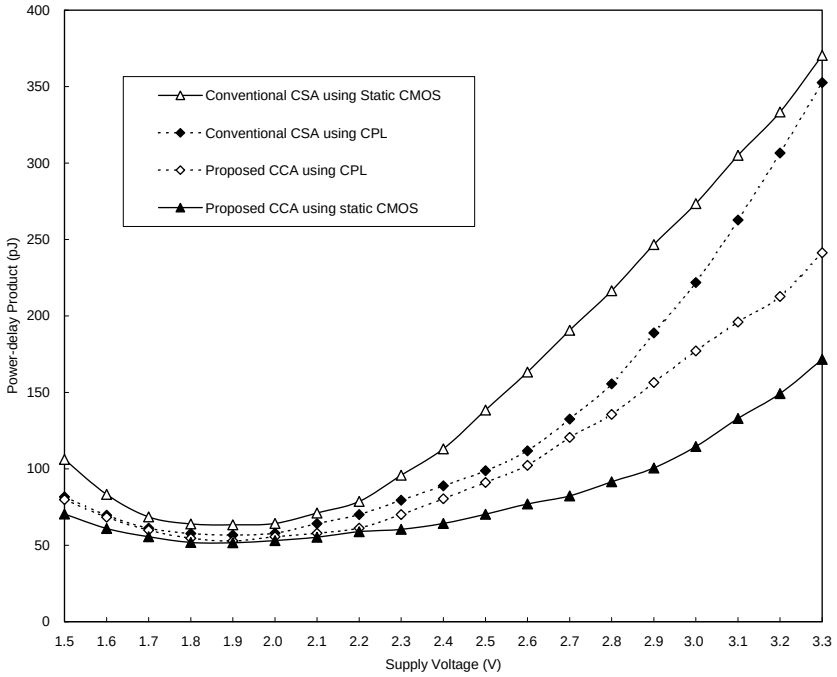

Fig. 10. Power-delay product comparison of 32-bit CSAs and CCAs.

Table 3. Comparison of various conditional carry/carry select adders.

Adder	V_{DD}	Tech.	Sim. delay/(meas.)
32-bit CSA using static CMOS	3.3V	0.5 μm CMOS	3.38/(7.03) ns
Proposed 32-bit CCA using static CMOS	3.3V	0.5 μm CMOS	2.73/(6.09) ns
32-bit CSA using CPL	3.3V	0.5 μm CMOS	4.05/(5.32) ns
Proposed 32-bit CCA using CPL	3.3V	0.5 μm CMOS	3.70/(4.42) ns
Proposed 32-bit CCA using CPL	3.3V	0.35 μm CMOS	2.10 ns
32-bit HSAC [9] (2002)	5V	1.0 μm BiCMOS	4.0 ns
32-bit [7] (1997)	5V	1.2 μm CMOS	3.28 ns
16-bit SICNB CCS [8] (2000)	1.5V	0.8 μm CMOS	14.56 ns

propagation delay of the CSA and the proposed CCA under standard supply voltage. It also shows data for various static conditional carry/carry select adders. Comparing the simulated data with the measurements indicates that inaccuracies in the simulation resulted mainly from the input/output pads and the package. The measurements revealed that the proposed 32-bit CCAs exhibited fewer delays and used less power than the 32-bit CSAs, and that the proposed 32-bit CCAs achieved better power-delay performance than the 32-bit CSAs, with either CMOS or CPL circuits implemented.

5. CONCLUDING REMARKS

This work has presented improved 32-bit conditional sum adders for high-speed low-power applications. The original 32-bit conditional sum adder uses 186 multiplexers, while the proposed 32-bit conditional carry adder (CCA) uses only 129 multiplexers, and the positions of the XOR gates are simply moved from the front-end to the back-end. The capacitance of the multiplexer network tree is therefore reduced, yielding benefits in power consumption and operation speed. Classical single-end CMOS and differential-end CPL circuits were utilized to implement the proposed 32-bit CCA scheme. It was found that the proposed adders always outperformed the old ones in these circuit implementations. The results of HSPICE post-layout simulations and measurements of the experimental chips indicate that these improved 32-bit adders can reduce the power-delay product by 10% to 25% and reduce the layout area of the CSA by about 20%.

ACKNOWLEDGMENT

The authors would like to thank the National Chip Implementation Center (CIC) of the National Applied Research Laboratories, Taiwan, for fabricating the four experimental 32-bit adder chips studied in this research.

REFERENCES

1. A. Bellaouar and M. I. Elmasry, *Low-Power Digital VLSI Design: Circuits and Systems*, Kluwer, MA, 1995.
2. N. H. E. Weste and K. Eshraghian, *Principle of CMOS VLSI Design*, 2nd ed., Addison Wesley, 1993.
3. K. Hwang, *Computer Arithmetic: Principles, Architecture, and Design*, John Wiley & Sons, Inc., 1976.
4. R. P. Brent and H. T. Kung, "A regular layout for parallel adders," *IEEE Transactions on Computers*, Vol. C-31, 1982, pp. 260-264.
5. O. J. Bedrij, "Carry-select adder," *IRE Transactions on Electronic Computers*, Vol. EC-11, 1962, pp. 340-346.
6. S. Perri, P. Corsonello, and G. Cocorullo, "A high-speed energy-efficient 64-bit reconfigurable binary adder," *IEEE Transactions on Very Large Scale Integration* (*VLSI*) *Systems*, Vol. 11, 2003, pp. 939-943.
7. J. C. Lo, "A fast binary adder with conditional carry generation," *IEEE Transactions on Computer*, Vol. 46, 1997, pp. 248-253.
8. Y. M. Huang and J. B. Kuo, "A high-speed conditional carry select (CCS) adder circuit with a successively incremented carry number block (SICNB) structure for low-voltage VLSI implementation," *IEEE Transactions on Circuit and System II: Analog & Digital Signal Processing*, Vol. 47, 2000, pp. 1074-1079.
9. Y. Wang, C. Pai, and X. Song, "The design of hybrid carry-lookahead/carry-select adders," *IEEE Transactions on Circuit and Systems II: Analog and Digital Signal Processing*, Vol. 49, 2002, pp. 16-24.
10. K. H. Cheng, S. M. Chiang, and S. W. Cheng, "The improvement of conditional Sum adder for low power applications," in *Proceedings of the IEEE International Application Specific Integrated Circuits Conference*, 1998, pp. 131-134.
11. J. Sklansky, "Conditional-sum addition logic," *IRE Transactions on Electronic Computers*, Vol. EC-9, 1960, pp. 226-231.
12. K. Yano, T. Yamanaka, T. Nishida, M. Saito, K. Shimohigashi, and A. Shimizu, "A 3.8-ns CMOS 16×16-b multiplier using complementary pass-transistor logic," *IEEE Journal of Solid-State Circuits*, Vol. 25, 1990, pp. 388-395.
13. R. E. Ladner and M. J. Fischer, "Parallel prefix computation," *Journal of the ACM*, Vol. 27, 1980, pp. 831-838.
14. O. L. MacSorley, "High-speed arithmetic in binary computers," in *Proceedings of the IRE*, Vol. 49, 1961, pp. 67-91.
15. A. Chandrakasan and R. W. Broersen, "Minimizing power consumption in digital CMOS circuit," in *Proceedings of the IEEE*, Vol. 83, 1995, pp. 498-523.
16. R. Zimmermann and W. Fichtner, "Low-power logic styles: CMOS versus pass-transistor logic," *IEEE Journal of Solid-State Circuits*, Vol. 32, 1997, pp. 1079-1090.

Kuo-Hsing Cheng (鄭國興) was born in Taipei, Taiwan, in 1962. He received the B.S. degree from the Department of *Electrical Engineering*, National Central University in 1985, and the M.S. and Ph.D. degrees from the Institute of Electronics Engineering,

National Chiao Tung University, Taiwan, in 1987 and 1992, respectively. During 1992 to 1993, he was an associate researcher in the Chip Implementation Center, the Nation Science Council, Taiwan. From 1993 to 2003, he was an Associate Professor at Tamkang University, Taiwan. In 2003, he joined the faculty of the National Central University, Chungli, Taiwan. He is currently an Associate Professor in the Department of Electrical Engineering. He has published more than 70 technical papers on integrated circuits. His research interests include LV/LP high-speed mixed-signal IC and systems, clock synchronization circuits, and ultra-high-frequency mixed signal circuits for wire communications.

Shun-Wen Cheng (鄭舜文) received his two B.S. degrees in Electrical Engineering and in Information Engineering from Tamkang University, Taipei, Taiwan, in 1996, and the Ph.D. degree in Electrical Engineering from Tamkang University in 2005. From 1994 to 1997, He designed and constructed the Juvenile Delinquent Pre-hearing Investigation Database System for the Taipei and Shih-Lin District Courts in Taiwan. Since 1999, he has served as a lecturer in Tze-Chiang Foundation of Science and Technology (TCFST), National Tsing Hua University, Hsinchu, Taiwan. In 2005, he joined the faculty of the Far East College, Tainan, Taiwan. He is currently an Assistant Professor in the Department of Electronic Engineering. He has published more than 20 technical papers on integrated circuits and system analysis. His present interests include system analysis and high-performance digital IC design.