(MAD) algorithm introduced in [1], [2] is used for coefficients quantization. The subfilter is based on canonical signed digit (CSD) structure and Carry-Save adders are used. Tables III, IV, and V show the results of area, power, and critical path delay, synthesized by Design Compiler [10] with 45-nm technology.

VI. Conclusion

In this paper, we have presented new parallel FIR filter structures, which are beneficial to symmetric convolutions when the number of taps is the multiple of 2 or 3. Multipliers are the major portions in hardware consumption for the parallel FIR filter implementation. The proposed new structure exploits the nature of even symmetric coefficients and save a significant amount of multipliers at the expense of additional adders. Since multipliers outweigh adders in hardware cost, it is profitable to exchange multipliers with adders. Moreover, the number of increased adders stays still when the length of FIR filter becomes large, whereas the number of reduced multipliers increases along with the length of FIR filter. Consequently, the larger the length of FIR filters is, the more the proposed structures can save from the existing FFA structures, with respect to the hardware cost. Overall, in this paper, we have provided new parallel FIR structures consisting of advantageous polyphase decompositions dealing with symmetric convolutions comparatively better than the existing FFA structures in terms of hardware consumption.

References

[1] D. A. Parker and K. K. Parhi, "Low-area/power parallel FIR digital filter implementations," *J. VLSI Signal Process. Syst.*, vol. 17, no. 1, pp. 75–92, 1997.

[2] J. G. Chung and K. K. Parhi, "Frequency-spectrum-based low-area low-power parallel FIR filter design," *EURASIP J. Appl. Signal Process.*, vol. 2002, no. 9, pp. 444–453, 2002.

[3] K. K. Parhi, *VLSI Digital Signal Processing Systems: Design and Implementation*. New York: Wiley, 1999.

[4] Z.-J. Mou and P. Duhamel, "Short-length FIR filters and their use in fast nonrecursive filtering," *IEEE Trans. Signal Process.*, vol. 39, no. 6, pp. 1322–1332, Jun. 1991.

[5] J. I. Acha, "Computational structures for fast implementation of L-path and L-block digital filters," *IEEE Trans. Circuit Syst.*, vol. 36, no. 6, pp. 805–812, Jun. 1989.

[6] C. Cheng and K. K. Parhi, "Hardware efficient fast parallel FIR filter structures based on iterated short convolution," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 51, no. 8, pp. 1492–1500, Aug. 2004.

[7] C. Cheng and K. K. Parhi, "Furthur complexity reduction of parallel FIR filters," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS 2005)*, Kobe, Japan, May 2005.

[8] C. Cheng and K. K. Parhi, "Low-cost parallel FIR structures with 2-stage parallelism," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 54, no. 2, pp. 280–290, Feb. 2007.

[9] I.-S. Lin and S. K. Mitra, "Overlapped block digital filtering," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 43, no. 8, pp. 586–596, Aug. 1996.

[10] *"Design Compiler User Guide,"* ver. B-2008.09, Synopsys Inc., Sep. 2008.

Low-Power and Area-Efficient Carry Select Adder

B. Ramkumar and Harish M Kittur

***Abstract*—Carry Select Adder (CSLA) is one of the fastest adders used in many data-processing processors to perform fast arithmetic functions. From the structure of the CSLA, it is clear that there is scope for reducing the area and power consumption in the CSLA. This work uses a simple and efficient gate-level modification to significantly reduce the area and power of the CSLA. Based on this modification 8-, 16-, 32-, and 64-b square-root CSLA (SQRT CSLA) architecture have been developed and compared with the regular SQRT CSLA architecture. The proposed design has reduced area and power as compared with the regular SQRT CSLA with only a slight increase in the delay. This work evaluates the performance of the proposed designs in terms of delay, area, power, and their products by hand with logical effort and through custom design and layout in 0.18-μm CMOS process technology. The results analysis shows that the proposed CSLA structure is better than the regular SQRT CSLA.**

***Index Terms*—Application-specific integrated circuit (ASIC), area-efficient, CSLA, low power.**

I. Introduction

Design of area- and power-efficient high-speed data path logic systems are one of the most substantial 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. The sum for each bit position in an elementary adder is generated sequentially only after the previous bit position has been summed and a carry propagated into the next position.

The CSLA is used in many computational systems to alleviate the problem of carry propagation delay by independently generating multiple carries and then select a carry to generate the sum [1]. However, the CSLA is not area efficient because it uses multiple pairs of Ripple Carry Adders (RCA) to generate partial sum and carry by considering carry input $\mathrm{Cin} = 0$ and $\mathrm{Cin} = 1$, then the final sum and carry are selected by the multiplexers (mux).

The basic idea of this work is to use Binary to Excess-1 Converter (BEC) instead of RCA with $\mathrm{Cin} = 1$ in the regular CSLA to achieve lower area and power consumption [2]–[4]. The main advantage of this BEC logic comes from the lesser number of logic gates than the n-bit Full Adder (FA) structure. The details of the BEC logic are discussed in Section III.

This brief is structured as follows. Section II deals with the delay and area evaluation methodology of the basic adder blocks. Section III presents the detailed structure and the function of the BEC logic. The SQRT CSLA has been chosen for comparison with the proposed design as it has a more balanced delay, and requires lower power and area [5], [6]. The delay and area evaluation methodology of the regular and modified SQRT CSLA are presented in Sections IV and V, respectively. The ASIC implementation details and results are analyzed in Section VI. Finally, the work is concluded in Section VII.

Manuscript received May 12, 2010; revised October 28, 2010; accepted December 15, 2010. Date of publication January 24, 2011; date of current version January 18, 2012.

The authors are with the School of Electronics Engineering, VIT University, Vellore 632 014, India (e-mail: ramkumar.b@vit.ac.in; kittur@vit.ac.in).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVLSI.2010.2101621

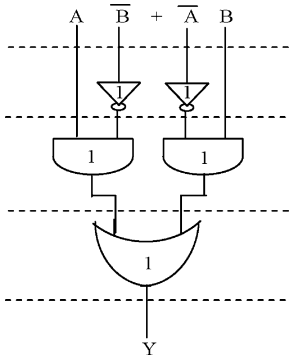

Fig. 1. Delay and Area evaluation of an XOR gate.

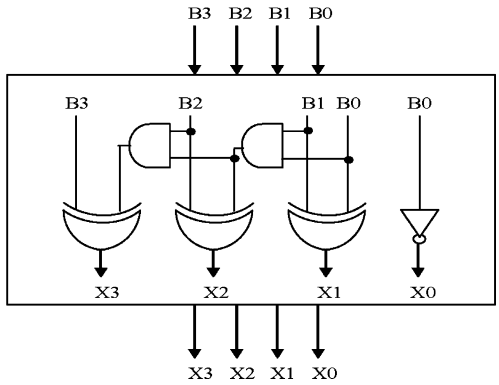

Fig. 2. 4-b BEC.

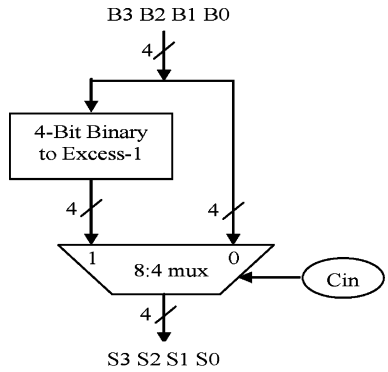

Fig. 3. 4-b BEC with 8:4 mux.

II. Delay and Area Evaluation Methodology of the Basic Adder Blocks

The AND, OR, and Inverter (AOI) implementation of an XOR gate is shown in Fig. 1. The gates between the dotted lines are performing the operations in parallel and the numeric representation of each gate indicates the delay contributed by that gate. The delay and area evaluation methodology considers all gates to be made up of AND, OR, and Inverter, each having delay equal to 1 unit and area equal to 1 unit. We then add up the number of gates in the longest path of a logic block that contributes to the maximum delay. The area evaluation is done by counting the total number of AOI gates required for each logic block. Based on this approach, the CSLA adder blocks of 2:1 mux, Half Adder (HA), and FA are evaluated and listed in Table I.

TABLE I
Delay and Area Count of the Basic Blocks of CSLA

Adder blocks	Delay	Area
XOR	3	5
2:1 Mux	3	4
Half adder	3	6
Full adder	6	13

TABLE II
Function Table of the 4-b BEC

B[3:0]	X[3:0]
0000	0001
0001	0010
⋮	⋮
1110	1111
1111	0000

III. BEC

As stated above the main idea of this work is to use BEC instead of the RCA with $\mathrm{Cin} = 1$ in order to reduce the area and power consumption of the regular CSLA. To replace the n-bit RCA, an $n+1$-bit BEC is required. A structure and the function table of a 4-b BEC are shown in Fig. 2 and Table II, respectively.

Fig. 3 illustrates how the basic function of the CSLA is obtained by using the 4-bit BEC together with the mux. One input of the 8:4 mux gets as it input (B3, B2, B1, and B0) and another input of the mux is the BEC output. This produces the two possible partial results in parallel and the mux is used to select either the BEC output or the direct inputs according to the control signal Cin. The importance of the BEC logic stems from the large silicon area reduction when the CSLA with large number of bits are designed. The Boolean expressions of the 4-bit BEC is listed as (note the functional symbols $\sim$ NOT, & AND,^XOR)

$$\mathrm{X0} = \sim \mathrm{B0}$$
$$\mathrm{X1} = \mathrm{B0} \hat{} \mathrm{B1}$$
$$\mathrm{X2} = \mathrm{B2} \hat{} (\mathrm{B0} \,\&\, \mathrm{B1})$$
$$\mathrm{X3} = \mathrm{B3} \hat{} (\mathrm{B0} \,\&\, \mathrm{B1} \,\&\, \mathrm{B2}).$$

IV. Delay and Area Evaluation Methodology of Regular 16-b SQRT CSLA

The structure of the 16-b regular SQRT CSLA is shown in Fig. 4. It has five groups of different size RCA. The delay and area evaluation of each group are shown in Fig. 5, in which the numerals within [] specify the delay values, e.g., sum2 requires 10 gate delays. The steps leading to the evaluation are as follows.

1) The group2 [see Fig. 5(a)] has two sets of 2-b RCA. Based on the consideration of delay values of Table I, the arrival time of selection input $c1[\mathrm{time}(t) = 7]$ of 6:3 mux is earlier than $s3[t = 8]$ and later than $s2[t = 6]$. Thus, $\mathrm{sum3}[t = 11]$ is summation of $s3$ and $\mathrm{mux}[\mathrm{t} = 3]$ and $\mathrm{sum2}[t = 10]$ is summation of $c1$ and mux.

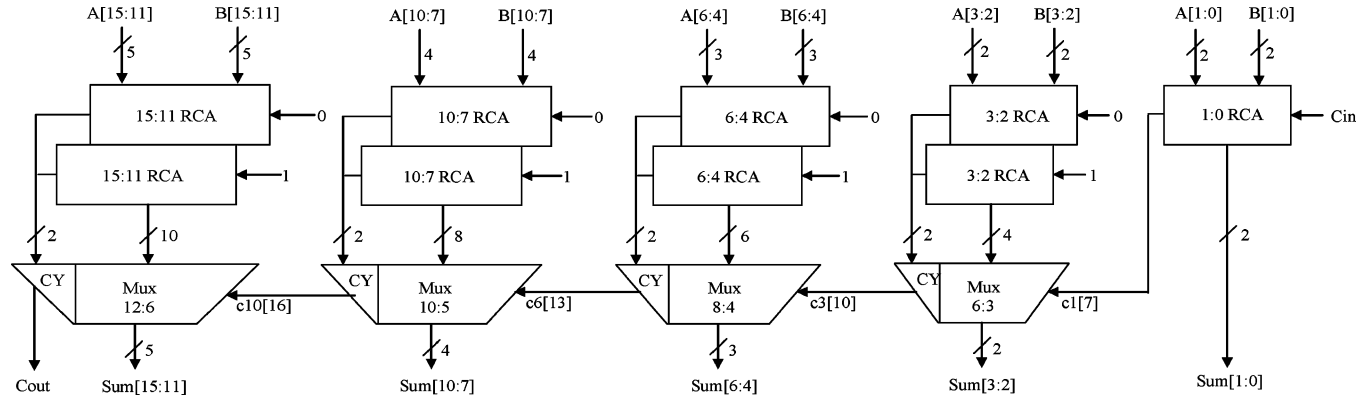

Fig. 4. Regular 16-b SQRT CSLA.

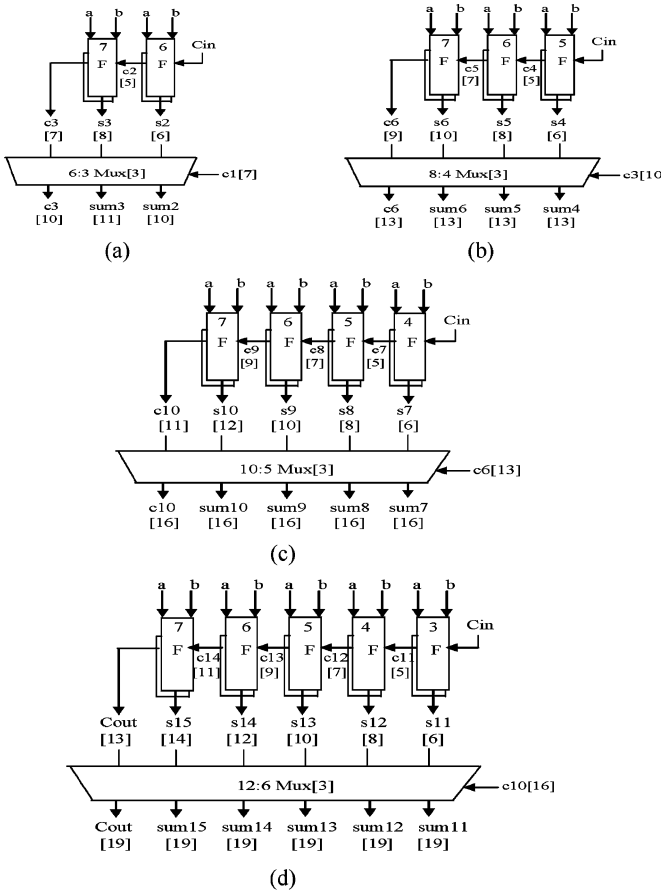

Fig. 5. Delay and area evaluation of regular SQRT CSLA: (a) group2, (b) group3, (c) group4, and (d) group5. F is a Full Adder.

2) Except for group2, the arrival time of mux selection input is always greater than the arrival time of data outputs from the RCA's. Thus, the delay of group3 to group5 is determined, respectively as follows:

$$\{c6, \text{sum}[6:4]\} = c3[t = 10] + \text{mux}$$
$$\{c10, \text{sum}[10:7]\} = c6[t = 13] + \text{mux}$$
$$\{\text{cout},\ \text{sum}[15:11]\} = c10[t = 16] + \text{mux}.$$

3) The one set of 2-b RCA in group2 has 2 FA for $\text{Cin} = 1$ and the other set has 1 FA and 1 HA for $\text{Cin} = 0$. Based on the area count of Table I, the total number of gate counts in group2 is determined as follows:

$$\text{Gate count} = 57\ (\text{FA} + \text{HA} + \text{Mux})$$
$$\text{FA} = 39(3 * 13)$$
$$\text{HA} = 6(1 * 6)$$
$$\text{Mux} = 12(3 * 4).$$

TABLE III
DELAY AND AREA COUNT OF REGULAR SQRT CSLA GROUPS

Group	Delay	Area
Group2	11	57
Group3	13	87
Group4	16	117
Group5	19	147

4) Similarly, the estimated maximum delay and area of the other groups in the regular SQRT CSLA are evaluated and listed in Table III.

V. DELAY AND AREA EVALUATION METHODOLOGY OF MODIFIED 16-B SQRT CSLA

The structure of the proposed 16-b SQRT CSLA using BEC for RCA with $\text{Cin} = 1$ to optimize the area and power is shown in Fig. 6. We again split the structure into five groups. The delay and area estimation of each group are shown in Fig. 7. The steps leading to the evaluation are given here.

1) The group2 [see Fig. 7(a)] has one 2-b RCA which has 1 FA and 1 HA for $\text{Cin} = 0$. Instead of another 2-b RCA with $\text{Cin} = 1$ a 3-b BEC is used which adds one to the output from 2-b RCA. Based on the consideration of delay values of Table I, the arrival time of selection input $c1[\text{time(t)} = 7]$ of 6:3 mux is earlier than the $s3[\text{t} = 9]$ and $c3[\text{t} = 10]$ and later than the $s2[\text{t} = 4]$. Thus, the sum3 and final $c3$ (output from mux) are depending on $s3$ and mux and partial $c3$ (input to mux) and mux, respectively. The sum2 depends on $c1$ and mux.
2) For the remaining group's the arrival time of mux selection input is always greater than the arrival time of data inputs from the BEC's. Thus, the delay of the remaining groups depends on the arrival time of mux selection input and the mux delay.

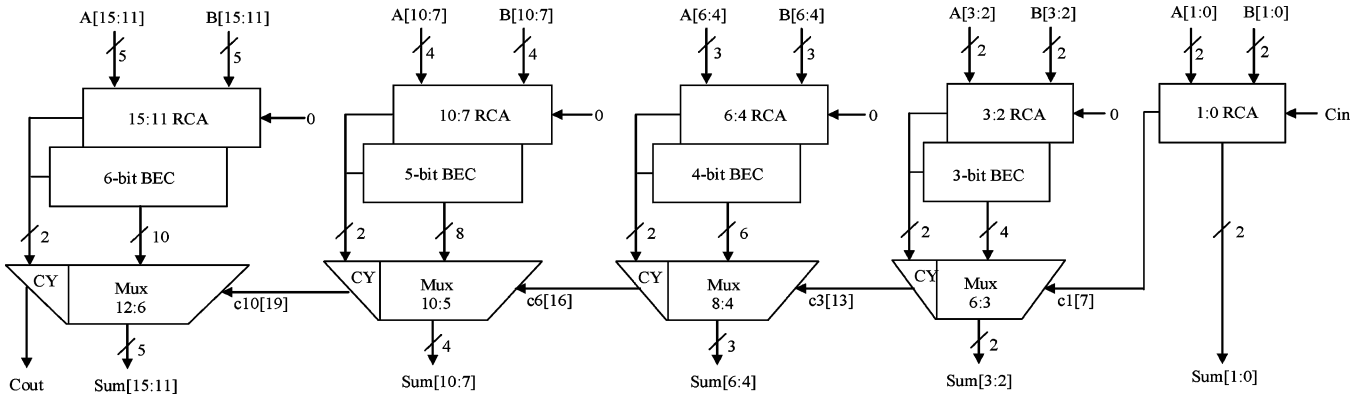

Fig. 6. Modified 16-b SQRT CSLA. The parallel RCA with Cin = 1 is replaced with BEC.

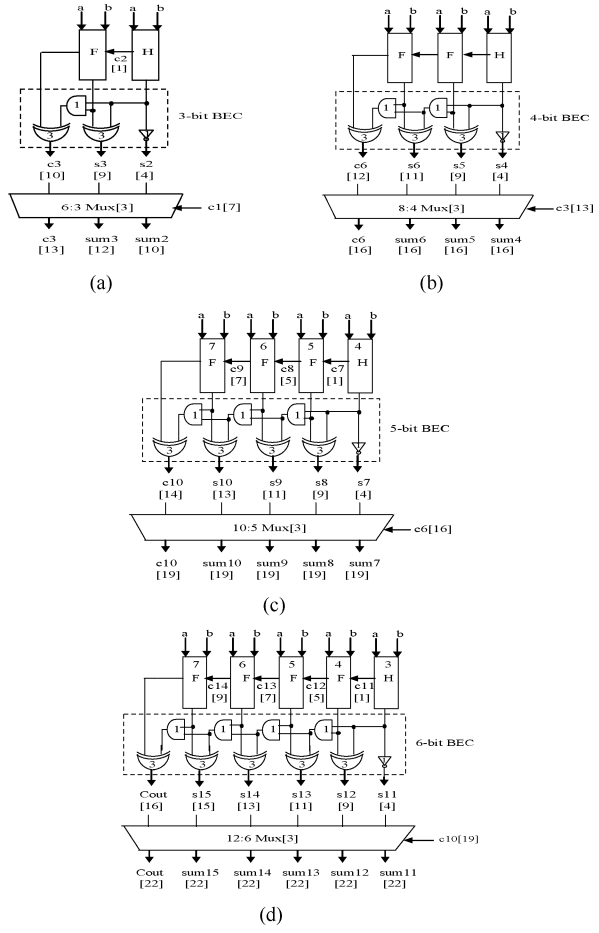

Fig. 7. Delay and area evaluation of modified SQRT CSLA: (a) group2, (b) group3, (c) group4, and (d) group5. H is a Half Adder.

3) The area count of group2 is determined as follows:

$$\text{Gate count} = 43\ (\text{FA} + \text{HA} + \text{Mux} + \text{BEC})$$
$$\text{FA} = 13(1 * 13)$$
$$\text{HA} = 6(1 * 6)$$
$$\text{AND} = 1$$
$$\text{NOT} = 1$$
$$\text{XOR} = 10(2 * 5)$$
$$\text{Mux} = 12(3 * 4).$$

TABLE IV
DELAY AND AREA COUNT OF MODIFIED SQRT CSLA

Group	Delay	Area
Group2	13	43
Group3	16	61
Group4	19	84
Group5	22	107

4) Similarly, the estimated maximum delay and area of the other groups of the modified SQRT CSLA are evaluated and listed in Table IV.

Comparing Tables III and IV, it is clear that the proposed modified SQRT CSLA saves 113 gate areas than the regular SQRT CSLA, with only 11 increases in gate delays. To further evaluate the performance, we have resorted to ASIC implementation and simulation.

VI. ASIC IMPLEMENTATION RESULTS

The design proposed in this paper has been developed using Verilog-HDL and synthesized in Cadence RTL compiler using typical libraries of TSMC 0.18 um technology. The synthesized Verilog netlist and their respective design constraints file (SDC) are imported to Cadence SoC Encounter and are used to generate automated layout from standard cells and placement and routing [7]. Parasitic extraction is performed using Encounter's Native *RC* extraction tool and the extracted parasitic *RC* (SPEF format) is back annotated to Common Timing Engine in Encounter platform for static timing analysis. For each word size of the adder, the same value changed dump (VCD) file is generated for all possible input conditions and imported the same to Cadence Encounter Power Analysis to perform the power simulations. The similar design flow is followed for both the regular and modified SQRT CSLA.

Table V exhibits the simulation results of both the CSLA structures in terms of delay, area and power. The area indicates the total cell area of the design and the total power is sum of the leakage power, internal power and switching power. The percentage reduction in the cell area, total power, power-delay product and the area–delay product as function of the bit size are shown in Fig. 8(a). Also plotted is the percentage delay overhead in Fig. 8(b). It is clear that the area of the 8-, 16-, 32-, and 64-b proposed SQRT CSLA is reduced by 9.7%, 15%, 16.7%, and 17.4%, respectively. The total power consumed shows a similar trend of increasing reduction in power consumption 7.6%, 10.56%, 13.63%, and 15.46 % with the bit size. Interestingly, the delay overhead also

TABLE V
COMPARISON OF THE REGULAR AND MODIFIED SQRT CSLA

Word Size	Adder	Delay (ns)	Area (um^2)	Power (uW)			Power-Delay Product(10^{-15})	Area-Delay Product(10^{-21})
				Leakage Power	Switching power	Total power*		
8-bit	Regular CSLA	1.719	991	0.007	101.9	203.9	350.5	1703.5
	Modified CSLA	1.958	895	0.006	94.2	188.4	368.8	1752.4
16-bit	Regular CSLA	2.775	2272	0.017	263.7	527.5	1463.8	6304.8
	Modified CSLA	3.048	1929	0.013	235.9	471.8	1438.0	5879.6
32-bit	Regular CSLA	5.137	4783	0.036	563.6	1127.3	5790.9	24570.2
	Modified CSLA	5.482	3985	0.027	484.9	969.9	5316.9	21845.7
64-bit	Regular CSLA	9.174	9916	0.075	1212.4	2425.0	22246.9	90969.3
	Modified CSLA	9.519	8183	0.057	1025.0	2050.1	19514.9	77893.9

*Total power = leakage power + Internal power + Switching power

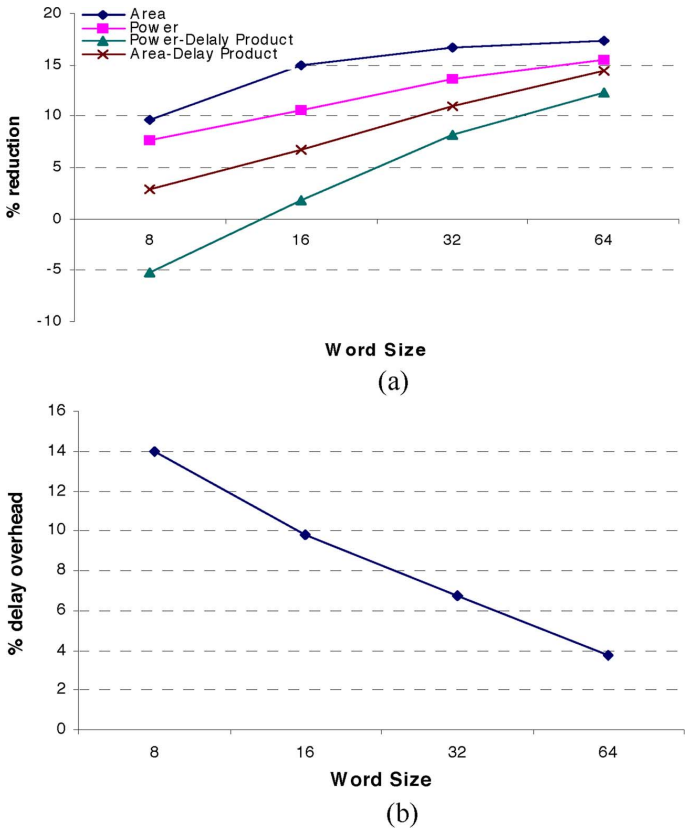

Fig. 8. (a) Percentage reduction in the cell area, total power, power–delay product, and area–delay product. (b) Percentage of delay overhead.

exhibits a similarly decreasing trend with bit size. The delay overhead for the 8, 16, and 32-b is 14%, 9.8%, and 6.7% respectively, whereas for the 64-b it reduces to only 3.76%. The power–delay product of the proposed 8-b is higher than that of the regular SQRT CSLA by 5.2% and the area-delay product is lower by 2.9%. However, the power-delay product of the proposed 16-b SQRT CSLA reduces by 1.76% and for the 32-b and 64-b by as much as 8.18%, and 12.28% respectively. Similarly the area-delay product of the proposed design for 16-, 32-, and 64-b is also reduced by 6.7%, 11%, and 14.4% respectively.

VII. CONCLUSION

A simple approach is proposed in this paper to reduce the area and power of SQRT CSLA architecture. The reduced number of gates of this work offers the great advantage in the reduction of area and also the total power. The compared results show that the modified SQRT CSLA has a slightly larger delay (only 3.76%), but the area and power of the 64-b modified SQRT CSLA are significantly reduced by 17.4% and 15.4% respectively. The power-delay product and also the area-delay product of the proposed design show a decrease for 16-, 32-, and 64-b sizes which indicates the success of the method and not a mere tradeoff of delay for power and area. The modified CSLA architecture is therefore, low area, low power, simple and efficient for VLSI hardware implementation. It would be interesting to test the design of the modified 128-b SQRT CSLA.

ACKNOWLEDGMENT

The authors would like to thank S. Sivanantham, P. MageshKannan, and S. Ravi of the VLSI Division, VIT University, Vellore, India, for their contributions to this work.

REFERENCES

[1] O. J. Bedrij, "Carry-select adder," *IRE Trans. Electron. Comput.*, pp. 340–344, 1962.
[2] B. Ramkumar, H. M. Kittur, and P. M. Kannan, "ASIC implementation of modified faster carry save adder," *Eur. J. Sci. Res.*, vol. 42, no. 1, pp. 53–58, 2010.
[3] T. Y. Ceiang and M. J. Hsiao, "Carry-select adder using single ripple carry adder," *Electron. Lett.*, vol. 34, no. 22, pp. 2101–2103, Oct. 1998.
[4] Y. Kim and L.-S. Kim, "64-bit carry-select adder with reduced area," *Electron. Lett.*, vol. 37, no. 10, pp. 614–615, May 2001.
[5] J. M. Rabaey, *Digtal Integrated Circuits—A Design Perspective*. Upper Saddle River, NJ: Prentice-Hall, 2001.
[6] Y. He, C. H. Chang, and J. Gu, "An area efficient 64-bit square root carry-select adder for low power applications," in *Proc. IEEE Int. Symp. Circuits Syst.*, 2005, vol. 4, pp. 4082–4085.
[7] Cadence, "Encounter user guide," Version 6.2.4, March 2008.