ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Comparison of Effective area Efficient Architectures for Modified SQRT CSLA

Gunja Naresh
M.Tech Student,
Department of ECE,
DRK Institute of Science and Technology.

Mrs.Sadguna
Assistant Professor,
Department of ECE,
DRK Institute of Science and Technology.

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-bit, 16-bit, 32-bit, 64-bit 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-m CMOS process technology. The results analysis shows that the proposed CSLA structure is better than the regular SQRT CSLA.

Keywords:

SQRT CSLA, area efficient, CSLA, low power, delay efficient.

I. INTRODUCTION:

Design of area and power efficient high speed data path-logic systems are one of the most substantial areas of researchin VLSI system design. In digital adders, the speed of additionis limited by the time required to propagate a carry throughthe adder. The sum for each bit position in an elementaryadder is generated sequentially only after the previous bitposition has been summed and a carry propagated into thenext position.The CLSA is used in many computational system is toalleviate the problem of carry propagation delay byindependently generating multiple carries and then select acarry to generate the sum [1].

However, the CSLA[3] is not areaefficient because it uses multiple pairs of Ripple Carry Adders(RCA) to generate partial sum and carry by considering carryinput and then the final sum and carry are selected by themultiplexers (mux). The basic idea of this work is to use Binary to Excess-1converted (BEC) instead of RCA with in the regular CSLA toachieve lower area and power consumption [2]-[4]. The mainadvantage of this BEC logic comes from the lesser number oflogic gates than the bit Full Adder (FA) structure.This brief isstructured as follows. This paper deals with thedelay and area evaluation methodology of the basic adder blocks. And also presents the detailed structure and thefunction of the BEC logic.The SQRT CSLA has been chosen for comparison with theproposed 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.The rest of the paper is organised as follows.In Section II, logic formulation is presented. In Section III, the proposed adder design is explained. In Section IV, the proposed scheme is compared to the previously proposed ones and results are shown. Finally, Section V concludes this paper.

II. LOGIC FORMULATION:

The CSLA has two units: 1) the sum and carry generator unit (SCG) and 2) the sum and carry selection unit [9]. The SCG unit consumes most of the logic resources of CSLA as shown in fig 1, and significantly contributes to the critical path. Different logic designs have been suggested for efficient implementation of the SCG unit. We made a study of the logic designs suggested for the SCG unit of conventional and BEC-based CSLAs of [6] by suitable logic expressions. The main objective of this study is to identify redundant logic operations and data dependence. Accordingly, we remove all redundant logic operations and sequence logic operations based on their data dependence which are discussed below.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

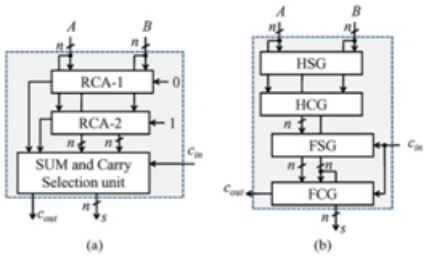

Fig. 1. (a) Conventional CSLA; n is the input operand bit-width. (b) The logic operations of the RCA.

A. Logic Expressions of the SCG Unit of th-eConventional CSLA:

The SCG unit of the conventional CSLA as shown in Fig.1(a), [3] is composed of two n-bit RCAs, where n is the adder bit-width. The logic operation of the n-bit RCA shown in fig.1(b) is performed in four stages:

•Half-sum generation (HSG);
•Half-carry generation (HCG);
•Full-sum generation (FSG); and
•Full carry generation (FCG).

Suppose two n-bit operands are added in the conventional CSLA, then RCA-1 and RCA-2 generate n-bit sum (s0 and s1) and output-carry (c0 out and c1 out) corresponding to input-carry (cin = 0 and cin = 1), respectively. Logic expressions of RCA-1 and RCA-2 of the SCG unit of the n-bit CSLA are given as

$$s_o^0(i)=A(i) \oplus B(i)\ \ c_o^0(i)=A(i).B(i)$$
$$s_1^0(i)= s_o^0(i) \oplus c_1^0(i-1)$$
$$c_1^0(i)=c_o^0(i)+s_o^0(i).c_1^0(i-1)\ \ c_{out}^0=c_1^0(n-1)$$
$$z_o^1(i)=A(i) \oplus B(i)\ \ c_o^1(i)=A(i).B(i)$$
$$s_1^1(i)=s_o^1(i) \oplus c_1^1(i-1)$$
$$c_1^1(i)= c_o^1(i)+s_o^1(i).c_1^1(i-1)\ \ c_{out}^1 = c_1^1(n-1)$$

..... 1

As stated above the main idea of this work is to use BEC instead of the RCA with Cin=1in 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.

B. Logic Expression of the SCG Unit of the BEC-Based CSLA

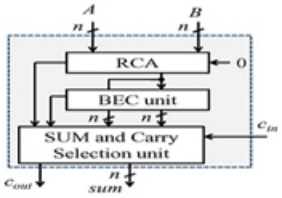

Fig.2. Structure of the BEC-based CSLA; n is the input operand bit-width.

The RCA as shown in Fig.2, calculates n-bit sum S_1^0 and C_{out}^0 corresponding to cin = 0. The BEC unit receives S_1^0 and C_{out}^0 from the RCA and generates (n + 1) bit excess-1 code. The most significant bit (MSB) of BEC represents c1 out, in which n least significant bits (LSBs) represent S_1^1. The logic expressions

$$s_1^1(i)= s_o^0(0)\ \ c_1^1(0)= s_1^0(0)$$
$$s_1^1(i)=s_1^0(i) \oplus c_1^1(i-1)$$
$$c_1^1(i)=s_1^0(i).c_1^1(i-1)$$
$$c_{out}^1=c_1^0(n-1) \oplus c_1^1(n-1) \text{..... 2}$$

The selected carry word is added with the half-sum (s0) to generate the final-sum (s). Using this method, one can have three design advantages:

1.Calculation of S_1^0 is avoided in the SCG unit;
2.The n-bit select unit is required instead of the (n+1) bit; and
3.Small output-carry delay.

All these features result in an area–delay and energy-efficient design for the CSLA.We have removed all the redundant logic operations of 2 and rearranged logic expressions of 2 based on their dependence. The proposed logic formulation for the CSLA is given as

$$s_o(i)=A(i) \oplus B(i)\ \ c_o^0(i)=A(i).B(i)$$
$$c_1^0(i)=c_1^0(i-1).s_o^0(i)+c_o(i)\ \ \text{for } c_1^0(0)=0$$
$$c_1^0(i)=c_0^1(i-1).s_o^0(i)+c_o(i)\ \ \text{for } c_1^0(0)=1$$
$$c(i)=c_1^0(i)\ \text{if}(c_{in}=0)$$
$$c(i)=c_1^1(i)\text{if}(c_{in}=1) \text{.....3}$$

III. PROPOSED ADDER DESIGN:

The proposed CSLA is based on the logic formulation given in equ.4, and its structure is shown in Fig. 3(a)where n is the input operand bit-width, and [*] represents delay (in the unit of inverter delay), n = max (t, 3.5n + 2.7).

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

It consists of one HSG unit, one FSG unit, one CG unit, and one CS unit. The CG unit is composed of two CGs (CG0 and CG1) corresponding to input-carry '0' and '1'. The HSG receives two n-bit operands (A and B) and generate half-sum word s0 and half-carry word c0 of width n bits each. Both CG0 and CG1 receive s0 and c0 from the HSG unit and generate two n-bit full-carry words c0 1 and c11 corresponding to input-carry '0' and '1', respectively. The logic diagram of the HSG unit is shown in Fig.3 (b). The logic circuits of CG0 and CG1 are optimized to take advantage of the fixed input-carry bits. The optimized designs of CG0 and CG1 are shown in Fig. 3(c) and (d), respectively.

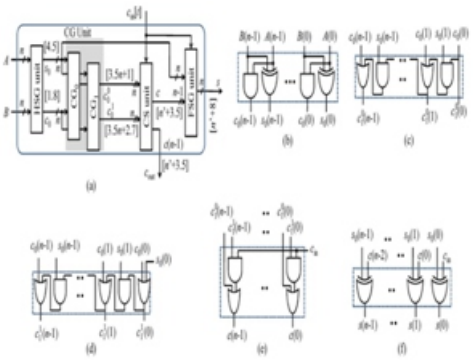

Fig.3(a). Proposed CS adder design,(b) Gate-level design of the HSG. (c) Gate-level optimized design of (CG0) for input-carry = 0. (d) Gate-level optimized design of (CG1) for input-carry = 1. (e) Gate-level design of the CS unit. (f) Gate-level design of the final-sum generation (FSG) unit.The CS unit selects one final carry word from the two carry words available at its input line using the control signal cin. It selects C_1^0 when cin = 0; otherwise, it selects C_1^1. The CS unit can be implemented using an n-bit 2-to-l MUX. However, we find from the truth table of the CS unit that carry words c0 1 and c11 follow a specific bit pattern. If C_1^0 (i) = '1', then C_1^1 (i) = 1, irrespective of s0(i) and c0(i), for $0 \le i \le n-1$. This feature is used for logic optimization of the CS unit. The optimized design of the CS unit is shown in Fig. 3(e), which is composed of n AND–OR gates. The final carry word c is obtained from the CS unit. The MSB of c is sent to output as cout, and (n − 1) LSBs are XORed with (n − 1) MSBs of half-sum (s0) in the FSG [shown in Fig. 3(f)] to obtain (n − 1) MSBs of final-sum (s). The LSB of s0 is XORed with cin to obtain the LSB of s.We have considered all the gates to be made of 2-input AND, 2-input OR, and inverter (AOI). A 2-input XOR is composed of 2 AND, 1 OR, and 2 NOT gates. The area and delay of the 2-input AND, 2-input OR, and NOT gates (shown in Table I) are taken from the Synopsys Armenia Educational Department (SAED) 90-nm standard cell library datasheet for theoretical estimation. The area and delay of a design are calculated using the following relations:

$$A = a \cdot N_a + r \cdot N_o + i - N_i$$
$$T = n_a \cdot T_a + n_o \cdot T_o + n_j \cdot T_i \quad \text{..... 4}$$

Where (Na, No, Ni) and (na, no, ni), respectively, represent the (AND, OR, NOT) gate counts of the total design and its critical path. (a, r, i) and (Ta, To, Ti), respectively, represent the area and delay of one (AND, OR, NOT) gate. We have calculated the (AOI) gate counts of each design for area and delay estimation the area and delay of each design are calculated from the AOI gate counts (Na, No, Ni), (na, no, ni), and the cell details of Table I. The path of the proposed CSLA, the delay of each intermediate and output signals of the proposed n-bit CSLA design of Fig. 3 is shown in the square bracket against each signal. We can observe that the proposed n-bit single-stage CSLA adder involves 6n less number of AOI gates than the CSLA of [6] and takes 2.7 and 6.6 units less delay to calculate final-sum and output-carry. Compared with the CBL-based CSLA of [7], the proposed CSLA design involves n more AOI gates, and it takes (n − 4.7) unit less delay to calculate the output-carry.

A.EXTENSION CONCEPT OF Multistage CSLA (SQRT-CSLA)

The multipath carry propagation feature of the CSLA is fully exploited in the SQRT-CSLA [5], which is composed of a chain of CSLAs. CSLAs of increasing size are used in the SQRT-CSLA to extract the maximum concurrence in the carry propagation path. Using the SQRT-CSLA design, large-size adders are implemented with significantly less delay than a single-stage CSLA of same size. However, carry propagation delay between the CSLA stages of SQRT-CSLA is critical for the overall adder delay.

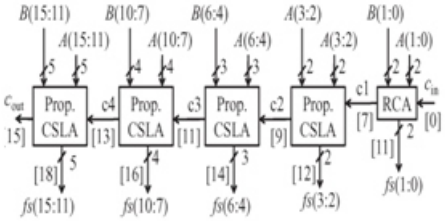

Fig.4. Proposed SQRT-CSLA for n = 16. All intermediate and output signals are labelled with delay

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Due to early generation of output-carry with multipath carry propagation feature, the proposed CSLA design is more favourable than the existing CSLA designs for area–delay[10] efficient implementation of SQRT-CSLA. A 16-bit SQRT-CSLA design using the proposed CSLA is shown in Fig. 4, where the 2-bit RCA, 2-bit CSLA, 3-bit CSLA, 4-bit CSLA, and 5-bit CSLA are used. We have considered the cascaded configuration of (2-bit RCA and 2-, 3-, 4-, 6-, 7-, and 8-bit CSLAs) and (2-bit RCA and 2-bit, 3-bit, 4-bit, 6-bit, 7-bit, 8-bit, 9-bit, 11-bit, and 12-bit CSLA's), respectively, for the 32-bit SQRTCSLA and the 64-bit SQRT-CSLA to optimize adder delay. To demonstrate the advantage of the proposed CSLA design in SQRT-CSLA, we have estimated the area and delay of SQRTCSLA using the proposed CSLA design and the BEC-based CSLA of [6] and the CBL-based CSLA of [7] for bit-widths 16, 32.

IV RESULTS& DISCUSSION:

In this section, we present the experimental results. In Section IV-A, the proposedmethod is compared with the conventional methods.We perform the simulation and synthesis and summarize the results of all the adders. The Functional verification (simulation) and synthesis (high level description is converted into RTL) of all the adders is performed and results are summarized.After the observation of simulation waveforms, synthesis is performed for calculation of delay and area and thereby the speed and power of the CSLA's are calculated and a comparison of regular, modified and improved CSLA is made in terms of delay, area and power

A. PERFORMANCE COMPARISON:

In this section, the proposed method is compared with the other 32-bit ripple carry adder.Area–Delay Estimation Method: The comparison of proposed system with 32-bit RCA is shown in Table 1.The delay can be calculated by adding up the number of gates in the longest path of logic block that contributes maximum delay.The area evolution is done by counting the total number of AOI gates required for each logic block. The main disadvantage of regular CSLA is high area usage that can be overcome by using modified CSLA.Table 1 shows the Area and delay of and, or, and not gates given in the90-nm standard cell library datasheet of proposed system compared with 32-bit RCA.Here, we show the result of power delays and critical path delays.

Table II exhibits the simulation results of both theCSLA structures in terms of delay, area and power.

Table I: path delays comparison

Adder type	No.of Success	No. of LTU	Critical Path Delay	Quisent Power (mw)	Dynamic IC power (mw)	Power Delay product terms (mw)
32 Bit ripple carry adder	46	63	61.5	203	241	14.842
Proposed ripple carry adder	71	141	47.66	203	254	12.107

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.

Table II: Comparison of the Regular and-Modified SQRT CSLA

Word size	Adder	Delay(ns)	Area	Power(uw)			Power delay product (10^{-15})	Area delay product (10^{-25})
				Leakage	switching	total		
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	1752.5
16 bit	Regular CSLA	2.775	2272	0.017	263.7	527.4	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	21848.5
64 bit	Regular CSLA	9.174	9916	0.075	1212.4	2425.0	22245.9	90969.3
	Modified CSLA	9.519	8183	0.057	1025.0	2050.1	19514.9	77893.9

Area and delay Comparison: Table III, depicts that the proposed SQRT CSLA has less number of gates and hence less area.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Table-III: Theoretical Estimation

Design	Width (n)	Delay(ns)	Area(um²)	ADP(um² us)	EADP(1%)
SQRT-CSLA [6]	16	7.38	1813.71	13.39	161.41
	32	14.58	3627.42	52.89	280.64
	64	28.98	7254.84	210.25	436.55
SQRT-CSLA (CBL) [7]	16	3.0	1706.80	5.12	---
	32	3.85	3608.98	13.89	--
	64	5.27	7435.46	39.18	--
SQRT-CSLA proposed	16	2.0	1574.54	4.10	--
	32	2.75	2989.99	11.89	---
	64	4.28	6553.24	32.10	--

B. Synthesis & Simulation Report

In this section, the proposed method synthesis and simulation results are reported.Synthesis Report:Synthesis report shows the design summary including number of LUT's, I/O buffers, Slice registers, flip flop pairs as shown in Table III.

Table -IV: Design Summary

Logic utilization	Used	Available	Utilization
Number of occupied slice	91	29504	1%
Number of 4 input LUTs	51	14752	1%
Number of slices containing only related logic	51	51	100%
Number of slices containing only un related logic	0	0	0%
Number of bonded IOBs	97	376	1%

Table-V : comparison of post layout- synthesis result

Design	Width(n)	Delay(ns)	Area(um²)	Power(uW)
SQRT-CSLA (conv)	16	5.61	2890.52	30.5673
	32	6.56	6100.34	60.2537
	64	8.37	12613.2	113.6457
SQRT-CSLA (BEC) [7]	16	5.96	2325.09	25.7858
	32	7.64	4801.33	50.0750
	64	10.18	9809.75	91.1774
SQRT-CSLA (CBL)	16	10.45	1722.96	12.8662
	32	18.72	2765.38	17.7900
	64	35.10	5530.56	30.1427
SQRT-CSLA proposed	16	5.55	1813.45	19.6652
	32	6.59	3735.36	38.1886
	64	8.35	7603.89	70.62442

V. CONCLUSION:

A simple approach is in this paper to reduce thearea and power of SQRT CSLA architecture. The reducednumber of gates of this work offers the great advantage in thereduction of area and also the total power. The comparedresults show that the modified SQRT CSLA has a slightlylarger delay, but the area and power of the 32-bmodified 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-b sizes which indicates thesuccess of the method and not a mere trade off of delay forpower and area. The modified CSLA architecture is therefore,low area, low power, simple and efficient for VLSI hardwareimplementation. It would be interesting to test the design ofthe modified SQRT CSLA.

REFERENCES:

[1] Low-Power and Area-Efficient Carry select Adder by B.Ram Kumar and Harish M Kitturin IEEE Transactions on Very Large scale Integration(VLSI) Systems, Volume 20 No.2, February 2012.

[2] An Area efficient static CMOS carry-select adder based on a compact carry look-ahead unit G.A.Ruiz, M.Granda in Micro- electronics Journal 35(2004) 939-944,2004 Elsevier Ltd.

[3] O.J.Badrij,"Carry-select Adder," IRETransaction Electronics Computers, pp 340- 344, 1962.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

[4] Y. Kim and L.S.Kim, "64-bit carry-selectadder with reduced area," Electron. Lett., vol.37, no. 10, pp. 614–615, May 2001.

[5] J.M. Rabaey,Digtal Integrated Circuits ADesign Perspective. Upper Saddle River, NJ:Prentice-Hall, 2001.

[6] Cadence, "Encounter user guide," Version6.2.4, March 2008.

[7] T.Y.Chang and M.J.Hsiao, "Carry-select adder using single ripple-carry adder," Electronics Letters, vol. 34, no. 22, pp. 2101 – 2103, Oct. 1998.

[8] Computer Arithmetic Algorithms and hardware designs by Behrooz parhami.

[9] Review on Carry Skip Adder and Gray/Black Cell Function Lecture 18 Datapath Subsystems Chapter 10 Copyright © 2005 Pearson Addison-Wesley. All rights reserved.

[10] Gray Yeap and Gilbert, Practical Lowpower Digital VLSI Design, Kluwer Academic Publishers. 1998.