Area efficient and high speed carry select adders

(Fast adders)

D Nikhil Chakravarthy and Jayasree Das
Center for VLSI Design,
Padmasri BV Raju Institute of Technology,
Hyderabad, India.
nik.mtech@gmail.com and jayasree.das@bvrit.ac.in

Balakrishna Islawath
NIT Calicut,
Kochi, Kerala,
Inida.
balu219418@gmail.com

Abstract— **Carry select adders (CSA) are the basic adders used in data processing applications. From the previous architecture, it is clear that the improvement can be done to reduce the delay and area of the CSA. The proposed architercture uses a ling adder/carry look ahead which reduces significantly the delay. The work is done in comparing the modified adders (CSLIA/CSLA) with regular carry select adder (CSA) of 32-bit, 64-bit. This work evaluates the performance in terms of area, delay, power with a logical effort and layout in 90 nm CMOS process technology.**

Keywords- delay, CSLIA, CSLA, area

I. INTRODUCTION

Digital processers require fast adders for their operation the carry select adder consists of ripple carry adders (RCA) which are a group of full adders. In the adders each Ripple carry adder block must wait for the carry from the Previous block thus making the ripple carry block to wait for the carry from the previous block , in the carry select adder each RCA block calculates the sum with predetermined carry i.e at cin =0,cin=1and multiplexed which is as in the section II in fig 2(a), the binary to excess 1 converter (BEC) is the excess one value of the binary input and this is shown in the fig 2(c), the input given to the BEC is the output of the ripple carry adder.

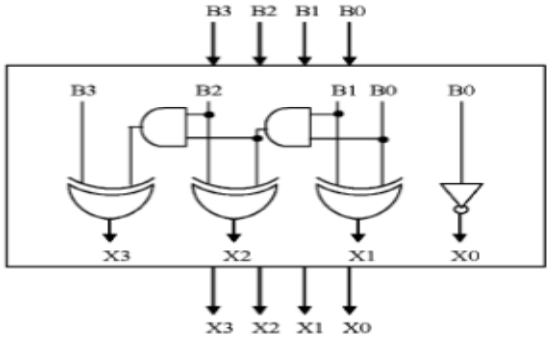
The modified carry select adders are carry select ling adder (CSLIA) where RCA is replaced with ling adder and the other is carry select look ahead adder (CLSA) where RCA is replaced with carry look ahead adder expalined in which are shown in the fig 2.(b)

The brief is explained as follows section III is about the funtioning of Ling adder and its Operation is explained, section IV is about the functioning of carry look ahead adder, section V which is about asic implementation, section VI is about the waveforms section VII is about the area, delay evaluation and are done in 32-bit and 64-bit adders and preceeding section is about and conclusion.

II. DESIGN METHODOLOGY AND ARCHITECTURE

A regular carry select adder shown in fig 2.1(a) consits of ripple carry adder which were the basic blocks of CSLA each ripple carry adder consists of a group of full adders which are responsible for carry and sum generation. The input to the BEC is the output of the ripple carry adder so; there is some delay for the input to appear at the BEC. It is important to reduce the output delay of RCA which is achieved by palcing other adders (ling adder or carry look ahead) instead of RCA.

The fig 2.1(b) shows the modified carry select adder (CSLIA/CSLA), which has their own advantages one has an increase in speed and other has a reduction in area and delay product. The work is done in comparing the area and delay of the regular carry select adder (CSA) with modified carry select adders (CSLIA/CSLA)



.

Fig 2(c). Binary to excess one converter (BEC)

X0= ~ B0
X1= B0 ^ B1
X2= B2 ^ (B0&B1)
X3= B3 ^ (B2&B1&B0)
Carry=B3& (B2&B1&B0)

The carry select adder is that in which prede-termined calculations for sum are done for Cin =0 and Cin=1 and the sum is multiplexed based on the previous carry the architecture of the regular carry select adder is modified inorder to reduce the delay of the sum generated by RCA blocks .The overall delay is reduced by

Reducing delay of the RCA which is done by replacing it with parllel-preffix ling adder and also carry look ahead

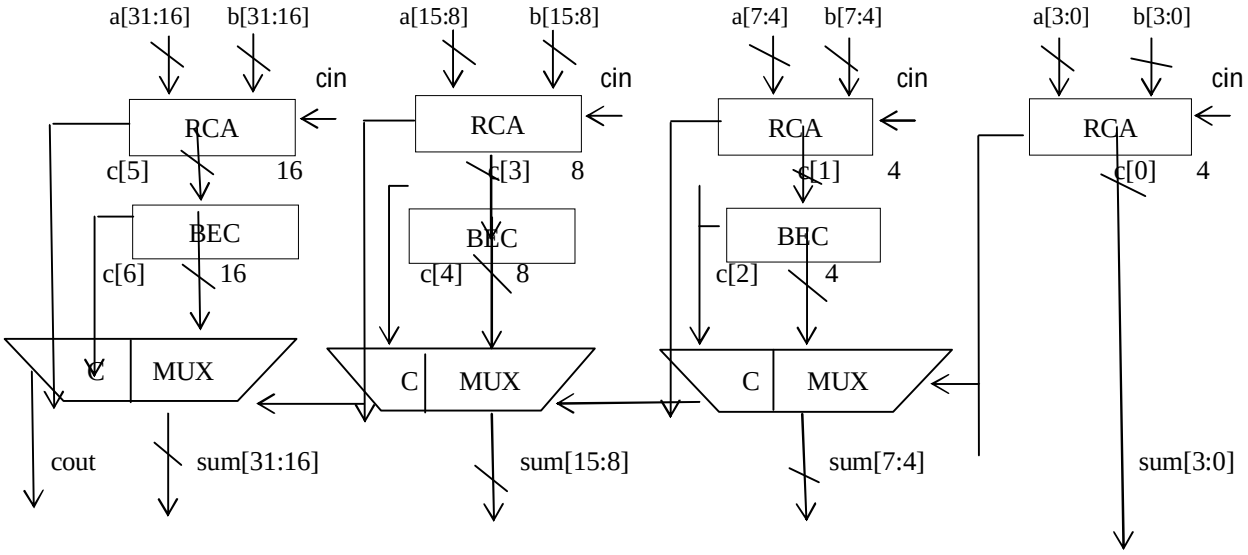
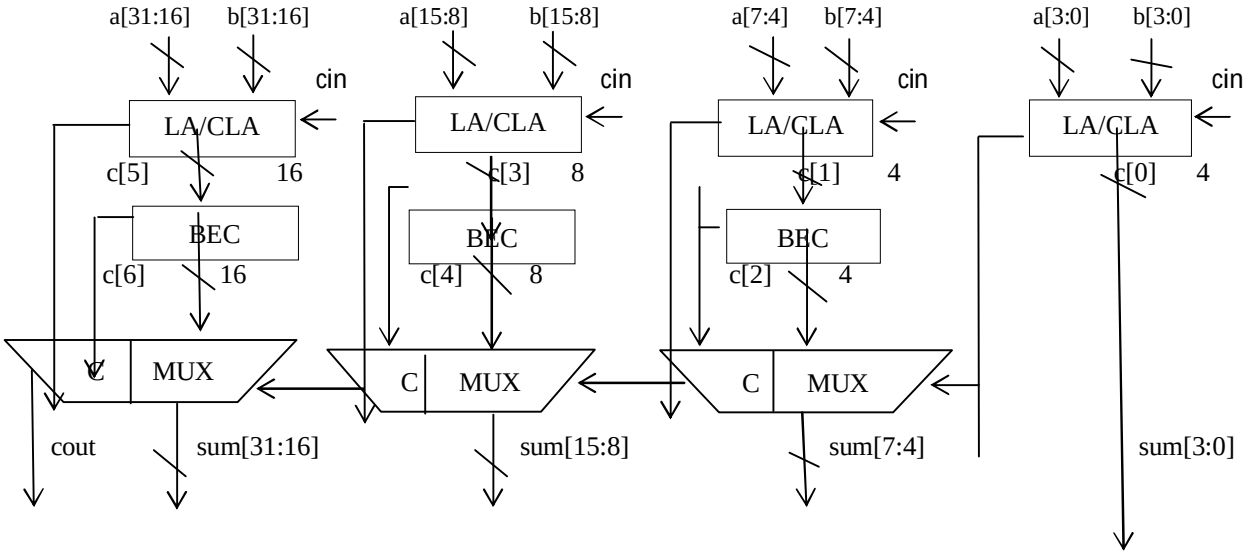


Fig 2(a) . Regular carry select adder 32-bit (CSLA)



Figs 2(b) modified carry select adder (CSLIA/CSLA) with LING ADDER (LA) / CARRY LOOK AHEAD ADDER (CLA)

adder which increases the speed compared to CSA. The carry generated by each block is placed as a select line of the multiplexer for the next stage and it select's the sum based on the select line .For both the Ling adder and carry look ahead adder.

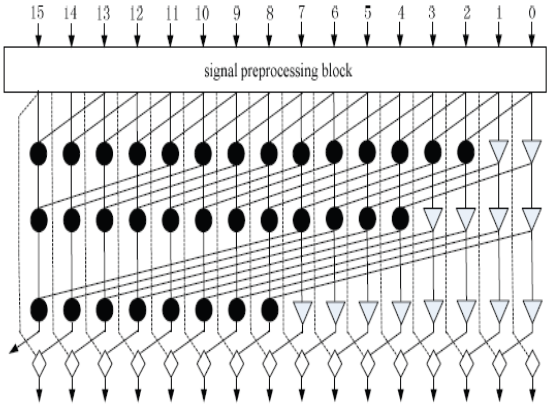
Ling adder has a tree type structure which uses the logrithmic calculations for its fanout and the number of the stages in the operation .Both the adders (LA and CLA) depends on the generate carry (G) and propogate (P) and also the number of stages in the CSLA are predetermined block, sum generator block and carry generator.

III. LING ADDER

Ling adder was first proposed by Huey Ling. It is an adder which improves the speed of traditional parallel-prefix carry signal calculation. A parallel prefix Ling adder can be considered as three blocks: the preprocessing block, the propagation block and the sum generation block. In the preprocessing block, the definitions of preprocessing signals are:

$$G_i = a_i \cdot b_i \;,\; P_i = a_i + b_i,\; D_i = a_i \wedge b_i$$

Assume that there are two n bit operands $A=a_{n-1}\, a_{n-2}\ldots\ldots\ldots\ldots a_0$ and $B = b_{n-1}b_{n-2}\ldots\ldots\ldots\ldots b_0$ representing two numbers are to be added and $S=s_{n-1}s_{n-2}\ldots\ldots\ldots\ldots s_0$ denotes the sum for the i^{th} bit ($0 \le i \le n-1$) the carry generate G_i and P_i are denoted by



3(a) Ling adder tree diagram

$$H_I = G_i + G_{i-1} + P_{i-1}.G_{i-2} + \ldots\ldots + P_{i-1}P_{i-2}\ldots\ldots P_1G_0$$

$$\text{Sum} = \overline{H_{i-1}} \cdot D_i + H_{i-1} \cdot (D_i \wedge P_{i-1})$$

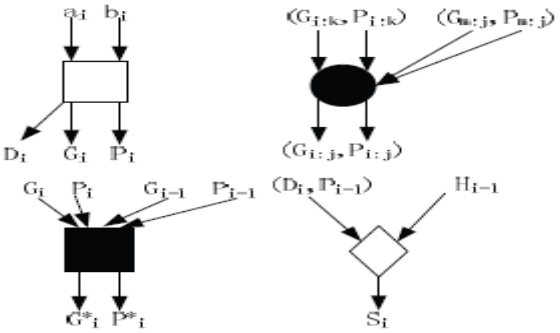
$$C_i = G_i + P_i C_{i-1}$$

In this model, the generate and propogate terms are given by the equations.which are calculated in the PG block.

$$G_i^* = \overline{\overline{G_i}\ \overline{G_{i-1}}}$$

$$P_i^* = \overline{\overline{P_i} + \overline{P_{i-1}}}$$

By using these equations the speed of carry generation is increased by moderate increase in area.



3(b). Ling adder blocks

IV. CARRY LOOK AHEAD ADDER

In this carry look ahead stucture the propogate (P) and the generate (G) computations are calculated to increase the speed of carry computation these two signals are given as

$$P_i = a_i \wedge b_i$$

$$G_i = a_i . b_i$$

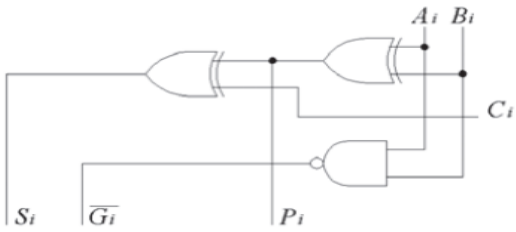


Fig 4(a) carry look ahead adder structure

The carry and sum calculations of the carry look ahead adder are given as

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

$$\text{Sum} = a_i \wedge b_i \wedge c_i$$

The carry look ahead adder reduces the delay and is one of the fast adders in now a day it serves the easy way of carry generation with a slight increase in area compared to ripple carry adder and so it serves a little increase in the chip area compared with ripple carry adders

V. ASIC IMPLEMENTATION

The design proposed in this paper was developed using verilog HDL and is synthezised using the cadace RTL compiler using design libraries of 90nm technology. The synthesized netlist and there respective designs constraints file are imported to cadance (soc) to generate placement and routing in the standard cell.

VI. WAVEFORMS

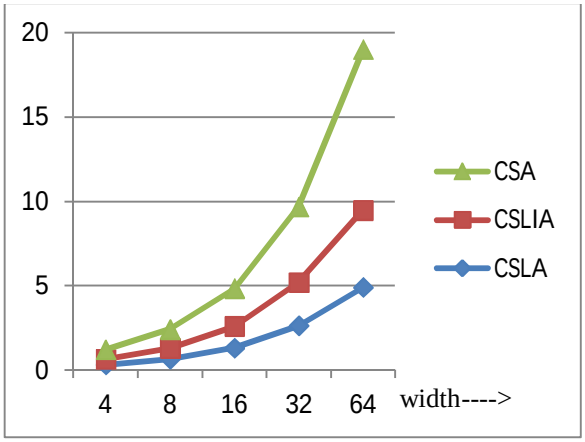


Fig 5(a). Width vs Delay curves

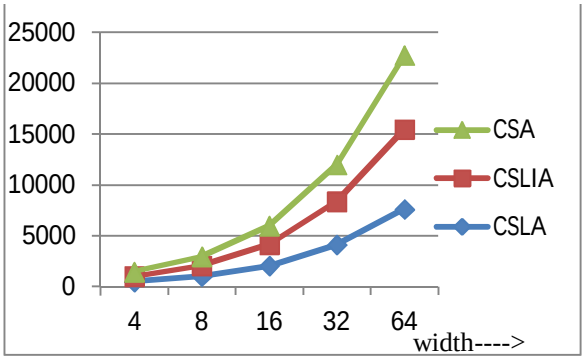


Fig 5(b). Width vs area curves

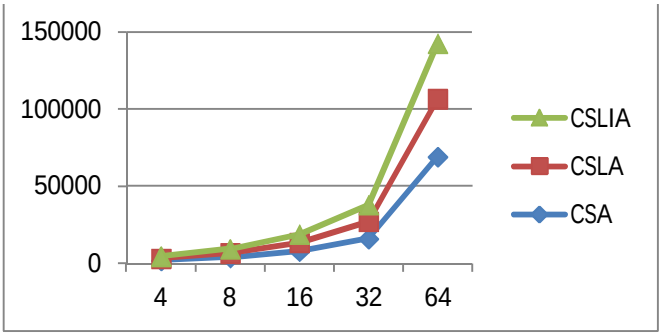


Fig 5(c). Width vs product (area-delay)

From the graph it is clear that the delay of the CSA is increasing as the width is increasing whereas the area of all the adders are increasing as the width is increasing. CSA takes less amount of area compared to all the adders, whereas the product of delay and area is less for CSLIA and CSLA compared with CSA

VII. Delay and Area Evaluation

The delay and area evaluation is calculated and tabulated for 32-bit, 64-bit using carry select ling adder (CSLA) and carry look ahead adder are shown in the TABLE1. We get high speed adder by using CSLIA and moderate reduction in delay by using CSLA in both 32-bit, 64-bit .The ling adder provides better performance as the bit size increases so there is a large amount of reduction in delay in a 32-bit and 64-bit CSLIA. CSA takes less area but the product of area and delay is reduced for CSLIA and CSLA compared with CSA.

VIII. Conclusion

The design provides a 42% reduction in delay in CSLIA compared to REGULAR CSA and is faster than all the carry select adders and the CSLA shows 40% reduction in delay compared REGULAR CSA. As the bit size increases for higher order bits of 64-bit the CSLIA shows upto 52% reductuction in delay compared to CSA and CSLA shows 48% reduction in delay compared with regular CSA .Thus as the bit size is increasing both CSLIA and CSLA shows better performance than CSA. Area is increased by 16% in CSLA and the area is increased to 14% in CSLIA but the product of area and delay is reduced in CSLIA and CSLA compared to CSA.

Name of the adder	WIDTH	DELAY (ns)	AREA(μm^2)	POWER(µw)			Area-delay product (10^{-21})
				LEKAGE POWER	DYNAMIC POWER	TOTAL POWER	
CSA	32-Bits	4.482	3626	0.0019	198.48	198.481	16251.73
CSLIA		2.562	4168	0.0047	520.97	520.974	11016.02
CSLA		2.643	4241	0.0040	520.14	520.144	10865.44
CSA	64-Bits	9.539	7264	0.0047	406.52	406.524	69291.296
CSLIA		4.56	8353	0.0086	1086.32	1086.328	41013.23
CSLA		4.91	8428	0.0089	1115.26	1152.268	38431.68

References

[1] B. Ramkumar and Harish M Kittur ,"low power and area efficient carry select adder," *IEEE Trans. Very large scale integrations (VLSI) pp* .1-5, 2011

[2] Yu-Ting Pai and Yu-Kumg Chen, "The fastest carry look ahead adder," *IEEE trans* . International Workshop on Electron-ic Design, Test and Applications, 2004

[3] Dayu Wang, Xiaoping Cui, Xiaojing Wang, "Optimized design of Parallel Prefix Ling Adder,"*IEEE* Trans. *pp*.941-944

[4] B.Bhaskar, M.Kanagasabapathy,V.S.Kanchana Bhaaskaran, "A hybrid adiabatic parllel prefi addition scheme for low pow-er", *IEEE Trans* , International Conference on Recent Trends in Information Technology, *pp.389-393 , june 2011*

[5] Dilip P. vasudevan, parag k.lala, james patrik parkerson, " Self-checking carry-select adder design based on two rail en-coding",*IEEE Trans*, CIRCUITS AND SYSTEMS—I , *Decem-ber 2007*