# Area and Power Efficiency of Carry Select Adder Using Gate Diffusion Input (GDI) Logic

K. SANGEETHA [1], S. KAYALVIZHI [2]
[1, 2] *Department of ECE, Government college of Technology, Coimbatore*

***Abstract -- In this paper an area and power efficient design of 8 bit Carry Select Adder (CSLA) has been proposed. Conventional and other CSLAs are designed using CMOS technology, which has complexity in terms of area and power consumption. So to overcome this problem a new technology is implemented on the CSLA. This paper shows the implementation and comparison of Carry Select Adder (CSA) using BEC (Binary Excess one Converter) and First Zero Finding (FZF) logic implementation techniques with optimization of GDI Logic by minimize number of transistors. All the designs are implemented 1.8Volt power supply and 180nm technology in Cadence Virtuoso environment.***

***Keywords -- Carry Select Adder (CSLA); GDI (Gate Diffusion Input); RCA (Ripple Carry Adder); MUX (Multiplexer); BEC (Binary to Excess one Convertor); FZF (First Zero Finding).***

## I. INTRODUCTION

In recent years a large amount of the research effort has been taken to improve the system performance such as ALU, FIR Filters, FFT implementation etc. However the backbone of a digital system is an adder block. Thus improvement of the adder block will lead to the improvement of the system as a whole without any change in the function of the architecture of the system. In the current VLSI industry optimization of both power and area are of prime importance since the demand of the Consumer is not only restricted to smaller size of the devices but also higher battery life. . The CSLA is the fastest adder among all the adder. However, due to the use of an extra Ripple Carry Adder (RCA) as the second stage and a MUX as a final, hence the area consumed by such a CSLA design is very huge, thus not only occupies a large area but also increases the power dissipation of the overall circuit, which can be a point of concern so as to make a device more and more portable. Hence in order to avoid the extra area and power consumption by the CSLA an intelligent design scheme was given in which gives the idea of using FZF logic circuit to eliminate the second stage RCA and final stage MUX in a traditional CSLA design for front end implementation.

This paper is organized as follows, in section II, we explain the architecture of Regular CSLA, in section III the Architecture of CSLA using BEC logic, in section IV the Architecture of CSLA using FZF logic, in section V Implementation of Gate Diffusion Input (GDI) logic for CSLA using FZF, in section VI Simulation and result analysis are given, The section VII gives conclusion.

## II. CONVENTIONAL CSLA

CSLA is one of the fastest adder since it previously calculates the sum and carry with carry '0' or carry '1' and by using the '2x1' Multiplexer (MUX) we select the output based on the previous carry. This carry select adder (CSLA) consists of two stages of Ripple Carry Adder (RCA) and Multiplexer (MUX). Adding two n-bit numbers with a carry select adder is done with two adders (Therefore two ripple carry adders), in order to perform the calculation twice, one time with the assumption of the carry being zero and the other assumption it will be one. After the two results are calculated, the correct sum as well as correct carry-out, is then selected with the multiplexer once if the previous carry-in is known. The below architecture shows the conventional carry select adder for 4 bits.

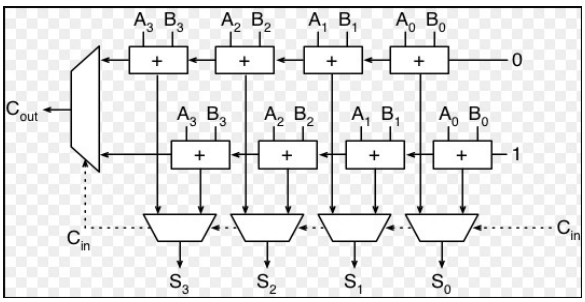


Fig .1: Block diagram of 4-bit carry select adder

## III. CARRY SELECT ADDER USING BEC LOGIC

First, a traditional CSLA will consist of the following major blocks that are first stage RCA, second stage RCA and final stage multiplexers for the final selection of the sum bit either for input carry being 0 or 1. In a unique design is given that gives implementation of CSLA with elimination of the second stage RCA. It uses a BEC logic block to perform the operation provided by the RCA as in Fig 1. When analyzed the second stage of a CSLA is nothing but a RCA with input carry as 1, thus any circuit that gives add one operation can be placed in place of the second stage RCA to achieve the desired operation. The operation is achieved by using BEC logic.

TABLE 2: Binary to Excess one Convertor

| BINARY NUMBER B3 B2 B1 B0 | EXCESS-1 LOGIC X3 X2 X1 X0 |
|---|---|
| 0000 | 0001 |
| 0001 | 0010 |
| 0010 | 0011 |
| 0011 | 0100 |
| 0100 | 0101 |
| 0101 | 0110 |
| 0110 | 0111 |
| 0111 | 1000 |
| 1000 | 1001 |
| 1001 | 1010 |
| 1010 | 1011 |
| 1011 | 1100 |
| 1100 | 1101 |
| 1101 | 1110 |
| 1110 | 1111 |
| 1111 | 0000 |

BOOLEAN EXPRESSION OF BEC LOGIC

$$X0 = B0$$
$$X1 = B0 \oplus B1$$
$$X2 = B2 \oplus (B0 \& B1)$$
$$X3 = B3 \oplus (B0 \& B1 \& B2)$$

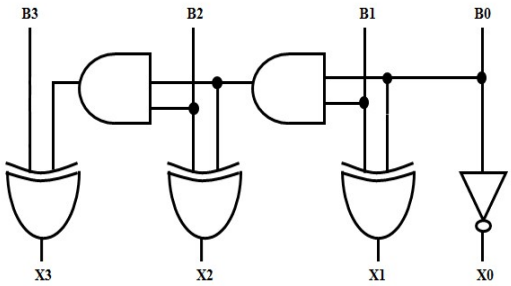


Fig. 2: Logic diagram of 4-bit BEC

The following figure shows the architecture of Carry Select Adder using Binary to Excess-1 Convertor.

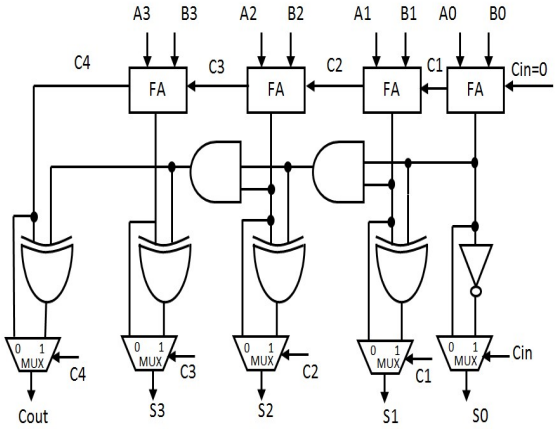


Fig .3: Architecture of CSLA with BEC

## IV. CARRY SELECT ADDER USING FZF

The architecture of a MUX Free CSA uses the advantage of a First Zero Finding (FZF) logic circuit. The operation of the CSA using FZF logic varies on $C_{in}$ being 1 and 0, Where $C_{in}$ is the carry of the previous stage. If $C_{in}$=0 the logic values of the output of RCA will be passed as such as the final output of CSLA. Now if $C_{in}$ =1 logic circuit will invert all the output bits of RCA from LSB to MSB until it encounters the first zero in the sequence.

The First Zero Finding Logic is nothing but the cascade of half adder Circuit. This Architecture eliminates the disadvantages of Carry Select Adder using BEC

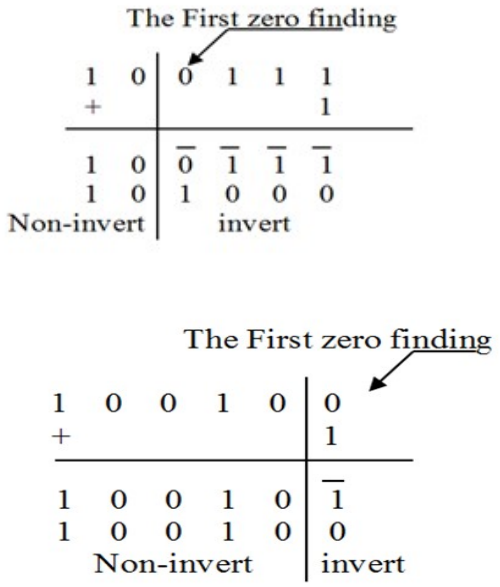


Fig .3: First Zero Finding Logic

The following figure shows the architecture of carry select adder using First Zero Finding (FZF) Logic.

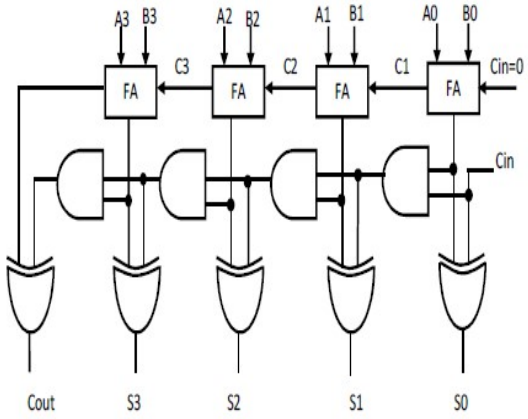


Fig .4: Architecture of CSLA with FZF

## V. PROPOSED CARRY SELECT ADDER

The CSLA using First Zero Finding logic is implemented using GDI logic to further reduction of area and power

The rapid development of portable digital applications, the demand for increasing speed, compact implementation, and low power dissipation triggers numerous research efforts. The wish to improve the performance of logic circuits, once based on traditional CMOS technology, resulted in the development of many logic design techniques during the last two decades. One form of logic that is popular in low-power digital circuits is pass-transistor logic (PTL).From the above two technology, there is the further reduction of transistor count by using GDI logic. The GDI approach allows implementation of a wide range of complex logic functions using only two transistors. This method is suitable for design of fast, low-power circuits, using a reduced number of transistors (as compared to CMOS and existing PTL techniques)

The GDI method is based on the use of a simple cell as shown in Fig. 6. At first glance, the basic cell reminds one of the standard CMOS inverter, but there are some important differences.

1) The GDI cell contains three inputs: G(common gate input of nMOS and pMOS), P (input to the source/drain of pMOS), and N (input to the source/drain of nMOS).

2) Bulks of both nMOS and pMOS are connected to N or P (respectively), so it can be arbitrarily biased at contrast with a CMOS inverter.

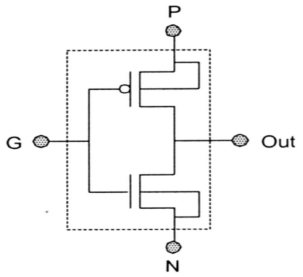


Fig .5: Basic GDI cell

The different combinational logic can be implemented using the GDI cell. The following table shows the different functions implementation using GDI logic.

Table II: Different functions implantation using GDI cell

| N | P | G | Out | Function |
|---|---|---|---|---|
| '0' | $B$ | $A$ | $\overline{A}B$ | F1 |
| $B$ | '1' | $A$ | $\overline{A}+B$ | F2 |
| '1' | $B$ | $A$ | $A+B$ | OR |
| $B$ | '0' | $A$ | $AB$ | AND |
| $C$ | $B$ | $A$ | $\overline{A}B+AC$ | MUX |
| '0' | '1' | $A$ | $\overline{A}$ | NOT |

## VI. RESULTS AND DISCUSSIONS

The simulations of all the adders have been performed using CADENCE 180nm technology. The parameters like power and area have been compared. From the Table III we can see that there is a drastic decrease in power consumption as well as the complexity in area. The power for 1-bit is reduced by 42.7% and 99% when compared with conventional CSLA and BEC based CSLA respectively while the power for 8-bit is reduced by 93.9% and 93.4% when compared with conventional CSLA and BEC based CSLA respectively.

TABLE III: Comparison of adder's power and Area

| TYPE OF ADDER | Power (uWatt) | | No. of Transistors | |
|---|---|---|---|---|
| | 1 Bit | 8 Bit | 1 Bit | 8 Bit |
| Conventional CSLA | 0.178 | 128 | 28 | 368 |
| BEC based CSLA | 11.57 | 117.6 | 76 | 440 |
| Proposed Adder | 0.102 | 7.7 | 14 | 112 |

## VII. CONCLUSION

On comparing the simulation results of the adders that have been calculated on the basis of power as well as the number of transistors in use we can conclude by saying that the Proposed CSLA that uses the GDI Technique is the most optimal and efficient. This is in comparison with the conventional CSLA and the BEC based CSLA. The conventional CSLA consumes a lot of power and also has a complexity in its area. The number of transistors used is also high. So, we then implemented the BEC based CSLA which consumed lesser power than the latter, but only by a smaller margin. There is complexity in this circuit too. Hence we implemented a adder using FZF in which the basic logic gates were designed using the GDI Technique which helped in reducing the power consumption drastically. The proposed design reduces the complexity in area as well as the power consumption. The power for 1-bit is reduced by 42.7% and 99% when compared with conventional CSLA and BEC based CSLA respectively and the power for 8-bit is reduced by 93.9% and 93.4% when compared with conventional CSLA and BEC based CSLA respectively. The numbers of transistors used in the circuit are also reduced due to the GDI technique. Hence, we can say that this proposed design of CSLA is area and power efficient and can be implemented in various circuits like multipliers which would need a lot of computations. All these circuits have been implemented using the CADENCE CMOS 180nm Technology.

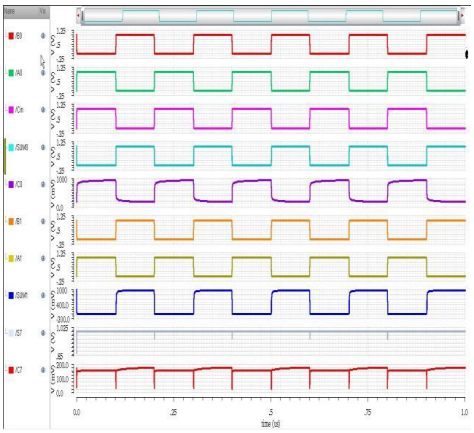

Fig. 6: proposed CSLA transient response and power

## REFERENCES

[1] B. Ramkumar and Harish M Kittur, "Low – Power and Area-Efficient Carry Select Adder", IEEE TRANSACTIONS ON VLSI SYSTEMS, Vol. 20, February 2012.

[2] Basant Kumar Mohanty and Sujit Kumar Patel, "Area–Delay–Power Efficient Carry-Select

Adder", IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: EXPRESS BRIEFS, Vol. 61, No. 6, JUNE 2014

[3] S. A. Mashankar, R. D. Kadam and P. R. Indurkar, "Power Efficient Carry Select Adder using D-Latch", IRJET, Vol. 02, Issue: 03, June 2015

[4] O. J. Bedrij, "Carryselect adder," IRE Trans. Electron, Comput., vol. EC-11, no. 3, pp. 340-344,Jun. 1962

[5] Edison A.J and C.S Manikandababu, "An Area Efficient CSLA Architecture for VLSI Hardware Implementation", IJMIE, Vol. 2, Issue 5, May 2012

[6] Prashanth Gurjar, Rashmi Solanki, Pooja Kansliwal and Mahendra Vucha, "VLSI Implementation of *Adders* for High Speed ALU", International Journal of Computer Applications, Volume 10 – No 29, September 2011

[7] Arkadiy Morgenshtein, Alexander Fish and Israel A. Wagner, "Gate-Diffusion Input (GDI): A Power-Efficient Method for Digital Combinational Circuits", IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS, Vol. 10, No. 5, October 2002