



Comparative Analysis for different models of Carry Select Adders

M.Soundharya, M.E Student & R.Arunkumar, Assistant Professor
Department of Electronics and Communication Engineering,
Karpagam College of Engineering,
Coimbatore

ABSTRACT

In this Era, the signal processing system is widely used. In the digital signal processing system, adder is the basic building block of all the major circuits. Today the requirements for minimizing the delay, area, and power of adder circuit improve the efficiency of whole system which drives the technology to the next level. Even though the Carry Select Adder (CSLA) occupies more area, it is been used instead of ripple carry adder to avoid propagation delay. In other models Binary to Excess-1 Converter (BEC) based Carry Select Adder was also used which uses less number of logic resources than conventional CSLA. But these CSLAs are not more efficient because it rejects one sum after the calculation. So the delay was not more effectively reduced. In order to overcome this problem the reduced logic CSLA is used. But by using Gate Diffusion Input (GDI) Technique can give less delay than this recently proposed reduced logic CSLA. The proposed technique provides low power consumption, less propagation delay. By using this GDI based CSLA the number of transistors required for the circuit also minimized. So an efficient adder design can be achieved through this technique.

I. INTRODUCTION

Today's modern electronics requires the design with high speed, low power, less area for VLSI circuits. The design with reduced delay, area, power will be very useful for portable and mobile devices. Reducing the delay of basic adder circuit will increase the efficiency of the entire system. But achieving these constraints are not an easy task. So based on the applications these constraints are modified to get an efficient design. Digital signal processing system has so many adders. So the efficient adder design is an important concern. There are several designs for the adder circuit. Ripple Carry Adder (RCA) uses the simple design but it has the main drawback of Carry Propagation Delay [2]. The delay will be linearly increases with the bit width of the numbers which are going to be taken for the addition process. But by using the

carry select adder the propagation delay is effectively reduced. Even though it has less delay, the design is not more attractive because of the

usage of dual RCAs, which means that the speed will be obtained with the penalty of area. In order to overcome this drawback the BEC based CSLA is used. It replaces one RCA with one add-one circuit. Hence the area will be saved. The BEC based CSLA uses less number of logic resources [2] but the routing delay is slightly higher than the conventional CSLA. The another one model for adder circuit design is Common Boolean Logic (CBL) which uses less number of logic resources and consumes low power but it has longer Carry Propagation Delay (CPD). The delay of CBL based CSLA is proportional to the bit width of the numbers [3] which are going to be add. To reduce the CPD of this adder CBL based SQR CSLA was proposed. SQR CSLAs are implemented for large bit-width [5], [6] adders with less delay. This CBL based SQR CSLA has longer delay than the BEC based SQR CSLA. From the analysis of Conventional and BEC based CSLAs, the data dependence and redundant logic operations are identified. The reduced logic CSLA overcomes the drawbacks of Conventional and BEC based CSLAs. The logic formulation of reduced logic CSLA is done based on the data dependence. The reduced logic SQR CSLA can have the ability to give optimized delay, area, and power than this recently proposed reduced logic CSLA.

II. EXISTING SYSTEM

A. Ripple Carry Adder

Ripple carry adder is designed by cascading the full adders in series. The full adder here is



responsible for adding two numbers with carry [2]. The carry out of first stage is given as the carry in of the next stage.

Ripple Carry Adder uses the compact Design. Even though it uses the simple adder design to add unrestricted bit length numbers, it is not more widely used because of its Carry Propagation Delay (CPD).

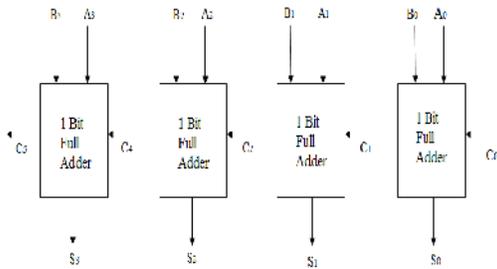


Figure 1: Ripple Carry Adder

The worst case delay of RCA is given by

$$t = (n - 1) tc + ts$$

Where,

tc – Delay for Carry

ts – Delay for Sum

From the above equation it is clear that, the delay of Ripple Carry adder linearly increases with the bit width of the numbers.

B. Conventional Carry Select Adder

In order to reduce the propagation delay, the Carry Select Adder is widely used as the adder circuit of any arithmetic unit. The Carry Select Adder contains the two main units such as the Sum and Carry Generation Unit (SCG) and the Sum and Carry selection Unit (SCS). Among these two units, the Sum and Carry Generation unit consumes more logic resources of CSLA. So the design of SCG unit is the main concern to reduce the logic resources. The SCG unit of four bit CSLA contains two Ripple Carry Adders (RCA). The Sum and Carry selection unit of four bit CSLA contains a multiplexers. It performs two additions in parallel, one assuming Carry-in zero, and the other carry-in is one. After the calculations of two additions one

sum is selected according to the carry in the multiplexer.

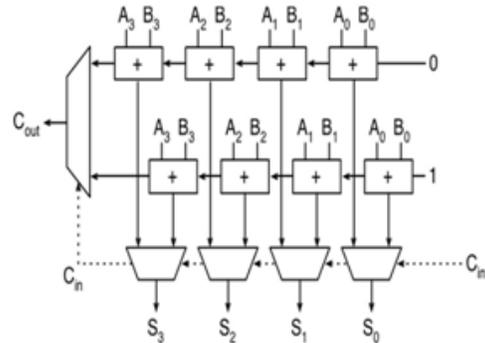


Figure 2: Conventional Carry Select Adder

Even though it reduces the propagation delay, the design is not attractive due to the usage of dual RCAs. Because if we use dual RCAs the area of the adder design will increase this is the main drawback of conventional CSLA.

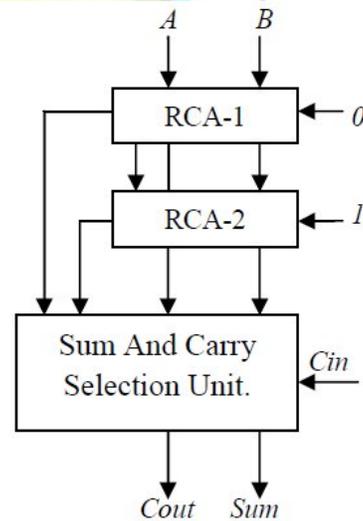


Figure 3: Conventional CSLA

In the above diagram, it is shown that a conventional CSLA has two RCAs. The n bit RCA performs the logical operations in four stages such as Half Sum Generation (HSG), Half Carry Generation (HCG), Full Sum Generation (FSG), and Full Carry Generation (FCG).

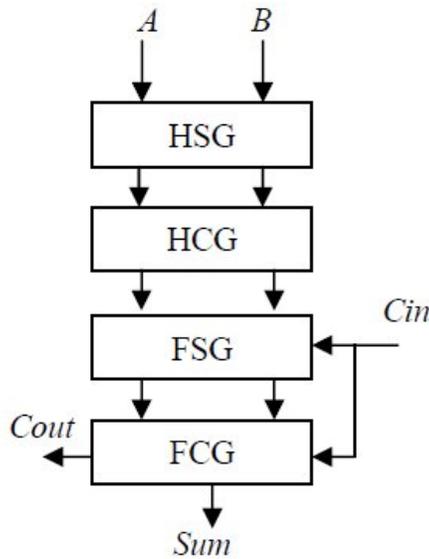


Figure 4: Logical operations of one RCA

The logical expressions of the two Ripple Carry Adders of the Conventional CSLA is given as

$$S_0^0(i) = A(i) \oplus B(i) \quad C_0^0(i) = A(i).B(i) \quad \dots(1a)$$

$$S_1^0(i) = S_0^0(i) \oplus C_1^0(i-1) \quad \dots(1b)$$

$$C_1^0(i) = C_0^0(i) + S_0^0(i).C_1^0(i-1) \quad C_{out}^0 = C_1^0(n-1) \quad \dots(1c)$$

$$S01(i) = A(i) \oplus B(i) \quad C01(i) = A(i).B(i) \quad \dots(2a)$$

$$S_1^1(i) = S_0^1(i) \oplus C_1^1(i-1) \quad \dots(2b)$$

$$C_1^1(i) = C_1^0(i) + S_0^1(i).C_1^1(i-1) \quad C_{out}^1 = C_1^1(n-1) \quad \dots(2c)$$

C. BEC Based CSLA

In order to reduce the area and power of the adder circuit the BEC based CSLA is used. The BEC based CSLA design is same as the Conventional CSLA design. But the only change is instead of using two RCAs it replaces one RCA with one add-one circuit [3], [7].

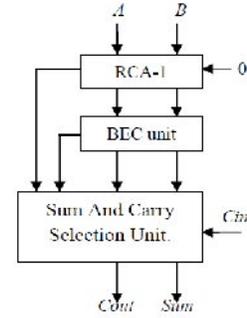


Figure 5: BEC Based CSLA

From the above figure 5, RCA calculates n bit sum and carry. The BEC unit receives the sum and carry generated by the RCA and produces (n+1) bit excess-1 code. The most significant bit of BEC represents carry and the least significant bits represent the sum. The logical expressions of the RCA are same as the expressions of conventional CSLA. The expressions for BEC unit are given as

$$S_1^1(i) = \sim[S_0^1(i)] \quad C_1^1(0) = S_1^0(0) \quad \dots(3a)$$

$$S_1^1(i) = S_0^1(i) \oplus C_1^0(i-1) \quad \dots(3b)$$

$$C_1^1(i) = S_1^0(i) . C_1^1(i-1) \quad \dots(3c)$$

$$C_{out}^1 = C_1^0(n-1) \oplus C_1^1(n-1) \quad \dots(3d)$$

For $1 < i < n-1$

From the expressions of conventional and BEC based CSLAs, it is known that the BEC based CSLA increases the data dependence of the circuit. The logic resources used by BEC based CSLA are less but the routing delay is slightly higher than the conventional CSLA.

D. Reduced Logic CSLA

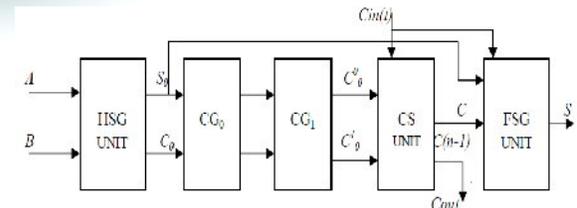


Figure 6: Reduced logic CSLA

The reduced logic CSLA has one Half Sum Generation (HSG) unit, one Full Sum Generation (FSG), one Carry Generation (CG) unit, and one



Carry Select (CS) unit The carry generation unit is constructed by two CGs (CG₀ and CG₁). Initially the n bit operands (A & B) are given as the input to the HSG unit. The HSG unit generates half sum and half carry. Both CG₀ and CG₁ receive the sum and carry generated by HSG. The CG₀ and CG₁ units have the optimized designs for fixed input carry bits. The Carry Selection unit selects the carry among the two carries based on the control signal. According to the selected carry the FSG unit generates the full sum word. The optimized design structures for CG₀ and CG₁ units are given as in the following

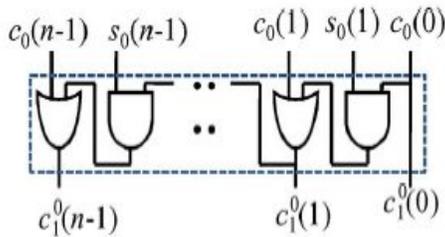


Figure 7: Optimized design of CG₀ unit

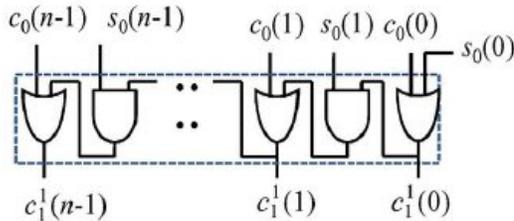


Fig 8: Optimized design of CG₁ unit

The designs of HSG unit, CS unit, and FSG unit are given as

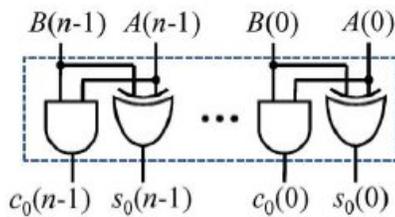


Figure 9: Design of HSG unit

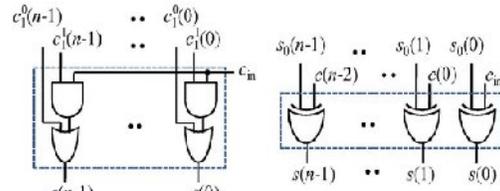


Figure 10: Design of the CS unit

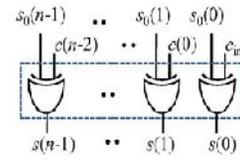


Figure 11: Design of the FSG unit

The logical expressions of reduced logic CSLA are

$$S_0(i) = A(i) \oplus B(i) \text{ and } C_0(i) = A(i) \cdot B(i) \dots (4a)$$

$$C_1^0(i) = C_1^0(i-1) \cdot s_0(i) + C_0(i) \text{ for } (C_1^0(0) = 0) \dots (4b)$$

$$C_1^1(i) = C_1^1(i-1) \cdot s_0(i) + C_0(i) \text{ for } (C_1^1(0) = 1) \dots (4c)$$

$$C(i) = C_1^0(i) \text{ if } C_{in} = 0$$

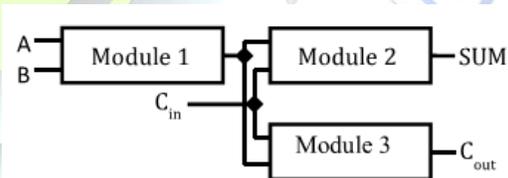
$$C(i) = C_1^1(i) \text{ if } C_{in} = 1$$

$$C_{out} = C(n-1)$$

$$S(0) = S_0(0) \oplus C_{in} \text{ and } S(i) = S_0(i) \oplus C(i-1)$$

III. GDI BASED CARRY SELECT ADDER:

The main objective of the GDI based CSLA is to improve the speed, power, and transistor count of the Carry select adder [1] when compared to the existing ones.

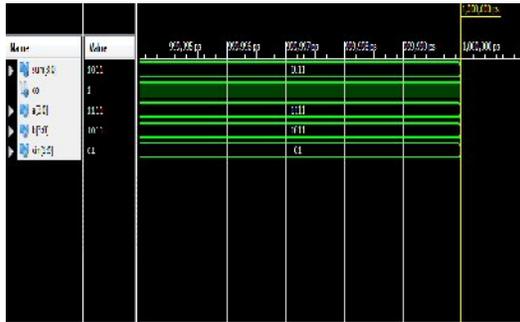


Module 1 and module 2 are the XNOR modules that generate the sum signal (SUM) and module 3 generates the output carry signal (C_{out}). Each module is designed individually such that the entire adder circuit is optimized in terms of power, delay, and area.



SIMULATION RESULTS:

OUTPUT WAVEFORM OF CSLA:



The above waveform gives the output of carry select adder. For this output waveform the synthesis report and RTL diagrams are given below in order to analyse the power, delay and area parameters of the adder circuit. From the Conventional CSLA synthesis report we can know that the power used by the adder circuit is 9.069 ns and the number of logical devices used in the circuit is high. From the BEC Based CSLA synthesis report we can know that the power used by the circuit is 10.294 and the logical resources used in the circuit is very less when compared to the previous conventional CSLA. But the delay is slightly higher than the conventional CSLA because of the routing delay.

SYNTHESIS REPORT OF BEC BASED CSLA:

```

Timing constraint: Default path analysis
Total number of paths / destination ports: 85 / 5

Delay: 10.294ns (Levels of Logic = 7)
Source: a<0> (PAD)
Destination: cout (PAD)

Data Path: a<0> to cout

Cell:in->out fanout Gate Delay Net Logical Name (Net Name)
-----
IBUF:I->O 6 1.218 0.748 a_0_IBUF (a_0_IBUF)
LUT4:I1->O 2 0.704 0.482 X2/Garry1 (C2)
LUT4:I2->O 3 0.704 0.566 x3/bxor_sum_xo<0>1 (s3)
LUT4:I2->O 2 0.704 0.451 x9/w77 (x9/w77)
LUT4:I3->O 1 0.704 0.000 x0/w93_F (N14)
MUXF5:I0->O 1 0.321 0.420 x9/w93 (cout_OBUF)
OBUF:I->O 3.272 cout_OBUF (cout)

Total 10.294ns (7.627ns logic, 2.667ns route)
(74.1% logic, 25.9% route)
    
```

SYNTHESIS REPORT OF CONVENTIONAL CSLA :

```

Timing constraint: Default path analysis
Total number of paths / destination ports: 49 / 5

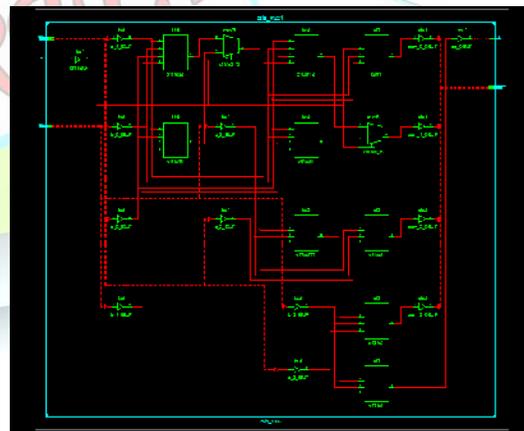
Delay: 9.069ns (Levels of Logic = 5)
Source: b<1> (PAD)
Destination: co (PAD)

Data Path: b<1> to co

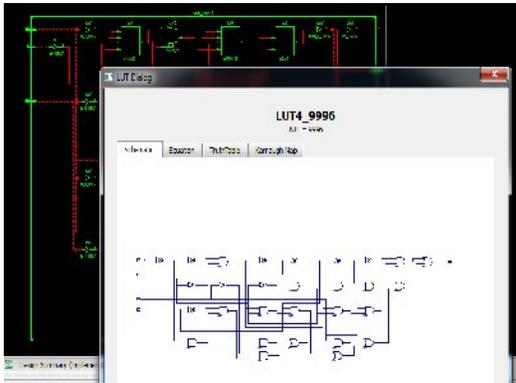
Cell:in->out fanout Gate Delay Net Logical Name (Net Name)
-----
IBUF:I->O 4 1.218 0.762 b_1_IBUF (b_1_IBUF)
LUT4:I0->O 1 0.704 0.000 x11/w31 (x11/w3)
MUXF5:I1->O 2 0.321 0.482 x11/w3_#5 (N10)
LUT4:I2->O 2 0.704 0.482 x12/w133 (N2)
LUT4:I2->O 1 0.704 0.420 x13/w1 (co_OBUF)
OBUF:I->O 3.272 co_OBUF (co)

Total 9.069ns (6.923ns logic, 2.146ns route)
(76.3% logic, 23.7% route)
    
```

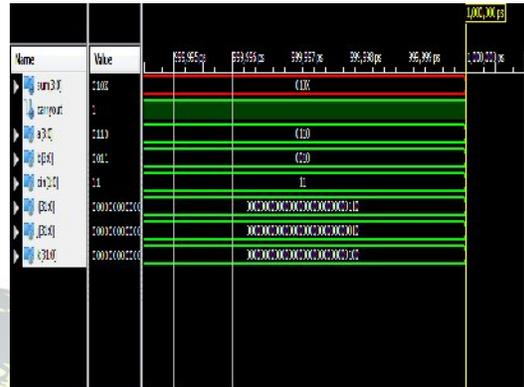
RTL DIAGRAM OF CONVENTIONAL CSLA:



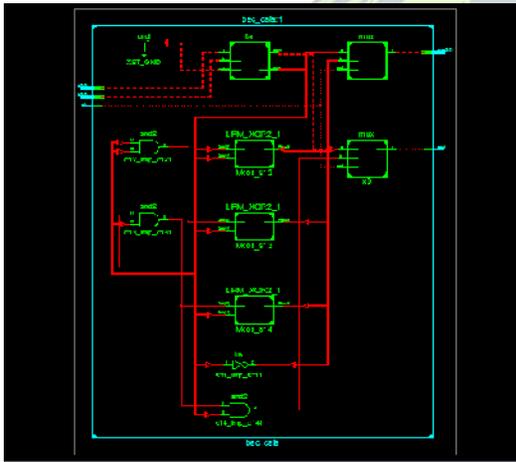
GATE LEVEL DIAGRAM OF ONE LUT IN CONVENTIONAL CSLA :



OUTPUT WAVEFORM OF REDUCED LOGIC CSLA:



RTL DIAGRAM OF BEC BASED CSLA :



SYNTHESIS REPORT OF REDUCED LOGIC CSLA :

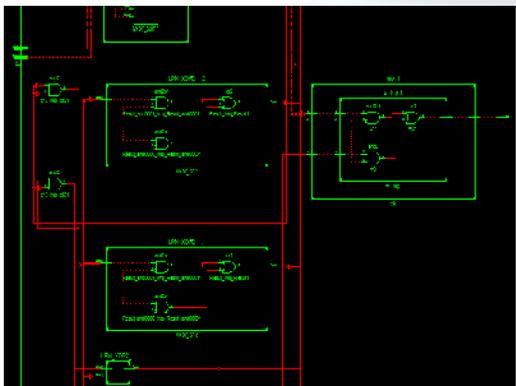
Timing constraint: Default path analysis
Total number of paths / destination ports: 15 / 4

Delay: 7.716ns (Levels of Logic = 5)
Source: a<1> (PAD)
Destination: carryout (EAD)

Data Path: a<1> to carryout

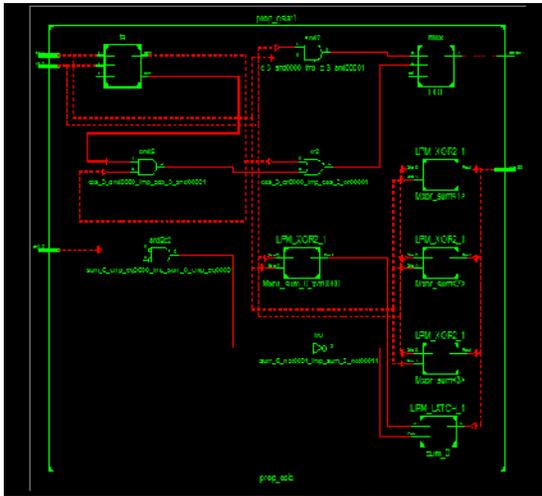
Cell:in->out	Fanout	Gate Delay	Net Delay	Logical Name (Net Name)
IBUF:I->O	2	1.218	0.622	a_1_IBUF (a_1_IBUF)
LUT4:I0->O	1	0.704	0.455	DOT/w47_SWO_SWO (N7)
LUT3:I2->O	1	0.704	0.000	DOT/w471 (DOT/w47)
MUXF5:I1->O	1	0.321	0.420	DOT/w47_f5 (carryout_OBUF)
OBUF:I->O		3.272		carryout_OBUF (carryout)
Total		7.716ns	(6.219ns logic, 1.497ns route)	(80.6% logic, 19.4% route)

GATE LEVEL DIAGRAM OF ONE LUT IN BEC BASED CSLA:



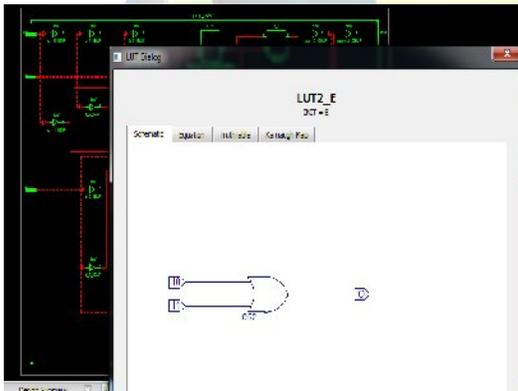
GATE LEVEL DIAGRAM OF ONE LUT IN REDUCED LOGIC CSLA:





From the synthesis report of reduced logic CSLA, it is known that the power used by the adder circuit is 7.716 ns which is very much less than the power used by conventional and BEC Based CSLAs. And the logic resources also much more reduced than the previous Conventional and BEC Based CSLAs.

GATE LEVEL DIAGRAM FOR ONE LUT IN REDUCED LOGIC CSLA:



REFERENCES

- [1]. Arkadiy Morgenshtein, Alexander Fish, Israel A. Wagner, "Gate Diffusion Input (GDI): A Power Efficient Method for Digital Combinatorial Circuits", IEEE Transactions on very large scale Integration Systems, Vol.10, No.5, 2002.
- [2]. Padma Devi, Ashima Girdher, Balwinder Singh, "Improved Carry Select Adder with reduced area and Low power Consumption", International Journal of Computer Applications, Vol.3, No.4, 2010.
- [3]. K.Allipeera, S. Ahmed Basha, "An Efficient 64-bit Carry Select Adder with less delay and reduced area application", International Journal of Engineering and Research applications", Vol.2, Issue 5, 2012.
- [4]. R.Uma, Vidya Vijayan, M.Mohanapriya, Sharon Paul, "Area Delay Power Comparison of Adder topologies", International Journal of VLSI design & Communication systems (VLSICS), Vol.3, No.1, 2012.
- [5]. M.Chithra, G.Omkareswari, "128-Bit Carry Select Adder having Less Area and Delay", International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, Vol.2, Issue 7, 2013.
- [6]. Damarla Paradhasaradhi, K. Anusudha, "An Area Efficient Enhanced Sqrt Carry Select Adder", International Journal of Engineering Research and Applications, Vol.3, Issue 6, 2013.
- [7]. K.Mariya Priyadarshini, N.V.N.Ravi Kiran, N.Tejasri, T.C.Venkat Anish, "Design of Area and Speed efficient Square Root Carry Select Adder using fast Adders", International Journal of Scientific & Technology Research, Vol.3, Issue.6, 2014.
- [8]. Kalapriya.K, KSN Raju, "Carry Select Adder using BEC and RCA", International journal of Advanced research in Computer and Communication Engineering, Vol.3, Issue.10, 2014.