Introduction:

Power consumption is an important efficeinecy factor in designing very large scale integrated circuit (VLSI). With the explosive growth of VLSI technology the demand and popularity of portable devices has driving designers to reduce the chip size and make them consume less power.

Adder are the central electronic circuit used for addition and are fundamenta for wide variety of digital system. There exists many adders with fast adding but Low area and Power is still challenging.

Techniques Used

1. Regular Carry Select Adder (CSLA)
2. BEC based CSLA
3. CBL based CSLA

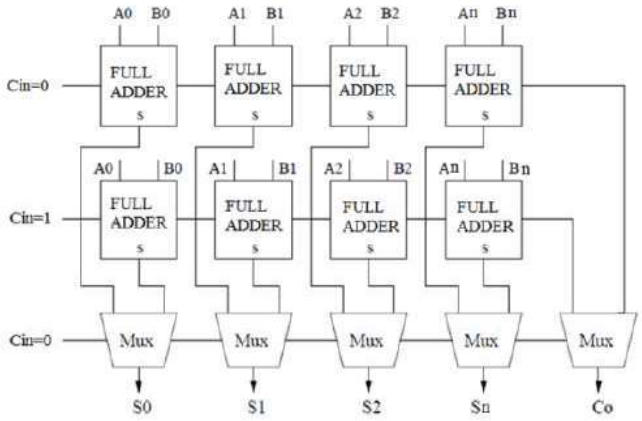
Tool used:

DSCH – Microwind (v.3.5) – Logic editor and simulator. Used to validate the architecture of the logic circuit.

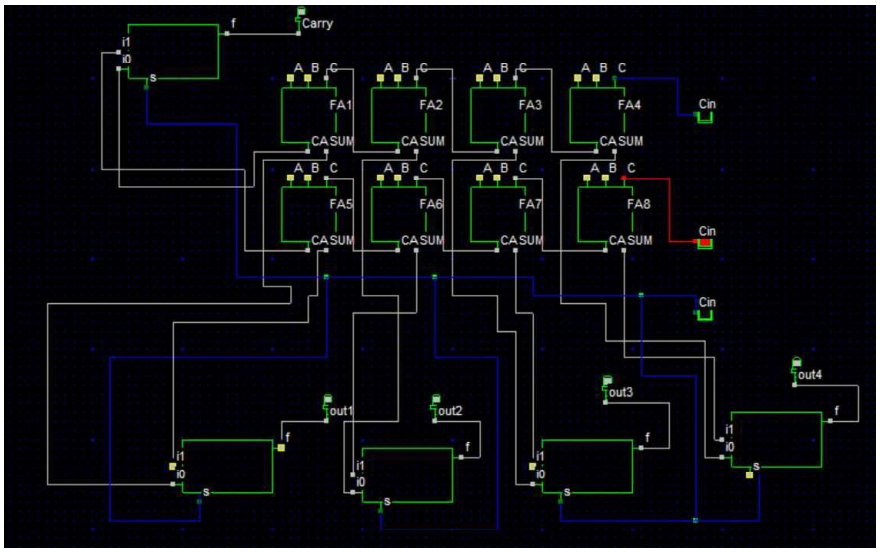
**ModelSim - ** Multi-language environment by Mentor Graphics, for simulation of hardware description language such as VHDL, Verilog and SystemC.

Circuit diagram :

I. Carry Select Adder



(i) Circuit of CSLA

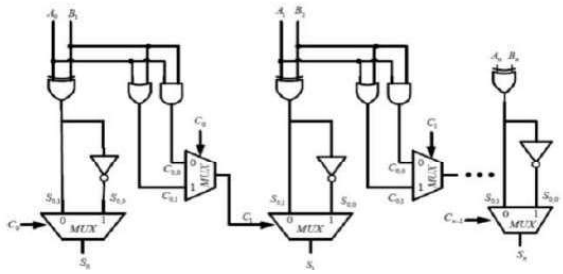


(ii) Schematic Circuit

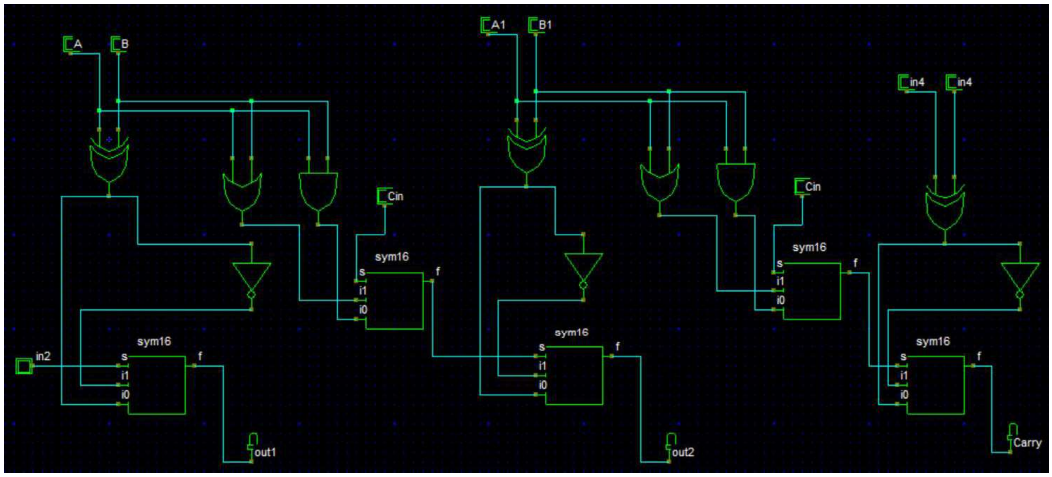
A CSLA breaks the addition problem into smaller groups. It is one of the fast adder. Each uit implements addition operation in parallel. Two carry are set as 0 and 1. It generally consists of two Ripple Carry Adder (RCA) and a multiplexer. Additio of two n-bit number is done using two adders.

Devices Used in Simulation: Full-Adders (FA) , Ripple Carry Adders (RCA), 2x1 MUX

II. CBL based CSLA



(i) Circuit Diagram

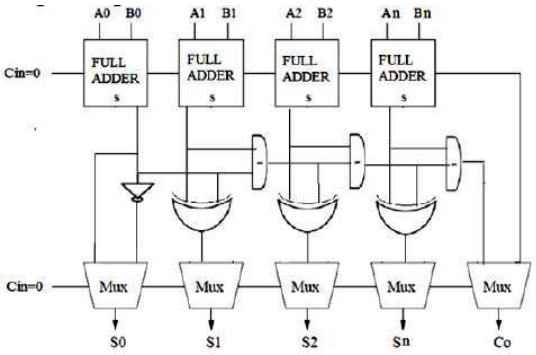


(ii) Schematic Circuit

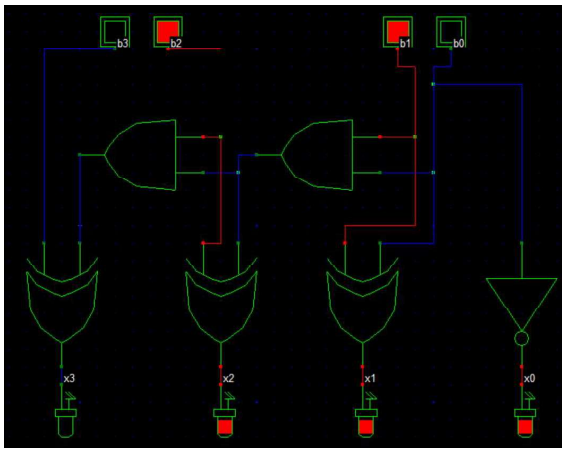
Common Boolean Logic is the sub area of algebra in which the values of the variables in the truth table are True and False. To share the CBL term, we implemented one XOR gate with INV gate to generate the summation signal pair.
And for carry propagation path, one OR gate and one AND gate is implemented to anticipate possible carry values in advance.

Devices Used in Simulations: XOR, 2x1 MUX, (AND and OR for Carry generation)

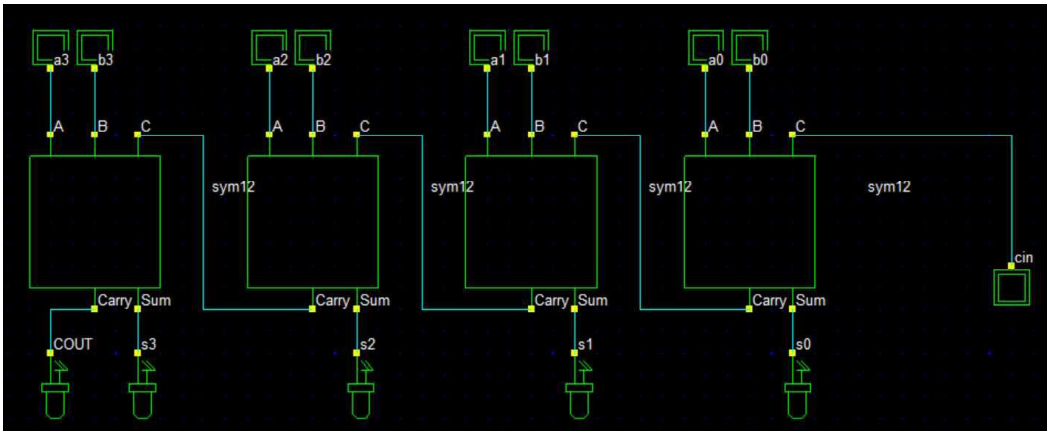
III. BEC based CSLA



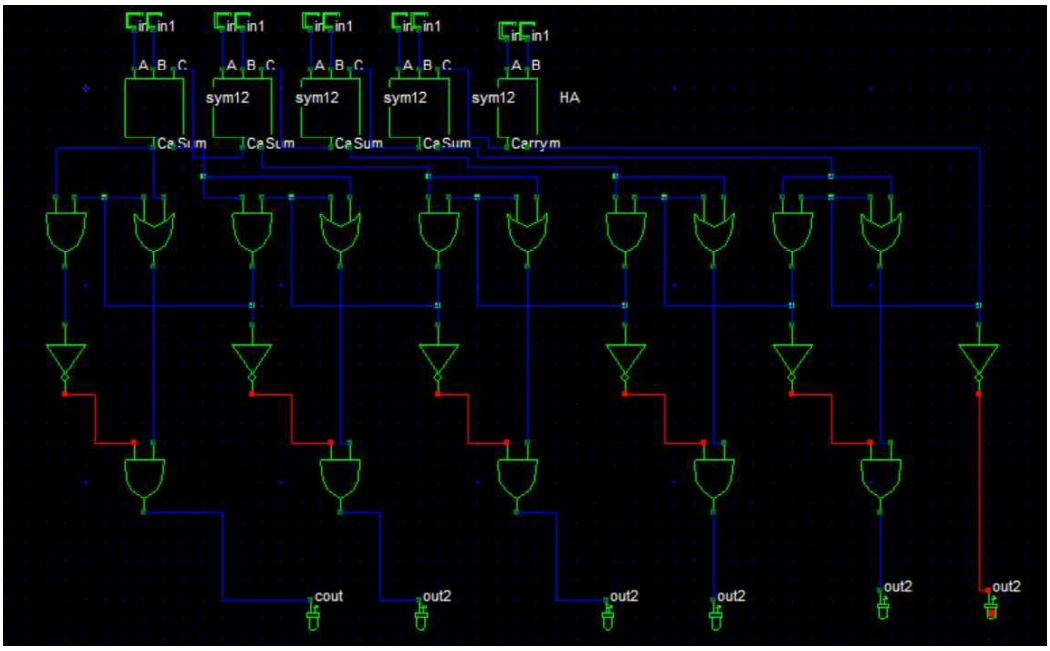
(i) Circuit Diagram of BEC based CSA



(ii) BEC-1 Schematic Diagram



(iii) RCA Schematic Diagram



(iv) Schematic Diagram

In this CSLA, the RCA which assume carry '1' is replaced using BEC (Binary to EXCESS-1 Convertor). In every FA cell in RCA, for every carry-in a carry-out is generated. Then we anticipate both possible values of carry in and evaluate the result for both possibility in advance. Once the correct value of carry is known the correct value is selected using multiplexer.

Devices Used in Simulations: BEC, RCA, 2x1 MUX

<u>Simulation Results</u>:

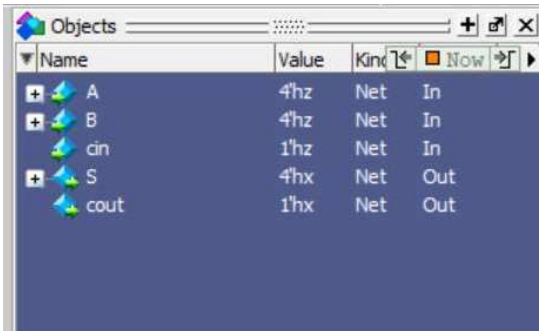
Generated Verilog files using DSCH and with minor changes we were able to compile those files in ModelSim and simulated the same, calculating delay time and wave diagram.

Objects then created are added to wave and given binary values as input.

```
# //
# Loading project cc
# Compile of adder.v was successful.
# Compile of csla.v was successful.
# Compile of mux2.v was successful.
# Compile of testbench2.v was successful.
# 4 compiles, 0 failed with no errors.

ModelSim>
```

(i) Compile transcript



(ii) Objects and input

(iii) Wave output

Cursor is then placed after every output. Delay calculated was **100ns.**

<u>Calculations and Results</u>:

The proposed design of CSLA is successfully tested and simulated in Microwind and ModelSim. Result table is given below, containing gate count (area count), power and delay. The total power consumed by the circuit is sum of leakage power, internal power and switching power.
Area or Gate count represent total cell area design.

BIT SIZE	ADDER	GATE COUNT	POWER (mV)
4	CSLA	136	0.867

4	BEC based CSLA	94	0.417
4	CBL based CSLA	88	0.388

Conclusion:

We discussed and implemented efficient approaches to reduce area and power of CSLA architecture. The compared results shows that CBL based CSLA has low power and area consumption. Also BEC based CSLA showed better results than Regular CSLA and very close to CBL based CSLA.
These both approaches can be used for low power, area and efficient VLSI hardware implementation.

Appendix:

```
module carry_select_adder
    (   input [3:0] A,B,
        input cin,
        output [3:0] S,
        output cout
        );


wire [3:0] temp0,temp1,carry0,carry1;

//for carry 0
fulladder fa00(A[0],B[0],1'b0,temp0[0],carry0[0]);
fulladder fa01(A[1],B[1],carry0[0],temp0[1],carry0[1]);
fulladder fa02(A[2],B[2],carry0[1],temp0[2],carry0[2]);
fulladder fa03(A[3],B[3],carry0[2],temp0[3],carry0[3]);

//for carry 1
fulladder fa10(A[0],B[0],1'b1,temp1[0],carry1[0]);
fulladder fa11(A[1],B[1],carry1[0],temp1[1],carry1[1]);
fulladder fa12(A[2],B[2],carry1[1],temp1[2],carry1[2]);
fulladder fa13(A[3],B[3],carry1[2],temp1[3],carry1[3]);

//mux for carry
multiplexer2 mux_carry(carry0[3],carry1[3],cin,cout);
//mux's for sum
multiplexer2 mux_sum0(temp0[0],temp1[0],cin,S[0]);
multiplexer2 mux_sum1(temp0[1],temp1[1],cin,S[1]);
multiplexer2 mux_sum2(temp0[2],temp1[2],cin,S[2]);
multiplexer2 mux_sum3(temp0[3],temp1[3],cin,S[3]);

endmodule
```

References:

I. Design and implementation of High Speed Carry Select Adder (IRJET 2015) by B.Gopinath1, N.Sangeetha2, S.Jenifer nancy3 and T.UmaraniAsst. Professor Dr. SJS Paul Memorial College of Engineering & Technology,Puducherry.

II. Analysis of Low Power High-S Carry Multi Adder (IRJET 2020) by M. Vinoth1, S. Selvakumar2, J. Vinothkumar3
1, 2, 3Assistant Professor SCSVMV, Kancheepuram, India ,234Student Dr. SJS Paul Memorial College of Engineering & Technology,Pondicherry.

III. An Efficient SQRT Architecture of Carry Select Adder Design by Common Boolean Logic.
Ms. S.Manjui, Mr. V. Sornagopae
IpG Scholar, Department of ECE, 2Assistant Professor, 1,2 Abdul Hakeem College of Engineering and Technology, Melvishram, Vellore.

IV. MODIFIED ENERGY AND AREA EFFICIENT CARRY SELECT ADDER USING BEC ON A RECONFIGURABLE HARDWARE (International Journal of Scientific & Engineering Research, Volume 6, Issue 9, September-2015) by JSSATEN Noida