INTERNATIONAL JOURNAL OF RESEARCH
In Advanced Engineering Technologies

Volume 3, Issue 2 OCT 2014

DESIGN OF LOW POWER ALU USING CARRY SELECT ADDER

M. Sandeep reddy [(1)] Katragadda Rajesh [(2)]

P.G.Scholor(VLSI System Design), CMR Institute of Technology,Hyderabad[(1)]

M.Tech, Asst.Professor, ECE Dept, CMR Institute of Technology,Hyderabad[(2)]

ABSTRACT— Most of the VLSI applications, such as DSP, image and video processing, and microprocessors use carry select adder(CSLA) for arithmetic functions. One of the existing solutions used in SQRT CSLA is replacement of second level RCA by BEC. Though increases the performance, very less percentage of improvement in reduction of area and power dissipation. And also the existing adder with BEC technique is not suitable for low power applications. Hence this paper proposes Special Hardware using Multiplexers (SHM) design in place of second level RCA. It is observed from the results that the area and power dissipation are reduced at comparable percentages with respect to the RCA and BEC techniques. In this paper we proposed an ALU that consists of Multiplexers adder and Logical Blocks like AND, OR, XNOR. Two 4X1multiplexers and one 2X1 multiplexer are used along with 8T full adder or 10T full adder using GDI technique. An 8-bit ALU consists of carry select adders to reduce the ripple in design, and square root carry select adder is used in place of linear carry select adder to reduce propagation delay. The final 8 bit ALU can be designed with square root carry select adder and sub blocks with GDI technique can reduce the power consumption by reducing the no of transistors. This proposed logic is designed in transistor level using 0.12µm technology in the Micro wind tool.

Key Words: Cary Select Adder, Regular SQRT CSLA**,** GDI, SHM, ALU

I. INTRODUCTION

Design of any Low power VLSI circuit with less area and high speed has become a main concern for digital designers. Building low power VLSI systems has emerged as highly in demand because of the fast growing technology in mobile communications and computation. The battery technology does not advance at the same rate as microelectronics technology. There is a limited amount of power available for the mobile systems. So designers are faced with more constraints such as high speed, high throughput, small silicon area, and at the same time, low power consumption. So building low power, high performance adder cells are of great interest [1]-[5]. To reduce the power and area requirements of the computational complexities, the size of transistors are shrunk into the deep submicron region [6] and predominantly handled by process engineering.

There are several Adder designs have been proposed to reduce the power consumption. Logic minimization not only results in better system throughput but also results in low power consumption designs. In digital adders, for speeds up the operation Ripple Carry Adder (RCA) is modified as CSLA. To achieve more speed CSLA is replaces by SQRT CSLA.

The arithmetic logic unit (ALU) is one of the main components inside a microprocessor. It is responsible for performing arithmetic and logic operations such as addition, subtraction, increment, decrement, logical AND, logical OR logical XOR and logical XNOR. An ALU is a digital circuit that performs arithmetic and logical operations. We have designed ALU using 4Xl mux, 2Xl mux and an 8T full adder. Here all the blocks in ALU are designed using Gate Diffusion Input (GDI).
This brief is structured as follows. Section II presents the detailed structure and the function of the BEC logic. The SQRT CSLA has been chosen for comparison with the proposed design as it has a more balanced delay, and requires lower power and area. The delay and area evaluation methodology of the regular and modified SQRT CSLA (SHM) are presented. Section III explains about GDI technology. Sections IV describes implementation of proposed ALU using efficient carry select adder. The ASIC implementation details and results are analyzed in Section V. Finally, the work is concluded in Section VI.

II. SQRT CSLA

In digital adders, for speed up the operation Ripple Carry Adder (RCA) is modified as CSLA. To achieve more speed

CSLA is replaced by SQRT CSLA. The CSLA is used in many computational systems to alleviate the problem of carry propagation delay by independently generating multiple carries and then select a carry to generate the sum [8]-[9

However, the CSLA is not area efficient because it uses multiple pairs of Ripple Carry Adders (RCA) to generate partial sum and carry input Cin=0 and Cin=1, the final sum and carry are selected by the multiplexers (mux).

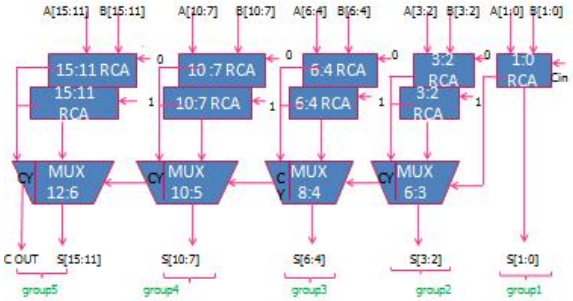

Fig 1: Block diagram for a 16-bit SQRT CSLA

The structure of the 16-b regular SQRT CSLA is shown in Fig 1. In general the complete SQRT CSLA is divided into different blocks. Block size and the number of blocks depend upon the size of SQRT CSLA according to the SQRT technique. From second block onwards, each block contains three different levels, first level is ripple carry adder with input carry zero, second level is ripple carry adder with input carry one and the third level is multiplexer which is used to select one of the ripple carry adders output according to the previous block carries. The disadvantage in SQRT CSLA is more area requirement as it uses two levels of RCAs.

I. BINARY TO EXCESS ONE CONVERTER (BEC)

For achieving better area efficiency Binary to Excess-1 Converter (BEC instead of the RCA with c_{in} =1 in 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. Second block of 16-bit SQRT CSLA with BEC logic is shown in fig.2. One input of third level multiplexers is the output of first level RCA and another input is BEC output. This produces the two possible partial results in parallel and the multiplexer is used to select either the BEC output or the direct inputs according to the control signal Cin.

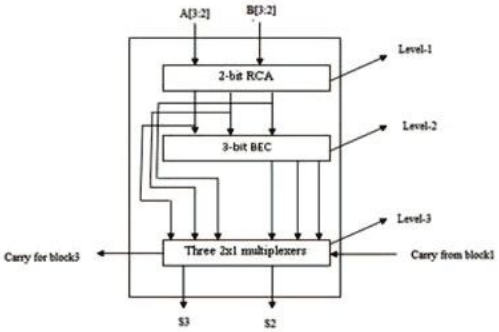

Fig. 2: Second block of SQRT CSLA with BEC in second level RCA

The structure of a 4-b BEC is shown in fig.3 and Functional block of CSLA with BEC is shown in fig.4.

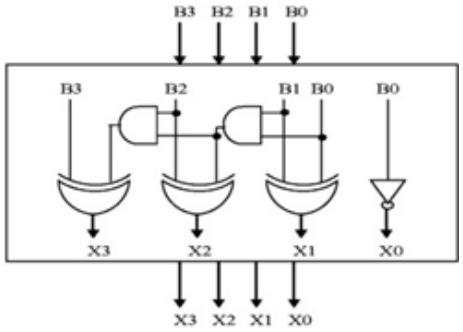

Fig 3: A 4- bit BEC

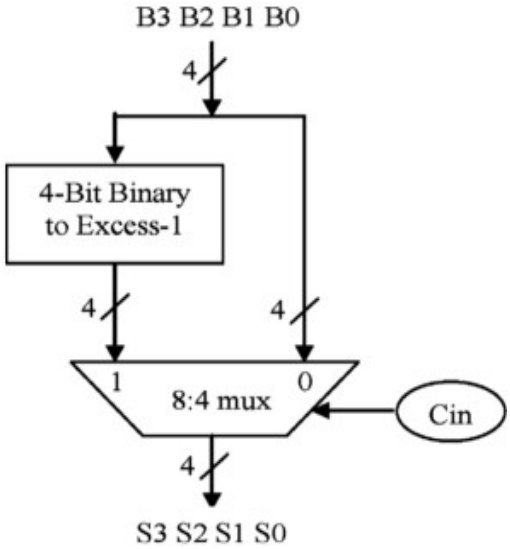

Fig 4: Functional block of CSLA

$$X0 = \sim B0$$
$$X1 = B0 \hat{} B1$$
$$X2 = B2 \hat{} (B0 \& B1)$$
$$X3 = B3 \hat{} (B0 \& B1 \& B2).$$

Fig 5: Boolean expressions of the 4-bit BEC

Fig.5 illustrates how the basic function of the CSLA is obtained by using the 4-bit BEC together with the mux. One input of the 8:4 mux gets as it input (B3, B2, B1, and B0) and another input of the mux is the BEC output. This produces the two possible partial results in parallel and the mux is used to select either the BEC output or the direct inputs according to the control signal Cin. The importance of the BEC logic stems from the large silicon area reduction when the CSLA with large number of bits are designed. The Boolean expressions of the 4-bit BEC is listed as (note the functional symbols NOT, &AND, XOR) in fig.5. The 16-bit SQRT CSLA using BEC is shown in fig.6.

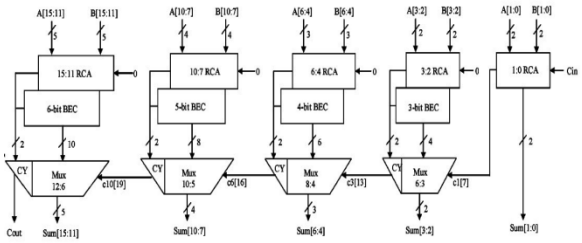

Fig 6: A 16-bit SQRT CSLA using BEC

A 3- bit BEC uses two XOR, one AND, one NOT gates, which takes 32 transistors overall whereas 2-bit RCA, which is the basic block in place of 3-bit BEC takes 56 transistors. 2-bit RCA using conventional full adder is shown in fig.7A 3-bit BEC is shown in fig8. Comparison between 2-bit RCA, 3- bit BEC is shown in table1.

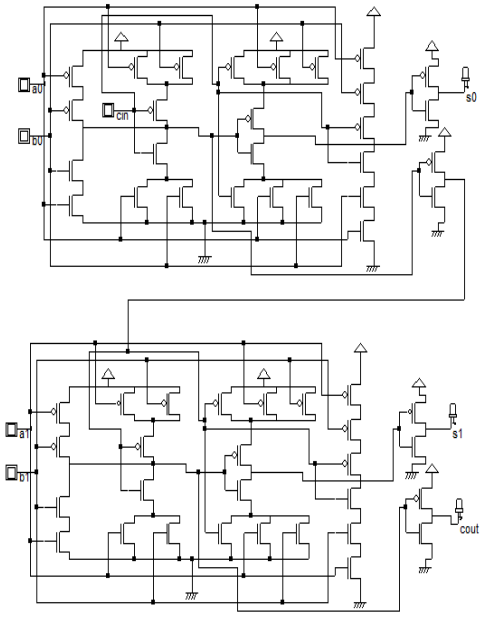

Fig 7: A 2-bit RCA using conventional full adders

Table 1: Comparison between 2-bit RCA and BEC

Logic for Second Level	Number of transistors	Critical path delay (ns)	Area (μm^2)	Power dissipation (µw)		
				Static	Dynamic	Total
RCA using CMOS	56	1.900	1342	6.706	42.565	49.271
BEC using CMOS	32	1.200	781	3.269	25.746	29.015

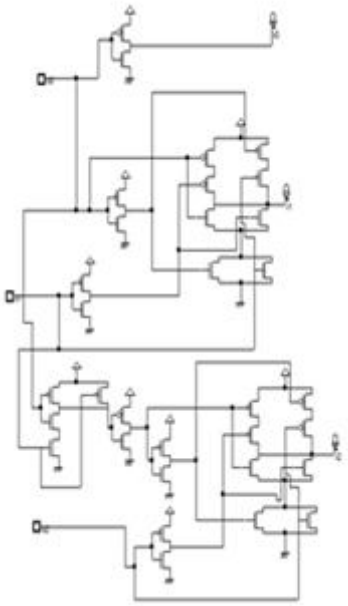

Fig 8: Transistor level 3-bit BEC

ii. SPECIAL HARDWARE USING MULTIPLEXERS (SHM)

Though BEC technique reduces area and power but not up to considerable amount and also the design is not suitable for sub threshold level modifications. The 16-bit SQRT CSLA using BEC in its second level requires 792 transistors. There is a scope to reduce the number of transistors along with the area reduction and power dissipation reduction by using proposed logic. For the implementation of a 16-bit SQRT CSLA, 736 transistors are required by using proposed logic.

The proposed logic implementation for second level RCA is Special Hardware using Multiplexers (SHM) as shown in fig.9. In this the inputs are applied to first level RCA. And the output of RCA is applied to second level SHM and then to third level multiplexer. Third level multiplexer selects either RCA output or SHM output according to the previous carry

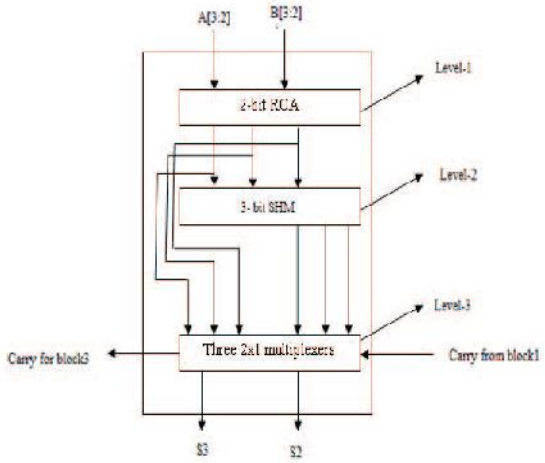

Fig. 9: Second block of SQRT CSLA with SHM in second level RCA

.

A simple 3-bit SHM is shown in Fig.10.and logic expressions of SHM shown in below the figure. A simple 3-bit SHM requires 3 multiplexers to implement b0, b1, b2 are the inputs to the 3-bit SHM and the x0, x1, x2 are corresponding outputs. SHM will take first level RCA output as input and appends its value by one. 3-bit SHM uses three multiplexers and three inverters. First inverter gives the first output bit x0 basing on input bit b0 and that output will be used as select line for the first multiplexer. First multiplexer passes either second bit b1 or inversion of second bit b1to the output because first inverter output acts like a carry to the second bit. First multiplexer gives the second output bit x1 and that will be used as second multiplexer select line. Basing on x1 output bit and b1 bit second multiplexer generates carry for input bit b2. One input to the second multiplexer is b1 and second input is grounded which will be selected when it is connected as select line to the third multiplexer. Third multiplexer passes third bit or inversion of third bit to the output according to the previous carry bit. This logic can be extended to any number of bits. It is implemented for second block with two inputs under consideration.

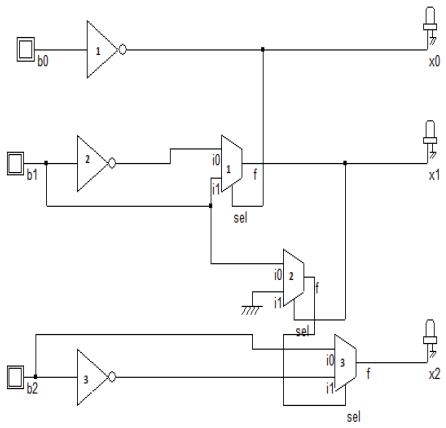

Fig 10: A 3-bit SHM

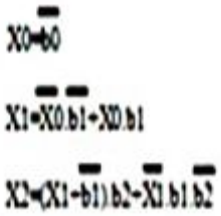

Fig 11: Boolean expressions of the 4-bit BEC

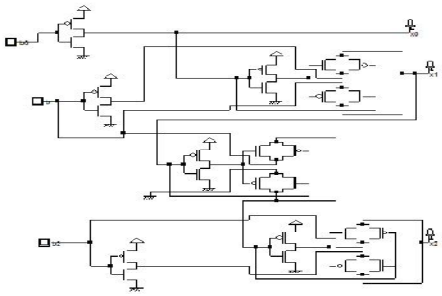

Fig 12: A 3-bit SHM using CMOS logic

Using CMOS logic 3-bit SHM is implemented and is shown in Fig.12. A 3-bit BEC uses two XOR Gates, one AND Gate and one inverter where as 3-bit SHM uses three multiplexers, three inverters. XOR Gate requires 12 transistors and AND Gate requires 6 transistors in CMOS logic. But a single multiplexer requires 6 transistors. From the Table I, it is observed that number of transistors reduced by 25% when compared to existing logic. For large number of inputs SHM is more advantageous. Fig.13 illustrates how the basic function of the CSLA is obtained by using the 4-bit SHM

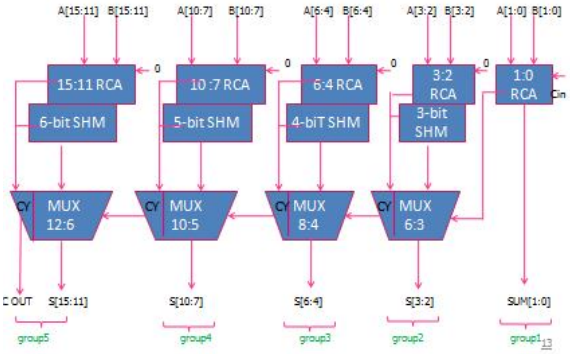

Fig 13: A 16-bit SQRT CSLA using SHM

III. GATE DIFFUSION INPUT (GDI)

Gate diffusion input (GDI) - a new technique of low-power digital combinatorial circuit design - is described. This technique allows reducing power consumption, propagation delay, and area of digital circuits while maintaining low complexity of logic design.

Gate Diffusion Input Technique here one of the inputs are directly diffused into the gates of the transistors of N-type and P-type devices so it is called as gate diffused input technique. Gate Diffusion Input technique reduces power dissipation, propagation delay, and area of digital

circuits. This method is based on the simple cell. A basic GDI cell contains four terminals they are G (common gate input of NMOS and PMOS transistors), P (the outer diffusion node of PMOS transistor), N (the outer diffusion node of NMOS transistor), and D (common diffusion node of both transistors). So it can be arbitrarily biased at contrast with a CMOS inverter. The GDI cell structure is different from the existing PTL techniques. The GDI method is based on the use of a simple cell as shown in Fig.14

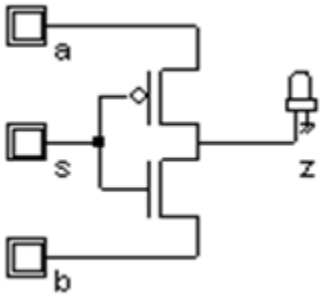

Fig 14: Basic GDI Cell

TABLE 1: BASIC FUNCTIONS USING GDI CELL

N	P	G	OUTPUT	FUNCTION
0	1	A	A'	INVERTR
0	B	A	A'B	F1
B	1	A	A'+B	F2
1	B	A	A+B	OR
B	0	A	AB	AND
C	B	A	A'B+AC	MUX
B'	B	A	A'B+B'A	XOR
B	B'	A	AB+A'B'	XNOR

AS there is a scope to reduce power, area and delay using GDI cell technique A simple GDI cell is shown in Fig.14. We can implement any bullion function using GDI cell. Low swing problems will arise, because we apply inputs directly to the sources of P and N transistors. N transistor weak to pass logic high and P transistor weak to pass logic low. When transition occur from the high to low at the P transistor source and the low to high at the N transistor source, low swing problem will arise. To avoid that demands special emphasis is that 50% of the cases, the GDI cell operates as regular CMOS inverter, which is widely used as a digital buffer for logic-level restoration. In some of these cases , when Vdd=1 without a swing drop from the previous stages, a GDI cell functions as an inverter buffer and recovers the voltage swing.

IV. PROPOSED ALU DESIGN

Here all the blocks in ALU are designed using Gate Diffusion Input (GDI). Table2 shows the truth table for the operations performed by the ALU based on the status of the select signals

Table 2: Truth table of one bit ALU

s2	s1	s0	Operation
0	0	0	AND
0	0	1	XOR
0	1	0	XNOR
0	1	1	OR
1	0	0	INCREMENT
1	0	1	ADDTION
1	1	0	SUBTRACTION
1	1	1	DECREMENT

ALU is designed using multiplexers and full adder circuit and shown in fig.15. The input and output sections consist of 4x1 and 2x1 multiplexers and logic is implemented by using full adder. A set of three select signals have been incorporated in the design to determine the operation being performed and the inputs and outputs being selected. Figure 3.13 shows the block diagram of 1-bit ALU using two 4x1 multiplexers and one 2x1 multiplexer. The complement of B is used for SUBTRACTION operation. The full adder performs the SUBTRACT operation by two's complement method

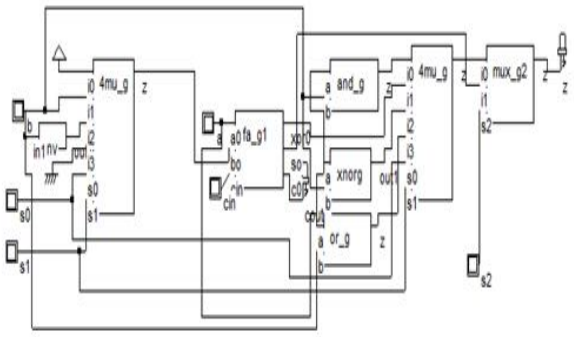

Fig 15: A 1-bit ALU

i. 1-bit ALU USING RIPPLE CARRY ADDERS

An 8- bit ALU is formed by connecting eight 1-bit ALUS in series. 8-bit ALU using 8-transistors is shown in fig.16.

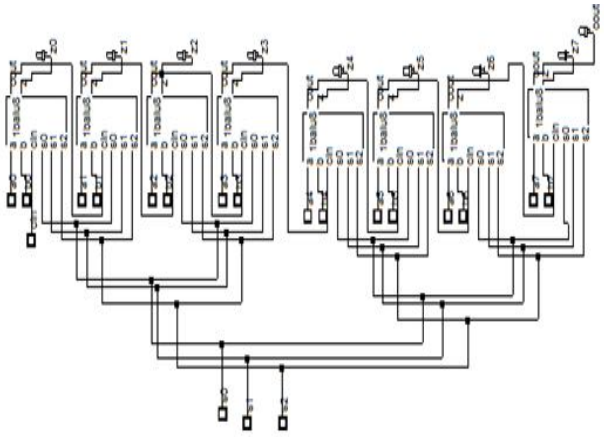

Fig 16: Eight bit ALU using 8 transistor full adders

An eight bit ALU using ripple carry adders takes more propagation delay.

The speed of ALU is limited by propagation of carry. To reduce the carry propagation the proposed design using carry select adder is implemented.

ii. DESIGN OF ALU USING MODIFIED SQRT CSLA

The proposed technique with 10-transistor full adder is applied to 8-bit ALU and corresponding circuit diagram shown in figure and for 8-transistor full adder, circuit diagram shown in fig17

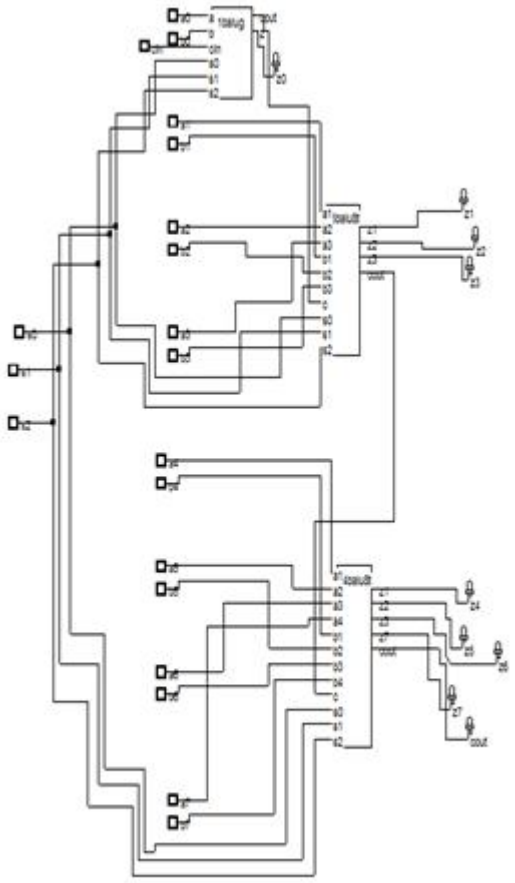

Fig 17: Eight bit ALU using modified SQRT CSLA

V. ASIC IMPLEMENTATION & RESULTS

In the designing of 8 bit ALU using efficient carry select adder, all the blocks of 16-bit SQRT CSLA, second level of second block such as 3-bit BEC and 3-bit SHM are implemented in Dsch2.6c – Logic Editor and synthesized in Micro wind 2.6a-Layout Editor under 0.12um technology with 1.2 volts as logic high voltage.

The total number of transistors required for the complete block 2 is only 98 when SHM is used. Otherwise it requires 106 Transistors with BEC technique. The number of transistors required for block3 is only 146, for block4 are 194 and for block5 are 242 when SHM is used. Otherwise block3 requires 158, block4 requires210 and block5 requires 262 ransistors with BEC technique. Using SHM for the implementation of a 16 bit SQRL CSLA 736 transistors are required where it requires 792 transistors with BEC technique. Finally the complete second block of16-bit SQRT CSLA with BEC and SHM is implemented using CMOS technology and observed the results and are shown from Table 3.

table 3: comparison of second level 2- bit rca, 3-bit bec and3- bit shm implemented using cmos technology

Logic for Second level	Number of transistors	Critical path delay (ns)	Area (μm^2)	Power dissipation (µw)		
				static	dynamic	total
RCA using CMOS	56	1.900	1342	6.706	42.565	49.271
BEC using CMOS	32	1.200	781	3.269	25.746	29.015
SHM using CMOS	24	2.350	486	3.100	22.843	25.943

Design of 8-bit ALU using efficient carry select adder is speed advantageous than the 8-bit ALU using ripple carry adders. ALU using efficient carry select adder gives 42% advantage for 10

transistors adder and 46% advantage for 8 transistor adder. Corresponding results are shown in Table 4

MODEL(ALU)	NUMBER OF TRANSISTORS	Critical path delay(ns)	Area(μm)	Power(mw)
8BIT ALU USING 8 TRANSISTOR CSLA	432	3.745	11832	0.221
8BIT ALU USING 8 TRANSISTOR RCA	494	2.070	20988	0.262

Table 4: Comparison of 8-bit ALU using 8 transistor adder

i. SIMULATION WAVEFORMS

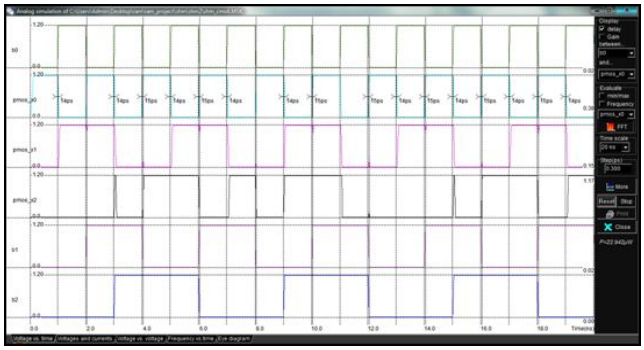

Fig 18: Power dissipation of a 3-bit SHM

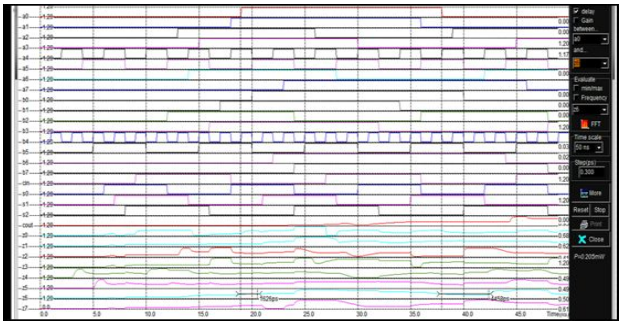

Fig 18: Wave forms of 8- bit ALU for 8- transistor full adder

REFERENCES

[1] Arun Prakash Singh, Rohit Kumar, "Implementation of 1-bit Full Adder Using Gate Diffusion Input (GDI) cell", International Journal of Electronics and Computer Science Engineering J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68-73.

[2] N. M. Chore, R. N. Mandavgane , " A survay of low power high speed one bit full adder",recent advances in networking, VLSI and signal processing, ISSN: 1790-5117. ISBN: 978-960-474-162-5.

[3] N. Weste and K. Eshraghian, Principles of CMOS VLSI Design, A System Perspective . Reading, MA: Addison- Wesley, 1993.

[4] Pardeep Kumar / International Journal of

Engineering Research and Applications(IJERA) ISSN: 2248-9622 Vol. 2, Issue 6, November- December 2012, pp.599-606

[5]. M.sreedevi and p.jeno.paul " Design and Optimization of a High Performance Low-Power CMOS Flex Cell ", International Journal of Signal System Control and Engineering Application, 2010, vol.3, no.4, pp.65-69. DOI: 10.3923/ijssceapp.2010.65.69.

[6].A good over view of leakage and reduction methods are explained in the book Leakage and reduction in Nanometer CMOS Technologies ISBN 0-387-25737-3.