IOSR Journal of VLSI and Signal Processing (IOSR-JVSP)
Volume 4, Issue 2, Ver. III (Mar-Apr. 2014), PP 31-38
e-ISSN: 2319 – 4200, p-ISSN No. : 2319 – 4197
www.iosrjournals.org

Design of Area optimized High Speed Adder Circuits in Self Resetting Logic Style

Mrs.K.Kalaiselvi, Dr.H.Mangalam, Mr.K.Manjunathan
Asst.Professor, Department of ECE, Hindusthan College of engineering and Technology, Coimbatore.
Professor and Head, Department of ECE, Sri Krishna College of Engineering and Technology, Coimbatore.
P.G Scholar (M.E., ECE), Department of ECE, Hindusthan College of engineering and Technology, Coimbatore.

***Abstract**: Dynamic logic families offer good performance over traditional CMOS logic. This is due to the comparatively high noise margins coupled with the ease of implementation. The main drawbacks of dynamic logic circuits are lack of design automation, charge sharing, feedthrough, charge leakage, single-event upsets. But these draw backs can be eliminated by using domino circuits. But it still lags in the application of clock distribution grid and routing to dynamic gates that creates problem to CAD tools. It also introduce issues on delay and skew into the circuit design process. A special dynamic logic circuit which resolves these issues is called Self-Resetting Logic (SRL). A new family of self-resetting logic (SRL) adder cells is presented in this paper which can eliminate the above sited issues. The operation of adder circuit is simulated using Tanner simulator. The analysis of modified adders designs are compared with existing Self-Resetting Logic (SRL) logic adder circuits in terms of transistor count and area, at 120nm CMOS technology is carried out.*
***Keywords:** Area, VLSI, Self-resetting logic (SRL), Full adder, High speed adders.*

I. Introduction

Different electronic devices such as mobile phones, DSPs, etc., are designed by using VLSI (Very Large Scale Integration) technology. In VLSI dynamic CMOS logic circuits are concentrating on the reducing the Power consumption, Area (portability of the system), and increasing the Speed by reducing the delay. To reduce the power consumption several parameters are to be considered, such as leakage power, feedthrough, single-event upsets, charge sharing by parasitic components while connecting source and drain of CMOS transistors.

ALU (Arithmetic Logic Circuits) are designed by using adder, subtractors, multiplier, divider, etc. Many different adder circuits designs are proposed over last few years with different logic styles. There are situations in a logic style that permit the use of circuits that can automatically precharge themselves (i.e., reset themselves) after a prescribed delays. These circuits are called postcharge or self-resetting logic which are widely in dynamic logic circuits. Overall performance of different adder designs is evaluated by using Tanner tool.

II. Related Work

In [1], low power CLA is designed using domino logic of 120nm channel length is more optimized than other channel lengths. In [2], the advancement in the modern CMOS design is superior when compared with CPL versions. In [3], full adder design is concentrated on area and delay efficient circuits. In [4], performance parameters of different types of full adders based on their transistor counts are designed and analyzed. In [5], Primitive gates and adder circuits are designed using self resetting logic and proved that their performances are good when compared with DYNAMIC and SRCMOS logic. In [6], basic gates and different adder circuits are designed using self resetting logic and their results are simulated using microwind tool. In [7], Area, Delay, Power parameters of different Adder topologies and simulation results are determined. In [8], full adder circuits with different structures are realized and the designs are implemented in FPGA and analyzed. In [9], full adder, parallel adder and flip-flop circuits are designed using self resetting logic and it is compared with conventional dynamic logic. In [10], static RAM circuits are designed using Self-Resetting logic and the falling edge are used stabilize the problems in DYNAMIC circuits. In [11], Minute delay Variations in the Self-Resetting logic are concentrated in all the circuits to increase the execution speed in the existing Full adder and high speed adders.

Advantage of our modified SRL circuits compared to all the existing designs in self-resetting logic is that area is reduced by 9.75 percentages.

III. Limitations Of Dynamic Circuits

In today‘s fast processing environment, the use of dynamic circuits is becoming increasingly popular. Dynamic CMOS circuits are defined as those circuits which have an additional clock signal inputs along with the default combinational circuit inputs of the static systems. Dynamic systems are faster and efficient than the static systems. MOS does not require the capacitances to be connected externally; thereby CMOS dynamic systems are very advantageous. However, they suffer from some major drawbacks [5].

They are:

1. Charge Leakage
2. Charge sharing

IV. Self-Resetting Dynamic Logic

Self-resetting logic is a commonly used piece of circuitry that automatically precharge themselves (i.e., reset themselves) after a prescribed delay. They find applications where a small percentage of gates switch in a cycle, such as memory decoder circuits. It is a form of logic in which the signal being propagated is buffered and used as the precharge or reset signal. By using a buffered form of the input, the input loading is kept almost as low as in normal dynamic logic while local generation of the reset assures that it is properly timed and only occurs when needed [5].

A generic view of a self-reset logic is shown in Fig 1. In the domino case, the clock is used to operate the circuit. In the self-resetting case, the output is fed back to the precharge control input and, after a specified time delay, the pull-up is reactivated. There is an NMOS sub block where the logic function performed by the gate is implemented which is represented as NMOS_LF through which the input data‘s are loaded. The output of the gate F provides a pulse if the logic function becomes true. This output is buffered and it is connected to PMOS structure to precharge. The delay line is implemented as a series of inverters.

The signals that propagate through these circuits are pulses. The width of the pulses must be controlled carefully or else there may be contention between NMOS and PMOS devices, or even worst, oscillations may occur.

One of the advantages of self-resetting logic is that when data present at evaluation does not require dynamic node to discharge, the precharge device is not active hence reduces power. In the circuit MP, MR and VSGR represent the precharge pull-up, reset pull-up and gate-source voltage of resetting transistor. During precharge phase clk=0, the transistor MP turns ON and the pull down network is OFF. Therefore the capacitor is charged to VDD. During evaluation phase clk=1, the transistor MP turns OFF and the pull down network is ON and evaluates the logic function. Therefore the capacitor is discharged making MR active which allows IDR to flow and recharge CX back up to a voltage of Vx=VDD as in Fig 2.

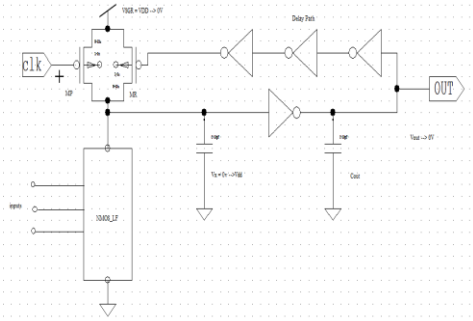

Fig 1: Precharge mode in self resetting logic

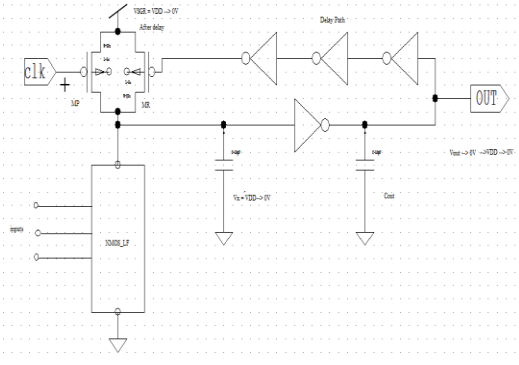

Fig 2: Evaluation mode in self resetting logic

V. Srl Full Adder

The SRL full adder circuit is shown in Fig. 3. This adder consists of sum and carry block. The sum block is implemented by SRL XOR gates. The carry block is implemented with SRL AND and SLR OR gates. The input to this full adder circuit are A ,B and Cin and the outputs are S(Sum) and C(Carry). Results of simulation of the adder, implemented in a 120nm CMOS process and supply voltage of VDD =5V [5].

The sum output can be obtained in output node S

$$S= ((A \text{ XOR } B) \text{ XOR } Cin) \quad (1)$$

The carry output is obtained by using the expression

$$C = AB+BC+AC \quad (2)$$

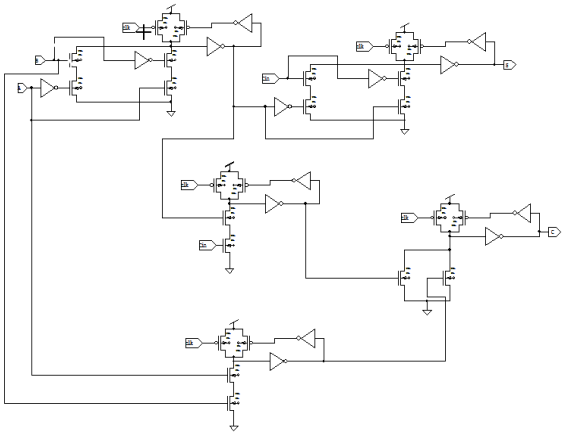

Fig 3: SRL Full adder

The sum output is obtained using two XOR gates and carry output is obtained by using two AND gates and one OR gate.

The design of modified Full adder in Self Resetting logic is shown in the Fig 4. Pull-up network is reconfigured and the clock signal applied to pull-up network has been eliminated from the existing circuit. In modified full adder structure number of transistors used to design the circuits is reduced.

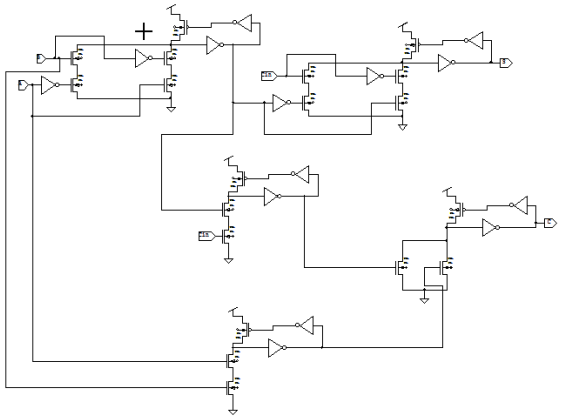

Fig 4: Modified SRL Full adder

VI. Srl Parallel Adder

Using the SRL full adder circuit, the SRL 4-bit parallel adder is as shown in Fig 5. In this adder circuit, the input to each full-adder will be Ai, Bi and Ci, and the outputs will be Si and Ci, where i ‘vary from 0 to 3. Also, the carry output of the lower order stage is connected to the carry input of the next higher order stage. In the least significant stage A0, B0 and C0 (which is grounded) are added resulting in S0 and C0. This carry C0 becomes the carry input to the second stage. Similarly the remaining stages are executed. The simulation setup cycle for is adder is 600 ns and the input combinations are fed into the system and its performance was analyzed. Results of TANNER simulation of the adder, implemented in a 120nm CMOS process, with VDD = 5V is shown in Fig 5.

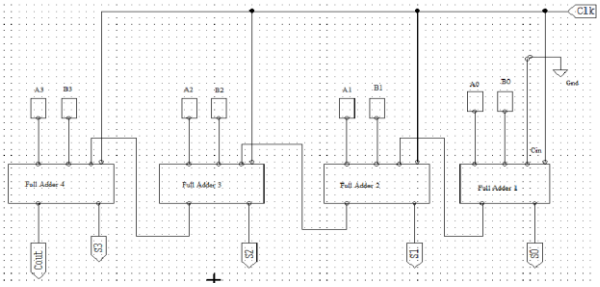

Fig 5: SRL Parallel adder

The Modified SRL Parallel adder is shown in the Fig 6. Modified Full adder circuits are used to design Parallel adder circuits, where clock signal is not needed which is eliminated from the existing circuit. Area of the Parallel adder is reduced because one transistor in the Pull-up network of all the basic gate designs are eliminated and re-configured.

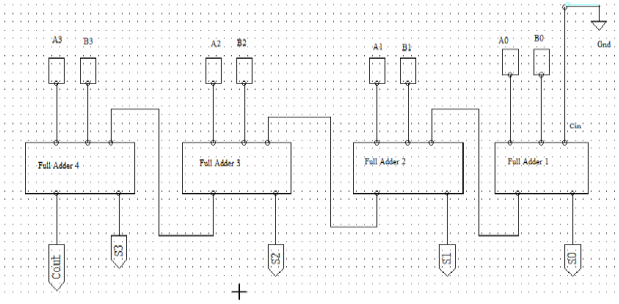

Fig 6: Modified SRL Parallel adder

VII. CARRY SKIP ADDER

A carry-skip adder consists of a simple ripple carry-adder with a special speed up carry chain called a skip chain. Carry skip adder is a fast adder compared to ripple carry adder when addition of large number of bits take place; carry skip adder has very less delay provides a good compromise in terms of delay, along with a simple and regular layout This chain defines the distribution of ripple carry blocks, which compose the skip adder. A carry-skip adder is designed to speed up a wide adder by aiding the propagation of a carry bit around a portion of the entire adder. Actually the ripple carry adder is faster for small values of N. However the industrial demands these days, which most desktop computers use word lengths of 32 bits like multimedia processors, makes the carry skip structure more interesting. The crossover point between the ripple-carry adder and the carry skip adder is dependent on technology considerations and is normally situated 4 to 8 bits. The carry-skip circuitry consists of two logic gates. The AND gate accepts the carry-in bit and compares it to the group propagate signal [6].

$$p[i , i + 3] = (pi + 3)*(pi + 2)*(pi + 1)* pi \quad (3)$$

Using the individual propagate values. The output from the AND gate is ORed with Cout of RCA to produce a stage output of Carry.

$$Carry = ci + 4 + p [i, i + 3] * ci \quad (4)$$

If p [i, i + 3] =0, then the carry-out of the group is determined by the value of ci + 4.
However, if p[i , i + 3] =1 when the carry-in bit is ci =1, then the group carry-in is automatically sent to the next group of adders. The design schematic of modified Carry Skip Adder is shown in Fig 7.

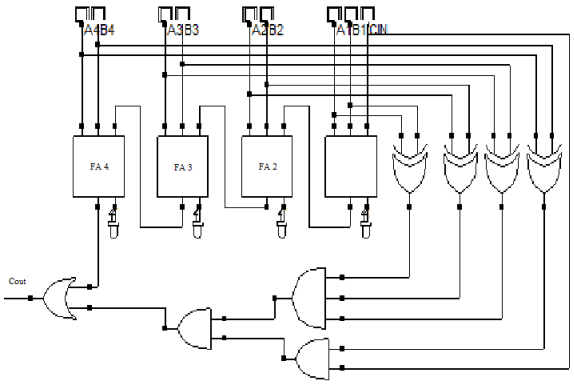

Fig 7: Modified Carry Skip adder

VII. Srl Carry Look Ahead Adder

The propagation delay occurred in the parallel adders can be eliminated by carry look ahead adder. This adder is based on the principle of looking at the lower order bits of the augends and addend if a higher order carry is generated. This adder reduces the carry delay by reducing the number of gates through which a carry signal must propagate. This adder consists of three stages: a propagate block/ generate block, a sum generator and carry generator [6]. The generate block can be realized using the expression

$$Gi = Ai*Bi \qquad \text{for } i=0, 1, 2, 3 \qquad (4)$$

Similarly the propagate block can be realized using the expression

$$Pi = A \text{ XOR } B \qquad \text{for } i=0, 1, 2, 3 \qquad (5)$$

The carry output of the (i-1) th stage is obtained from

$$Ci\ (Cout) = (Gi + Pi)*Ci\text{-}1 \qquad \text{for } i=0, 1, 2, 3 \qquad (6)$$

The sum output can be obtained using

$$S= (A \text{ XOR } B)*Ci\text{-}1 \qquad \text{for } i=0, 1, 2, 3 \qquad (7)$$

The 4-bit modified carry look ahead adder with the primitive gates and output of the adder are processed with 120nm CMOS processing technology, with VDD = 5V are shown in Fig 8.

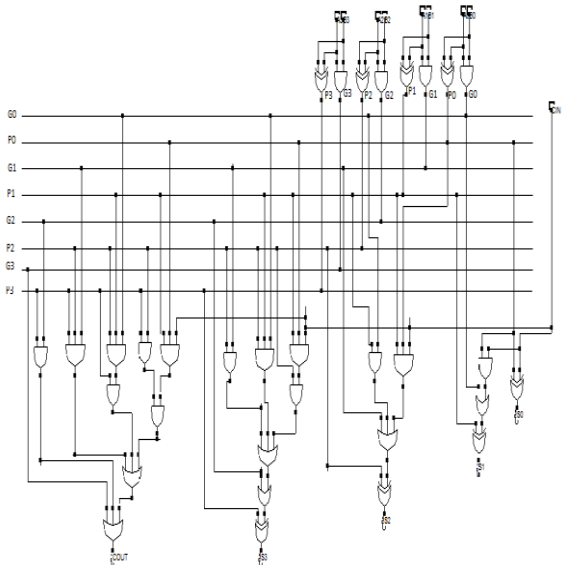

Fig 8: Modified Carry Look ahead adder

VIII. Simulation Results And Discussion

The circuits simulated using Tanner tool. Fig 9 to 13 shows the simulation result of modified full adder, parallel adder, carry skip adder, carry look ahead adder circuits. Table 1 and Chart 1 gives the area of the above said adder circuits.

It can be noted that the area of modified full adder and parallel adder using SRL is 9.72% lesser than the area of existing Full adder and Parallel adder circuits using SRL. The area of modified carry skip adder circuit using SRL is 9.42% lesser than the area of existing carry skip adder using SRL. The area of modified carry look ahead adder circuit using SRL is 10.19% lesser than the area of existing carry look ahead adder using SRL.

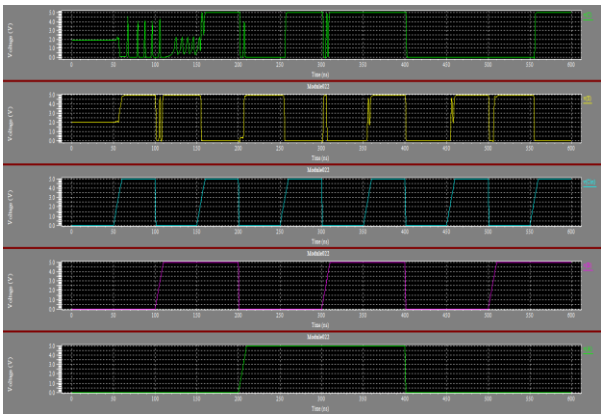

Fig 9: Simulation result of modified Full adder

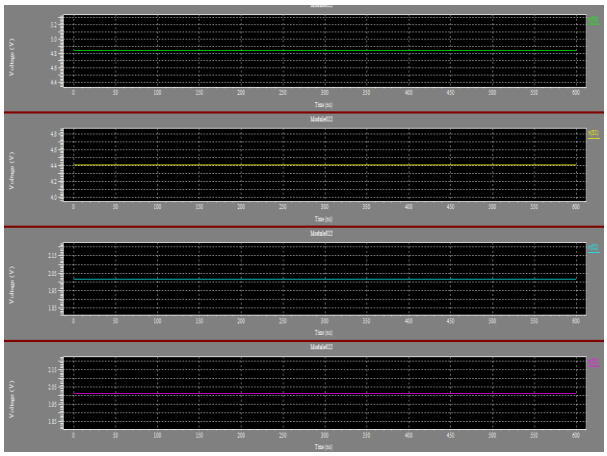

Fig 10: Simulation result of modified Parallel adder

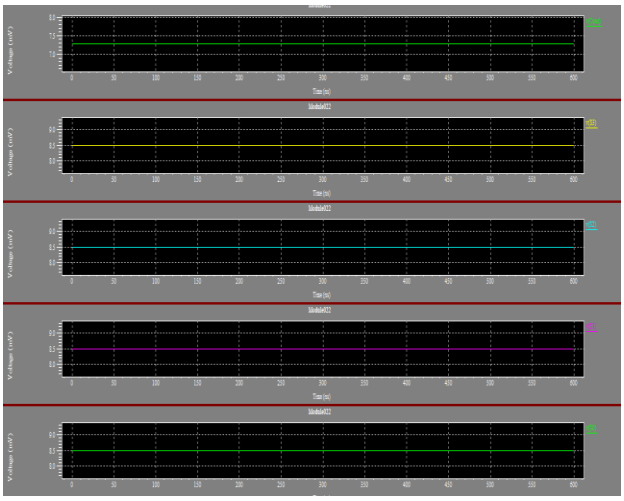

Fig 11: Simulation result of modified Carry Skip adder

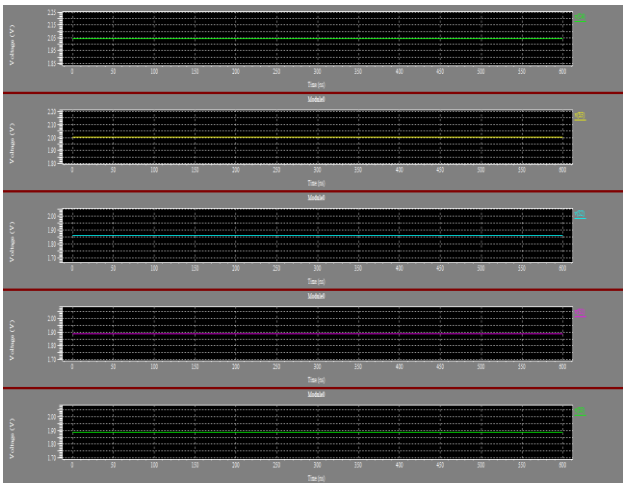

Fig 12: Simulation result of modified Carry Look ahead adder

Table.1 – Area Comparison of various adders

Parameters	Transistor Count	Area in μm^2
SRL Full adder	52	2288
Modified SRL Full adder	47	2068
SRL Parallel adder	208	9152
Modified SRL Parallel adder	188	8272
SRL Carry skip adder	297	13068
Modified SRL Carry skip adder	269	11836
SRL Carry Look ahead adder	314	13816
Modified SRL Carry Look ahead adder	282	12408

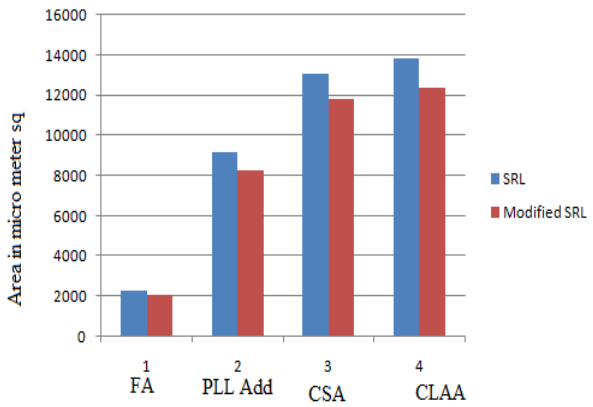

Fig 13: Comparison of Area in high speed adders.

IX. Conclusion

In the proposed paper the high speed adder circuits designed in Self resetting logic are re-configured and the number of transistors in the modified design is reduced. The goal is to obtain a family of adders that could simplify the implementation of fast processing circuit which overcomes the restriction due the pulses being elongated and shortened as signal traverse the logic stages. The circuits are designed in 120nm processing CMOS processing technology. Modification is done in the pull up network and clock signal also eliminated. Even though average power consumed has been increased slightly the total area of the circuits are reduced by 9.75% percentages from the existing design. Hence it is concluded that the proposed design of Self resetting logic provide effective way to reducing the area in DYNAMIC logic circuits.

References

[1]. Mousam Halder, (2012), 'Implementation of high speed low power carry look ahead using Domino logic', International Journal of Applied Sciences and Engineering Research, Vol. 1, No. 3. ISSN 2277 – 9442.

[2]. Reto Zimmermann and Wolfgang Fichtner, (1997), 'Low-Power Logic Styles: CMOS versus Pass-Transistor Logic', Journal of Solid-state Circuits, (IEEE) Vol. 32, No. 7, 1.

[3]. G.Shyam Kishore, "A Novel Full Adder with High Speed Low Area", 2nd National Conference on Information and Communication Technology (NCICT) 2011 Proceedings published in International Journal of Computer Applications® (IJCA).

[4]. Saradindu Panda, A.Banerjee, B.Maji, Dr.A.K.Mukhopadhyay, (2012), 'Power and Delay Comparison in between Different types of Full Adder Circuits', International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, Vol. 1, No 3, ISSN 2278 – 8875.

[5]. Uma.R, (2011), '4-Bit Fast Adder Design: Topology and Layout with Self-Resetting Logic for Low Power VLSI circuits', International Journal of Advanced Engineering Sciences and Technologies, Vol. 7, No. 2, pp.197 – 205.

[6]. Uma.R, Vidya Vijayan, M.Mohanapriya, Sharon Paul, (2012), 'Area, Delay and Power Comparison of Adder Topologies', International Journal of VLSI design & Communication Systems Vol.3, No.1.

[7]. Uma.R and P.Dhavachelvan, (2012), 'Logic Optimization using Technology Independent MUX based Adders in FPGA', International Journal of VLSI design & Communication Systems, Vol.3, No.4.

[8]. Vijaya Bhaskar M, Suparshya Babu Sukhavasi, Susrutha Babu Sukhavasi, G Santhi Swaroop Vemana, (2012), 'Implementation of Synchronous up Counter by using Self Resetting Logic', International Journal of Engineering Research and Applications, Vol 2, No.1, pp.972-981

[9]. Suparshya Babu Sukhavasi, Susrutha Babu Sukhavasi, Vijaya Bhaskar M, B Rajesh Kumar, (2012), 'Analysis and Comparison of combinational Circuits by using low Power Techniques', International Journal of Advances in Engineering & Technology, Vol. 3, No.1, ISSN: 2231-1963.

[10]. Woo Jin Kim, Yong-Bin Kim, 'A Localized Self-Resetting Gate Design Methodology for Low Power', Electrical and Computer Engineering Department, Northeastern University.

[11]. Miguel E. Litvin and Samiha Mourad, Senior Member, (2005), 'Self-Reset Logic for Fast Arithmetic Application' VLSI Systems, IEEE, Vol.13, No.4.