International Journal of Research
Available at https://edupediapublications.org/journals
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 02 Issue 01
January 2015

DESIGN OF LOW-POWER FULL ADDER USING GDI STRUCTURE AND HYBRID CMOS LOGIC STYLE

Banda Srikanth
Assistant Professor, Department of E.C.E
University College of Engineering Kakatiya University,Kothagudem
Telangana State, India.

***Abstract*— In this paper, a hybrid 1-bit full adder design using GDI structure is employed. The circuit was implemented using TANNER EDA tools in 180 and 90-nm technology. Performance parameters such as power, delay were compared with the existing designs such as conventional cmos adder, hybrid cmos adder, mirror adder, transmission gate adder, and so on. In comparison with the existing full adder designs, the present implementation was found to offer significant improvement in terms of power and speed.**

***Index Terms*— Adders, Hybrid CMOS, GDI, Tanner EDA**

I. INTRODUCTION

Due to rapid advances in electronic technology, electronics market is becoming more competitive, which results in consumer electronic products requiring even more stringently high quality. The design of consumer electronic products requires not only light weight and slim size, but also low power and fast time-to-market. Therefore, the integrated circuit (IC) designers have to consider more important issues such as chip area, power consumption, operation speed, circuit regularity, and so on. Due to these design issues relevant to the key competitive factors of electronic systems, IC designers and electronic design automation (EDA) vendor are very concerned about the development of effective methodologies to fetch smaller chip area design, lower power consumption, faster operation speed and more regular circuit structure.

The arithmetic circuit is the important core in electronic systems. If the arithmetic circuit has good characteristics, the overall performance of electronic systems will be improved dramatically. Obviously, the performance of the arithmetic circuit directly determines whether the electronic system in market is competitive. It is well known that full adder is the crucial building block used to design multiplier, microprocessor, digital signal processor (DSP), and other arithmetic related circuits. In addition, the full adder is also dominant in fast adder design. Therefore, to effectively design a full adder with smaller chip area, low power consumption, fast operation speed and regular circuit structure, are the common required for IC designers.

Since full adder plays an extremely important role in arithmetic related designs, many IC designers puts a lot of efforts on full adder circuit research. Consequently, there are many different types of full adders have been developed for a variety of different applications. These different types of full adders have different circuit

structures and performance. Full adder designs have to make trade off among many features including lower power consumption, faster operating speed, reduced transistor count, full-swing output voltage and the output driving capability, depending on their applications to meet the needs of electronic systems.

One important kind of full adder designs focus on adopting minimum transistor count to save chip area [1, 2, 3, 4, 5]. These full adder designs with fewer transistors to save chip area does have excellent performance, however, due to MOS transistors reduced, these full adders have threshold voltage loss problem and poor output driving capability. Some full adders are designed to emphasize making up for threshold voltage loss to improve circuit performance [6, 7, 8]. These full-swing full adder designs insist on using fewer MOS transistors to reduce circuit complexity to go along with reduced power consumption and delay time. However, the full-swing full adders have no output driver in design leading to signal attenuation problems when they are connected in series to construct multi-bit adders. Therefore, many studies focus on gathering many features such as full-swing voltage, powerful output driving capability and good power delay product [9, 10, 11, 12, 13] in the meantime to boost the performance of full adder circuit design as a whole. However, the penalties have to pay for taking too many design issues into consideration are increased circuit complexity, larger chip area, difficult layout design, and increased transistor count. Therefore, how to design a full adder circuit with better performance and simpler structure is the main goal of full adder design field. In order to design a full adder with low circuit complexity, good circuit performance and the modularized structures, a multiplexer-based full adder is proposed in this study. The multiplexer-based full adder has not only regularly modularized structure, but also superior circuit performance.

II. PREVIOUS WORKS ON FULL ADDER DESIGN

The full adder function is to sum two binary operands A, B and a carry input C_i, and then generate a sum output (S) and a carry output (C_o). There are two factors affecting the performance of a full adder design: one is the full adder logic architecture, and the other is the circuit design techniques to perform the logic architecture function. Therefore, the full adder design approach requires using different types of logic architecture and circuit design technique to improve the total performance.

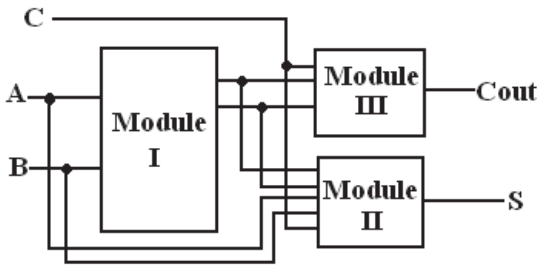

Fig 1. Schematic structure of hybrid full adder.

The traditional full adder logic architecture can be divided into three modules [6, 7, 8], and the architecture block diagram is shown in Fig 1. 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.

OPERATION OF THE HYBRID FULL ADDER:

International Journal of Research
Available at https://edupediapublications.org/journals
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 02 Issue 01
January 2015

Fig. 2 shows the detail diagram of the hybrid full adder. The sum output of the full adder is implemented by XNOR modules. The inverter comprised of transistors Mp1 and Mn1 generate B', which is effectively used to design the controlled inverter using the transistor pair Mp2 and Mn2. Output of this controlled inverter is basically the XNOR of A and B. But it has some voltage degradation problem, which has been removed using two pass transistors Mp3 and Mn3. pMOS transistors (Mp4, Mp5, and Mp6) and nMOS transistors (Mn4, Mn5, and Mn6) realize the second stage XNOR module to implement the complete SUM function. Analyzing the truth table of a full adder, the condition for *C*out generation has been deducted as follows:

If, $A = B$, then $C_{out} = B$; else, $C_{out} = C$in.

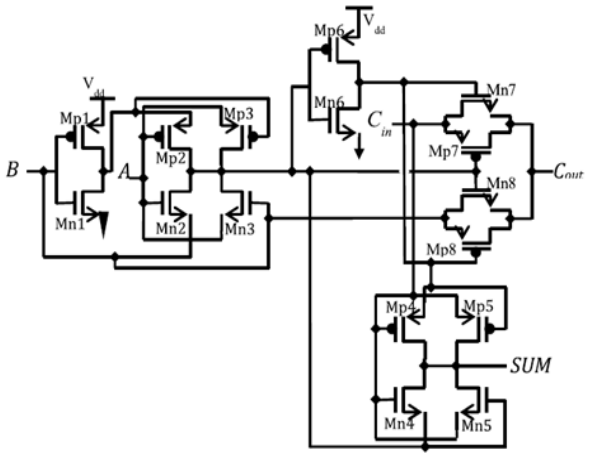

Fig 2. Circuit diagram of hybrid full adder.

The parity between inputs A and B is checked by *A* XNOR *B* function. If they are same, then C_{out} is same as B, which is implemented using the transmission gate realized by transistors Mp8 and Mn8. Otherwise, the input carry signal (*C*in*)* is reflected as C_{out} which is implemented by another transmission gate consisting of transistors Mp7 and Mn7.

III. DESIGN CONSIDERATIONS AND TECHNOLOGIES

A. Impact of Logic Style

The logic style used in logic gates basically influences the speed, size, power dissipation, and the wiring complexity of a circuit. The circuit delay is determined by the number of inversion levels, the number of transistors in series, transistor sizes [3] (i.e., channel widths), and intra- and inter-cell wiring capacitances. Circuit size depends on the number of transistors and their sizes and on the wiring complexity. Power dissipation is determined by the switching activity and the node capacitances (made up of gate, diffusion, and wire capacitances), the latter of which in turn is a function of the same parameters that also control circuit size. Finally, the wiring complexity is determined by the number of connections and their lengths and by whether single-rail or dual-rail logic is used. All these characteristics may vary considerably from one logic style to another and thus make the proper choice of logic style crucial for circuit performance. As far as cell-based design techniques (e.g., standard-cells) and logic synthesis are concerned, ease-of-use and generality of logic gates is of importance as well. Robustness1 with respect to voltage and transistor scaling as well as varying process and working conditions, and compatibility with surrounding circuitries are important aspects influenced by the implemented logic style.

International Journal of Research
Available at https://edupediapublications.org/journals
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 02 Issue 01
January 2015

B. Logic Style Requirements for Low Power

According to the formula_

$$P_{dyn} = V_{dd}^2 . f_{clk} . \sum a_n . c_n + V_{dd} . \sum_n i_s c_n$$

the dynamic power dissipation of a digital CMOS circuit depends on the supply voltage, the clock frequency, the node switching activities the node capacitances, the node short circuit currents and the number of nodes. A reduction of each of these parameters results in a reduction of dissipated power [4]. However, clock frequency reduction is only feasible at the architecture level, whereas at the circuit level frequency is usually regarded as constant in order to fulfill some given throughput requirement. All the other parameters are influenced to some degree by the logic style applied. Thus, some general logic style requirements for low-power circuit implementation can be stated at this point.

Switched capacitance reduction: Capacitive load, originating from transistor capacitances (gate and diffusion) and interconnect wiring, is to be minimized. This is achieved by having as few transistors and circuit nodes as possible, and by reducing transistor sizes to minimum. In particular, the number of (high capacitive) inter-cell connections and their length (influenced by the circuit size) should be kept minimal. Another source for capacitance reduction is found at the layout level [4], which, however, is not discussed in this paper. Transistor downsizing is an effective way to reduce switched capacitance of logic gates on noncritical signal paths [5]. For that purpose, a logic style should be robust against transistor downsizing, i.e., correct functioning of logic gates with minimal or near-minimal transistor sizes must be guaranteed (ratio less logic).

Supply voltage reduction: The supply voltage and the choice of logic style are indirectly related through delay-driven voltage scaling. That is, a logic style providing fast logic gates to speed up critical signal paths allows a reduction of the supply voltage in order to achieve a given throughput. For that purpose, a logic style must be robust against supply voltage reduction, i.e., performance and correct functioning of gates must be guaranteed at low voltages as well. This becomes a severe problem at very low voltages of around 1 V and lower, where noise margins become critical.

Switching activity reduction: Switching activity of a circuit is predominantly controlled at the architectural and register transfer level (RTL). At the circuit level, large differences are primarily observed between static and dynamic logic styles. On the other hand, only minor transition activity variations are observed among different static logic styles and among logic gates of different complexity, also if glitching is concerned.

Short-circuit current reduction: Short-circuit currents (also called dynamic leakage currents or overlap currents) may vary by a considerable amount between different logic styles. They also strongly depend on input signal slopes (i.e., steep and balanced signal slopes are better)[7] and thus on transistor sizing. Their contribution to the overall power consumption is rather limited but still not negligible (10–30%), except for very low voltages where the short-circuit currents disappear. A low-power logic style should have minimal short-circuit currents and, of course, no static currents besides the inherent CMOS leakage currents.

C. Logic Style Requirements for Ease-of-Use:

For ease-of-use and generality of gates, a logic style should be highly robust and have friendly electrical characteristics, that is, decoupling of gate inputs and outputs (i.e., at least one inverter stage per gate) as well as good driving capabilities and full signal swings at the gate outputs, so that logic gates can be cascaded arbitrarily and work reliably in any circuit configuration. These properties are prerequisites for cell-based design and logic synthesis, and they also allow for efficient gate modelling and gate-level simulation. Furthermore, a logic style should allow the efficient implementation of arbitrary logic functions and provide some regularity with respect to circuit and layout realization. Both low-power and high-speed versions of logic cells (e.g., by way of transistor sizing) should be supported in order to allow flexible power-delay tuning by the designer or the synthesis tool.

D. Static versus Dynamic Logic Styles:

A major distinction, also with respect to power dissipation, must be made between static and dynamic logic styles. As opposed to static gates, dynamic gates are clocked and work in two phases, a precharge and an evaluation phase. The logic function is realized in a single NMOS pull-down or PMOS pull-up network, resulting in small input capacitances and fast evaluation times. This makes dynamic logic attractive for high-speed applications. However, the large clock loads and the high signal transition activities due to the precharging mechanism result in an excessive high power dissipation. Also, the usage of dynamic gates is not as straightforward and universal as it is for static gates, and robustness is considerably degraded. With the exception of some very special circuit applications, dynamic logic is no viable candidate for low-power circuit design.

There are many sorts of techniques that intend to solve the problems mentioned above. The sum and carry out expressions can be written as

So = H′Ci + HC′o…….. (1)

Co = HCi + H′A……. (2)

where H = A XOR B and H′ = A XNOR B. A Full Adder is made up of an XOR–XNOR module, a sum module and a carry module. The XOR–XNOR module performs XOR and XNOR logic operations on inputs A and B, and then generates the outputs H and H′. Subsequently, H and H′ both are applied to the sum and the carry modules for generation of sum output S and carry output Co.

In hybrid CMOS Full Adder the

Module I (XOR/XNOR module) was designed using 6 transistors using CPL with swing restored logic was added in order to generate full output swing.

Module II XOR between H and Cin was designed in such a way that all the input logics should have a high logical swing output.

Module III Multiplexer was designed with same as the Module II for every input it should have high logic swing.

IV.DESIGN APPROACH OF THE PROPOSED FULL ADDER

International Journal of Research
Available at https://edupediapublications.org/journals
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 02 Issue 01
January 2015

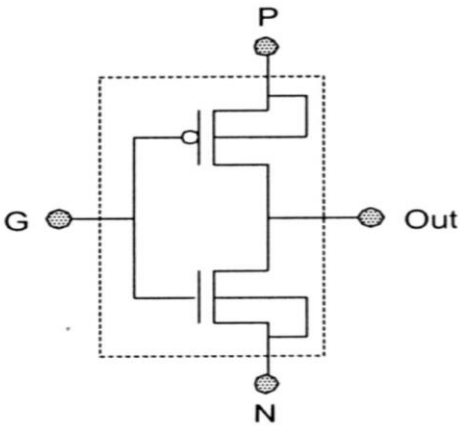

Fig 3. GDI basic cell.

The GDI method is based on the use of a simple cell as shown in Fig 3. At first glance, the basic cell reminds one of the standard CMOS inverter, but there are some important differences.

1) The main difference between the CMOS and GDI based design is that the source of the PMOS in a GDI cell is not connected to VDD and the source of the NMOS is not connected to GND. This feature gives the GDI cell two extra input pins for use which makes the GDI design more flexible than CMOS design.

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

3) 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. This technique allows reducing power consumption, propagation delay, and area of digital circuits while maintaining low complexity of logic design.

Here the three modules are designed using GDI based design.

In this cell, the Gate Diffusion Input (GDI CELL) technique has been used for generating of intermediate function of XOR. The new circuit is the most energy efficient cell compared to several CMOS circuits. The issue of sequential logic design with Gate Diffusion Input (GDI CELL) is currently being explored, as well as technology compatibility for twin-well CMOS process. More work was done in automation of a logic design methodology based on Gate Diffusion Input (GDI) cells.

GDI Consists three modules XOR/XNOR block and a XOR block and a Multiplexer

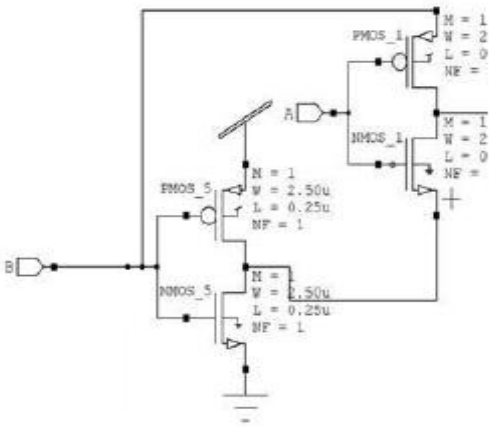

Fig 4. (Module 1) XOR/XNOR Module

XOR module was designed using GDI by making Gate input as A and P input as B and N inut as Bbar. So this generates (A xor B Xor C).

International Journal of Research
Available at https://edupediapublications.org/journals
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 02 Issue 01
January 2015

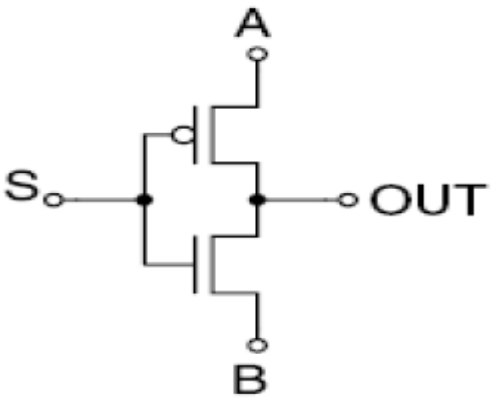

Fig 5. GDI multiplexer

Module 2 is a XOR gate operation between (A xor B) output and XOR with CIn

So this generates (A xor B Xor C).

Module 3 is a Multiplexer whose selection line is A xor B and two inputs are A and Cin which generates the Carry.

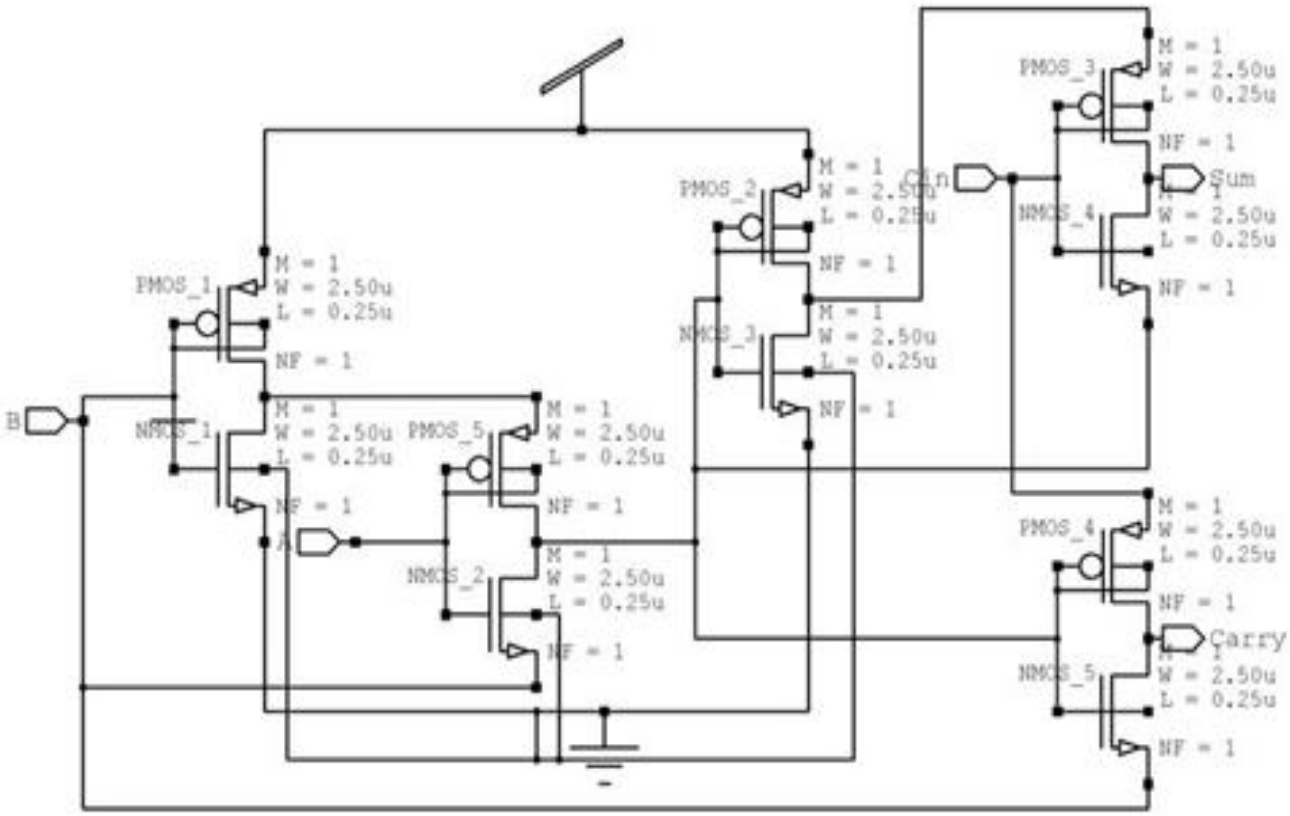

Fig 6. GDI Full Adder design

The full adder operation can be given as follows. Given the three 1-bit inputs *A*, *B*, and Cin, it is desired to compute the two 1-bit outputs Sum and Cout, given by

Sum = *A XOR B XOR* Cin.

Cout = $A \cdot B$ + Cin (*A XOR B*).

For generating the Sum output in the proposed design, the truth table has been segmented into two parts, one for input *A* = "0" and another for *A* = "1" rather than implementing the conventional Sum module. From the

truth table shown in Table 1 it is clear that when *A* = "0", Sum can be produced by XORing inputs *B* and Cin. Similarly, when *A* = "1", Sum focusing the XNORing between inputs *B* and Cin. Therefore, the operation of Sum module depends on implementing XOR operation and XNOR operation between inputs *B* and Cin .

V. PERFORMANCE ANALYSIS OF THE PROPOSED FULL ADDER

TABLE I. SIMULATION RESULTS FOR FULL ADDERS IN 180 nm TECHNOLOGYWITH 1.8 V SUPPLY

designs	Average power(µw)	Delay (ns)	Transistor count
c-cmos	185.3711	150	28
mirror	15.04689	50	28
Hybrid	8.034121	3.25	16
proposed	5.047875	5.4	10

VI.CONCLUSION

In this paper, a low-power hybrid 1-bit full adder using GDI structure has been proposed. The simulation was carried out using Tanner EDA tools with 180/90-nm technology and compared with other standard design approaches like CMOS, CPL, TFA, TGA, and other hybrid designs. The simulation results established that the proposed adder offered improved PDP compared with the earlier reports. The proposed full adder offered 20.56% improvement with respect to the best reported design [23] in terms of PDP (180-nm technology at 1.8 V). Corresponding PDP improvement was 27.36% when the same design was implemented in 90-nm technology at 1.2-V power supply.

REFERENCES

[1] C.-K. Tung, Y.-C. Hung, S.-H. Shieh, and G.-S. Huang, "A low-power high-speed hybrid CMOS full adder for embedded system," in *Proc. IEEE Conf. Design Diagnostics Electron. Circuits Syst.*, vol. 13.
Apr. 2007, pp. 1–4.

[2] S. Goel, A. Kumar, and M. A. Bayoumi, "Design of robust, energyefficient full adders for deep-submicrometer design using hybrid-CMOS logic style," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 14, no. 12, pp. 1309–1321, Dec. 2006.

[3] N. H. E. Weste, D. Harris, and A. Banerjee, *CMOS VLSI Design: A Circuits and Systems Perspective*, 3rd ed. Delhi, India: Pearson Education, 2006.

[4] J. M. Rabaey, A. Chandrakasan, and B. Nikolic, *Digital Integrated Circuits: A Design Perspective*, 2nd ed. Delhi, India: Pearson Education, 2003.

[5] D. Radhakrishnan, "Low-voltage low-power CMOS full adder," *IEEE Proc.-Circuits Devices Syst.*, vol. 148, no. 1, pp. 19–24, Feb. 2001.

[6] R. Zimmermann and W. Fichtner, "Low-power logic styles: CMOS versus pass-transistor logic," *IEEE J. Solid-State Circuits*, vol. 32, no. 7, pp. 1079–1090, Jul. 1997.

[7] C. H. Chang, J. M. Gu, and M. Zhang, "A review of 0.18-μm full adder performances for tree structured arithmetic circuits," *IEEE Trans.*

Very Large Scale Integr. (VLSI) Syst., vol. 13, no. 6, pp. 686–695, Jun. 2005.

[8] A.M. Shams, T. K. Darwish, and M. A. Bayoumi, "Performance analysis of low-power 1-bit CMOS full adder cells," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 10, no. 1, pp. 20–29, Feb. 2002.

[9] M. L. Aranda, R. Báez, and O. G. Diaz, "Hybrid adders for high-speed arithmetic circuits: A comparison," in *Proc. 7th IEEE Int. Conf. Elect. Eng. Comput. Sci. Autom. Control (CCE)*, Tuxtla Gutierrez, NM, USA, Sep. 2010, pp. 546–549.

[10] M. Vesterbacka, "A 14-transistor CMOS full adder with full voltageswing nodes," in Proc. IEEE Workshop Signal Process. Syst. (SiPS), Taipei, Taiwan, Oct. 1999, pp. 713–722.

[11] M. Zhang, J. Gu, and C.-H. Chang, "A novel hybrid pass logic with static CMOS output drive full-adder cell," in Proc. Int. Symp. Circuits Syst., May 2003, pp. 317–320.

[12] S. Wairya, G. Singh, R. K. Nagaria, and S. Tiwari, "Design analysis of XOR (4T) based low voltage CMOS full adder circuit," in Proc. IEEE Nirma Univ. Int. Conf. Eng. (NUiCONE), Dec. 2011, pp. 1–7.

[13] S. Goel, M. Elgamel, and M. A. Bayoumi, "Novel design methodology for high-performance XOR-XNOR circuit design," in Proc. 16th Symp. Integr. Circuits Syst. Design (SBCCI), Sep. 2003, pp. 71–76.

[14] J.-M. Wang, S.-C. Fang, and W.-S. Feng, "New efficient designs for XOR and XNOR functions on the transistor level," IEEE J. Solid-State Circuits, vol. 29, no. 7, pp. 780–786, Jul. 1994.

[15] P. Prashanth and P. Swamy, "Architecture of adders based on speed, area and power dissipation," in Proc. World Congr. Inf. Commun. Technol. (WICT), Dec. 2011, pp. 240–244.

[16] M. J. Zavarei, M. R. Baghbanmanesh, E. Kargaran, H. Nabovati, and A. Golmakani, "Design of new full adder cell using hybrid-CMOS logic style," in Proc. 18th IEEE Int. Conf. Electron., Circuits Syst. (ICECS), Dec. 2011, pp. 451–454.

[17] I.Hassoune, D. Flandre, I. O'Connor, and J. Legat, "ULPFA: A new efficient design of a power-aware full adder," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 57, no. 8, pp. 2066–2074, Aug. 2010.

[18] K. Navi, M. Maeen, V. Foroutan, S. Timarchi, and O. Kavehei, "A novel low-power full-adder cell for low voltage," VLSI J. Integr., vol. 42, no. 4, pp. 457–467, Sep. 2009.

[19] M.Aguirre-Hernandez and M. Linares-Aranda, "CMOS full-adders for energy-efficient arithmetic applications," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 19, no. 4, pp. 718–721, Apr. 2011.

[20] J. L. Wyatt, Jr., "Signal propagation delay in RC models for interconnect," in Circuit Analysis, Simulation and Design, Part II, VLSI Circuit Analysis and Simulation, vol. 3, A. Ruehli, Ed. Amsterdam, The Netherlands: North Holland, 1987, ch. 11.

[21] M. Alioto, G. Di Cataldo, and G. Palumbo, "Mixed full adder topologies for high-performance low-power arithmetic circuits,"

Microelectron. J. vol. 38, no. 1, pp. 130–139, Jan. 2007.

[22] X. Wu and F. Prosser, "Design of ternary CMOS circuits based on transmission function theory," Int. J. Electron., vol. 65, no. 5, pp. 891–905, 1988.

[23] Vinukonda Srilatha, Rakesh K,"Design of Low Power TPG with LP-LFSR", International Journal of Computer Engineering in Research Trends(ijcert), Vol.1,no.6,pp480-484.