International Journal of Engineering Research and General Science Volume 5, Issue 1, January-February, 2017
ISSN 2091-2730

Analysis and Performance Evaluation of 1-bit Full Adder Using Different Topologies

Vinny Wilson

M.Tech Graduate, SSET Karukutty, MG University, vinnywilson2012@gmail.com, 8943174331

Abstract— An adder is a digital circuit that performs addition of numbers and it plays an important role in today's digital world. In processors and other kinds of computing devices, Adders are used in the arithmetic logic units. They are also utilized in other parts of the processors for calculating addresses, table indices, increment and decrement operations and other similar operations because it is the basic building block of on-chip libraries. Also, it can be used for the construction of many number representations and it is a trivial to modify an adder into an adder-subtractor. Full adder reduces circuit complexity and can be integrated in the calculators for addition and subtraction operations. At DSP oriented system and at networking side full adder is used mostly. Full adders can be cascaded (e.g.: ripple carry adder) easily so that one can make a cascade to add any number of bits that form the word-width of a system. In recent years, low power circuit design has been encountered as a major issue in VLSI design areas. This paper focuses on the Performance evaluation of 1-bit full adder for low power in CMOS technology using three different topologies as Static or Conventional CMOS, Gate Diffusion Input (GDI), and Hybridizing PTL (Pass Transistor Logic) techniques. This paper describes comparison of these three topologies of 1-bit full adder based on area, delay, power consumption and transistor count. Simulations are done using NgSPICE and MICROWIND DSCH tool.

Keywords— CMOS Technology, Full Adder, Conventional or static logic, GDI logic, PTL, Hybridizing PTL, NMOS, PMOS, Pull up and pull down transistors, Multiplexer, sleep transistor, Xor gate, Xnor gate, SPICE, MICROWIND.

INTRODUCTION

Need for low power VLSI chips arise from evolution forces of integrated chips. VLSI design is a modular technology originated by Carver Mead and Lynn Conway for saving microchip area by minimizing the interconnect fabric area. Low power is needed because of desirability of portable devices like cell phones batteries and in biomedical field like in heart pacemakers, however large power dissipation requires larger heat sinks hence increased area. So a new solution is to be provided for low power applications for VLSI designers. Especially, this work focuses on the reduction of power, which is showing an ever-increasing growth with the scaling down of the technologies.

A new low power design technique that solves almost all of the problems known as Gate-Diffusion-Input (GDI) is implemented. This technique allows in the reduction of power consumption, propagation delay, and area of digital circuits. The advantage of using Conventional CMOS is that it has layout regularity, high noise margins and stability at low voltage due to complementary transistor pair and smaller number of interconnecting wires and disadvantage is that it has weak output driving capability due to series transistors in output stage and consumes more power and large silicon area.

Main idea behind PTL (Pass Transistor Logic) is to use purely NMOS Pass Transistors network for logic operation. In this design style, transistor acts as switch to pass logic levels from input to output and this design requires less transistor count because one pass transistor network (either NMOS or PMOS) is sufficient to perform the necessary logic operation. Speed is increased because less number of transistors is used for design. PTL has some advantage over static CMOS that it has the capability to implement a logic function with smaller number of transistor, smaller area and less power consumption.

CONVENTIONAL OR STATIC LOGIC

The CMOS logic circuits are categorized into two categories: - static (conventional) and dynamic logic circuits. These different logic styles are used according to different design requirements such as power consumption and dissipation, speed and area. In a static logic circuit a logic value is retained by using the circuit states while in a dynamic logic circuit a logic value is stored in the form of charge.

So, in static logic circuit [1][2] each output of the gate assume at all the times the value of Boolean function implemented by the circuit. So, at every point the output will be connected to either Vdd or Gnd via a low resistance path. The static logic eliminates pre-charging and decreases extra power dissipation, thus, widely used for low power circuit designs.

Static logic is further divided into two types: - single rail and dual rail logic. In single rail logic its input is either true or complementary output signals. However dual rail uses its input either true or false and yield both true and false signals as their output at the same time. Most commonly used static logics are Pseudo-nMOS, fully CMOS, transmission gate logic (TGL), Complementary pass transistor logic (CPL), Gate diffusion input logic (GDI), Double pass transistor logic (DPL) and Pass transistor logic (PTL).

A) STATIC FULL ADDER

In digital electronics an adder or summer, is a circuit that performs addition of any numbers. An adder is used to calculate address, table indices and similar operations in other parts of processors. A full adder is a combinational circuit that performs the arithmetic sum of three input bits: augends (Ai), addend (Bi) and carry in (Cin) from the previous adder. Its results contain the sum out, Sout and the carry out, Cout, to the next stage.

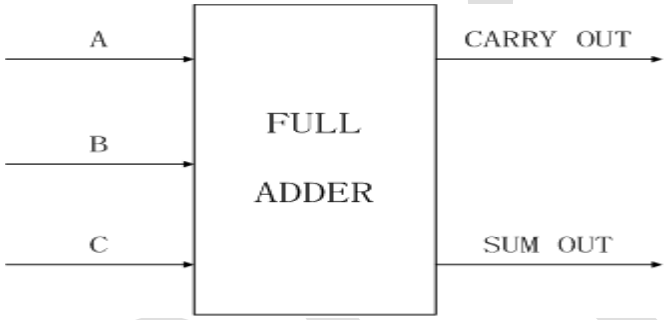

Fig: 1 Full adder block diagram

A	B	Cin	Sout	Cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Table: 1 Truth table of full adder

The Boolean equations of a full adder are given by:

Sout = ABC + AB'C' + A'B'C + BA'C' = (AB'+BA') C +AB+A'B') C'
Sout = A xor B xor C
Cout = AB + AC + BC = AB + C (A xor B)

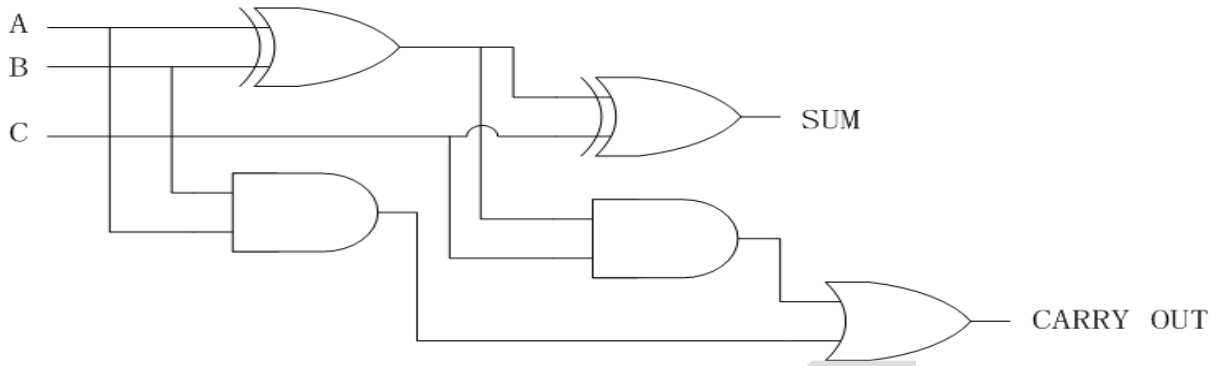

Fig: 2 Gate level representation of full adder

B) 28T STATIC LOGIC

Full adder is a circuit that performs the addition of a given three 1-bit inputs A, B, C and two 1-bit outputs sum and carry. Many full adder cells designed with the help of static logic.

$$\text{Sum} = A \oplus B \oplus C$$
$$\text{Sum} = A'B'C + AB'C + A'BC' + ABC$$
$$\text{Sum} = ABC + (A+ B + C)\ \text{Carry bar}$$

$$\text{Carry} = AB + BC + CA$$
$$\text{Carry} = AB+C\ (A+B)$$

This CMOS full adder cell uses 28 transistors which consist of 14 pull-up and 14 pull-down networks. Complementary transistor pairs make the circuit design simple but it makes some complexity in the layout area. Complementary CMOS (CCMOS) generates carry through a static gate. Due to complementary transistor pair and smaller number of interconnecting wires this CCMOS design has layout regularity, high noise margins and stability at low voltage. Disadvantage is that in this design it uses Cout signal inorder to generate the sum output which is responsible for the production of an unwanted additional delay in the circuit. It has weak output driving capability due to series transistors in output stage and consumes more power and large silicon area.

Cout is generated first using above Cout equation. Then the sum is derived from the sum equation shown in above. One of the most significant advantages of the 28T full adder was its high noise margins and was responsible for reliable operation at low voltages. The layout of CMOS gates get somewhat simplified due to the complementary transistor pairs. But the use of same number of pull-up and pull-down transistors results in high input loads, increase in power consumption and larger silicon area.

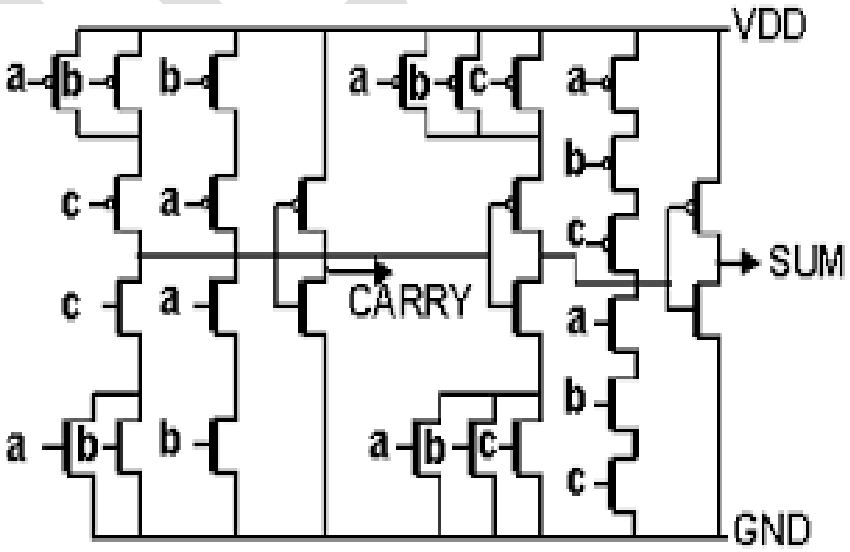

Fig: 3 Static 1-bit full adder

GDI LOGIC

Gate Diffusion Input (GDI CELL) [6][8] make use of a pure and simple cell as shown in figure below. At first sight itself this basic cell reminds us the standard CMOS inverter but there are some major and important differences:
1. GDI CELL contains three inputs: G (gate input common for both NMOS and PMOS), P (input to the source of PMOS), and N (input to the source of NMOS).
2. Bulks of both NMOS and PMOS transistors are connected to N or P inputs (respectively), so we can say that it can be arbitrarily biased at contrast with CMOS inverter.

One of the major differences between the CMOS and GDI based design techniques [9][10] is that the input P (source) of the PMOS in a GDI cell is not connected to supply voltage (VDD) and the input N (source) of the NMOS is not connected to ground (GND). This idealistic feature gives the GDI cell two extra input pins for making the GDI design more flexible than CMOS design. The basic GDI cell shown in Figure 4 was proposed and implemented by Morgenshtein. It is a new approach for the reduction of power in digital combinational circuit design. This technique helps to reduce power consumption, propagation delay and area of digital circuits while maintaining low complexity of logic design.

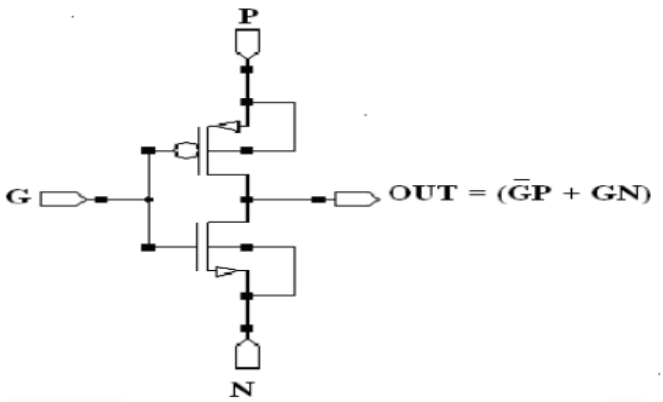

Fig: 4 basic GDI cell

Table below shows, how a simple change of the input configuration of the simple GDI cell corresponds to very different Boolean functions. Most of these functions are complex (6-12 transistors) in CMOS, as well as in standard PTL (Pass Transistor Logic) implementations, but very simple (only 2 transistors per function) in GDI design method. A simple change of the input configuration of simple Gate Diffusion Input (GDI) CELL is shown in table corresponds to six different Boolean functions.

N	P	G	D	FUNCTION
0	B	A	A'B	F1
B	0	A	AB	AND
1	B	A	A+B	OR
B	1	A	A'+B	F2
0	1	A	A'	NOT
C	B	A	A'B+AC	MUX
B'	B	A	AB'+A'B	XOR
B	B'	A	AB+A'B'	XNOR

Table: 2 Some logic functions that can be implemented using a single GDI cell

XOR and XNOR functions are the key elements and operations in adder equations. If the design, implementation and generation of them is optimized, this will greatly affect the performance and function of the full adder cell. In this new cell has used the GDI technique for generating of XOR and XNOR functions. It uses only eight transistors to generate the balanced XOR and XNOR functions, as shown in figure 5.

XOR =A'B+AB' and XNOR=AB+A'B'

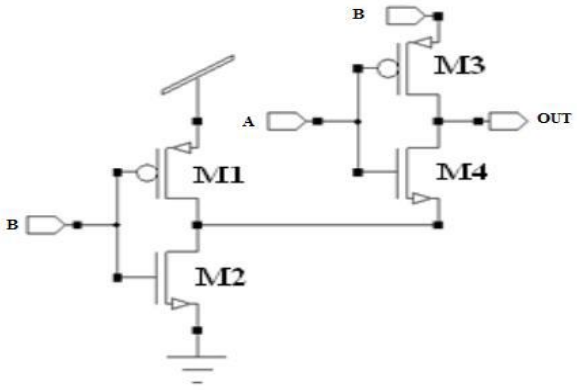

Fig: 5 XOR cell with GDI technique

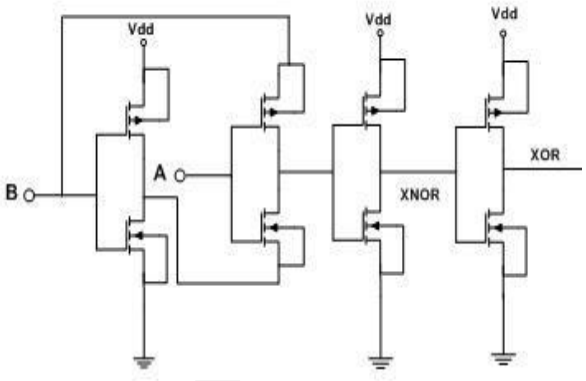

Fig: 6 XNOR/ XOR cell with GDI technique

A one-bit full adder takes three one-bit inputs: A, B and Cin and generate two outputs: sum and carry.

Sum= (A xor B xor Cin)
Carry= A. B + Cin (A xor B)

The goal is to design a high performance and low power full adder cell with the GDI technique [11][12]. In the first stage of this cell, the GDI technique is used for generating of XOR and XNOR functions. This stage shows full swing with low voltage. These complementary outputs, together with other inputs, will be fed to the second stage. The Sum and Carry are generated from the second stage of the cell design. Adder cells are normally cascaded to form a normal arithmetic circuit and their capabilities must be ensured with proper design approaches. Fig: 5 show the implementation of XOR gate using GDI technique. It is the main building block of full adder circuit. So if we can optimize XOR gate then it can improve the overall performance of the 1 bit full adder circuit. It uses less number of transistors as compared to conventional or static CMOS logic design of XOR gate. Fig: 7 show the detail circuit of GDI XOR based 10T Full Adder.

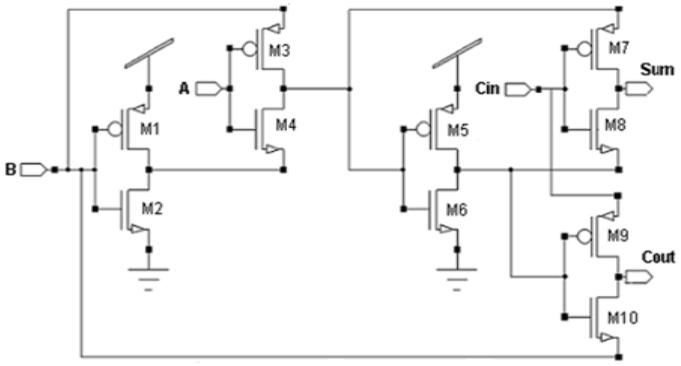

Fig: 7 1-bit full adder (10 T) using GDI technique

In fig: 7 M1, M3, M5, M7, and M9 are the PMOS transistor whereas M2, M4, M6, M8, and M10 are the NMOS transistors. A, B and Cin are taken as input and output of the circuit is drawn from the Sum and Cout.

The proposed GDI 11-T Full Adder is shown in Fig: 8 M1, M3, M5, M7, and M9 are the PMOS transistor. M2, M4, M6, M8, and M10 are the NMOS transistors. A, B and Cin are the inputs of the full adder and output of the circuit is taken from the Sum and Cout nodes. Here M11 act as a sleep transistor which is responsible for the average power consumption reduction of the entire circuit. When the circuit is in active mode, sleep transistor M11 will be in OFF state and when the circuit is in standby mode, the sleep transistor will be in ON state. By using modified GDI based 11T full adder design, there will be a drastic reduction in the power consumption and the delay of the circuit compared to GDI 10T based full adder circuit.

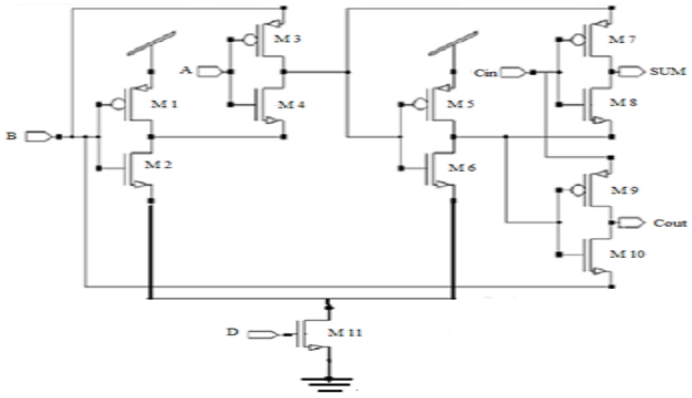

Fig: 8 11T full adder using GDI technique

"Sleep transistors" are designed to minimize the power consumption and heat dissipation of a given circuit. Basically, the chip will contain specialized circuitry that will shut down some specific areas of the chip that aren't immediately needed for the proper function of the circuit. The problem with this approach is that it will degrade the performance of the circuit to a small extend, and will lead to an increase in the number of transistors.

The sleep transistors can be implemented as: "coarse-grain" or "fine-grain" power gating styles. In the "fine-grain" power gating implementations, the sleep transistor will be accomplished inside each and every standard cell which is often called MTCMOS cell. The advantage of the fine-grain sleep transistor approach is that the virtual power networks (VVSS or VVDD) are very short and are hidden in the circuit during the time of implementation. However, the fine-grain sleep transistor implementation adds a sleep transistor to every cell that results in significant area increase. In the "coarse-grain" power gating design styles, the sleep transistors are connected together between the permanent and the virtual power supply networks. One of the main advantage of the "coarse-grain" power gating is that sleep transistors share (charged or discharged) current. Also, the area overhead is significantly smaller among sleep transistors due to their charge sharing behavior. Most power-gating designs are based on "coarse-grain" sleep transistor implementation because the "fine-grain" implementation which incurs large area penalty.

PASS TRANSISTOR LOGIC

In electronics, pass transistor logic (PTL) [3][4] describes several logic families used in the design of IC chips. It reduces the number of transistors used in the construction of different logic gates, by eliminating redundant transistors. Here transistors are used as switches inorder to pass the logic levels between the nodes of a circuit, instead of switches that are connected directly to supply voltages. This reduces the number of active devices, but it has the disadvantage that the voltage difference between high and low logic levels will dramatically decrease at each stage. Each series transistor is less saturated at its output than at its input. When several devices are chained in series in a logic path, inorder to restore the signal voltage to a full value a conventionally constructed gate may be required. By contrast, conventional CMOS logic switches the transistors so that the output connects to one of the power supply rails and the logic voltage levels in a sequential chain do not decrease or reduce drastically. Circuit simulation may be required or it is essential to ensure adequate performance and functionality of the circuit.

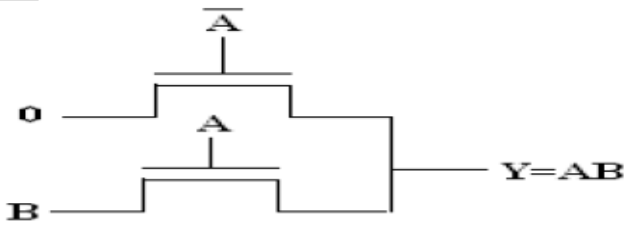

Fig: 9 AND gate using PTL logic

Pass transistor logic often uses fewer transistors and thus it runs faster. It requires less power than the same function implemented with the same transistors in fully complementary CMOS logic. XOR has the worst-case Karnaugh map -- if implemented from simple

gates, it requires more transistors than any other function. The designers and many other chip manufacturers save a few transistors by implementing the XOR gate using pass-transistor logic rather than simple gates.

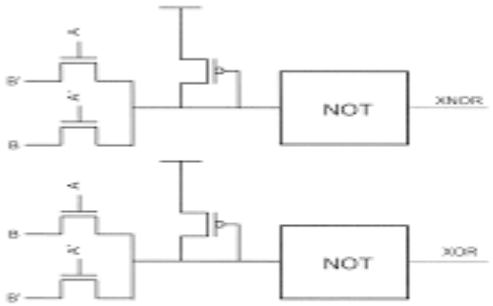

Fig: 10 XOR gate using PTL logic

Some writers use the term "Complementary Pass Transistor Logic" (CPL) to point out the style of implementing logic gates that uses transmission gates which consist of both NMOS and PMOS pass transistors. Other authors use the term CPL to indicate a style of implementing logic gates where each gate consists of a NMOS-only pass transistor network, followed by a CMOS output inverter. Other authors use the term CPL to indicate a style of implementing logic gates using dual-rail encoding. Every CPL gate has two output wires, both the positive signal and the complementary signal, eliminating the need for inverters.

A) HYBRIDIZING PTL 9T FULL ADDER

The design of hybridizing PTL full adder consists of three modules. Module 1 consists of an XOR –XNOR module which was implemented by 5 transistors. Module 1 produces two intermediate signals which are given as the inputs to the module 2 and module 3 to obtain sum and carry output as shown in Fig: 11. Module 2 and 3 are GDI 2:1 MUX with different input and select lines which produce carry and sum respectively. Module 1 produces two outputs: *A* xnor *B* and A xor B. For module 2, *A* xor *B* act as a select line and A, C as inputs. On the other hand for module 3, C act as a select line and AOB and *A* xor *B* as inputs. Proposed full adder design is hybridized design because two different logic styles have been used to make the full adder. Module 1 was implemented by PTL logic and GDI technique has been used for the implementation of module 2 and 3.

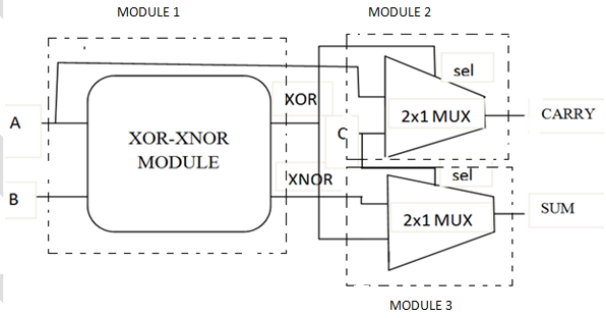

Fig: 11 Proposed adder schematic

Hybridizing PTL full adder has been implemented by using only 9 transistors i.e. five transistors are used in module 1 stage and module 2 and 3 uses 2T GDI multiplexer. Sum is realized by module 1 and module 3 as per equation 1 and carry is realized by module 1 and module 2as per equation 2. Module 1 has been implemented by proposed 5T XOR-XNOR module and GDI technique has been used for module 2 and 3. In module 1 a XOR-XNOR cell is used to drive the selection lines and control signal lines of the multiplexer in module 2 and 3. The XOR-XNOR module in Fig: 12 have been designed by PTL logic and consist only 5 transistors which shows least transistor count as compared to all previous discussed designs.

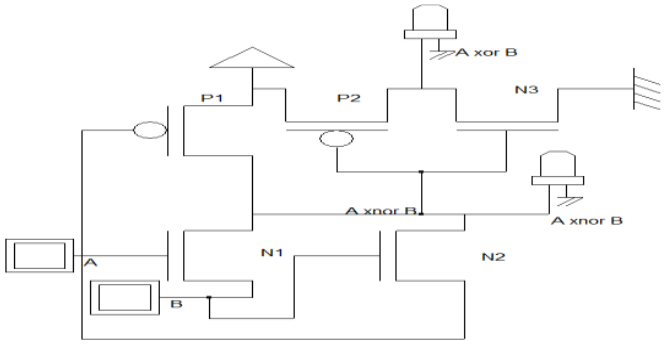

Fig: 12 Proposed PTL based XOR-XNOR Module

Proposed design XOR-XNOR module has been made by 3 NMOS and 2 PMOS transistors which provide an area efficient circuit design as compared to previous discussed design models. MOS logic states on four different input combination has been shown in table 1 for both XOR and XNOR output. Comparative analysis of proposed XOR-XNOR module in terms of area with other existing XOR-XNOR module has been shown in Table: 3.

A	B	N1	N2	N3	P1	P2	A xor B	A xnor B
0	0	OFF	OFF	ON	ON	OFF	0	1
0	1	OFF	ON	OFF	ON	ON	1	0
1	0	ON	OFF	OFF	OFF	OFF	1	0
1	1	ON	ON	ON	OFF	ON	0	1

Table: 3 Analysis of Proposed XOR-XNOR Module

Verification and simulation of the functionality of proposed XOR-XNOR module is first done by using DSCH 3.5 designing tool. Channel width should be accurate for efficient working of the design. Channel width can be changed in DSCH schematic editor. The timing simulated for 5T XOR –XNOR module will show the accurate functioning of the proposed design.
The 5T XOR-XNOR module has been compared with the previous discussed XOR-XNOR designS in terms of area in Microwind designing tool. Microwind deals with both front end and back end designing. In front end it has DSCH in which both transistor level and gate level designing can be done. DSCH generate a Verilog file which can be compiled by the Microwind back end designing tool to get power and area consumption. The 5T XOR-XNOR module is compared with the discussed XOR-XNOR designs in terms of area on 45nm technology.

Outputs of module 1 act as inputs for the module 2 and module 3. The logical Boolean expression for module 2 and 3 can be expressed as:

SUM = C(A xnor B) + C bar(A xor B)
CARRY = C(A xor B) + A(A xnor B)

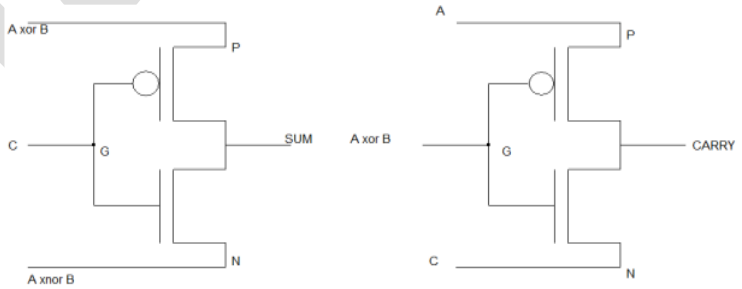

Fig: 13 Use of GDI cell as Module 2& 3

GDI technique is an efficient technique for designing area and power efficient digital circuits as compared to PTL, TG, CPL and DPL designing approaches. Schematic of proposed full adder has been designed and simulated in DSCH 3.5 logic editor and simulator.

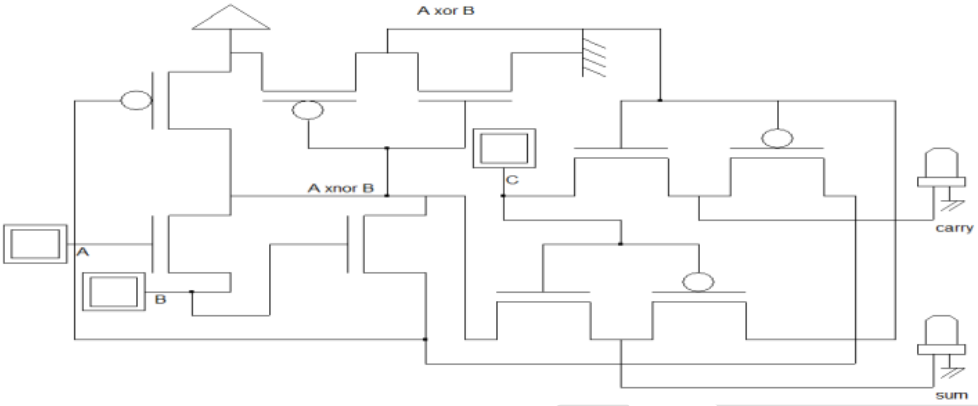

Fig: 14 9T Hybridizing PTL Full Adder Design

B) 8T FULL ADDER DESIGN

In the proposed 8T full adder [8] sum is generated using 3T XOR module twice, and carry is generated using NMOS and PMOS pass transistor logic devices as shown in Fig.15. The equations are modified so as to visualize the 8T full adder design. The modified equations for 8T full adder design are:

$$SUM = A \text{ xor } B \text{ xor } C$$

$$= (A \text{ xor } B) \text{ xor } C$$

$$CARRY = AB + BC + CA$$

$$= AB + BC\,(A + \bar{A}) + AC\,(B + \bar{B})$$

$$= AB + (A \text{ xor } B)\,C$$

$$= (\bar{A} * B)\bar{B} + (A \text{ xor } B)\,C$$

Instead of using two NMOS pass transistor devices we have used one NMOS and one PMOS pass transistor device, because of ease of the design and as according to the equation as shown in Fig.15.

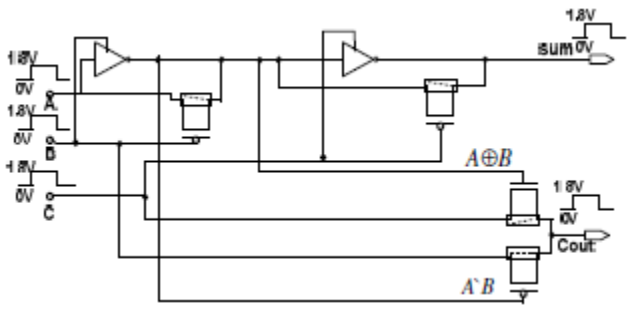

Fig: 15 8T Hybridizing PTL Full Adder Design

It must be noted that PMOS transistor passes ' l' clearly, but cannot pass '0' completely thus, the carry output has weak '0'. NMOS transistor passes '0' very precisely, but cannot pass '1' completely therefore, the carry output has weak ' 1 ', Having weak '0' and '1' at carry outputs is one of the disadvantages of 8T full adder circuit and also the PMOS transistor (5) count is high compared to NMOS transistor (3) count. In practical situations, one solution for this problem is using an inverter at carry output, but this solution will increase the power and area of the circuit.

SIMULATION RESULTS

It is difficult to implement a 28T full adder using DSCH tool. By using NgSPICE we can solve that problem. All other simulations were done using MICROWIND DSCH tool. 45nm technology was used for this purpose. Power consumption is calculated by the equation,

Power, P = Supply voltage (Vdd) * Current (I)

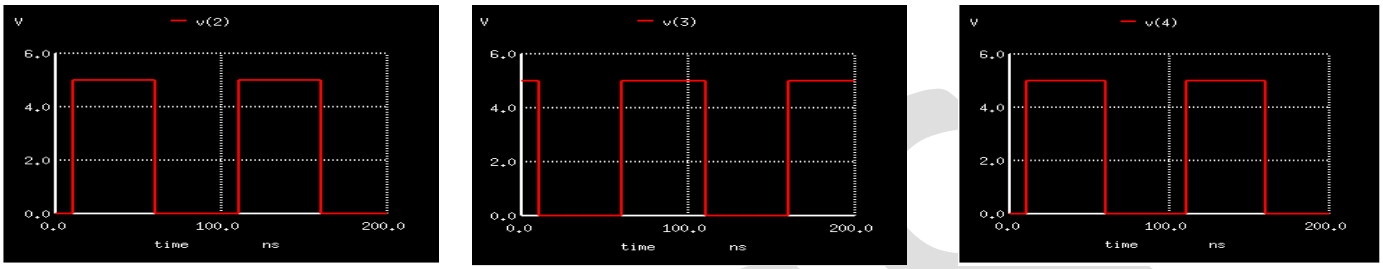

Fig: 16 Simulation of 28T Static adder using NgSPICE a) input A b) input B c) input C

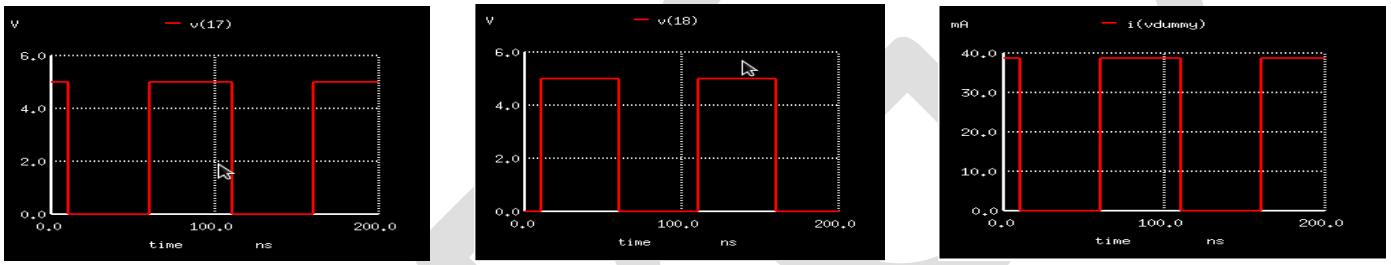

Fig: 17 a) sum b) carry c) current at sum and carry node

A) MICROWIND SIMULATION OF 10T FULL ADDER USING GDI

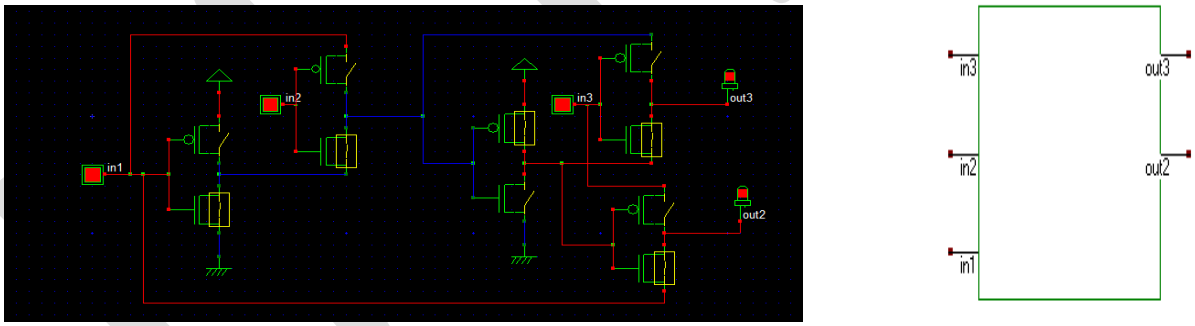

Fig: 18 a) Transistor Level Simulation b) Schematic symbol

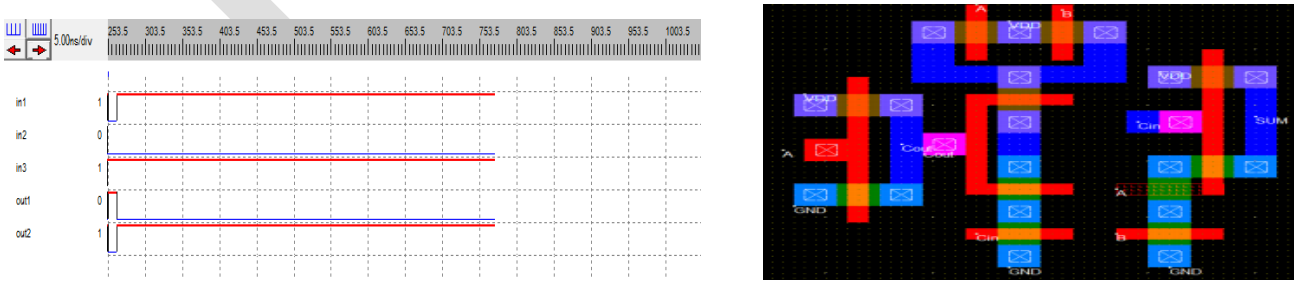

Fig: 19 a) Timing Diagram b) Layout

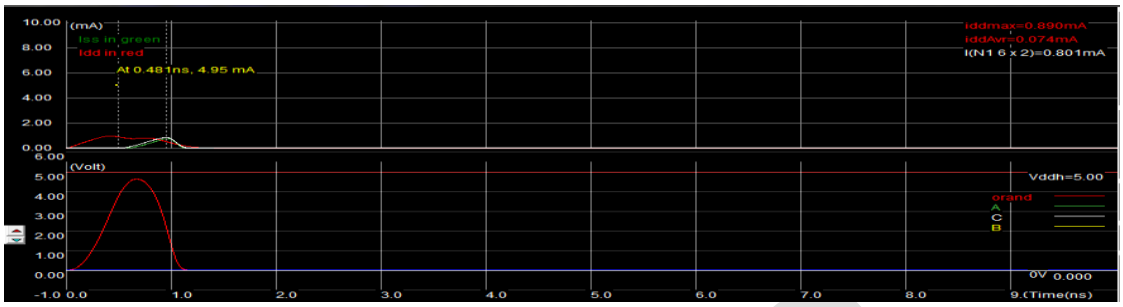

Fig: 20 Voltage, Current versus Time

B) MICROWIND SIMULATION OF 11T FULL ADDER USING GDI

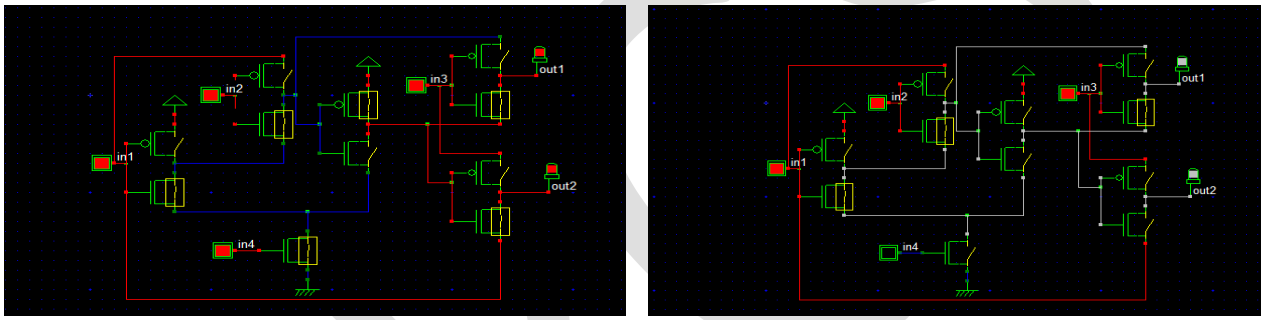

Fig: 21 a) Sleep transistor working as in4 b) Sleep transistor not working

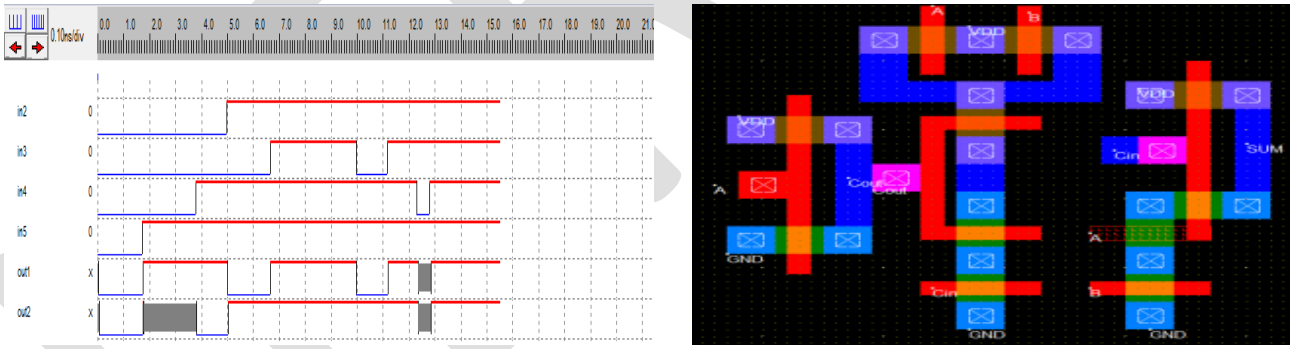

Fig: 22 a) Timing Diagram b) Layout

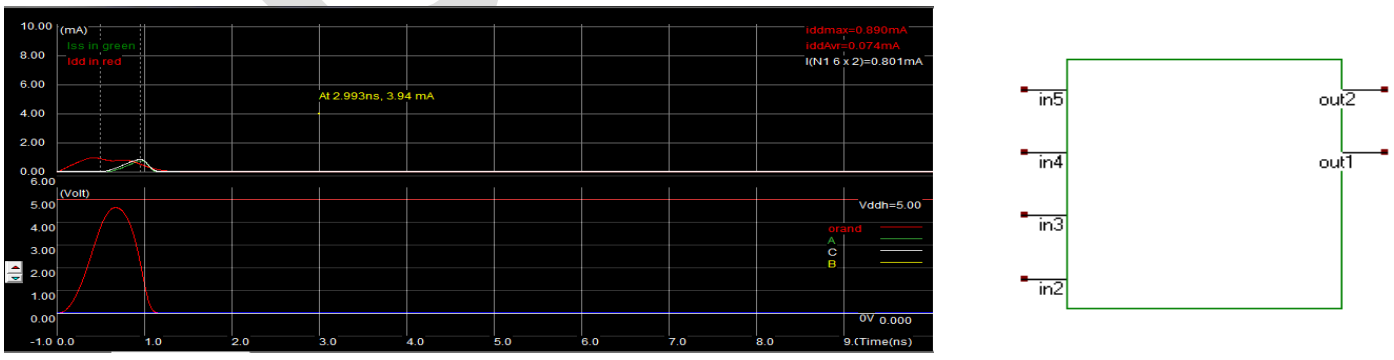

Fig: 23 a) Voltage, Current versus Time b) Schematic Symbol

C) MICROWIND SIMULATION OF 9T FULL ADDER USING HYBRIDIZING PTL AND GDI LOGIC

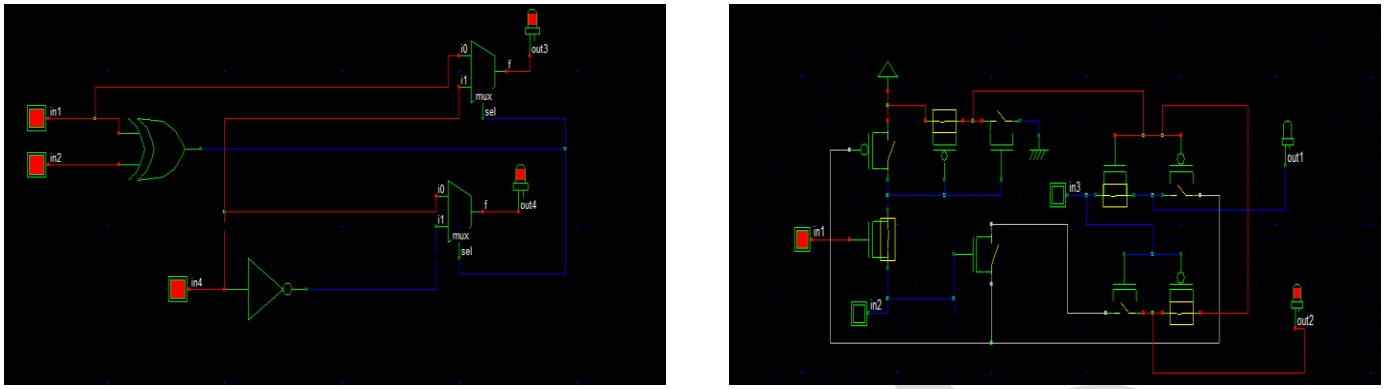

Fig: 24 a) Gate level simulation b) Transistor level simulation

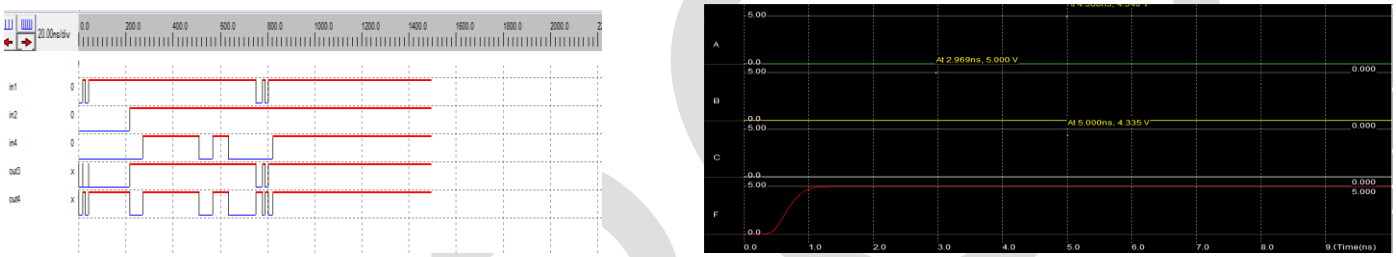

Fig: 25 a) Timing diagram b) Voltage Vs Time

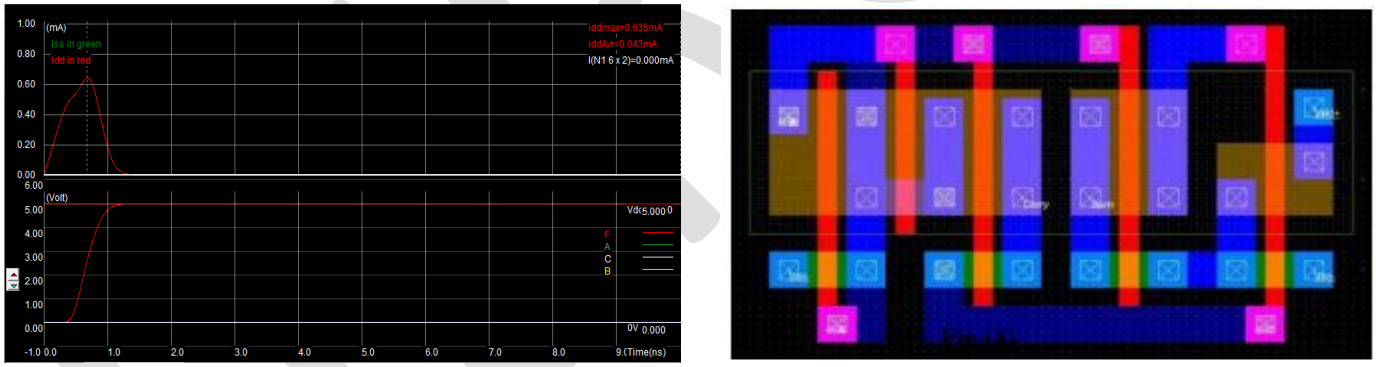

Fig: 26 Voltage, Current Vs Time b) Layout

D) MICROWIND SIMULATION OF 8T FULL ADDER USING HYBRIDIZING PTL AND GDI LOGIC

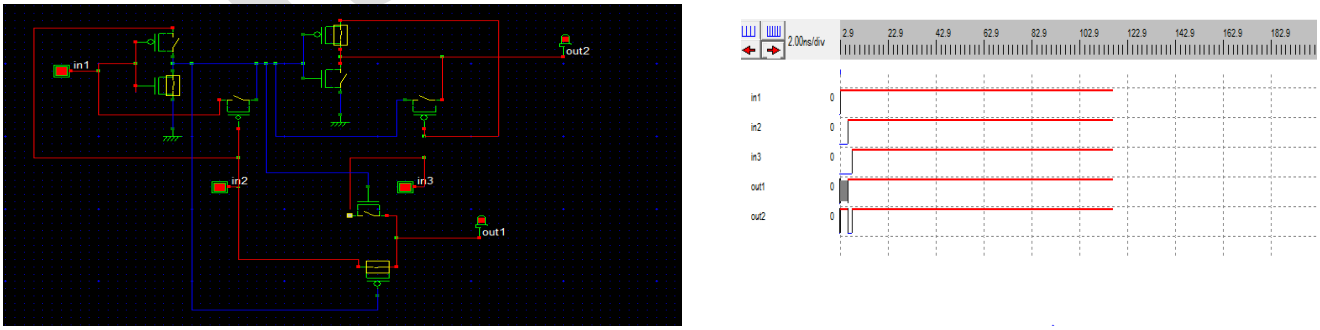

Fig: 27 a) Transistor level simulation b) Timing diagram

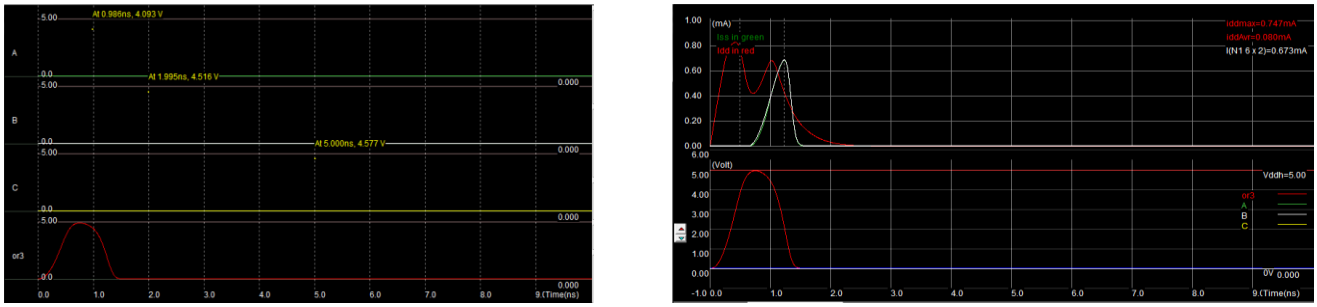

Fig: 28 a) Voltage Vs Time b) Voltage, Current Vs Time

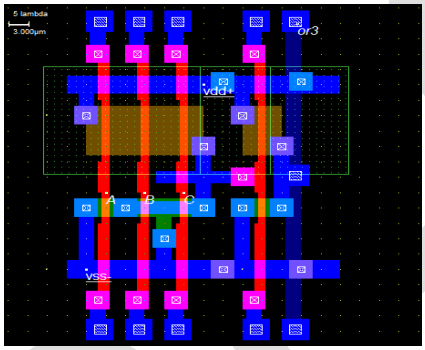

Fig: 29 Layout

E) MICROWIND SIMULATION OF 28T STATIC FULL ADDER

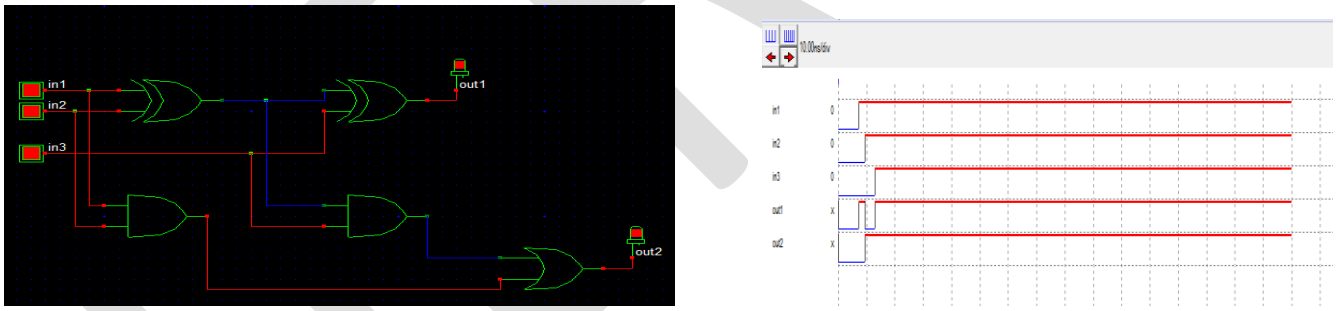

Fig: 30 a) Gate Level Simulation b) Timing Diagram

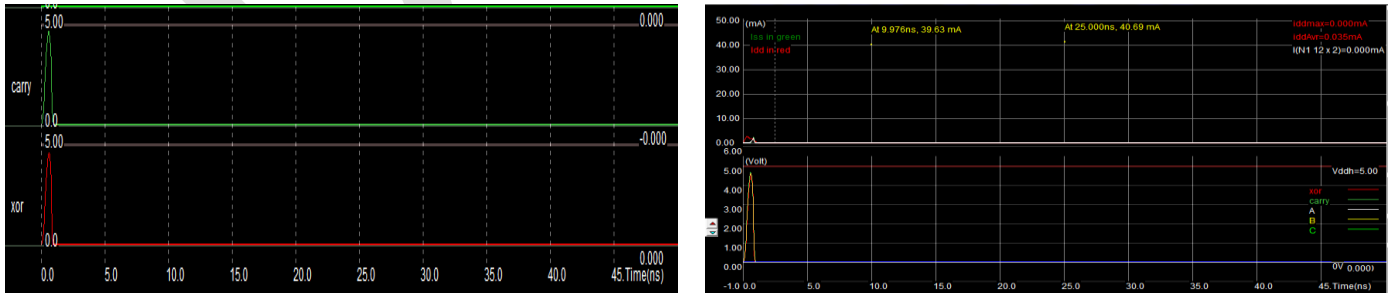

Fig: 31 a) Voltage Vs Time of output (same for all simulations) b) Voltage, Current Vs Time

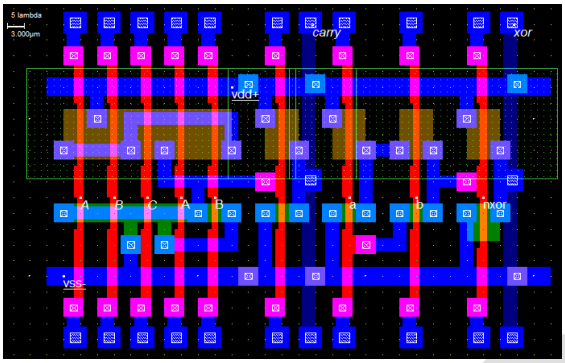

Fig: 32 Layout

	STATIC LOGIC	GDI LOGIC WITH SLEEP TRANSISTOR	GDI LOGIC W/O SLEEP TRANSISTOR	PTL & GDI	PTL & GDI
NO:OF TRANSISTORS	**28T**	**11T**	**10T**	**9T**	**8T**
NMOS	**14**	**6**	**5**	**5**	**3**
PMOS	**14**	**5**	**5**	**4**	**5**
VOLTAGE	**5V**	**5V**	**5V**	**5V**	**5V**
CURRENT(mA)	**39.63**	**3.94**	**4.95**	**0.68**	**0.89**
POWER (mW)	**198.15**	**19.70**	**24.95**	**3.40**	**4.45**

Table: 4 Simulation Table

Layout area was calculated as per design rules (in terms of lambda). 28T static full adder has highest Layout area & it has highest power dissipation. Here the supply voltage was taken as 5V for three topologies. So, the delay didn't change very much. When supply voltage is reduced delay will increase drastically. Also, we can compare 1-bit full adder design with 7T and 6T full adder circuits.

CONCLUSION

In this paper conventional complementary metal oxide semiconductor (CMOS), gate diffusion input (GDI) and hybridizing PTL adder circuits are analyzed in terms of power. In this paper I have presented three different logic circuits in which power delay product is improved more than 50% compared with that of conventional CMOS circuits. The simulation result and comparative performance revealed that power dissipation in hybridizing PTL logic produces considerably lower power than conventional CMOS logic. Also, the transistor count decreased drastically from 28 to 9 which in-turn helps to reduce the layout area of the circuit with better performance.

REFERENCES:

[1] S. Hanson, B. Zhai, K. Bernstein, D. Blaauw, A. Bryant, L. Chang, K. K. Das, W. Haensch, E. J. Nowak and D. Sylvester, "Ultralow-voltage, minimum-energy CMOS," IBM Journal of Research and Development, vol. 50, no. 4-5, pp. 469–490, 2006.

[2] Sumeer Goel, Ashok Kumar, Magdy A. Bayouni, "Design of Robust, Energy-Efficient Full Adders for Deep-Submicrometer Design Using Hybrid-CMOS Logic Style," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 14, No. 12, pp. 1309- 1321,2006.

[3] Subodh Wairya, Rajendra Kumar Nagaria, Sudarshan Tiwari, "Comparative Performance Analysis of XOR/XNOR Function Based High-Speed CMOS Full Adder Circuits For Low Voltage VLSI Design", International Journal of VLSI Design & Communication System, pp.-221-242, 2012.

[4] R. Zimmermann and W. Fichter, "Low –power logic styles: CMOS versus pass-transistor logic," IEEE J. Solid-State Circuits, Vol. 32, , pp.1079-90, July 1997.

[5] Khatibzadeh and K. Raahemifar, "A Study and Comparison of Full Adder Cells based on the Standard Static CMOS Logic.", IEEE CCECE 2004 – CCGEI 2004, Niagara Falls, May 2004.

[6] Morgenshtein, A.; Fish, A.; Wagner, A., "Gate diffusion input (GDI)-A novel power efficient method for digital circuits: A Design Methodology," IEEE International Conference, pp. 39 – 43, 2001.

[7] Mohammad Javad Zavarei, Mohammad Reza Baghbanmanesh, Ehsan Kargaran, Hooman Nabovati, Abbas Golmakani, "Design of New Full Adder Cell Using Hybrid- CMOS Logic Style", IEEE International Conference on Electronics, Circuits and Systems (ICECS), pp.-451-454, Nov 2011.
[8] Karthik Reddy.G and Kavita Khare, "Low Power-Area GDI & PTL Techniques Based Full Adder Designs", IEEE International Conference on Electronics, Circuits and Systems (ICECS), pp.-249-257, Oct 2013.
[9] Anjali Sharma and Rajesh Mehra, "Area and Power Efficient CMOS Adder Design by Hybridizing PTL and GDI Technique", International Journal of Computer Applications (0975 – 8887) Volume 66– No.4, March 2013.
[10] Pankaj Kumar and Poonam Yadav, "Design and Analysis of GDI Based Full Adder Circuit for Low Power Applications", International Journal of Engineering Research and Applications ISSN: 2248-9622, Vol. 4, Issue 3 (Version 1), pp.462-465, March 2014.
[11] J. Dhanasekar and et.al, "Design of Full Adder using PTL & GDI Technique", International Journal of Advanced Research Trends in Engineering and Technology (IJARTET), Vol. 2, Issue 3, March 2015.
[12] Arkadiy Morgenshtein,; Michael Moreinis and Ran Ginosar, "Asynchronous Gate Diffusion- Input (GDI) Circuits," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol.12, No.8, August 2004.

IJERGS