

Designing Of Novel Low Power Signed And Unsigned Multiplier Using 180nm CMOS Technology In CADENCE

V.G.Sanathi Swaroop¹, E.Pavani², Ch.Vasundhara³

¹Assistant Professor, SSCE, SRIKAKULAM.

²UG Student, SSCE, SRIKAKULAM.

³UG Student, SSCE, SRIKAKULAM.

Abstract: Power consumption is the bottle neck of system performance in VLSI design. Minimization of power consumed by the circuit tends to improve the performance and reduce the cost of the system. Power consumption is mainly due to increased number of transistors and leakage power. The reduction of transistor count and leakage power is done by using a technique like “GATE DIFFUSION INPUT”. Multipliers are vital components of any processor (or) computing machine. Performance of microcontrollers and digital signal processors are evaluated on the basis of number of multiplications performed in unit time. Hence, better multiplier architectures are bound to increase the efficiency of the system. This paper presents different types of Multiplier architectures based on “GDI technique” and Low power Baugh-Woolley multiplier is proposed for both unsigned and two’s complement signed multiplication. The total architecture is designed in 180nm CMOS technology using CADENCE tool and analyze power dissipation.

Keywords: Array multiplier, Tree multiplier, Baugh-Woolley multiplier, GDI technique, Cadence virtuoso tool, Low power consumption.

1. Introduction

Electronics as we know today is characterized by reliability, low power dissipation, extremely low weight and volume, and low cost, coupled with an ability to cope easily with high degree of sophistication and complexity. MULTIPLIER is an electronic circuit used extensively in modern processors. It is building block of many high speed systems such as ‘Digital signal processor’. The performance of any digital system is mainly evaluated by performance of the multiplier. Present processor aim is to design low power multiplier. Therefore power consumption and dissipation of multiplier should be minimized. Power dissipation in a CMOS circuit is determined by two components.

1) Static power dissipation , 2) Dynamic power dissipation

Static power dissipation becomes an issue when the circuit is inactive mode or in a power down mode. Dynamic power dissipation occurs when the circuit is in operation mode, i.e. the circuit performing some task.

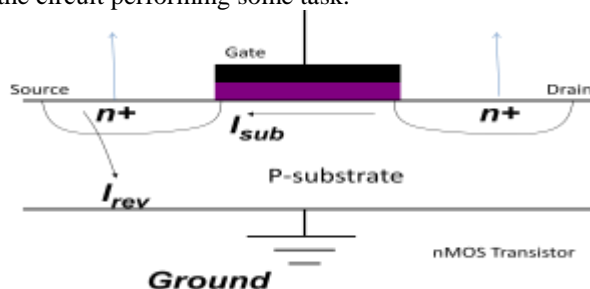


Figure 1: leakage power components in NMOS transistor

The static power components become important when the circuits are rest, i.e. when there is no activity in the circuits. It is defined as product of supply voltage and leakage Current. The static power dissipation includes Sub-threshold leakage current and Reverse biased diode leakage current. Sub-threshold leakage current has a strong dependence on the threshold voltage. It is the current between source and drain of the MOSFET when transistor is in sub-threshold region or weak- inversion region, that is, gate to source voltage is below the threshold voltage of device. The reverse bias diode leakage is due to the reverse bias current in the parasitic diodes that are formed between diffusion region of transistor and substrate. It results from minority charge carriers and drifts near the edge of the depletion regions, and also form the generation of electron hole pairs in the depletion regions of reverse bias junctions.

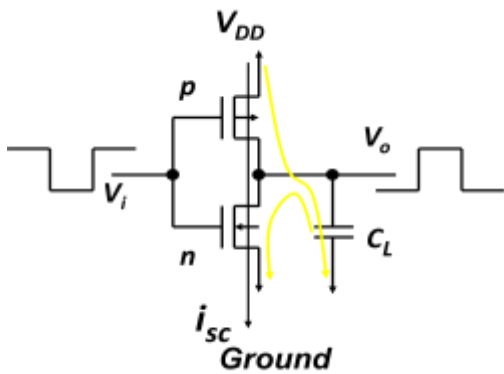


Figure 2: Dynamic power dissipation in CMOS inverter

Dynamic power dissipation can be divided into three mechanisms: switched, short circuit and glitch power dissipation. Switched power dissipation comes due to repeated charging and discharging of output capacitor. Short circuit dissipation is occur in a circuit when, both PMOS and NMOS transistors are ‘ON’. In real circuits signals have non-zero rise and fall times, which causes both P-net and N-net of CMOS gate to conduct current simultaneously. This leads to the flow of a short circuit current for a short period of time.

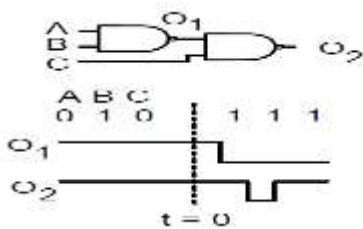


Figure 3: example for glitch power dissipation

Glitches are undesired transitions which do not contribute any useful information. In digital circuits glitch is occur before the signal settles to its intended value. In other words glitch is an electrical pulse of short duration that is usually the result of a fault or design error. As shown in fig 3, there is some delay at the output O1, which results in glitch at output O2. As there is some capacitance associated with the output O2, it leads to switching power dissipation. However glitch can be minimized by scale down the supply and threshold voltages probably at different phases.

GATE DIFFUSION INPUT is the VLSI technique to reduce the dynamic and static power dissipation in digital circuits. By using this technique various digital circuits can be designed with low transistor count as compare to CMOS designs which results in low power dissipation.

2. GATE DIFFUSION INPUT TECHNIQUE

Gate diffusion input technique is named itself because of one of the inputs are directly diffused into the gates of NMOS and PMOS transistors. GDI reduces power dissipation and area of digital circuits. This method is based on the simple cell which looks exactly like basic inverter. It is shown in fig 4.

The GDI cell contains 3 inputs.

- 1) G (Common input to the gate of PMOS and NMOS)
- 2) N (input to the source/drain of NMOS)
- 3) P (input to the source/drain of PMOS)

In GDI technique N,P,G terminals are could be given to a power supply ‘Vdd’ or can be grounded or can be supplied with input signal depending upon the circuit to be designed and hence effectively minimizing the number of transistors used in case of most logic circuits (eg. AND, OR, NOT, XOR, etc). Bulks of PMOS and NMOS are connected to the ‘P’ or ‘N’ terminals respectively.

2.1 Basic Logic Functions

The implementation of different logic functions using GDI technique are as follows.

TABLE I
FUNCTIONS THAT CAN BE IMPLEMENTED USING BASIC GDI CELL

N	P	G	Out	Function
0	1	A	A'	INVERTER
0	B	A	A'B	FUNCTION1
B	1	A	A'+B	FUNCTION 2
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

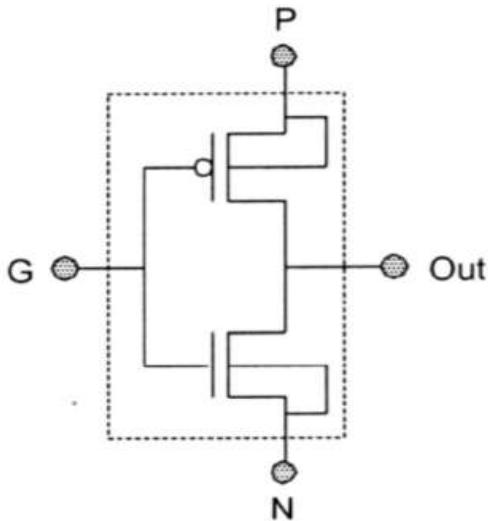


Figure 4: BASIC GDI CELL

The TABLE I shows information about the logic functions which are implemented by using GDI technique. So any logic function can be implemented with GDI technique.

The operation of GDI technique based two input AND gate can be explained with respect to basic GDI cell, 'P' of the transistor is given to the ground, it will cut-off from its operation, hence the logic either '1' or '0' at the input 'a' will be reflected at the output. Thus the output will be $a \cdot b$.

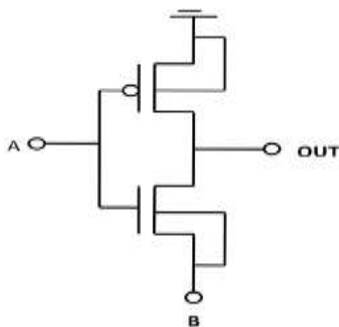


Figure 5: GDI AND GATE

To obtain the two input OR logic using GDI cell, both the inputs are given to the 'G' and 'P' terminals and power supply is given to the terminal 'N'. Thus, output will be $A+B$.

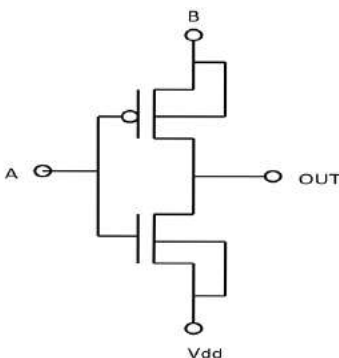


Figure 6: GDI OR GATE

To obtain the inverter logic using GDI cell, 'P' terminal is connected to the logic '1'; 'G' terminal is connected to the input 'A' and terminal 'N' is connected to the ground. Thus the output will be A' .

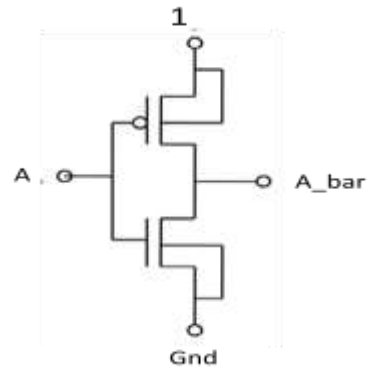


Figure 7: GDI INVERTER

To obtain the two input XOR logic using GDI cell, one input 'A' is given to the terminal 'G', second input 'B' is given to the terminal 'P', and for terminal 'N', complement of 'B' is given. Thus the output will be $(A'B+AB')$.

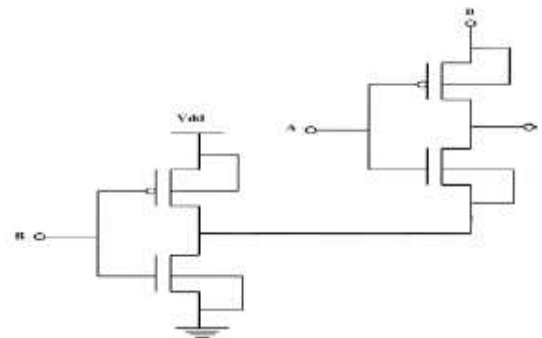


Figure 8: GDI XOR

2.2 ADDERS

By using GDI XOR and GDI AND gates we can design GDI half adder as shown in fig 9.

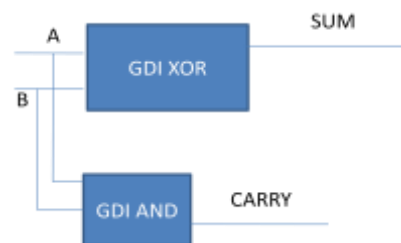


Figure 9: GDI HALF ADDER

Low power 6 transistor GDI full adders are obtained from sum and carry expressions of full adder as follows.

$$\begin{aligned}
sum &= (a \oplus b) \oplus c \\
\Rightarrow &(a \oplus b)c' + (a \oplus b)c \\
carry &= ab + bc + ca \\
\Rightarrow &(a \oplus b)a + (a \oplus b)c
\end{aligned}$$

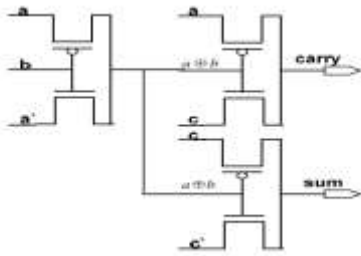


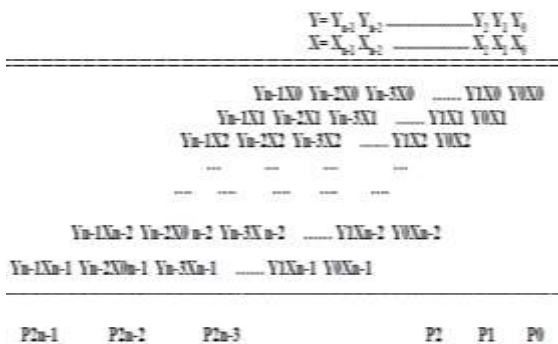
Figure 10: 6 TRANSISTOR GDI FULL ADDER

By using the logic functions based on GDI technique, we can design multiplier with low power consumption.

3. MULTIPLIER

Multiplication is one of the basic functions used in various VLSI applications. The multipliers are widely used in arithmetic logic unit, DSP processors, math processors, etc. digital multiplication is series of bit shifts and bit additions, where two numbers, the multiplicand and the multiplier are combined into a result. Considering the bit representation of the multiplicand $Y_0, Y_1, Y_2, \dots, Y_{n-1}$ and the multiplier $X_0, X_1, X_2, \dots, X_{n-1}$, in order to form the product, up to 'n' shifted copies of the multiplicand is to be added for unsigned multiplication.

The multiplication algorithm for N bit multiplicand by N bit multiplier is shown below.



In this paper we are presenting 3 multiplier approaches. They are,

- 1) Array multiplier
- 2) Tree multiplier
- 3) Signed and unsigned Baugh-Woolley multiplier

3.1 ARRAY MULTIPLIER

Array multiplier is well known due to its regular structure. The array topology is a two dimensional structure, that fits nicely on the VLSI planar process. 4-bit array multiplier

architecture is shown in fig 11. It uses short wires that go from one full adder to adjacent full adder horizontally, vertically, or diagonally. For N-bit array multiplier, 'NxN' array of AND gates can compute all the partial product terms simultaneously. The terms are summed by an array of 'N[N-2]' full adders and 'N' half adders. The shifting of partial products for their proper alignment is performed by simple routing and does not require any

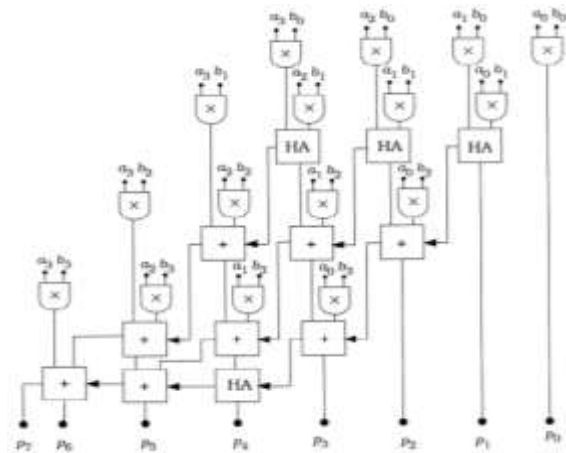


Figure 11: 4-BIT ARRAY MULTIPLIER ARCHITECTURE

logic. The number of rows in array multiplier denotes the length of the multiplier and width of each row denotes width of multiplicand. The output of each row of adders acts as input to the next row of adders. The advantage of array multiplier is comes from its regular structure. Since it is regular, it is easy to layout and has a small size. The design time of array multiplier is much less than that of a tree multiplier. The limitation of array multiplier is that they are very large. As operand size increase, linear array grows in size at a rate equal to the square of operand size. This is because number of rows in array multiplier is equal to the length of the multiplier, with the width of each row equal to width of multiplicand. The large size of full arrays typically prohibits their use, except for small operand sizes, or on special purpose math chips where a major portion of silicon area can be assigned to the multiplier array.

3.2 TREE MULTIPLIER

The first tree multiplier was introduced by C.S.WALLACE. He suggested a fast technique to perform multiplication in 1964. Tree multiplier is an extremely fast structure for summing partial products. In linear array each row sums one additional partial product. Such linear arrays require order 'N' stages to reduce 'N' partial products. In contrast by doing the additions in parallel, tree structure requires only order of 'log N' stages to reduce 'N' partial products. The result of multiplication is obtained by first generating partial products and then adding the partial products.

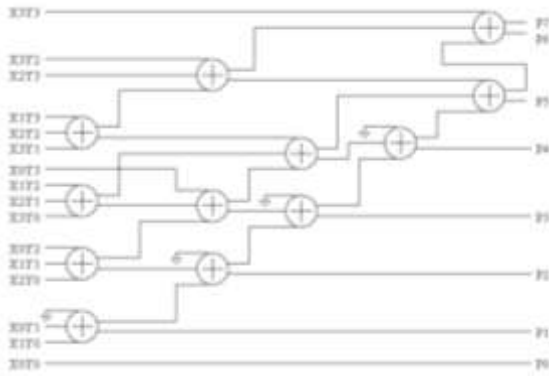


Figure 12: 4-BIT TREE MULTIPLIER ARCHITECTURE

In tree multiplier partial-sum adders are arranged in a tree like fashion, reducing both the critical path and number of adders needed as shown in fig 12 .

3.3 BAUGH WOOLLEY MULTIPLIER

Baugh and Woolley have proposed an algorithm for direct two's complement array multiplication. The principal advantage of their algorithm is that the signs of all summands are positive, thus allowing the array to be constructed entirely with the conventional Type 0 full adders. This uniform structure is very attractive for VLSI. Baugh-Woolley multiplier uses AND gates and NAND gates to generate partial products. This Baugh-Woolley multiplier multiplies both positive and negative numbers.

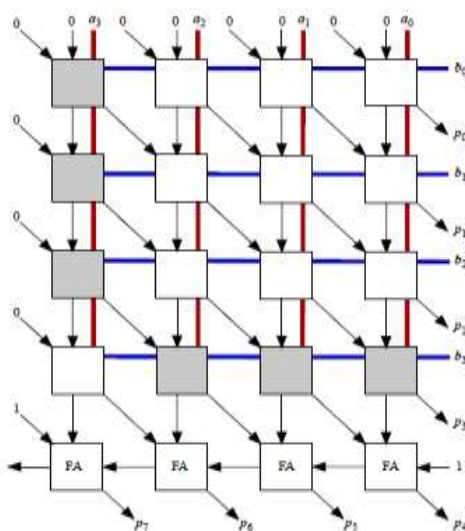


Figure 13: 4-BIT BAUGH WOOLLEY MULTIPLIER

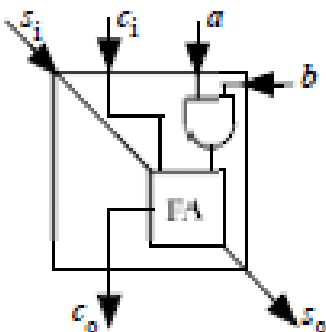


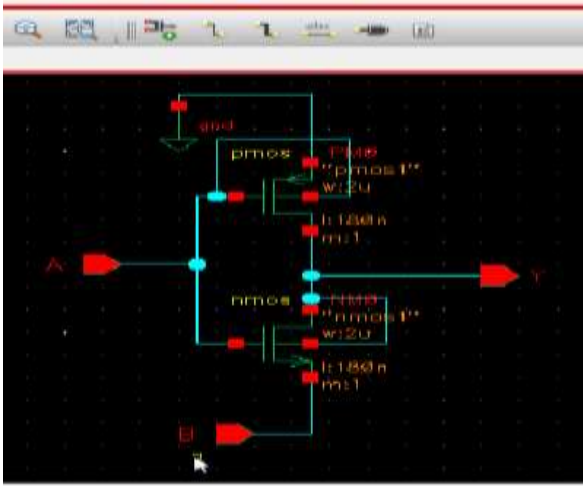
Figure 13(a): Baugh-Woolley multiplier white cell

Figure 13(b): Baugh-Woolley multiplier gray cell

4. SCHEMATICS

Schematics for GDI technique based different logic functions and 2-Bit Array, Tree, Signed and unsigned Baugh-Woolley multiplier architectures are designed in 180nm CMOS technology in CADENCE. Cadence is an Electronic Design Automation (EDA) environment. This is a powerful tool that allows the user to design circuits and simulate them. Designs very similar to Cadence are extensively used in industry and research. Cadence tool kit consists of several programs for different applications such as schematic drawing, layout, verification, and simulation. These applications can be used on various computer platforms.

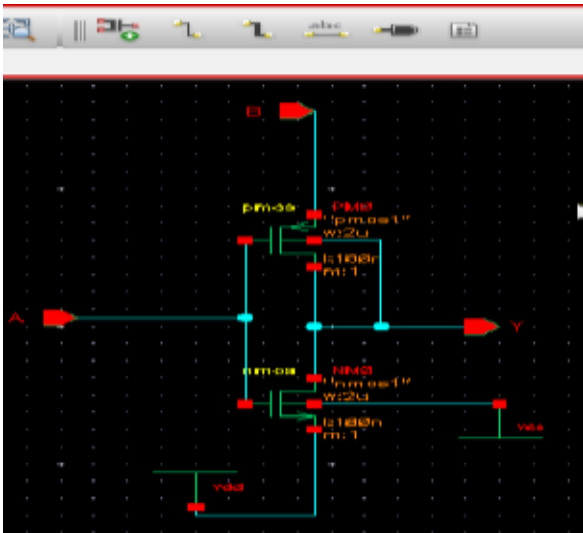
4.1 GDI BASED AND GATE



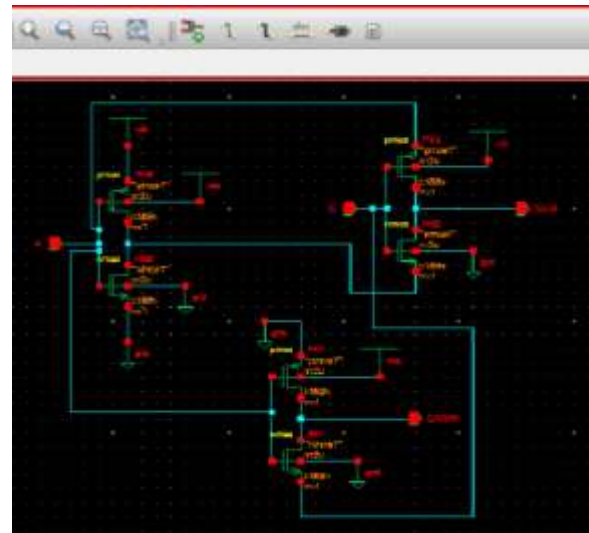
4.2 GDI BASED OR GATE



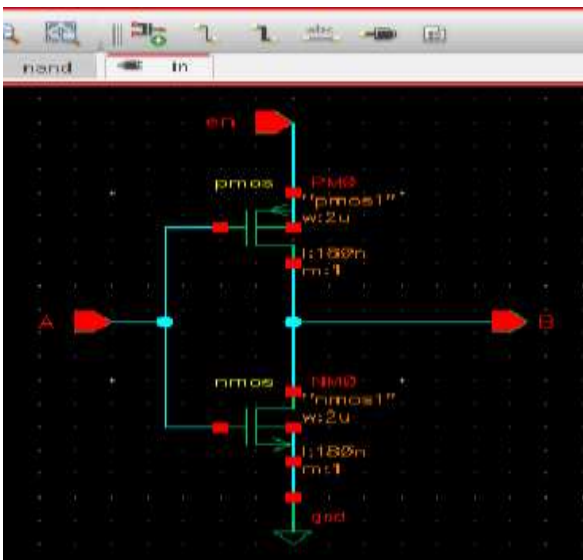
4.5 GDI BASED HALF ADDER



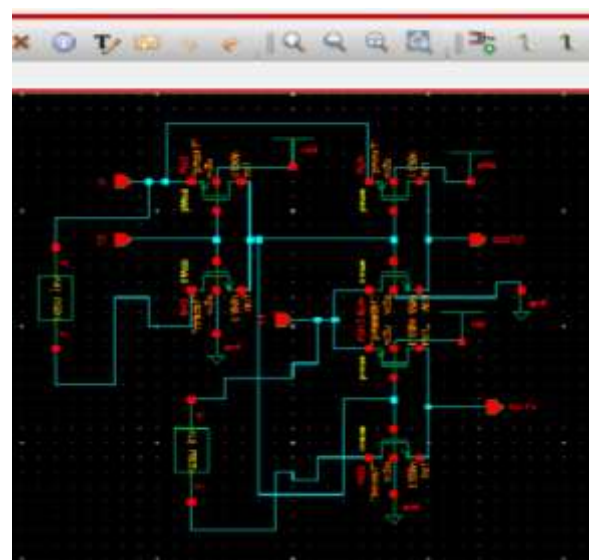
4.3 GDI BASED INVERTER



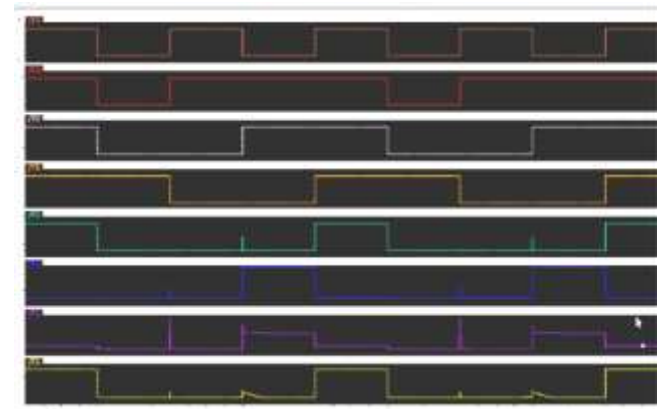
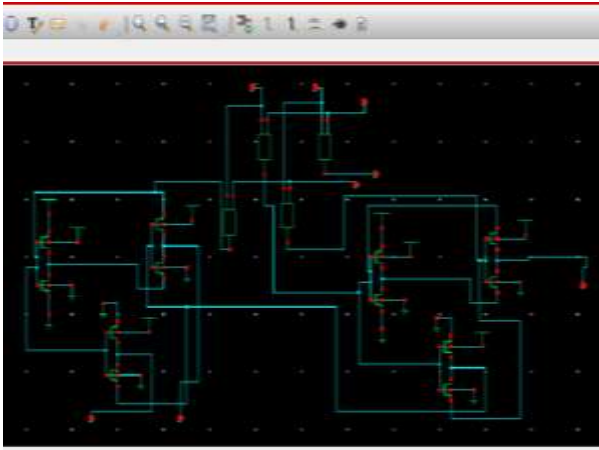
4.6 GDI BASED FULL ADDER



4.4 GDI BASED XOR

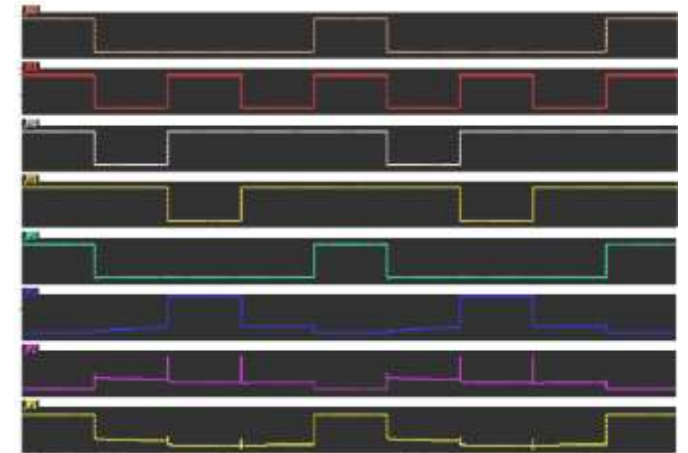
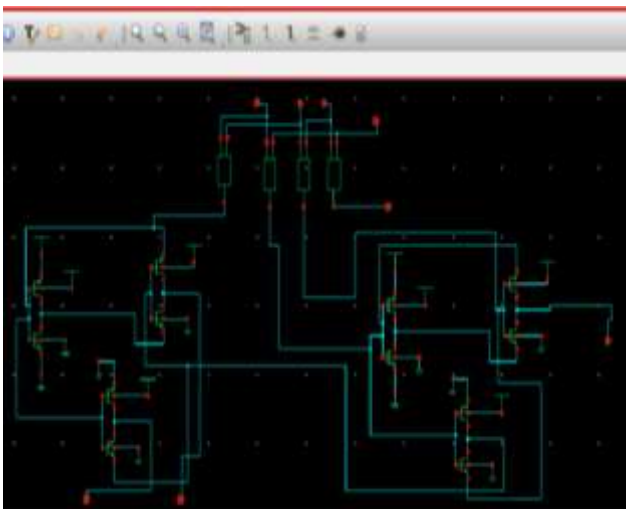


4.7 GDI BASED 2-BIT ARRAY MULTIPLIER



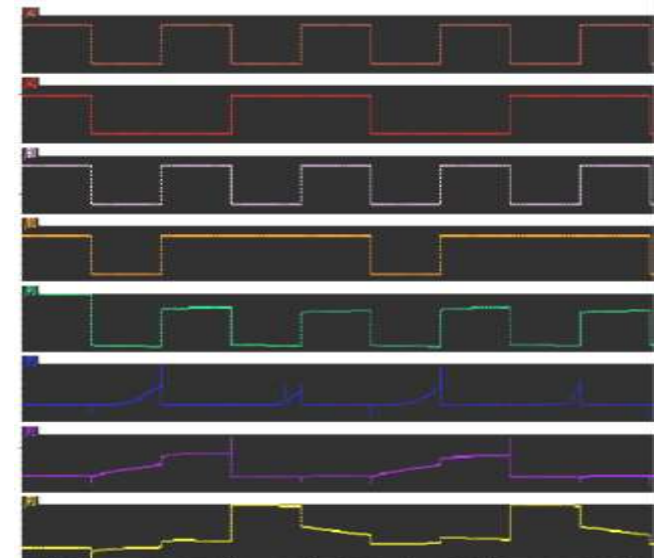
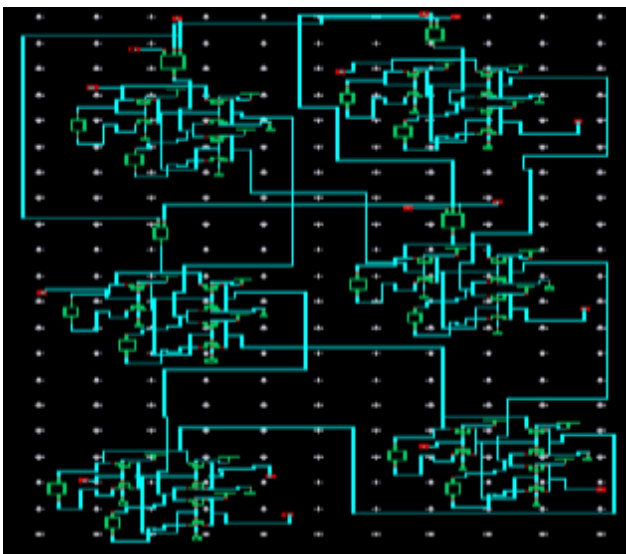
Output wave forms for GDI based 2-bit Tree multiplier

4.8 GDI BASED 2-BIT TREE MULTIPLIER



Output wave forms for GDI based 2-bit signed unsigned Baugh-Woolley multiplier

4.9 GDI BASED 2-BIT BAUGH WOOLLEY MULTIPLIER



5. SIMULATION RESULTS

Output wave forms for GDI based 2-bit array multiplier

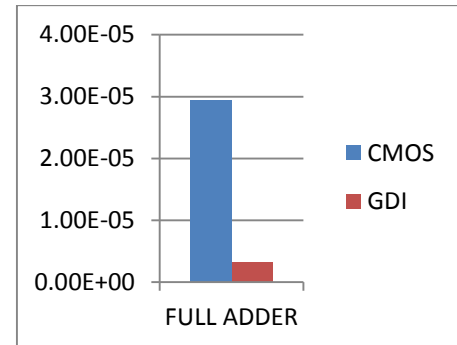
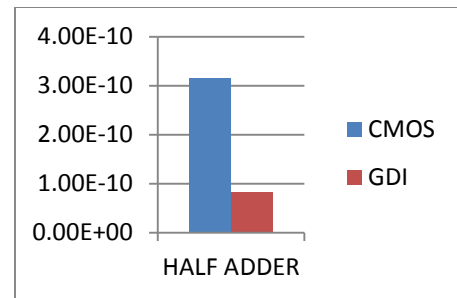
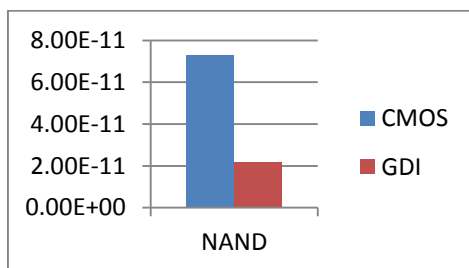
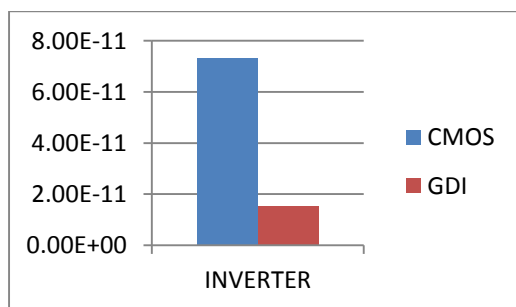
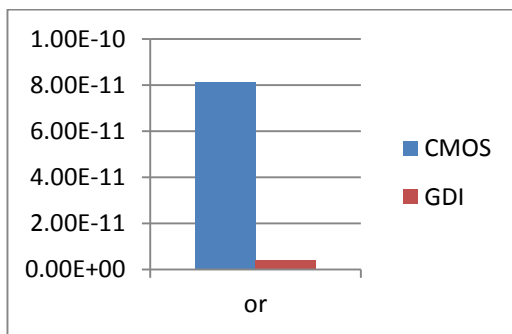
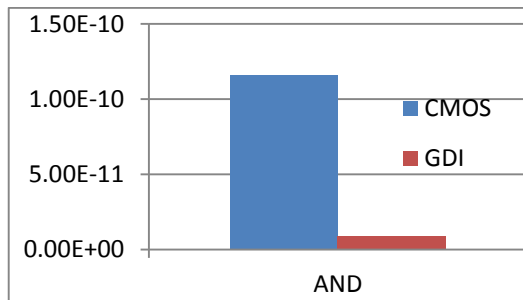
6. PERFORMANCE ANALYSIS

6.1 COMPARITIV PERFORMANCE ANALYSIS OF GDI AND CMOS BASED DIGITAL CIRCUITS

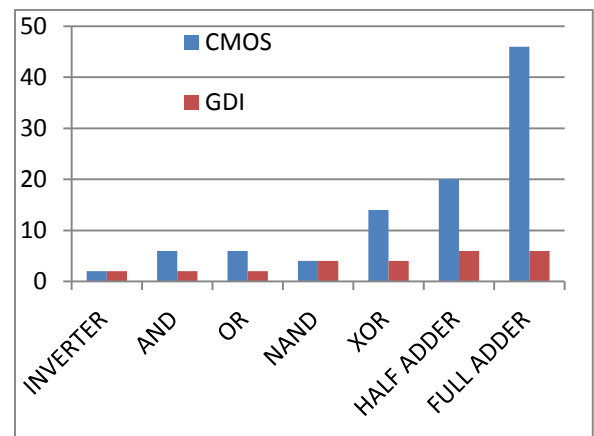
MODULE NAME	POWER IN CMOS (watts)	TRANSI STOR COUNT	POWER IN GDI	TRANSI STOR COUN
-------------	-----------------------	-------------------	--------------	------------------

		IN CMOS	(watts)	T IN GDI
INVERTER	73.25E-12	2	15.26E-12	2
AND	115.8E-12	6	9.013E-12	2
OR	81.34E-12	6	3.923E-12	2
NAND	73.06E-12	4	21.61E-12	4
XOR	206.2E-12	14	92.19E-12	4
HALF ADDER	315.2E-12	20	82.5E-12	6
FULL ADDER	29.35E-6	46	3.303E-6	6

6.2 COMPARISON GRAPH OF POWER BETWEEN CMOS AND GDI CIRCUITS



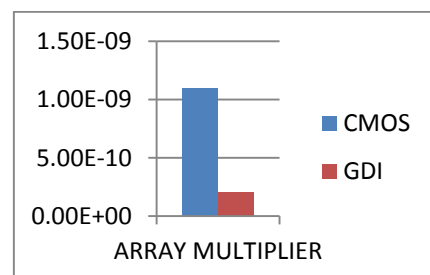
6.4 COMPARISON GRAPH OF TRANSISTOR COUNT BETWEEN GDI AND CMOS CIRCUITS

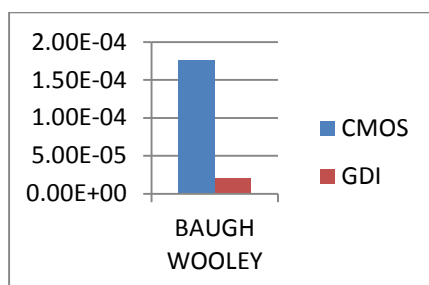
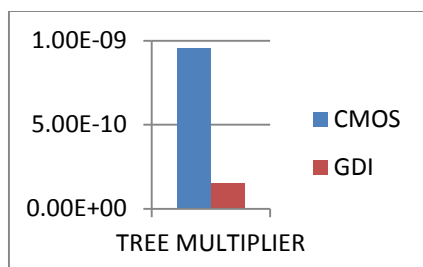


6.5 COMPARITIVE PERFORMANCE ANALYSIS OF GDI AND CMOS BASED MULTIPLIERS

DIFFERENT MULTIPLIERS	CMOS DESIGN	GDI DESIGN
Array multiplier	1093.6E-12	207.057E-12
Baugh-Woolley multiplier	1.76E-4	0.198E-4

6.6 COMPARISON GRAPH OF MULTIPLIERS BETWEEN CMOS AND GDI DESIGNS





G Santhi Swaroop Vemana was born in A.P, India. He received **B.TECH** degree in **Electronics and Communication Engineering** from Jawaharlal Nehru technological university in 2008. He worked as OFC engineer at united telecoms ltd at GOA during 2009-2010. he received M.Tech degree in VLSI Design in KL University. His research interests include FPGA Implementation, Low Power Design.



Pavani Ejju is pursuing B.Tech degree in Electronics and Communications Engineering from Jawaharlal Nehru technological university. Her research interests in low power VLSI Design.



Vasundhara chilakalapudi is pursuing B.Tech degree in Electronics and Communications Engineering from Jawaharlal Nehru technological university. Her research interests in low power VLSI Design.

CONCLUSION

We are primarily focused on the design of low power and high performance multiplier. GDI cell provides digital designs superior than other CMOS techniques in terms of power and transistor count. The disadvantage of the GDI technique is that, it is not possible to obtain a strong 0 and strong 1 at the output under certain combinations of inputs. The three types of Multipliers are used in many applications and processors, like DSP. Here “Baugh Woolley” multiplier is support for both signed and unsigned numbers and Tree multiplier has less number of adder stages compare to array multiplier.

REFERENCES

- [1] N. Weste, K. Eshraghian, Principles of CMOS VLSI Design: A System Perspective, Reading MA: Addison-Wesley, (1993).
- [2] Chandrakasan, A., and Brodersen, Low Power Digital Design, Kluwer Academic Publishers, 1995.
- [3] Issam S. Abu-Khatter, Abdellatif bellaouar, and M.I.Elmasry, Circuit Techniques for CMOS Low-Power High-Performance Multipliers, IEEE J.
- [4] Low Power Digital design using modified GDI method; Padmanabhan Balasubramanian and John, 2006, IEEE.
- [5] A. Morgenshtein, A. Fish, I.A. Wagner, ”Gate-Diffusion Input (GDI)- A Power Efficient Method for Digital combinatorial circuits”, IEEE Transactions on VLSI Systems, vol.10, no.5, October 2002.
- [6] J. P. Uyemura, Circuit Design for CMOS VLSI (Norwell, MA: Kluwer Academic, 1992, pp. 88–129).
- [7] R. De Mori, “Suggestion for an IC Fast Parallel Multiplier,” Electronics Letters, vol. 5, pp. 50-51, Feb. 1969.
- [8] H. Guilt, “Fully Iterative Fast Array for Binary Multiplication”, Electronics Letters, vol. 5, p. 263, 1969.

Author Profile