International Journal of Scientific Research Engineering & Technology (IJSRET)
Volume 1 Issue5 pp 179-183 August 2012 www.ijsret.org ISSN 2278 – 0882

Analysis and Design of Logic Gates Using Static and Domino Logic Technique

[1]Mr.Om Prakash, [2]Mr.R.K.Prasad, [3]Dr.B.S.Rai, [4]Mr.Akhil Kaushik
[1]Associate Professor, Electronics & Comm. Dept., IAMR Engg. College, Meerut, U.P. (India)
[2]Associate Professor, Electronics & Comm. Dept., M. M. M. Engg. College, Gorakhpur, U.P. (India)
[3]Professor, HOD of Electronics & Comm. Dept., M. M. M. Engg. College, Gorakhpur, U.P (India)
[4]Sr.Lecturer, Computer Science & Technology Dept., SIT, Engg. College, Meerut, U.P. (India)
E-mail: sagar.op@gmail.com, rkp_sikarpur@yahoo.com, bsrec@mmmec.net, akhil.cse.07@gmail.com

ABSTRACT
This paper presents a new design of static and Domino logic using CMOS. Domino logic is a CMOS based evolution of the dynamic logic techniques based on either PMOS or NMOS transistors. It allows a rail-to-rail logic swing. It was developed to speed up circuits in Dynamic Logic. They have smaller areas than conventional CMOS logic (as does all Dynamic Logic) and parasitic capacitances are smaller so that higher operating speeds are possible. It consumes less power than static logic. TSPICE simulation result at a frequency of 400MHz shows that proposed Domino logic speed up design.

Keywords: *Static CMOS and Dynamic CMOS design etc.*

I. INTRODUCTION
Domino logic is a CMOS-based evolution of the dynamic logic techniques based on either PMOS or NMOS transistors. It allows a rail-to-rail logic swing. It was developed to speed up circuits. Operation of dynamic CMOS consist of precharge and evaluation phase. In Dynamic Logic, a problem arises when cascading one gate to the next. The precharge "1" state of the first gate may cause the second gate to discharge prematurely, before the first gate has reached its correct state. This uses up the "precharge" of the second gate, which cannot be restored until the next clock cycle, so there is no recovery from this error.

There are many solutions to the problem of how to cascade dynamic logic gates. One way is Domino Logic, which inserts an ordinary static inverter between stages. While this might seem to defeat the point of dynamic logic, since the inverter has a PFET (one of the main goals of Dynamic Logic is to avoid PFETs where possible, due to speed), there are two reasons it works well. First, there is no fan out to multiple PFETs. The dynamic gate connects to exactly one inverter, so the gate is still very fast. And since the inverter connects to only NFETs in dynamic logic gates, it too is very fast. Second, the PFET in an inverter can be made smaller than in some types of logic gates.
In a domino logic cascade structure consisting of several stages, the evaluation of each stage ripples the next stage evaluation, similar to a domino falling one after the other. Once fallen, the node states cannot return to "1" (until the next clock cycle) just as dominos, once fallen, cannot stand up. The structure is hence called Domino CMOS Logic. It contrasts with other solutions to the cascade problem in which cascading is interrupted by clocks or other means. Important Domino Logic features include:

- They have smaller areas than conventional CMOS logic (as does all Dynamic Logic).
- Parasitic capacitances are smaller so that higher operating speeds are possible.
- Operation is free of glitches as each gate can make only one transition.
- Only non-inverting structures are possible because of the presence of inverting buffer.

Hence in this paper, we first analyze the Static OR, AND Gate .In section 2, we discuss of the problem of static logic. In section 3, we provide the experimental results of these two logics. Then we compute Average Power dissipation and Delay. Section 4 Experimental result show that average power of domino based logic is less than Static based logic.

II. STATIC LOGIC
A static CMOS gate is a combination of two networks, called the pull-up network (PUN) and the pull-down network (PDN) (Figure1.1). The figure shows a generic N input logic gate where all inputs are distributed to both the pull-up and pull-down networks. The function of the PUN is to provide a connection between the output and V_{DD} anytime the output of the logic gate is meant to be 1 (based

on the inputs). Similarly, the function of the PDN is to connect the output to V_{SS} when the output of the logic gate is meant to be 0. The PUN and PDN networks are constructed in a mutually exclusive fashion such that one and only one of the networks are conducting in steady state. In this way, once the transients have settled, a path always exists between VDD and the output F, realizing a high output "one", or, alternatively, between VSS and F for a low output "zero". This is equivalent to stating that the output node is always a low impedance node in steady state.

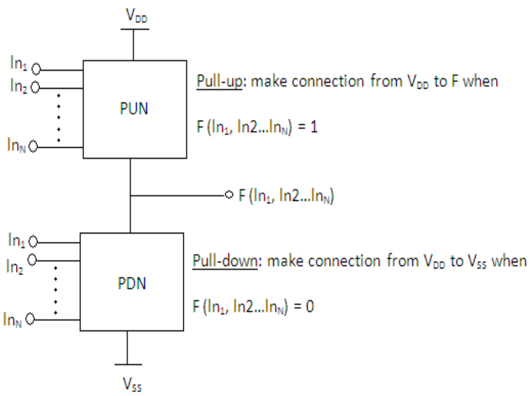


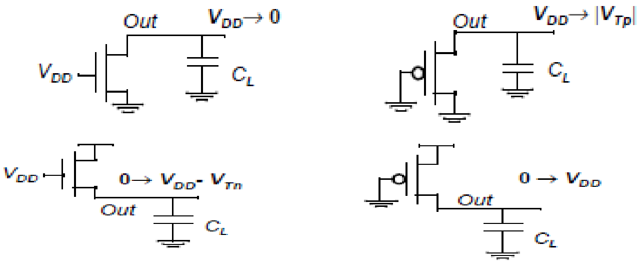
Figure 1.1: Complementary logic gate as a combination of a PUN (pull-up network) and a PDN (pull-down network).

In constructing the PDN and PUN networks, the following observations are:

1. A transistor can be thought of as a switch controlled by its gate signal. An NMOS switch is on when the controlling signal is high and is off when the controlling signal is low. A PMOS transistor acts as an inverse switch that is on when the controlling signal is low and off when the controlling signal is high.

2. The PDN is constructed using NMOS devices, while PMOS transistors are used in the PUN. The primary reason for this choice is that NMOS transistors produce "strong zeros," and PMOS devices generate "strong ones". To illustrate this, consider the examples shown in Figure 1.2. In Figure 1.2a, the output capacitance is initially charged to V_{DD}. Two possible discharge scenarios are shown. An NMOS device pulls the output all the way down to GND, while a PMOS lowers the output no further than $|V_{Tp}|$ the PMOS turns off at that point, and stops contributing discharge current. NMOS transistors are hence the preferred devices in the PDN. Similarly, two alternative approaches to charging up a capacitor are shown in Figure 1.2b, with the output initially at GND. A PMOS switch succeeds in charging the output all the way to V_{DD}, while the NMOS device fails to raise the output above V_{DD}-V_{Tn}. This explains why PMOS transistors are preferentially used in a PUN.

(a) Pull down a node using NMOS and PMOS switch



(b) Pull up a node using NMOS and PMOS switch

Figure 1.2: Simple examples illustrate why NMOS should be used a pull-down, and PMOS should be used a pull-up device.

3. A set of construction rules can be derived to construct logic functions (Figure 1.3). NMOS devices connected in series corresponds to an AND function. With all the inputs high, the series combination conducts and the value at one end of the chain is transferred to the other end. Similarly, NMOS transistors connected in parallel represent an OR function. A conducting path exists between the output and input terminal if at least one of the inputs is high. A series connection of PMOS conducts if both inputs are low, representing a NOR function, while PMOS transistors in parallel implement a NAND.

4. Using De Morgan's theorems, it can be shown that the pull up and pull-down networks of a complementary CMOS structure are dual networks. This means that a parallel connection of transistors in the pull-up network corresponds to a series connection of the corresponding devices in the pull-down. Network, and vice versa. Therefore, to construct a CMOS gate, one of the networks (e.g. PDN) is implemented using combinations of series and parallel devices.

5.

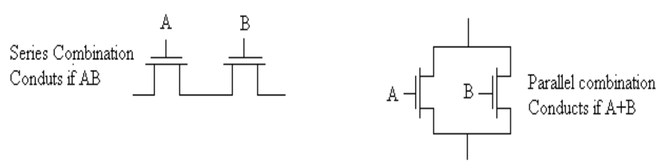


Figure1.3: NMOS logic rules- series devices implement and AND, and parallel device implement an OR

International Journal of Scientific Research Engineering & Technology (IJSRET)
Volume 1 Issue5 pp 179-183 August 2012 www.ijsret.org ISSN 2278 – 0882

The other network (i.e., PUN) is obtained using duality principle by walking the hierarchy, replacing series sub-nets with parallel sub-nets, and parallel sub-nets with series subnets. The complete CMOS gate is constructed by combining the PDN with the PUN. The complementary gate is naturally inverting, implementing only functions such as NAND, NOR, and XNOR. The realization of a non-inverting Boolean function (such as AND OR, or XOR) in a single stage is not possible, and requires the addition of an extra inverter stage. The number of transistors required to implement an N-input logic gate is 2N.

Problems in static logic

A typical static logic gate generates its output corresponding to the applied input voltages after a certain time delay and it can preserve its output level (or state) as long as the power supply is provided. This approach may require a large number of transistors to implement function and may cause a considerable time delay.

Dynamic Logic Gates

Dynamic logic (properly designed) is over twice as fast as normal logic. It uses only fast N transistors, and is amenable to transistor sizing optimizations. Static logic is slower because it has twice the loading, higher thresholds, and actually uses slow P transistors to compute things. Dynamic logic may be harder to work with, but if you need the speed, there is no other choice. A dynamic logic circuit running at 1/2 voltage will consume 1/4 the power of normal logic. Also each rail can convey an arbitrary number of bits, and there are no power-wasting glitches. Also power-saving clock gating and asynchronous techniques are much more natural in dynamic logic.

III. DOMINO LOGIC

Domino logic is a commonly used alternative to CMOS logic for designing circuits with high speed and or low area requirements. Although it provides higher speed and lower area, domino logic has relatively higher dynamic power consumption than CMOS logic due to precharge evaluate based operation and also proposed a novel low-power domino gate design and also a methodology to use these low-power hut slower gates with regular domino logic gates in combinational circuits to achieve low-power operation without changing the circuit delay.

Domino Logic Design:

Domino logic AND gate

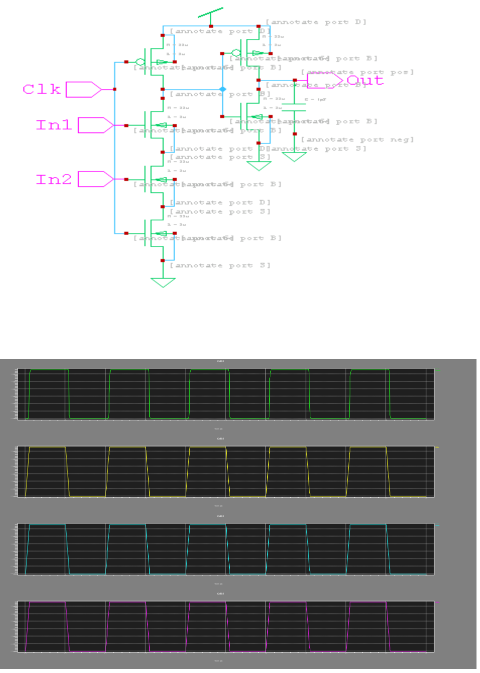


Fig.2 Circuit and waveform of Domino AND gate.

Domino OR gate

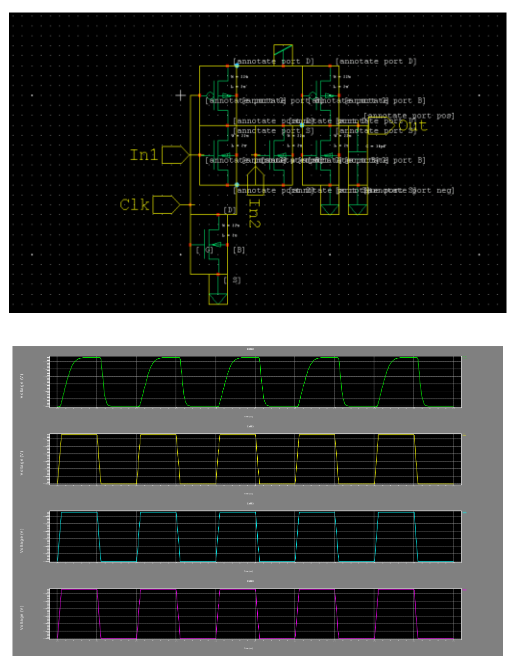


Fig.2.1 circuit and wave form of Domino OR gate.

International Journal of Scientific Research Engineering & Technology (IJSRET)
Volume 1 Issue5 pp 179-183 August 2012 www.ijsret.org ISSN 2278 – 0882

Static AND gate

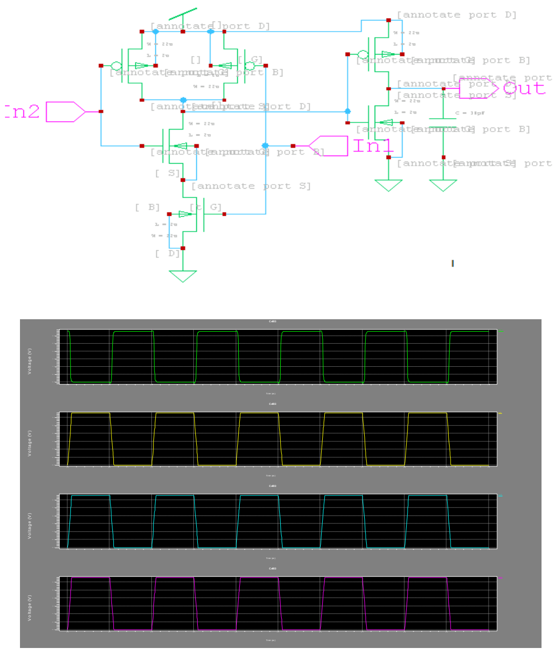


Fig.2.2 Circuit and waveform of static AND gate

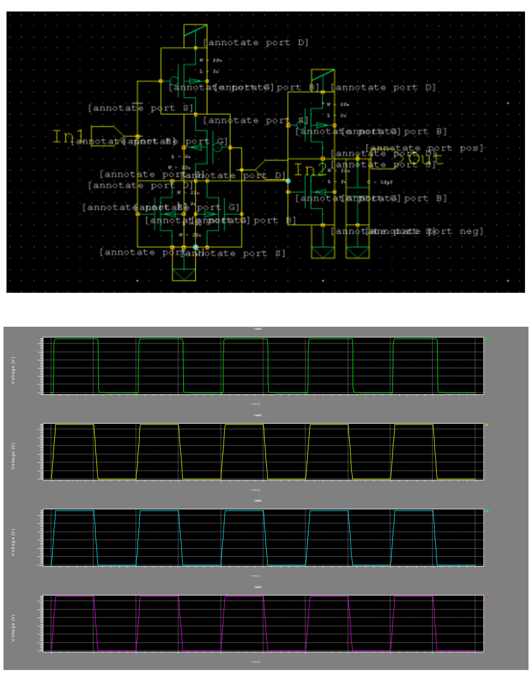


Fig.2.3 Circuit and waveform of static OR gate

IV. EVALUATION AND RESULTS

I have simulated static and Domino circuit on TANNER TOOL. I have simulated my devices using T-Spice. For simulation we have calculated area of devices, gate capacitances, widths of P-MOSFETs and N-MOSFETs, threshold voltage and also calculated time delay in CMOS circuits.

In case of Domino AND gate, calculation provides the value of Wn = 6.123µm ≈ 6µm and Wp =4.233µm ≈ 4µm. For the inverter at output side of the circuit, I found the values of Wn=4.485µm≈4.5µm and Wp=15.17µm≈15µm. T-SPICE provides time delay on this CMOS logic circuit for the calculated values as t_{plh}= 1.0333e-009 and t_{phl}=1.0964e-009.

In case of Domino OR gate, calculation provides the value of Wn=3.354µm≈3µm and Wp=3.969µm≈4µm. For the inverter at output side of the circuit, I found the values of Wn=4.485µm≈4.5µm and Wp=15.17µm≈15µm. T-SPICE provides time delay on this CMOS logic circuit for the calculated values as t_{plh}= 1.0444e-009 and t_{phl}=9.5983e-010

On comparing these values with static logic it is found that the silicon area covered by the domino circuit is less than static counterpart hence reduces the power consumption, despites its use of clock signals. The time delay in domino logic is less than static counterpart hence it is used in high speed devices.

Table: Comparison between Static and Domino logic Gates

	Static OR Gate	**Domino OR Gate**	**Static AND Gate**	**Domino AND Gate**
Time delay (ns)	5.0485 e^{-004}	1.4596 e^{-010}	1.4596 e^{-008}	3.9937 e^{-009}
Max. Power (watts)	4.336588 e^{-003}	3.463550 e^{-003}	3.463550 e^{-003}	3.990001 e^{-003}
Min. Power (watts)	2.294866 e^{-012}	6.311984 e^{-011}	6.311984 e^{-011}	1.772035 e^{-011}
Average Power(watts)	1.395826 e^{-004}	6.697686 e^{-005}	6.697686 e^{-004}	1.231027 e^{-004}

V. CONCLUSION

Domino logic is developed to speed up circuits in Dynamic Logic. They have smaller areas than conventional CMOS logic (as does all Dynamic Logic) and parasitic capacitances are smaller so that higher operating speeds are possible. On comparing these values with static logic it is found that the silicon area covered by the domino circuit is less than static counterpart hence reduces the power consumption, despites its use of clock signals. The time delay in domino logic is less than static counterpart hence it is used in high speed devices.

REFERENCES

1. Madhuban Kishor et. al., "Threshold Voltage and Power-Supply Tolerance of CMOS Logic Design Families," IEEE, 2000.
2. Massimo Alioto et. al., "Design of High-Speed Power-Efficient MOS Current-Mode Logic Frequency Dividers, "IEEE Transactions On Circuits And Systems—Ii: Express Briefs, Vol. 53, No. 11, November 2006.
3. R. H. Krambeck, C. M. Lee, and H. S. Law, "High-Speed Compact Circuits with CMOS", in IEEE Journal of Solid State Circuits, pp 614-619, Vol. SC-17, No. 3, June 1982.
4. N. F. Goncalves and H. J. De Man, "NORA: A Race free Dynamic CMOS Technique for Pipelined Logic Structures", in IEEE Journal of Solid State Circuits, pp 261-266, Vol. 18, No. 3, June 1983.
5. M. R. Prasad, D. Kirkpatrick, and R. K. Brayton, "Domino Logic Synthesis and Technology Mapping", in International Workshop on Logic Synthesis, 1987.
6. Min Zhao and Sachin S. Sapatnekar, "Technology Mapping for Domino Logic", in IEEE/ACM Proc. of Design Automation Conference, pp 248-251, 1998.
7. R. K. Brayton, R. L. Rudell, and A. L. Sangiovanni-Vincentelli, "MIS: A Multiple-Level Logic Optimization System", in IEEE Trans. on Computer Aided Design, pp. 1062-1081, Vol. 6, No. 6, 1987.
8. Tyler Thorp, Gin Yee and Carl Sechen, "Domino Logic Synthesis Using Complex Static Gates", in Proc. of Int'l Conference on Computer Aided Design, pp 242-247, 1998.
9. Abul Sarwar, "CMOS Power Consumption and Cpd Calculation", Texas Instruments Incorporated, June 1997.
10. Ge Yang et. al, "Low Power and High Performance Circuit Techniques for High Fan-in Dynamic Gates", IEEE Computer Society, 2004.