EFFICIENT DESIGN OF THRESHOLD LOGIC GATE USING PNAND CELL

P.Pandeeswari, B.Sathya & R.Marishri
Students, Department of Electronic and Communication Engineering,
P.S.R Rengasamy College of Engineering for Women, Virudhunagar, Tamilnadu, India

P.Vedhanayagi
Assistant Professor, Department of Electronic and Communication Engineering,
P.S.R Rengasamy College of Engineering for Women, Virudhunagar, Tamilnadu, India

Abstract

In this paper, we describe a method for reducing dynamic power, leakage and area of application-specified integrated circuits without sacrificing performance. Abstractly the threshold gate act as a multi input single output, edge-triggered flip-flop which computes a threshold function of inputs on the clock edge. The library consist of small number of cells, each cell can compute a set of complex threshold function. The flip-flop stores a data in complementarily programmed resistive memory device during inactive period while power supply is tuned off and then restores the data to flip-flop outputs once power supply is turned back on. We present a method for the assignment of signals to inputs of a threshold gate to release a given threshold function, Next, we present an algorithm that replace a subset of flip-flops and portion of their logic cones in a conventional logic net list, with threshold gates from the library. An algorithm of obtaining efficient architectures composed of threshold gates that implement arbitrary Boolean function is introduced. The method reduces the complexity of a target function by splitting it according to the variable with the highest influence. We demonstrate significant reduction in power, leakage, area of the hybrid circuits when compared with the conventional logic circuits.

Index Terms: *Low power technology, Threshold logic, Scan implementation*

Introduction

ASIC is an integrated circuit of reducing dynamic power include logic synthesis are reduce switching activity, gate sizing, technology mapping, retiming manufactured for a particular use rather than intended for general purpose use. The new standard cell ASICs is targeted at high volume consumer computing and communication markets which are driven by end-user requirements for low cost and power. Some of the ways, and voltage scaling and so on. Similarly the uses of dual supply and device threshold voltages, adaptive body biasing, clock and power gating, transistor stacking and so on are some of the well-known ways to reduce the power due to leakage. We have focused on the higher levels of design, including power efficient micro architectures, memory, compilers and OS, and system level control, thread migration among processor cores. The complimentary logic is a preferred choice to build complex logic circuits for its low static power and large noise margin. A CMOS Application Specified Integrated Circuit using static logic is a multilevel network of AND/OR logic gates. To improve the robustness to parametric variability, advanced device or circuit architectures such as dual-Vth TFT logic, pseudo CMOS logic, differential logic and positive- feedback level shifter logic were proposed.

A BOOLEAN function *f(x1, x2 ,..., xn)* is called a threshold function if its own and offsets are linearly separable. Synthesis of two level threshold logic networks has not been practical for circuits of present-day complexity. A linear programming algorithm can be used to determine the minimal weight assignment. Once selected, a candidate is checked for the threshold property by solving a linear program with constraints derived from the truth table. This can be fundamentally computed by different mechanisms, which presents the possibility of further improvements in power consumption, performance, and area.

$$f(X)=\{ =\text{sign}^{+} (w`\ X-T)^{1}$$

In area of the circuit design we use threshold gates instead of standard Boolean logic AND, NOT, and OR gates because that threshold elements are more powerful ; in the sense that size of the circuits that can be constructed to compute the desired functions can be smaller. A threshold function can be implemented in the same way as any logic function, i.e., as a network of logic primitives or a full-up network and full-down network of pFETS and nFETS the implementation a threshold logic gate. In this paper compute the predicate in by performing a comparison of some electrical quantity, such as charge, voltage, or current. Different threshold circuits have been implemented in hardware but the commercial application of threshold circuits is still in its infancy. To reduce the number of gates by replacing the Boolean gates by threshold gates using a node collapsing algorithm. This algorithm works directly with the truth table of the Boolean function, and, thus, all techniques for function splitting and checking linearly seperability are optimized for this type of representation. Currently, near /sub-threshold circuits have been drawing a lot of attentions for ultra-low power applications, because an operation at ultra- low supply voltage near or below threshold voltage achieves an energy minimum operation. We use the term "threshold gate" to denote a single, primitive, or non-decomposable circuit that realizes a threshold function, and one that physically embodies the linear seperability characteristic of threshold function. Possibility of releasing a threshold functions by a static, complex CMOS cell that consists of a pull up-pull down network. Such an approach would offer no real advantage over a traditional CMOS logic structure, and in fact would be quite poor in terms of speed, power, and area as the original Boolean functions. Current mode logic compares the strength of conducting paths in pull up and pull down transistor networks. These are generally very fast. More practical designs that employ static logic, single phase clock, and utilize the full voltage swing binary outputs have been proposed. For instance, symmetric functions have been identified to be efficiently and systematically represent able in terms of threshold components.

The proposed non volatile flip-flop design in detail along with transient simulations performed using the compact model to demonstrate functionality, energy consumption and data restore operation reliability for a range of voltage supply levels. Energy consumption of proposed flip-flop is compared with an existing leakage optimized CMOS flip-flop to obtain an optimum inactive period duration. Most of the properties and characteristics of the circuits can be extended and applied to neural networks. Based on the predicted yield result, the most cost-effective solution can then be selected to obtain the required functional circuits of good manufacturability. In this paper, a methodology for ASIC design with TLGs is described. The resulting circuit, referred to as a hybrid circuit, will have a mixture of conventional logic cells.

New Architecture for a Threshold Gate

In this section, the architecture for a threshold gate that employs differential logic, referred to as a pNAND cell, is described. A conventional D-type ETFF computes the identify function *f(x) =x* on a clock edge, a pNAND cell computes a threshold function *f(x1, x2... xn)* on a clock edge.

a) Cell Operation

The circuit consist of three main components:

- Two groups of parallel pFET transistor referred to us the left input network and right input network
- A sense amplifier, which consist of a pair of cross-coupled NAND gates
- A set-reset latch, the cell operated in two phases: reset (CLK=0) and evaluation (CLK0→1).For the moment, ignore the transistor M9 and M10 in the LIN and RIN.

1. Reset phase: When CLK=0, the two discharge devices M18 and M19 pull nodes N5 and N6 low, which turn off M5 and M6, disconnecting all parts from N1 and N2 to ground. In addition, M7 and M8 are active, which results in N1 and N2 being pulled high. The nFETs M3 and M4 are ON, with N1 and N2 being high, the state of the SR latch does not change.
2. Evaluation phase: This corresponds to when CLK0→1. An input that results in *l* active devices in the LIN and r active devices in RIN is denoted by *l/r*

1 will start to discharge through M3 and M5.The delay in the starts to discharge through M3 and M5. The delay in the start time for charging N6 due to the lower conductance of As the discharge devices M188 and M19 are turned OFF, both N5 and N6 will rise to 1. Due to the higher conductivity of the LIN, node N5 will start to rise first, which turns ON M5. With M3= 1, the RIN allows for N1 to turn ON M2 and turn OFF M4. Thus, if N2 starts to discharge, its further discharge is impeded as M2 turns ON, resulting in N2 getting pulled back to 1. As a result, the output node N1 is N2 getting pulled back to 1. As a result, the output node N1 is 0 and N2 is 1. As the circuit and its operation symmetric, if *l<r,* then the evaluation will result in N1=1 and N2=0.

The active low SR latch stores the signal N1 and N2.During reset, when (N1,N2)=1,the SR latch retains its state. After evaluation, if(N1,N2)=(0,1), the output Q=0, and if (N1,N2)=(1,0), Q=1, providing a dual-rail output for the threshold function to be computed. Therefore, once evaluated after the rising edge of the CLK, the output Q of the cell is stable for the remaining duration of the clock cycle. Hence, it operates like an ETFF that computes a threshold function.

In the layout of the pNAND cell, various steps were taken for ensuring robustness to process variations and signal integrity. A symmetric SR latch was used to ensure near similar load on nodes N1 and N2 and near equal rise and fall delays. Some nodes of M16 and M17 are shorted. So that the transistors in the LIN and RIN have nearly similar Vd, Vg , Vs. Before the clock rises. The sizes of pull-down devices in the differential amplifier were optimized, as were the sizes of the input transistors in the LIN and the RIN, to improve the robustness of the cell, an internal feedback is created with transistors M9 and M10 in the LIN and the RIN, driven by N1 and N2 respectively.

Assume that M9 and M10 are not present, and consider the situation in which there are k devices in the LIN and none in the RIN. After reset, N5 and N6 are both 0.When the clock rises to 1,N5 will rise to 1, and N6 will be HiZ0, and the circuit will correctly values N1=0 and N2=1.Note that M4 is inactive and M3 is active. Now suppose that while CLK=1, the input change, and all transistors in the LIN will be inactivated and some k transistors in the RIN become active.N5 is now HiZ1, and N1 will remain at 0, keeping M4 inactive. However N6 rise to 1, turning ON M6. As long as M4 remains inactive, no change will take place. N5 being HiZ1, is susceptible to being discharged. If that happens, N1 rises, activates M4, and discharging N2, which causes the output being complemented.

Transistors M9 and M10 ensure that N5 and N6 do not become HiZ0 or HiZ1. In the above situation, once N1=0 during evaluation, the presence of M9 driven by N1 ensures that N5=1. Hence, after evaluation and when CLK=1, any change in the input will not affect the output. The simulation starts with applying a 5/4 input, which results in N1=0, N2=1, and Q=1. While CLK is held at 1, the input is switched to 0/5, so that N5=HiZ1. Next, N5 discharged to ground through a capacitor, which turns OFF M5 and turns ON M7, pulling N1 to 1. This turns ON M2 and turns OFF M4, and N2 discharges to ground. The result is that while CLK =1, if the input switched leaving N5 and N6 in a HiZ1 state, it is possible to switch the output due to coupling noise. The same process when repeated with the presence of the keeper transistors M9 and M10, does not result in the output being disturbed.

b) Scan Implementation

If pNAND cell replaces flip-flop and logic cones feeding them, scan capability is necessary. There exist several ways to implement scan for a p NAND cell has negligible impact on the cell's performance and robustness during normal operation. T he additional transistors for scan are labled S1 through S6. In the normal mode, the signals TE and T1 are both 0, which disables the scan-related transistors (S1-S4), and reduces the circuit function. If a circuit has a mix of D-FFs and pNAND cells, the pNAND cells must be part of a separate scan chain. The procedure to scan in a steam of bits into a scan chain consisting of pNANDs is as follows. Signal global TI is the entry point for the scan data input to the pNAND chain.

1. Set CLK = 0 and TE= 0.
2. Set GTI = *ith* bit of the input (*i=0* initially).
3. Set TE = 1. Each pNAND registers its TI input.
4. Set |TE = 0.
5. Increment i and repeat until the end of stream

The pull-up transistor S5 and S6 are included to eliminate dc path during testing. In the option of these transistors, when TE is asserted (0→1), while CLK = 0, M7 is active, and there is a dc path VDD→M7→M3→S1→S2→GND. Scanning of hybrid circuits, i.e. one with both D-FF and pNANDs, requires two separate scan chains –one for the D- FFs and one for the pNANDs. A common TE signal is used for both the scan chains.

First the signal TE is held high, and the data are scanned into regular flip-flops. Once this is done, the common clock signal is held low, and the data are scanned through pNAND chain only using TE signal as described above note that the toggling signal TE does not affect the data stored in the first scan chain consisting of regular flip-flops. At the end of this process, both scan chains will have the required data, and regular clocking can proceed.

Results

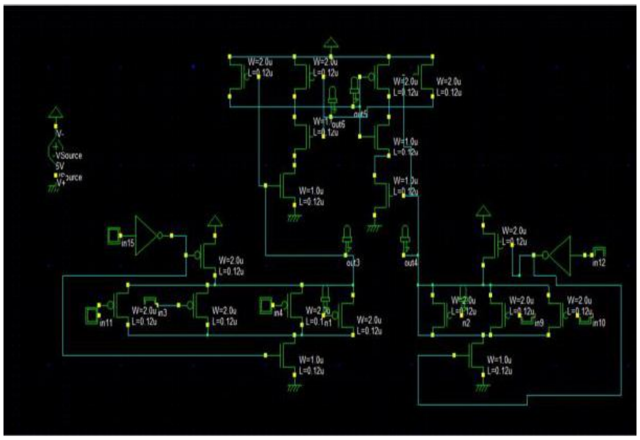
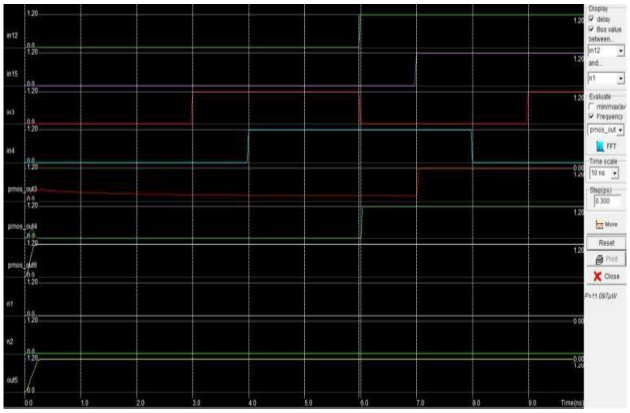


Fig. Circuit Design of pNAND Cell

- Open the Schematic Editor in Microwind (DSCH3). Click on the transistor symbol in the symbol Library on the right.
- Instantiate NMOS or PMOS transistor from the symbol library and place them in the editor window.
- Connect the drain and sources of transistors. Connect VDD and GND to the schematic.
- Open the Schematic Editor in Microwind [DSCH3]. Click on the transistor symbol in the symbol Library on the right.
- Instantiate NMOS or PMOS transistor from the symbol library and place them in the editor window.
- Connect the drains sources of transistor. Connect VDD and GND to the schematic.



☑ delay
☑ Bus value
between... in12
and... n1
□ min/max/av
☑ Frequency pmos_out
Time scale 10 ns
Step(ps) 0.300
P=11.097µW

Fig. Analog simulatio

- Use your logic simulator to verify the functionality of your schematic.
- The next step is to simulate the circuit and check for functionality.
- Click on, Simulate→ Start simulation.
- This brings up a Simulation Control Window.
- Click File → Make Verilog File. The Verilog Hierarchy and Netlist window appears. This window shows the Verilog representation of NOR gate.
- Click OK to save the Verilog as a.txt file.
- Click Compile and then Back to editor in the Verilog File Window. This creates a layout in layout editor window using automatic layout generation procedure
- Click Simulate → Run Simulation. A simulation window appears with input and output, shows the tphl, tplh and tp of circuit. The power consumption is also shown on the right bottom portion of the window

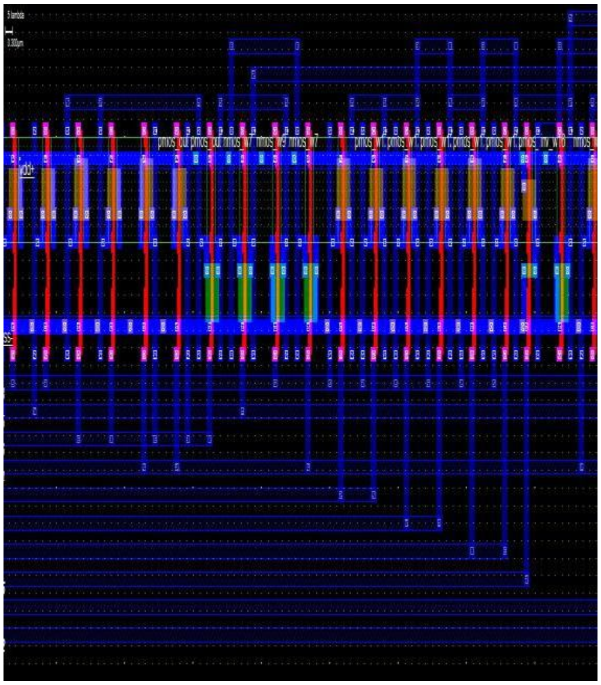


Fig. Layout of pNAND Cell

- If pNAND cells are to replace flip-flops and logic cones feeding them, scan capability is essential.

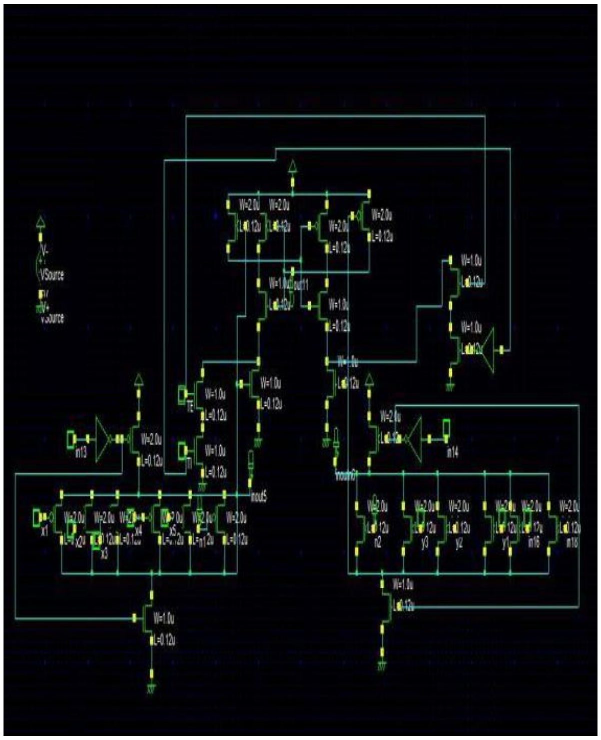
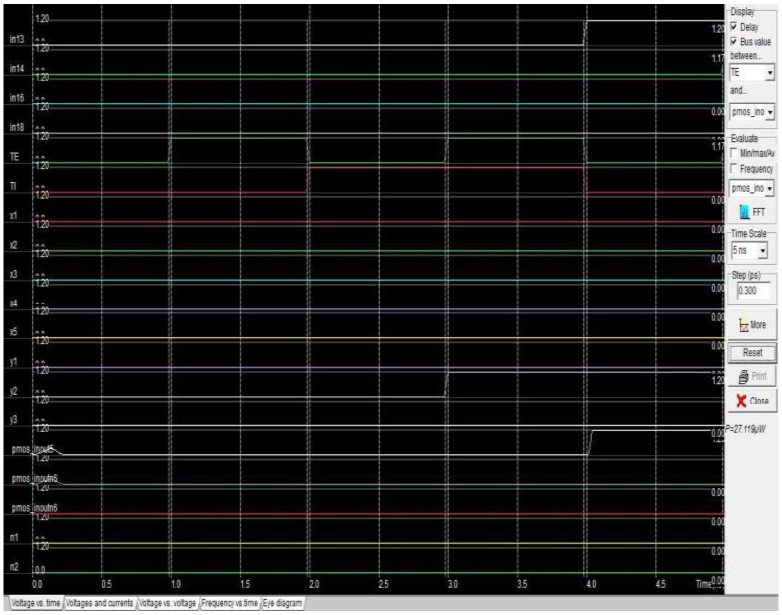


Fig. pNAND Cell Scan Implementation Circuit



Display
☑ Delay
☑ Bus value between...
TE
and...
pmos_ino

Evaluate
□ Min/max/Av
□ Frequency
pmos_ino

Time Scale
5 ns

Step (ps)
0.300

P=27.119µW

Fig. Analog Simulation

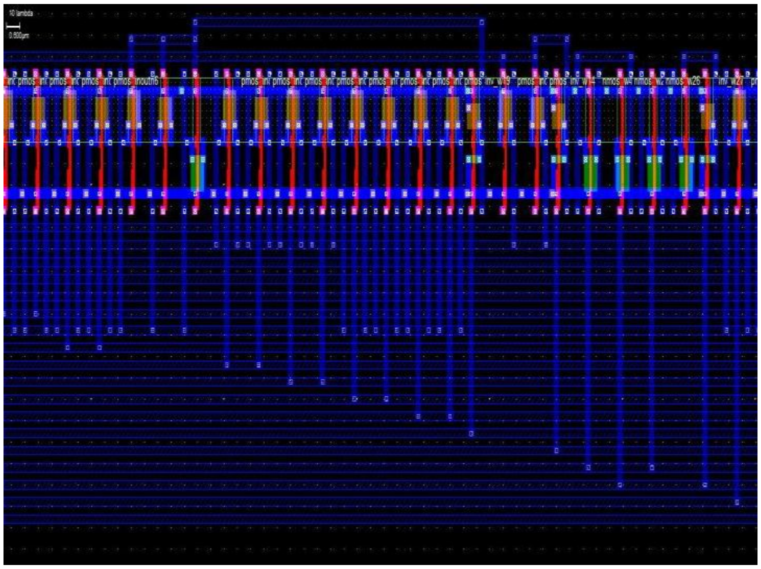


Fig. Layout of pNAND Cell with Scan Implementation

Conclusion

The approach depicted in this paper was practiced on various complex functional blocks, and significant. Improvements in dynamic power, leakage, area, and power variation were demonstrated. Our experimental results also demonstrates that the proposed threshold gates, when operated at the nominal voltage, can be made robust in the presence of process variations. However, dynamic voltage scaling, which is now an integral part of the power management of most digital circuits, must be limited when applied to threshold gates due to the presence of the latch-based SA. The degree to which the voltage of a pNAND-k cell can be reduced depends on k- with lower voltage for smaller k.

References

1. V. Beiu, "A survey of preceptor circuit complexity results," in Proc. Int. Joint Conf. Neural Netw. (IJCNN), Jul. 2003, pp. 989–994.
2. V. Beiu, J. M. Quintana, and M. J. Avedillo, "VLSI implementations of threshold logic—A comprehensive survey," IEEE Trans. Neural Netw., vol. 14, no. 5, pp. 1217–1243, Sep. 2003.
3. B. Nikoli´ , V. G. Oklobdžija, V. Stojanovi´ , W. Jia, J. K.-S. Chiu, and ccM. M.-T. Leung, "Improved sense- amplifier -based flip-flop: Design and measurements," IEEE J. Solid-State Circuits, vol. 35, no. 6, pp. 876–884, Jun. 2000.
4. M. Padure, S. Cotofana, and S. Vassiliadis, "Design and experimental results of a CMOS flip-flop featuring embedded threshold logic," in Proc. Int. Symp. Circuits Syst., May 2003, pp. V-253–V-256.
5. N. Kulkarni, N. Nukala, and S. Vrudhula, "Minimizing area and power of sequential CMOS circuits using threshold decomposition," in Proc. Int. Conf. Comput.-Aided Design, 2012, pp. 605–612.