Asian Journal of Applied Science and Technology (AJAST)
Volume 1, Issue 1, Pages 73-78, February 2017

Low Power Threshold Logic Designing Approach for High Energy Efficient Flip-Flop

K.Thulasimani[#] and M.Hemalatha[*]

[#]*PG Student, Department of ECE, Vivekanandha College of Engineering for Women, Tiruchengode, India. Email: thulasiece13@gmail.com*
[*]*Assistant Professor, Department of ECE, Vivekanandha College of Engineering for Women, Tiruchengode, India. Email: engineerhemalatha@gmail.com*

Article Received: 07 February 2017 *Article Accepted: 17 February 2017* *Article Published: 21 February 2017*

ABSTRACT

Improving energy efficiency has always been the prime objective of the custom and automated digital circuit design techniques. However, as the field of design automation has matured over the last few decades, there have been no new automated design techniques, that can provide considerable improvements in circuit power, delay. Although emerging nano-devices are expected to replace the existing MOSFET devices, they are far from being as mature as semiconductor devices and their full potential and promises are many years away from being practical. The research described in this dissertation consists of four main parts. First is a new circuit architecture of a differential threshold logic flip-flop called PNAND. The PNAND gate is an edge-triggered multi-input sequential cell whose next state function is a threshold function of its inputs. Second a new approach, called hybridization, that replaces flip-flops and parts of their logic cones with PNAND cells is described. The resulting hybrid circuit, which consists of conventional logic cells and PNANDs, is shown to have significantly less power consumption, smaller area, less standby power and less power variation. Third, a new architecture of a field programmable array, called field programmable threshold logic array (FPTLA), in which the standard lookup table (LUT) is replaced by a PNAND is described. The FPTLA is shown to have as much as 50% lower energy delay product compared to conventional FPGA using well known FPGA modeling tool called VPR. Fourth, a novel clock skewing technique that makes use of the completion detection feature of the differential mode flip-flops is described.

Keywords: PNAND cell, threshold logic and clock edge trigger.

1. INTRODUCTION

As an effort to reduce power consumption of digital CMOS circuits have been in progress for nearly three decades. As a result, a number of well understood and proven techniques for low power energy, efficient flip-flop design using threshold logic have been incorporated into modern design tools. For us, some of the ways to reduce dynamic power include logic synthesis and restructuring to reduce switching activity, gate sizing, technology mapping, retiming, 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.

Thus, it appears that the techniques for reducing power at the logic and circuit levels have been thoroughly explored, leaving little opportunity for improvement. Consequently, the focus has shifted to the higher levels of design, including power-efficient micro architectures, memory, compilers, and OS, and system level control, including thermal-aware dynamic frequency and voltage control, thread migration among processor cores, and so on. One aspect of digital CMOS circuits that has not changed is how logic functions are computed. A CMOS application specified integrated circuit (ASIC) using static logic is a multilevel network of AND/OR logic gates or more complex cells, in which each node computes a Boolean function of its inputs by establishing a conducting path from the supply rails to its output. However, there exists a proper subset of unite Boolean functions, called threshold functions, which can be fundamentally computed by different mechanisms, which presents the possibility of further improvements in power consumption, performance, and area, which has not been sufficiently explored. Let $X = (x_1, x_2, \ldots, x_n)$, $x_i \in \{0, 1\}$, $w = (w_1, w_2, \ldots, w_n)$, $w_i \in R$, and $T \in R$. A unite Boolean function f (X) is called a threshold function if there exist weights w and a fixed threshold T ,The reason for examining threshold gates as logic primitives stems from the fact that they are computationally more powerful than the standard AND/OR logic primitives. Many common logic functions, such as the n-bit parity, n-bit multiplication, division, powering, sorting, and so on, can be computed by polynomial size threshold networks of a fixed number of levels, while the same would require exponential size AND/OR networks. A detailed treatment of the complexity of threshold networks and constructive methods for various types of arithmetic functions, including size-depth and weight depth tradeoffs. An updated survey of the same appears in [3], and an extensive survey of circuit architectures of threshold gates. These results suggest that the threshold gates and networks can potentially lead to significant reductions in circuit size and delay.

1.1 MCCULLOCH-PITTS

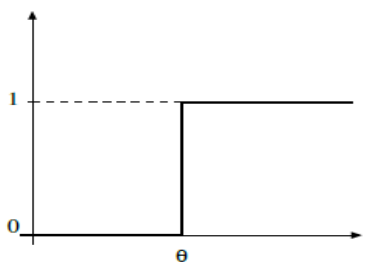

Fig.1. Threshold function

© 2017 AJAST All rights reserved.

Threshold function implementation of and /or us for (McCulloch-pitts) as referred as to process.

1.2 Threshold Logic

Networks of functions

We deal in this chapter with the simplest kind of computing units used to build artificial neural networks. These computing elements are a generalization of the common logic gates used in conventional computing and, since they operate by comparing their total input with a threshold, this field of research is known as threshold logic.

This rule implies that a McCulloch–Pitts unit can be inactivated by a single inhibitory signal, as is the case with some real neurons. When no inhibitory signals are present, the units act as a threshold gate capable of implementing many other logical functions of n arguments. Figure 2.7 shows the activation function of a unit, the so-called step function. This function changes discontinuously from zero to one at θ. When θ is zero and no inhibitory signals are present, we have the case of a unit producing the constant output one. If θ is greater than the number of incoming excitatory edges, the unit will never fire. In the following subsection we assume provisionally that there is no delay in the computation of the output.

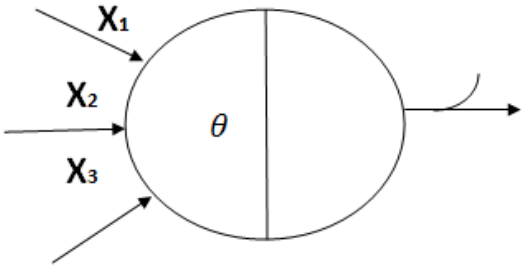

Fig.2. McCulloch-pitts

2. THRESHOLD LOGIC FLIP-FLOP

2.1 Basic operation

A schematic diagram of the Threshold Logic flip-flop (TLFF) is presented in Figure 1. The circuit is composed of a semi-dynamic front-end comprising a differential current-switch Threshold Logic gate (DCSTL) [4] followed by a static back-end comprising an RS latch. DCSTL front-end comprises a fast latched comparator and two parallel-connected sets of unit nMOS transistors, referenced herein as input data bank and threshold mapping bank.

The nMOS transistors from the threshold mapping bank have the gates hardwired to ground or power supply With respect to the circuit from Figure 1, the TLFF has 3 data inputs and 3 threshold mapping inputs. The data inputs, X_0, X_1, X_2, and the threshold mapping inputs, T_0, T_1, T_2, have the weights 1, 3, 4 respectively.

The weights are implemented using parallel connected sets of 1, 3 and 4 unit transistors respectively. The total conductances of the transistor banks are compared each other by the latched comparator and therefore the node X is logic zero if the current generated by the data bank is greater than the current generated by the threshold mapping bank and logic one otherwise. Please note that, by design, the data bank is prevented to have similar conductance with the threshold mapping bank, when the threshold is reached, since an nMOS transistor with weight 0.5 is always on. This prevents the latch comparator entering in a metastable state.

The circuit in operates as follows. On the falling edge of the clock, the flip-flop enters in precharge phase. Therefore, M_{10}, M_{11}, are on, nodes X and Y are precharged high and the outputs Q and Q_{bar} and hold their previous evaluation values; since X and Y are high, M_6, M_7 are on pulling their sources to weak high level. On the rising edge of the clock, the flip-flop enters the evaluation phase. Therefore, M_5, $M_{8,9}$ are on and M_6, M_7 (shutoff devices) start drawing currents from nodes X and Y. If $I_{data} \geq I_T$ then the voltage at node X will start to drop faster than the voltage at node Y.

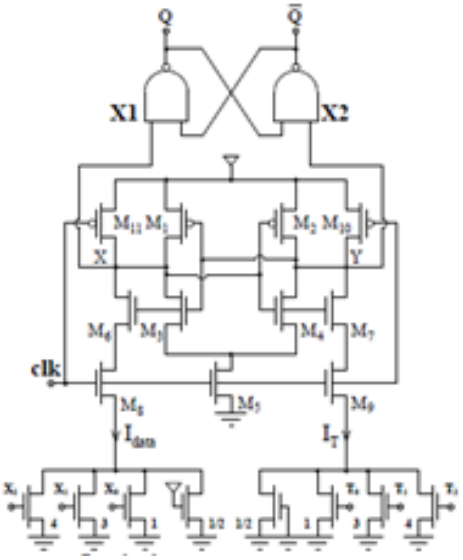

Fig.3. Threshold logic bank

Data bank threshold logic bank threshold logic with embedded ØTherefore, X crosses first the latch switching threshold which regenerates rapidly to X low and Y high, causing Q high. Conversely, if $I_{data} < I_T$ then Y low and X high, causing Q low. At the end of the evaluation phase, the high-rising node among X and Y will be decoupled from being connected to ground by one of the shutoff transistors M_6, M_7 going off. Therefore no DC power is dissipated at the end of the evaluation phase. Additionally, any change on the inputs after the gate has ended the evaluation will not affect nodes X and Y and consequently TLFF is an edge-triggered flip-flop

.

2.2 Embedding Threshold Logic functions

One distinctive advantage of the proposed TLFF is that complex TL functions can be embedded easily. Indeed, most logic functions available in Domino logic, such as OR/AND functions can be embedded in TLFF. Additionally, in comparison with Domino logic, wide OR/AND and their complements can be incorporated with no prohibitive latency.

© 2017 AJAST All rights reserved.

2.3 P-NAND Technology Mapping

In ASIC implementation technologies that use cell generators to create circuit elements, the set of available circuit elements consists of a parameterized family of cells rather than a specific library of functions. This cell family contains all members of a class of functions, such as And Or Inverts (AOIs), which do not exceed parameters defining the family. Library-based technology mapping is inappropriate for cell generator technologies when the number of cells in the family is too large to be practically expressed in a library. Examples of technology mapping that deals specifically with cell generators are the approaches of Berkelaar and Jess [Berk88] and Liem and Lefebvre [Liem91].

The key to cell generator technology mapping is the completeness of the cell family. This simplifies the matching of network sub-functions to circuit elements. If a sub-function does not exceed the parameters defining the family, it can be implemented by a cell in the family. In addition, simplified matching makes it possible to improve the final circuit by combining decomposition and matching.

Berkelaar addresses technology mapping for a cell generator that creates NMOS or CMOS And Or Invert gates. The set of available circuit elements includes all AOI gates that meet limits on the maximum number of transistors in series and in parallel. The network is first partitioned into a forest of trees and a circuit implementing each tree is then constructed by traversing the tree proceeding from the root node to the leaf nodes. The decomposition of each AND or OR node in the tree is determined by the parameters defining the cell family. When the in-degree of the node exceeds the limits of the cell family, the node is decomposed into a tree of nodes that match the largest available cell. When the in-degree of the node does not exceed these limits, the node is implemented by a single cell. If this cell is not the largest cell in the family, then the remaining unused capacity is passed on to the fanin nodes. In this case, the cell also implements part of the functions of the fanin nodes.

The original network is first partitioned into a forest of trees and each tree is decomposed into a minimum-depth binary tree. The circuit implementing each tree is then constructed using a dynamic programming approach similar to the DAGON approach. At each node, the set of matching circuit elements is constructed using a recursive traversal that is pruned by the limits defining the cell family.

While latency is increased, the merger allows the elimination of one or more levels of logic from the path leading to the flip-flop. The result is a reduction of the overall latency of the circuit employing such a flip-flop.

With regard to Figure 1, an 8-input AND function can be implemented in TLFF by mapping all threshold mapping inputs to Vdd. Therefore, T=8 and all data inputs have to be logic one in order to have a logic one output. An 8-input OR function can be implemented with T=1 and consequently, only one data input is necessary to be logic one in order to have a logic one output.

2.4 Dynamic Reconfigurability

Another attractive advantage of TLFF is the ability to change between two evaluations the TL function embedded in TLFF. This property comes from the fact that, in contrast with other TL gates (e.g., [3]), threshold mapping inputs, T_0, T_1, T_2 are accessible externally and can be treated as data inputs with negative weight, there are presented the Spice waveforms of a reconfigurable TLFF as in Figure 1 having applied the following set of input vectors: $[X_0, X_1, X_2,]$ = {[1, 1, 1], [0, 1, 1], [1, 1, 0], [0, 1, 0]} while threshold T is reprogrammed each four clock cycles as follows: $T = 8 \rightarrow 7 \rightarrow 5 \rightarrow 3$. Please note, that TLFF from Figure1 has $\Omega = [1, 3, 4]$ and $T \in \{0, 1, 3, 4, 5, 7, 8\}$.

2.5 Mapping technology

After logic optimization has produced the optimized network, technology mapping selects circuit elements to implement sub functions within this network. When wired together these circuit elements form a circuit implementing the entire network. This circuit is optimized to reduce a cost function that typically incorporates area and delay. Conventional approaches to technology mapping can be categorized as rule-based, library- based and cell generator approaches. The following sections briefly describe each of these approaches.

3. P-NAND CELL OPERATION

A threshold function can be implemented in the same way as any logic function, i.e., as a network of logic primitives or a pull-up network and pull-down network of pFETs and nFETs. As implementations of threshold logic gate (TLG) considered.

In this paper compute the predicate by performing a comparison of some electrical quantity, such as charge, voltage, or current. This is what distinguishes such implementations of a threshold gate with any of the conventional implementations of CMOS logic functions. However, the use of TLGs in conventional ASIC design has not been thoroughly explored due to the lack of efficient and reliable gate implementations. And the infrastructure required for automated synthesis and physical design the schematic of the threshold gate with k inputs, henceforth, referred to as pNAND-k.

It consists of three main components: 1) two groups of parallel pFET transistors as referred to as the left input network (LIN) and the right input network (RIN); 2) a sense amplifier (SA), 3) a set–reset (SR) latch. The cell is operated 10 is clocked, and its behavior can be abstracted to be that of a multi-input edge-triggered flip-flop (ETFF). Whereas a conventional D-type ETFF (D-FF) 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. Furthermore, like the D-FF, a pNAND cell can be made scannable and have other features, such as asynchronous preset and clear.

1) The waveforms from the SPICE simulation of a pNAND-3 extracted from layout, with and without transistors M9 and M10.

© 2017 AJAST All rights reserved.

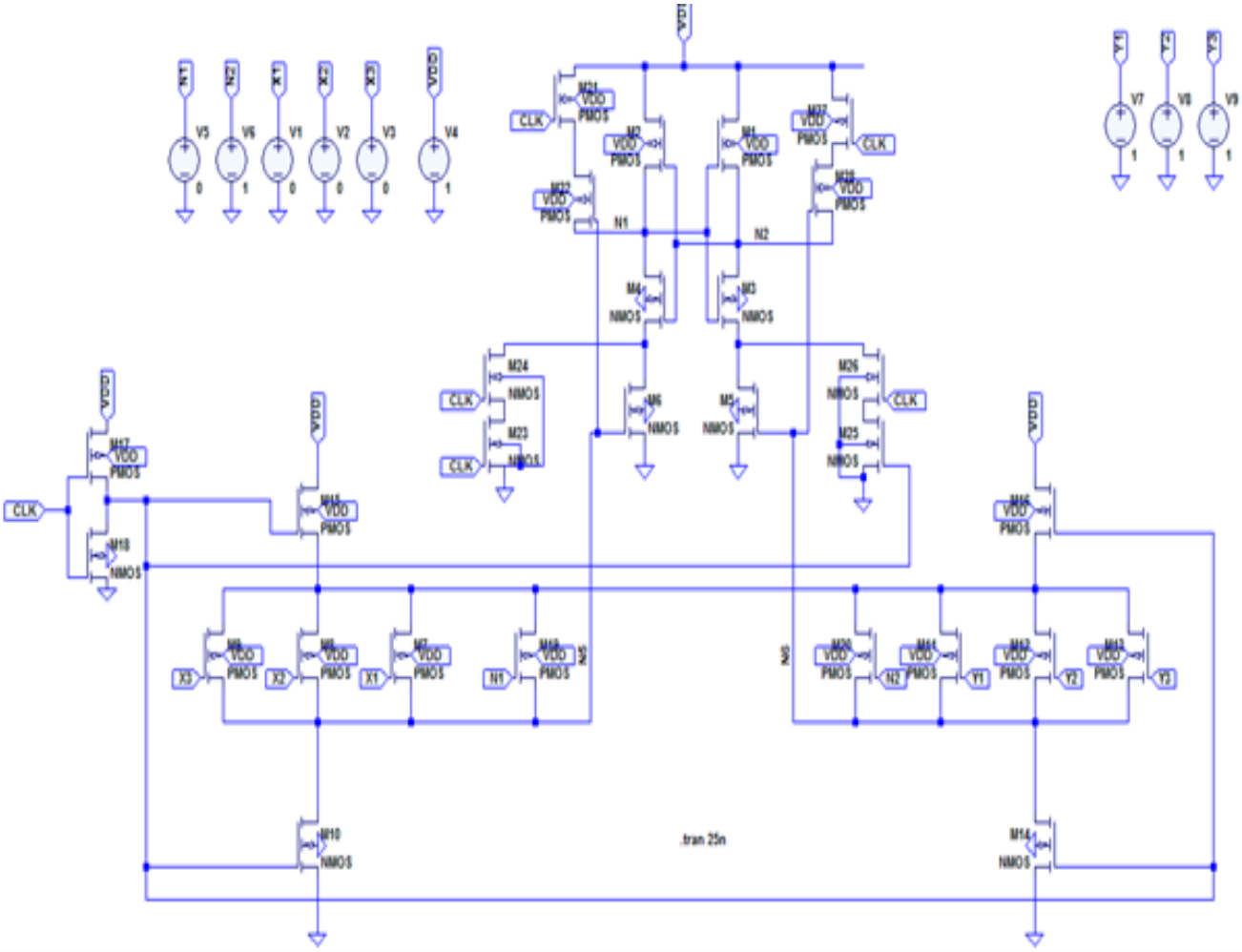

Fig.4. P-NAND without scan Cell Operation

2) For the specific signal assignment used in the technology mapping, the maximum number of active devices in the LIN or RIN among all the functions realized by a pNAND-3.

3) Therefore, the simulation starts with applying a CLK-0 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.

4) Next, N5 is discharged to ground through a capacitor, which turns OFF M5 and turns ON M7, pulling N1 to 1. This corresponds to when CLK 0 → 1. An input that results in ℓ active devices in the LIN and r active devices in the RIN is denoted by ℓ/r. The signal assignment procedure (explained in Section III) will ensure that ℓ ≠ r. Assume that ℓ > r.

As a result, the conductance of the LIN is higher than that of the RIN. As the discharge devices M18 and M19 are turned OFF, both N5 and N6 will rise to 1. Discharge is impeded as M2 turns ON, resulting in N2 getting pulled back to 1. As a result, the output node N1 is 0 and N2 is 1. As the circuit, its operation are symmetric, if ℓ < r , then the evaluation will result in N1 = 1 and N2 = 0.

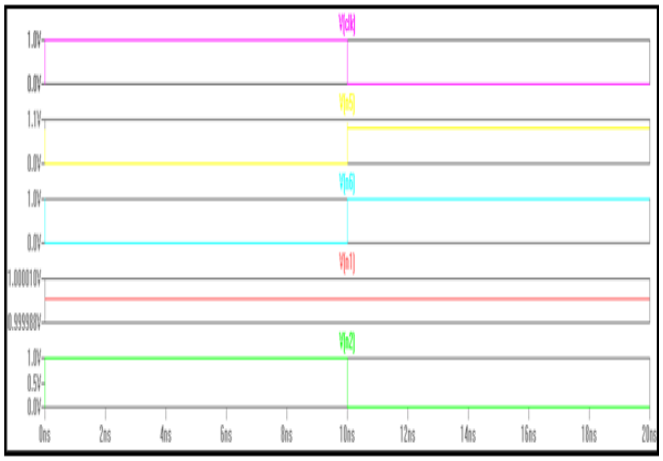

Fig.5. Simulation of p-NAND cell

For the efficiency power and analysis improvement of the Wallace tree multiplier increased from 33% and that of the FIR filter improved from 30%. Furthermore, the present results are obtained using an improved VLSI design flow that considers multiple PVT corners for tool-based optimization.

© 2017 AJAST All rights reserved.

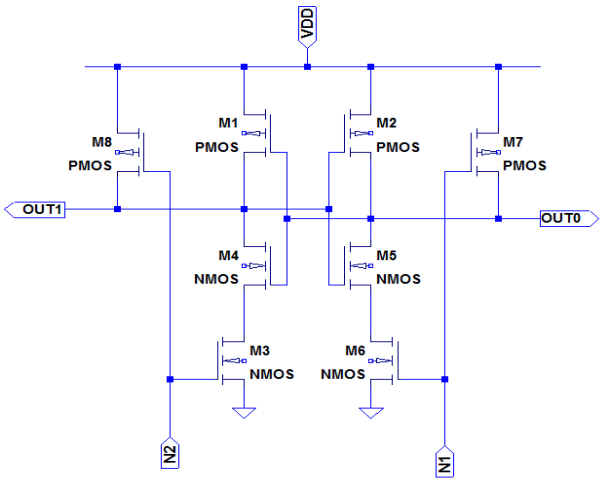

Fig.6. SR LATCH design

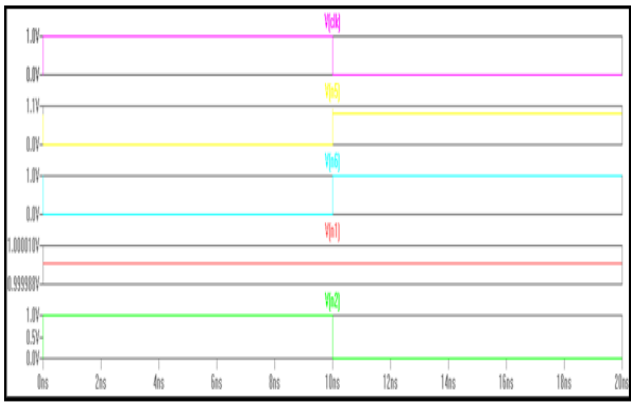

Fig.7. Simulation of SR LATCH

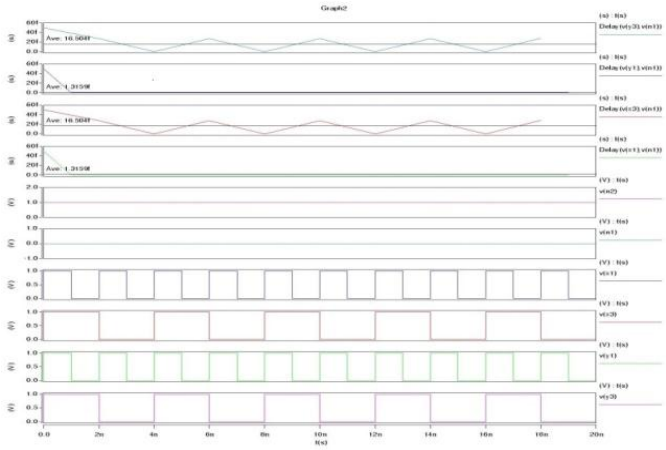

Fig.8. Power and analysis p-NAND

3.1 Scan Amplifier Implementation

If pNAND cells are to replace flip-flops and logic cones feeding them, scan capability is essential. The simplest way to make a D-FF scannable is to use a 2:1 mux that selects between the input D and the test input (TI), depending on whether or not the test mode is enabled (TE). This is not practical for a multi-input flip-flop, such as the pNAND cell. Although there exist several ways to implement scan for a pNAND cell, the one shown in Fig. 3 has negligible impact on the cell's performance and robustness during normal operation. All other variations were significantly worse in this regard. The additional transistors for scan are labeled as S1 through S6. In the normal mode, the signals TE and TI are both 0, which disables the scan-related transistors (S1–S4), and reduces the circuit function to the one shown in Fig. 1. In the scan mode, the TE signal itself acts as a clock. Therefore, 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 stream of bits into a scan chain consisting of pNANDs is as follows. Signal global TI (GTI) is the entry point for the scan data input to the pNAND chain.

1) Set CLK = 0 and TE = 0.
2) Set GTI = i^{th} 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 transistors S5 and S6 are included to eliminate a dc path during testing. In the absence 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.

Power analysis in the waveforms obtained via SPICE simulation of a scan chain of four pNAND-9 cells. The CLK is set to 0. The nodes Q1–Q4 are the output nodes of four cells that are initialized to 0. The scan pattern being registered is (Q1, Q2, Q3, Q4) = 1010.

Scanning of hybrid circuits, i.e., one with both D-FFs 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 (conventional way).

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.

4. CONCLUSION

In our experimental results shows 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 voltages for smaller k. For the 32-nm LTSPICE and the net list create with answer is HSPICE, that the result is accurate. In that current research in the use of threshold logics flip-flops includes new retiming algorithms, the design of asynchronous circuits, threshold logic-based field-programmable gate arrays, nonvolatile threshold logic flip-flops, and the combinations conventional logics of these different design approaches.

© 2017 AJAST All rights reserved.

REFERENCES

[1] Niranjan Kulkarni, Jinghua Yang, Jae-Sun Seo, (2016), "Reducing Power, Leakage, and Area of Standard-Cell ASICs Using Threshold Logic Flip-Flops" *IEEE transactions on very large scale integration (VLSI) systems.*

[2] Borivoje Nikolic, Vojin G. Oklobd zija, Vladimir Stojanovic, Wenyan Jia, James Kar-Shing Chiu and Michael Ming-Tak Leung (2000), "Improved Sense-Amplifier-Based Flip-Flop:Design and Measurements", *IEEE Journal of Solid-State Circuits*, vol. 35, no. 6, pp. 876-884.

[3] Celinski P., Lopez J.F., Al-Sarawi S. and Abbott D. (2001), "Low power, high speed, charge recycling CMOS threshold logic gate", *Electronics Letters*, vol. 37, no. 17, pp. 1067-1069.

[4] Hakan Ozdemir, Asim Kepkep, Banu Pamir, Yusuf Leblebici, and Ugur Cilingiroglu (1996), "A Capacitive Threshold-Logic Gate*", IEEE Journal of Solid-State Circuits*, Vol. 31, No. 8, pp. 1141-1150.

[5] Hidialgo-Lopez J. A, Tejero J.C, Fernhdez J. and Gago A. (1995), "New Types of Digital Comparators", *IEEE International Symposium on Circuits and Systems*, Vol. 1, pp. 29-32.

[6] Javier Lopez-Garcia, Jose Fernandez-Ramos and Alfonso Gagobohorquez (2004), "A Balanced Capacitive Threshold-Logic Gate", *Analog Integrated Circuits and Signal Processing*, Vol. 40, pp. 61–69.

[7] Jose M. Quintana, Maria J. Avedillo, Raul Jimenez and Esther Rodriguez-Villegas (2001), "Practical Low-Cost CPL Implementations of Threshold Logic Functions", *Proceedings of the 11th Great Lakes symposium on VLSI*, New York, pp. 139-144.

[8] Koji Kotani, Tadashi Shibata, Makoto Imai and Tadahiro Ohmi (1998), "Clock-Controlled Neuron-MOS Logic Gates", *IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal Processing*, Vol. 45, No. 4, pp. 518-522.

[9] Marius Padure, Sorin Cotofana, and Stamatis Vassiliadis (2003), "Design And Experimental Results of a CMOS Flip-Flop Featuring Embedded Threshold Logic", *Proceedings of the International Symposium on Circuits and Systems*, Vol. 5, pp. 253-256.

[10] Preeti Ranjan Panda, Aviral Shrivastava, Silpa B.V.N. and Krishnaiah Gummidipudi (2010), "Power-efficient System Design", *Springer*, pp. 1-253.

[11] Reto Zimmermann and Wolfgang Fichtner (1997), "Low-Power Logic Styles: CMOS versus Pass-Transistor Logic", *IEEE Journal of Solid-State Circuits*, Vol. 32, No. 7, pp. 1079-1090.

[12] Roland Strandberg and Jiren Yuan (2000), "Single Input Current-Sensing Differential Logic (SCSDL)", *IEEE International Symposium on Circuits and Systems*, Geneva, Switzerland, pp. 764-767.

[13] Samuel Leshner, Niranjan Kulkarni, Sarma Vrudhula and Krzysztof Berezowski (2010), "Design of a robust, high performance standard cell threshold logic family for DSM technology", *International Conference on Microelectronics*, pp. 52-55.

[14] Tadashi Shibata and Tadahiro Ohmi (1991), "An Intelligent MOS Transistor Featuring Gate-Level Weighted Sum and Threshold Operations", *International Electron Devices Meeting*, pp. 919-922.

© 2017 AJAST All rights reserved.