Power and Aging Characterization of Digital FIR Filters Architectures

Calimera, Andrea; Liu, Wei; Macii, Enrico; Nannarelli, Alberto; Poncino, Massimo

Published in:
First MEDIAN Workshop 2012

Publication date:
2012

Document Version
Publisher's PDF, also known as Version of record

Power and Aging Characterization of Digital FIR Filters Architectures

Andrea Calimera · Wei Liu · Enrico Macii · Alberto Nannarelli · Massimo Poncino

Abstract With technology scaling, newer metrics have been introduced, in addition to delay, area, and power dissipation, to characterize the behavior of digital systems. While dynamic and static power dissipation still remain the most serious concern at nanometer lengths (65nm and below), process-variation, temperature and aging induced variations pose new challenges in the fabrication of the next generation of ICs.

This work presents a detailed power and aging characterization of digital FIR filters in an industrial 45nm CMOS technology, and a design space exploration of different filter architectures with respect to throughput, area, power dissipation and aging. The exploration is intended to provide new design guidelines when considering aging of components in power/performance tradeoffs.

1 Introduction

With rapid scaling in CMOS technology, negative biased temperature instability (NBTI) is becoming one of the major reliability concerns that can limit device's lifetime. The NBTI effect primarily affects PMOS transistors and can lead to a shift in the threshold voltage up to 50 mV over time. The delay increase induced by NBTI aging can severely degrade performance and in the worst case result in system failure [1], [2].

Recent works have shown that NBTI-induced aging may benefit from the application of traditional power management implementations, namely *voltage scaling* and *power gating*. There exist however special classes of circuits for which very few power management options are available, because of the nature of their computation. Digital filters are one relevant example of this class: they implement a sort of streaming computation, in which generally no *structural* idleness is present.

Since the structure of a digital filter is quite fixed (e.g., direct or transpose forms), the most relevant degrees of freedom in their implementation are (a) the representation of data, (b) different approaches in implementing arithmetic operations, and (c) the frequency characteristics.

In this work, we characterize aging for a selection of architectures normally used to implement Finite Impulse Response, or FIR, filters and perform a design space exploration by comparing maximum delay (frequency/throughput), area, dynamic and static power dissipation with aging.

2 NBTI Effects on pMOS Transistor

Negative Bias Temperature Instability, or NBTI is a time-dependent degradation mechanism which affects p-type MOS transistors. NBTI arises when a pMOS, operating at high temperature, is negative biased (i.e., $V_{gs} = -V_{dd}$). Under this condition, called *stress-state*, the electric field across the gate dielectric causes the generation of traps at the Si/SiO_2 interface. This affects the threshold voltage V_{th}, whose absolute value increases over time, thus causing the shift of other electrical parameters, such as the drive current I_{ds} and the transconductance G_m. The resulting effect is the progressive slow down of CMOS standard gates. However, as soon as the stress is removed (i.e., $V_{gs} = 0$), *recovery-state*, a significant fraction of traps are annealed, and V_{th} appears to partially relax.

A. Calimera, E. Macii, M. Poncino
Politecnico di Torino, Italy

W. Liu, A. Nannarelli
Technical University of Denmark, Denmark

While there is no consensus on the exact quantum-mechanics mechanisms which govern the NBTI effects, the characterization of such effects is very challenging, and several fast measurement techniques have been developed recently [3]. In the meantime the reactivation-diffusion (R-D) model [4] has emerged as the most accredited model for pMOS NBTI. However, since a detailed treatment of NBTI models is out of the scope of this paper, we limit our contribution to list a few basic aspects that are essential for the understanding of the NBTI induced effects on digital circuits.

Operating Condition For a given set of technological parameters of a device (e.g., thickness oxide, channel strain, and nitrogen concentration), NBTI effects are mainly dependent on temperature (ΔV_{th} increases with increasing T), supply voltage (ΔV_{th} increases with increasing V_{dd}). Therefore, each library cell instance has its own specific NBTI-induced curve of V_{th} degradation in a multi-parameter space (V_{dd}, temperature, size and elapsed time). Therefore, a customized characterization of cell libraries is required for the estimation of delay.

Static Signal Probability The alternation of stress and recovery periods complicates the modeling of NBTI, since each single device should in principle be explicitly simulated by collecting the temporal profile of stress/ recovery cycles. Things are even more complicated for generic gates, in which each pMOS device is connected to a distinct input with its own time-dependent waveform. Fortunately, this behavior can be approximated with negligible error. It has been show in [5] that a generic waveform can be modeled as a periodic one with the same amount of stress time, and that aging is independent of the frequency of the applied waveform. Together, these properties imply that it is the *total stress time* that matters, (rather than the actual waveform), thus allowing to use *signal probabilities* in the simulation for the evaluation of the effective aging.

From the above considerations it is possible to derive a simplified model that describes the V_{th} variation induced by NBTI:

$$\Delta V_{th} = K \cdot \alpha \cdot t^{1/4}$$

where K is a parameter which lumps all the technological constants and considers the operating conditions of the device, α is the static zero-probability of the gate signal, and t is the elapsed time.

3 Digital Filters

Finite Impulse Response (FIR) filters are among the most popular components used in Digital Signal Processing (DSP). A FIR filter of order N is described by the expression

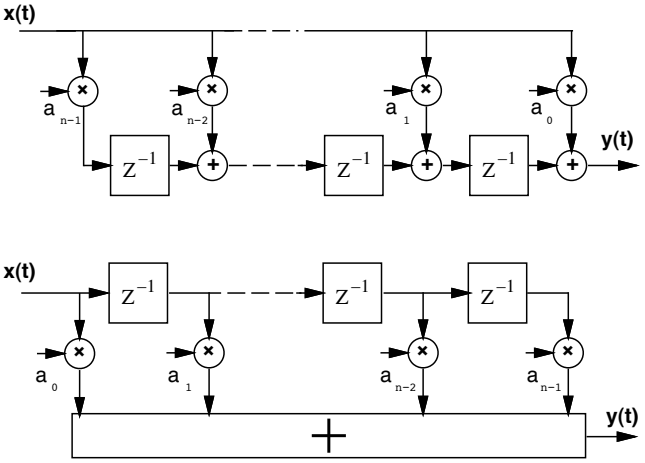

Fig. 1 FIR filters in transposed (top) and direct (bottom) form.

$$y(n) = \sum_{k=0}^{N-1} a_k x(n-k) \tag{1}$$

which is implemented in hardware with a sequence of multiply-add operations.

Normally, DSP systems process data in fixed-point format as the corresponding arithmetic units (circuits) are simpler (smaller) and faster. One of the first steps in designing a digital filter is to determine the dynamic range of the system according to the filter specifications. That is, to determine the bit-width of the datapath.

In the following, we present some alternatives and their tradeoffs in terms of the conventional metrics: delay, area and power dissipation, for implementing the FIR filter of expression (1). We consider a sample FIR filter of order $N = 16$ with dynamic ranges of 12, 10 and 22 bit for x, a and y, respectively. However, by (1), the results can be easily extended to different dynamic ranges (datapath bit-width) and filter's order (N).

FIR filters can be realized in either transposed or direct form, as shown in Fig. 1.

A FIR filter can be seen as the connection of N sections (referred as *"taps"*) containing a multiplier, a delay line (implemented by a register) and some adding structure: a regular adder for the transposed form, and an adder tree for the direct one.

We briefly describe the filters composing blocks and some implementation alternatives.

3.1 Multiplication

Multipliers are present in each filter tap. Therefore, it is crucial they are implemented efficiently with respect to delay, area and power dissipation.

Parallel multiplication (combinational) is a three steps computation [6]. We indicate with

$p = a \times x$

the product p ($n+m$ bits) of a n-bit operand x and a m-bit operand a.

1. First, m partial products
   $p_i = 2^i x \cdot a_i \qquad i = 0, \ldots, m-1$
   are generated. Because $a_i = \{0, 1\}$, this step can be realized with a $n \times m$ array of AND-2 gates[1]
2. Then, the m partial products are reduced to 2 by an adder tree
   $\sum_{i=0}^{m-1} 2^i x \cdot a_i = p_s + p_c \ .$
3. Finally, the carry-save product p_s, p_c is assimilated by a carry-propagate adder (CPA).
   $p = p_s + p_c \ .$

Because in a FIR filter, the product is added to the value y coming from the previous tap for transposed form, or to the other products for direct form, the final CPA of the multiplier is eliminated to save time. This is illustrated in Fig. 2 for the FIR in transposed form where the multiplication and the addition are fused and only one CPA per tap is used. The block marked CSA (Carry-Save Adder) in Fig. 2 is an array of full-adders which reduce 3 (n+m)-bit operands to 2 (n+m)-bit operands without carry propagation [6].

The delay in the adder tree and its area depend on the number of addends to be reduced ($m : 2$). By radix-4 recoding the multiplier a, often referred as Booth's recoding, the number of partial products is halved $\frac{m}{2}$. As, a consequence the multiplier's adder tree is smaller and faster. However, in terms of delay, the reduction in the adder tree is offset by a slower partial product generation, due to the recoding [6]. On the other hand, the reduction in area is significant, and the power dissipation is reduced as well due to both the reduced capacitance (area) and the nodes' activity because for two's complement representation, sequences of 1's are recoded into sequences of 0's resulting in less transitions when positive and negative values are alternated at the multiplier inputs.

In the evaluation of the different FIR filter architectures, we include filters with both radix-2 and radix-4 multiplication.

[1] Shifting (2^i) is done by hard-wiring the AND-2 array's output bits.

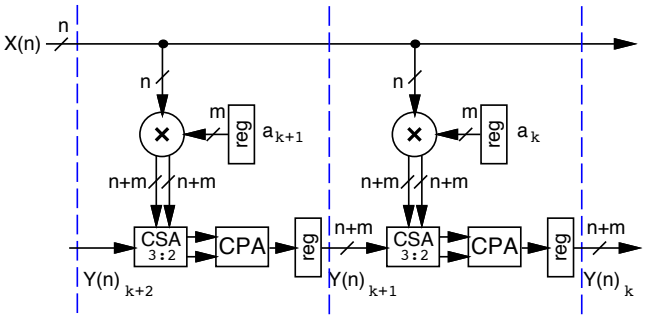

Fig. 2 Tap for FIR filters in transposed form.

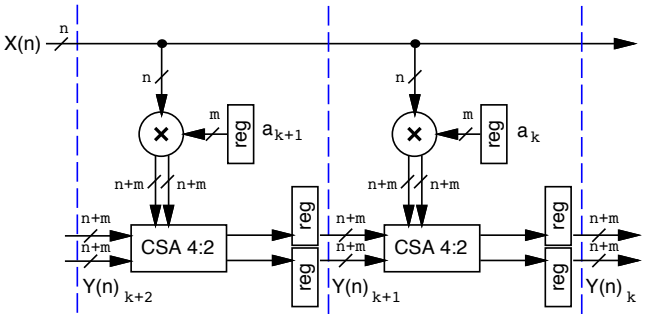

Fig. 3 Carry-save tap for FIR filters in transposed form.

3.2 Addition

The delay of the critical path, of the transposed FIR filter of Fig. 2 is

$t_{maxCPA} = t_{MULT} + t_{CSA} + t_{CPA} + t_{REG}$

This delay determines the maximum clock frequency and the throughput of the filter. Computing the carry-propagate addition in each tap can be avoided by storing the output of the CSA. This requires the doubling of the registers storing y in each tap, as shown in Fig. 3. In this way, the delay of the critical path for the transposed FIR filter is reduced to

$t_{maxCS} = t_{MULT} + t_{CSA4:2} + t_{REG} \ .$

Because we have now a carry-save representation in both $p = p_s + p_c$ and $y = y_s + y_c$, the CSA 3:2 is replaced by a CSA 4:2 which is slightly slower [6].

To give a numerical example, for a transposed FIR filter with radix-4 multipliers the speed-up for carry-save over carry-propagate is 25%.

$$\text{speed-up} \ = \frac{t_{maxCPA}}{t_{maxCS}} = \frac{1.27\ ns}{1.01\ ns} = 1.25$$

The carry-save representation of y, requires an additional stage at the filter output where a CPA assimilates $y_s + y_c = y$.

The above considerations apply for FIR filter in transposed form only. For filters in direct form (Fig.

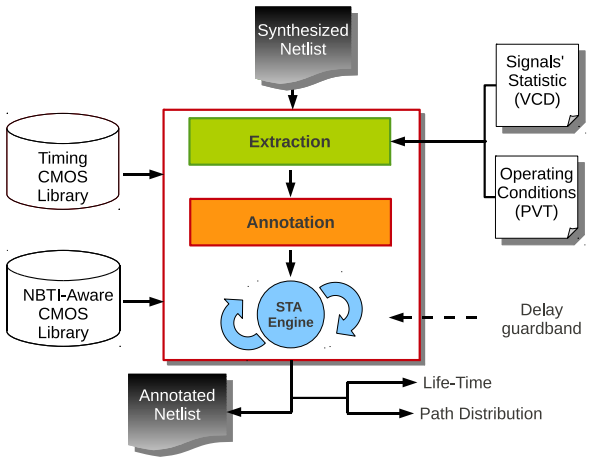

Fig. 4 Flow of the Proposed NBTI-Aware Exploration Framework for logic Circuits.

1 bottom), the carry-save output of the N multipliers is reduced by an adder tree $2N : 2$, and the final y is computed by a final stage consisting in a CPA, as for the case of the carry-save transposed FIR filter.

3.3 FIR filter architectures

Summarizing, we list the alternative architectures for the FIR filter implementation:

- **FIR-T-R2-CPA** is the transposed form implementation using radix-2 multipliers and one CPA per tap (Fig. 2).
- **FIR-T-R2-CSA** same as FIR-T-R2-CPA, but without CPA in taps (Fig. 3). An additional stage, implementing a single CPA, is inserted at the filter output.
- **FIR-T-R4-CPA** is the same as FIR-T-R2-CPA except that radix-4 multipliers are used within each tap.
- **FIR-T-R4-CSA** is the same as FIR-T-R2-CSA except that radix-4 multipliers are used within each tap.
- **FIR-D-R2** is the direct form implementation using radix-2 multipliers. An additional stage, implementing a single CPA, is inserted at the bottom of the tree (filter's output).
- **FIR-D-R4** is the same as FIR-D-R2, but with radix-4 multipliers.

4 NBTI-Aware Exploration Framework

Fig. 4 shows the implemented NBTI-aware exploration framework for the aging profile of standard cells based digital circuits.

After obtaining a synthesized circuit, a post-synthesis simulation[2] is needed to extract the statistical information of all the internal nodes. These information, which are stored on a dedicated ASCII file, formatted using the Value Change Dump (VCD) format, are the actual input of the implemented framework.

Depending on the operating PVT corner (i.e., Process, supply Voltage, and Temperature), and the static 0-probability of internal signals, the NBTI-induced delay degradation of each standard cell is extracted and annotated. This is done with the support of new NBTI-aware timing libraries that support time-dependent variations. Since today's design kits do not provide designers with such type of libraries, we filled new look-up tables containing the NBTI-induced delay degradation of each cell. The characterization, which is made under several operating conditions, i.e., Static 0-probabilities of the inputs, stress voltage (i.e., V_{gs}), temperature, and aging time, was run by using a dedicated SPICE-based aging analysis flow consisting of a two-phase simulation: the *pre-stress* simulation phase, in which we estimate the aging effects of the p-type transistors contained in the standard cell, and the *post-stress* simulation phase, where the stress information are integrated into the pMOS device parameters. At this point the delay degradation of the aged cell is measured and stored in a dedicated look-up table.

The netlist, annotated with the NBTI information, is then loaded into a standard Static Timing Analysis (STA) engine that provides timing information of the aged circuit, Fig. 4. The collected aging curves are than used to calculate the lifetime of the circuit. The lifetime is measured as the time at which the aging curve crosses a user defined *delay guard-band*.

5 Experimental Results

The FIR filter architectures previously described can have different characteristics in terms of power, delay and aging. To explore the design space of these filter architectures, we implemented the six units listed in Section 3.3.

The units are synthesized by Synopsys's Design Compiler with a 45 nm standard cell library in topographical mode. The topographical mode gives us better estimations of parasitics associated with interconnects which has an increasing contribution to path delay in nanometer technologies. The whole design flow is illustrated in Fig. 5. After we obtain the synthesized netlist, a

[2] Post-synthesis simulations are obtained applying dedicated testbenches that emulate the actual workload

Unit	$Area$ $[\mu m^2]$	$Delay$ $[ns]$	$MaxFreq$ $[MHz]$	P_{leak} $[\mu W]$	P_{dyn} $[mW]$	$Lifetime$ $[years]$
FIR-D-R2	47137	2.10	476	46.5	31.2	2.92
FIR-D-R4	35852	2.63	380	33.7	17.6	2.49
FIR-T-R2-CPA	51019	1.45	692	49.5	24.6	2.79
FIR-T-R2-CSA	54315	1.13	882	82.6	20.5	3.09
FIR-T-R4-CPA	43799	1.27	787	45.3	13.6	2.32
FIR-T-R4-CSA	43600	1.01	989	69.5	10.1	3.29

P_{dyn} is dynamic power measured at 100 MHz.

Table 1 Implementation results of different FIR filter architectures.

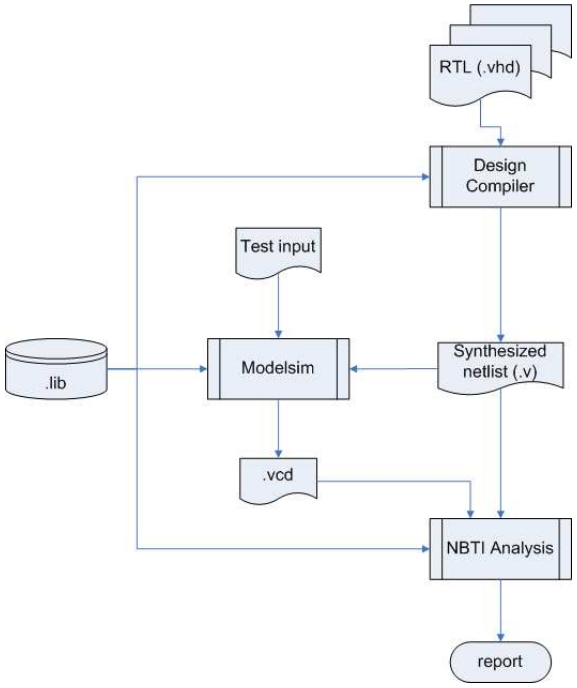

Fig. 5 Design space exploration flow.

post-synthesis simulation is run using Mentor Graphics' Modelsim to extract the toggling information of all the internal signals. The test-patterns used to extract the switching activity, program the FIR units as a low-pass filter and apply white noise (random vectors) at the filter input.

Our NBTI analysis tool then takes the synthesized design, the statistics about internal signals and library information as input and produces an aging profile of the design.

Table 1 shows the implementation results of the units. $MaxFreq$ is the maximum frequency at which the design can be clocked. To have a fair comparison, dynamic power dissipation P_{dyn} is normalized for all units at 100 MHz. In addition to the traditional metrics, we define $Lifetime$ as the time elapsed until when the delay of the critical path exceeds, due to aging effects, by 15% the delay of the critical path at age-0 (chip production).

From the data in Table 1, the direct form implementations, due to their large reduction tree, have the largest delay and thus the lowest maximum frequency. However, they occupy less area especially in the case of FIR-D-R4 which has the smallest area, and consequently, the smallest leakage power. The transposed form implementations have a much better performance over FIR-D-R2 and FIR-D-R4 at a price of larger area. For example, the maximum operating frequency in FIR-T-R4-CSA is two times than that of FIR-D-R2.

In general, units that use radix-4 multipliers are faster, smaller and more power efficient than their radix-2 counterparts which makes the radix-2 implementations less attractive. Additionally, FIR-T-R2-CSA and FIR-T-R4-CSA which save the results at each tap in carry-save format have a shorter delay and consume less power than FIR-T-R2-CPA and FIR-T-R4-CPA. This makes the CSA implementations more favorable. In fact, within all the units FIR-T-R4-CSA is the fastest in speed and lowest in dynamic power consumption.

To understand the aging characteristics of different architectures and the lifetime shown in Table 1, we plot the aging curves of all units in Fig. 6. The x-axis shows the cumulative operating time in years and the y-axis shows the critical delay in ns.

For each unit, the initial climb (as years moves from 0 to 1) has a larger slope than the other segments (as years moves from 1 to 6), showing a shift of the critical path. This means that a path other than the critical path at age-0 is more stressed under NBTI effects, and eventually, it obtains a larger delay induced by aging. For different units, the slope of the initial climb is ordered from high to low as: FIR-D-R4, FIR-D-R2, FIR-T-R2-CPA, FIR-T-R4-CPA, FIR-T-R2-CSA, FIR-T-R4-CSA which is consistent with the delay without

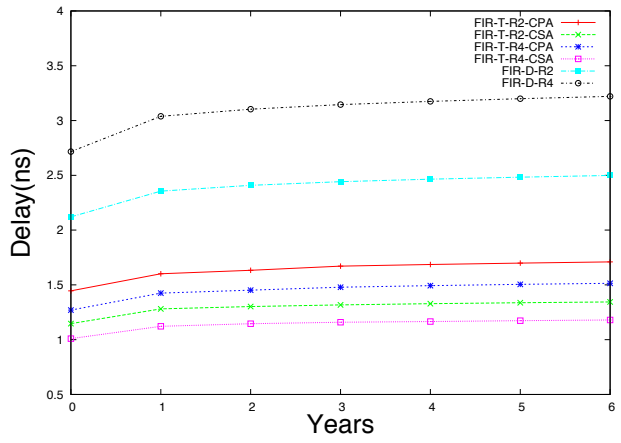

Fig. 6 Aging curve for different FIR filter architectures.

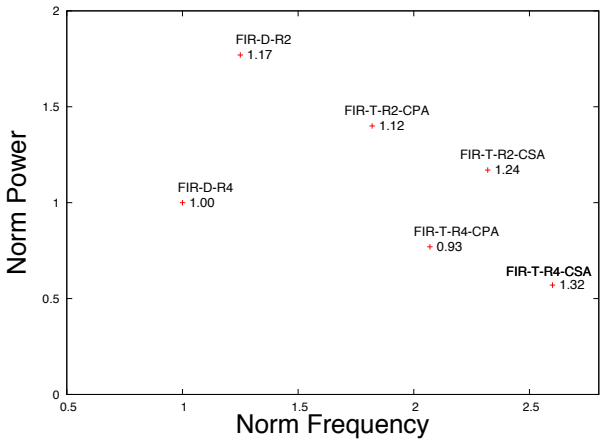

Fig. 7 Design space exploration of FIR filter architectures. Frequency, power and lifetime from Table 1 are normalized.

aging effects considered. This is as intuitively expected, because units with large logic depth have longer paths. After 1 year of operation time, the relative delay increase in longer paths is also larger than shorter paths.

However, over time, this does not necessarily to be true. Transistors along a longer path can age at a different speed than transistors along a shorter path. As explained in the previous section, the speed of aging is determined by the signal probabilities (duration of 0's which stress PMOS). Therefore, as time elapses a shorter path could have a longer delay, as we can see in the case of FIR-T-R2-CPA in Table 1 where its lifetime is shorter than FIR-D-R2.

Three metrics of the most interest, frequency, power and lifetime, of all units are plotted in Fig. 7. All numbers are normalized to unit FIR-D-R4. The x-axis and y-axis shows normalized frequency and power, respectively. Normalized lifetime is shown in the plot. Filter FIR-T-R4-CSA, shown in the lower right corner, performs best in terms of speed, power and lifetime, making it the most attractive architecture overall.

The characterization of aging in FIR filters showed that the architectures with shallower logical depth (transposed form and CSA) are the ones with longer aging time. Therefore, they provide the highest throughput sustainable over a longer period of time.

On the other hand, our design space exploration provided no evidence of a relationship between aging and power dissipation for the FIR filter architectures implemented. This is probably due to the high activity in filters (little idleness) which produces evenly distributed power dissipation in the different parts of the system independently of the number of paths affected by the aging.

6 Conclusions

In this paper, we described a design space exploration framework which besides traditional metrics also takes NBTI induced aging as another dimension. The exploration on a set of FIR filter architectures shows that transposed form implementations perform better than direct forms in terms of delay (they can sustain a higher throughput) and dynamic power dissipation, while filters in direct form have smaller area and, consequently, consume less static power.

The results also show that significant differences in lifetime exist in these units and that FIR filters providing higher throughput (frequency) are the ones that can sustain this throughput for a longer period of time. As for the power dissipated, the results of the characterization do not show any dependency of aging.

References

1. Borkar, S., "Electronics beyond nano-scale CMOS," Design Automation Conference, 2006 43rd ACM/IEEE, pp.807-808.
2. Schroder, Dieter K., Babcock, Jeff A., "Negative bias temperature instability: Road to cross in deep submicron silicon semiconductor manufacturing," *Journal of Applied Physics*, vol.94, no.1, pp.1-18, Jul 2003.
3. Ming-Fu Li, et.al., "Understand NBTI Mechanism by Developing Novel Measurement Techniques," *IEEE Transaction on Device and Materials Reliability*, vol. 8, no.1, pp. 62–71, Mar. 2008.
4. M. Alam, "Reliability- and process-variation aware design of integrated circuits," *Microelectronics Reliability*, vol. 48, no. 8, pp. 1114–1122, Aug. 2008.
5. S. V. Kumar *et al.*, "An analytical model for negative bias temperature instability," *Proc. of IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, pp. 493–496, Nov. 2006.
6. M. Ercegovac and T. Lang, *Digital Arithmetic*. Morgan Kaufmann Publishers, 2004.