ADAPTIVE FIR FILTER DESIGN USING AGING AWARE RELIABLE MULTIPLIER

Alvin K Varghese
JCT college of Engineering and Technology, Department of ECE,Pichanur,Coimbatore,
alvinkallely@gmail.com

Abstract — **The adaptive FIR filters have finite impulse responses and it consists of delay elements, shift registers, adders and multipliers. Today, the digital multipliers are the most important arithmetic functional units in every system design and the performance of all these systems depends upon the output of the multiplier. The Bias Temperature Instability affects both n-MOS and p-MOS transistors performances, by degrading its speed so, Adaptive hold logic based aging-aware reliable multiplier circuit design is proposed as a solution for this problem. Which will changes one cycle patterns to two cycle patterns thus, reducing aging effect and increase the performance of the overall system. Here the proposed adaptive FIR filter design using aging-aware reliable multiplier is based on LMS algorithm it provide higher accuracy through the variable latency method and will adjust the adaptive hold logic circuit to reduce the degradation in performance mainly due to the aging effect.**

Keywords— Bias temperature instability (BTI), variable latency, least mean square (LMS), reliable multiplier, adaptive FIR filter.

I. INTRODUCTION

Adaptive digital FIR filters can be widely used for many applications. The FIR filters are made by using series of delays, multipliers and adders. When the pMOS transistor is under negative bias then, an increase in threshold voltage is measured and is called negative bias temperature instability (NBTI) effect. Then the device drive current, noise margin, circuit speed, and the matching property will degrade. Similar effect in the nMOS transistor is called positive bias temperature instability (PBTI) and this effect is usually small as compared with NBTI effect so it can be ignored. These effects will degrades transistor speed and creates timing violations for long term and thus the system may fails. Here we propose an adaptive FIR filter design with adaptive hold logic (AHL) based aging-aware reliable multiplier circuit and LMS algorithm.

The variable latency design [1] divides the circuits in to shorter path and longer path. Here compares the worst case speed with the fastest existing multiplier and improves the performance of the system. The main problem of this method is that the number of pipelined stages increases then the power consumption increased and hence, to reduce power consumption [2] proposed. It describes a method to reduce the power consumption of digital multiplier based on dynamic bypassing of partial products. Because of charging and discharging of capacitors the transition activity dominates the total energy dissipation in static CMOS. Here by adding zero partial products generate signal transitions in the carry adder array. Another method for power reduction is described in [3] for the razor flip-flop. It operates at subcritical supply voltages and can dynamically detect and corrects the delay failures with the shadow flip-flop using a delayed clock with every flip-flop. Using this method, power reduction can be achieved but the propagation delay is large and will reduce the switching speed. The NBTI effect for a nanometer design is evaluated [4] and predictive model for degradation of both static and dynamic operations. VDD tuning, PMOS sizing and duty cycle reduction are the most effective techniques to mitigate the NBTI degradation. In this approach the area and power inefficient. Many methodologies for NBTI reduction have been proposed in [5]. If the multiplier outputs are known then some columns of the multiplier array can be turned off. However, a significant charge trapping can be seen in case for high-k/metal-gate nMOS transistor, so here the PBTI effect can’t be neglected because for 32-nm high-k/ metal-gate it is more significant than NBTI effect [6].

The conventional burn-in tests for nano scale designs in [7] will be effective to predict the degradation due to NBTI and there is no need need to test individual transistors. The threshold voltage shift (vt) for a field effect transistor (FET) and the temporal delay degradation of logic circuits due to NBTI is analyzed in [8] and [9]. A variable latency adder (VL-adder) technique for NBTI tolerance in [10], [15] detects the circuit failures and will reduce the NBTI-induced delay degradation on critical timing paths. It uses a fixed supply voltage and clock period and will reduce the manufacturing cost incurred by existing NBTI-tolerant techniques. The variable latency design, uses for reducing the timing waste of all circuits. Several variable latency adders are proposed using speculation with error correction and recovery in [22]. However no adaptive FIR filter designs using aging aware reliable multiplier that considers the aging effect has been done.

The remaining portion of this paper is organized as follows. Section II describes the array, column and row bypassing multipliers. Section III gives details about the adaptive FIR filter with aging aware reliable multiplier using LMS algorithm. The setup for the experiment and results are properly entered in section IV. The Section V contains conclusion of the proposed adaptive FIR filter.

II. PRELIMINARIES

A. ARRAY MULTIPLIER

Today the multipliers play almost every important role in digital signal processing and many other applications. The number of partial products is added as the main parameter of parallel multipliers that determines the performance of the multiplier. For reducing the number of partial products the Modified Booth algorithm is used. Similarly Wallace Tree algorithm provides speed improvements. To get the advantages of both algorithms we combine these two algorithms in one multiplier.

The existing array multiplier is well known due to its regular structure and multiplier circuit is based on add and shift algorithm. Here each partial product is generated by the multiplication of multiplicand with one multiplier bit and the partial products are shifted according to their bit orders and then added. Normal carry propagate adder can be used for addition. *N* is the multiplier length then *N-1* adders are needed. The 4*4 array multiplier is shown in Fig. 1. It consists of carry save adder (CSA) and each row contains (*n-1*) full adder (FA) cells. For propagating the carry the ripple carry adder can be used in the last row of the multiplier architecture.

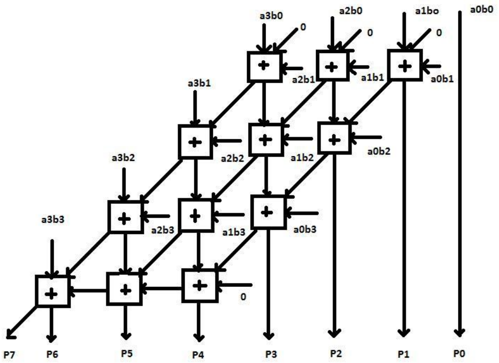

Fig. 1. 4*4 array multiplier.

B. COLUMN-BYPASSING MULTIPLIER

The existing column bypassing multiplier is a modification of normal array multiplier and here the full adder is always active regardless of input states. In [5] a low power column bypassing multiplier is proposed and if the corresponding bit in the multiplicand is 0 then the full adder (FA) operations are disabled. The existing 4*4 column-bypass multiplier is shown in Fig. 2. Here the full adder is modified as tri-state gates and one multiplexer. Here the multiplicand bit ai can be used as the selector of the multiplexer and the selector of the tri-state gate to decide the output of FA and can also be used as to turn off the input path of the FA. If second input of FA is 0 then the disabled inputs and the sum bit of the current FA is equal to the sum bit from its upper FA and thus power consumption can be reduced. If second input of FA is 1 then the normal sum result is selected.

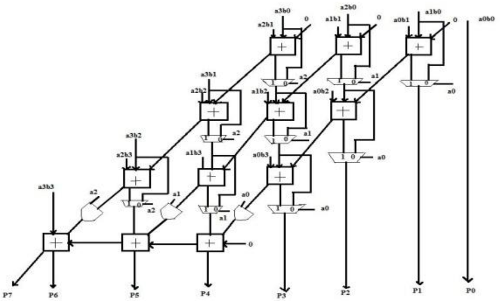

Fig. 2. 4*4 Column-bypass multiplier.

C. ROW-BYPASSING MULTIPLIER

The activity power of the array multiplier will reduce by using this row bypassing multiplier and the operation is similar to that of column-bypassing multiplier, but here the multiplicator is used by selector of the multiplexers and the tri-state gates. Fig. 3 shows the existing 4*4 Row-bypass multiplier.

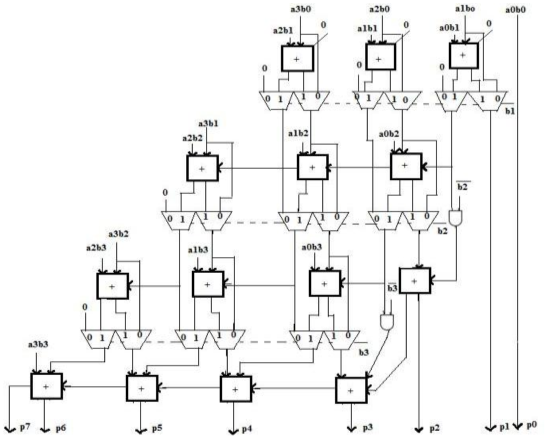

Fig. 3. 4*4 Row-bypass multiplier.

Here considering 11112 * 10012 are the two inputs and in the first and second rows are 0 for full adders. That menace, *b1* is 0 then the first row multiplexers select *aib0* as the sum bit and 0 as the carry bit. The inputs are bypassed to FAs in the second rows and at that time the tri-state gate turn off the input paths to the FAs and therefore no switching activities will take place in the first row FAs. Thus the power consumption can be reduced by this method. Because *b2* is 0 then also no switching activities occur in the second row FAs. The FAS must be active in the third row because b3 is not 0. More details can be found in [2].

III. PROPOSED ADAPTIVE FIR FILTER

This section details the proposed adaptive FIR filter using aging–aware reliable multiplier design and introduces the overall architecture and the functions of each component and describes how the AHL will adjust the circuit when significant aging occur.

A. PROPOSED ARCHITECTURE

Fig. 4 shows the proposed adaptive FIR filter using aging-aware multiplier, which includes multipliers, adders and delay elements.

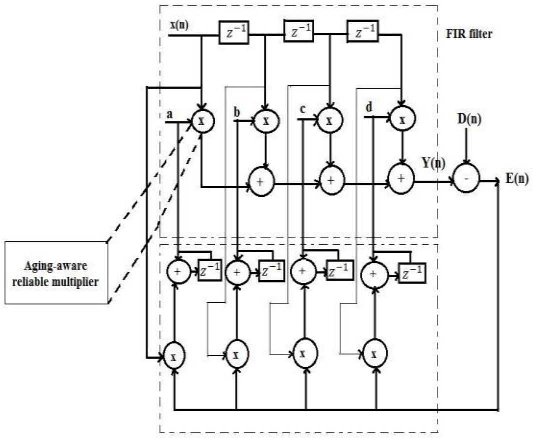

Fig. 4 proposed adaptive FIR filter.

B. ADAPTIVE FIR FILTER

Digital signal processors usage is increased since the consumers demand for low power portable multimedia and computing devices and which the filters are largely used component in devices such as mobiles and other communication devices. The filters perform manipulation or signal processing for eliminating any unwanted noise induced in the signal.

By using variable parameters the transfer function of a filter can be controlled and such filters are called an adaptive filters and it is a linear filter whose parameters can be adjust according to an optimizing algorithm. For example in case of heartbeat recording, the ECG wave is free from the noise. In this case adaptive filters are applicable and they are characterized for their flexibility and accuracy. Fig. 5 shows adaptive filter. The commonly used algorithms are Least mean square and Recursive Least mean square. There are four types of adaptive filtering configurations and all the systems have same general parts that are input x(n), a desired result d(n), an output y(n), an adaptive transfer function w(n), and actual output y(n).

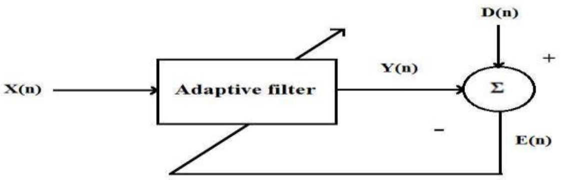

Fig. 5. Adaptive filter.

C. LMS ALGORITHM

The LMS algorithm uses a gradient descent to estimate a time varying signal. Fig. 6 shows the adaptive filter with LMS algorithm. The gradient descent method always finds a minimum. If it exists then the negative direction of the gradient is taken. Thus by adjusting the filter coefficients to minimize the error and the gradient is the Del operator (partial derivative). In this case it is applied to find the divergence of a function which is an error with respect to the nth coefficient.

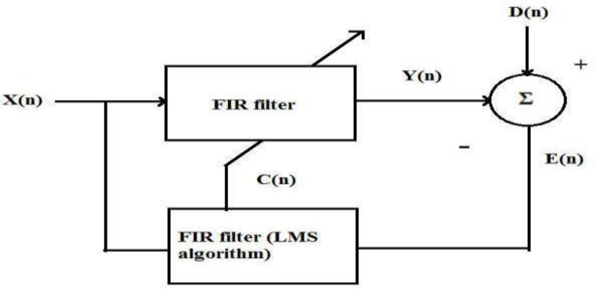

Fig. 6. adaptive filter with LMS algorithm.

The LMS algorithm takes negative gradient of a function to minimize error. The d(n) is the desired signal and it is tracked from the filter coefficients c(n) or w(n). The known signal x(n) is the input signal that is fed to the FIR filter. The difference between desired signal d(n) and output y(n) is the error signal. The error e(n) is then fed to the adaptive filter to compute the filter coefficients c(n+1) iteratively to minimize the error. The convergence time of the LMS algorithm depends on the step size µ. The value of µ should be scientifically calculated based on the effects the environment will have on d(n).

D. AGING –AWARE MULTIPLIER

The adaptive hold logic (AHL) in the novel variable latency multiplier architecture can decide whether the input patterns requires one or two cycles. It will use judging criteria for performance evaluation after aging effect. An adaptive hold logic based aging-aware reliable multiplier design method that is suitable for large multipliers and the experiment is performed for both 16- and 32-bit multipliers and the proposed architecture can be easily extended to large designs. Fig. 7 shows the aging-aware reliable multiplier used here. It consists of two registers, column/row bypassing multipliers, razor flip-flop, AHL and one AND gate.

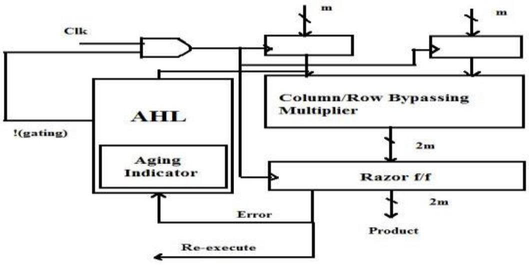

Fig. 7. Aging-aware reliable multiplier.

E. RAZOR FLIP-FLOP

The razor flip-flop contains a main flip-flop, shadow latch, XOR gate and MUX. The main flip-flop will catches the execution result using normal clock signal and the shadow latch uses delayed clock signal for catching the execution result. The delayed clock signal is slower than the normal clock signal. Fig. 8 shows the razor flip-flop.

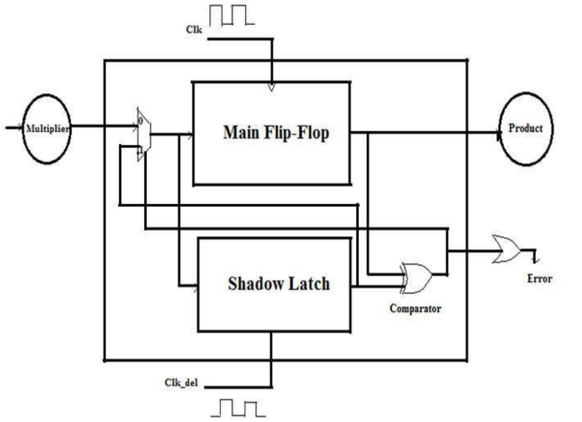

Fig. 8. Razor flip-flop.

In the shadow latch the latched bit is different from that of the main flip-flop, this means the current operation have path delay larger than the cycle period then the main flip-flop will give incorrect result. If there is any errors occurs then the razor flip-flop will gives 1 as output and notify the system to re-execute the operation. Thus inform the AHL circuit that an error has occurred. The razor flip-flop will detect which operation needs one cycle to complete the operation and if not then it is re-executed using two clock cycles.

F. ADAPTIVE HOLD LOGIC

The AHL circuit is the main component in the aging-aware variable-latency multiplier. Fig. 9 shows the details of AHL circuit. The aging indicator, two judging blocks, one MUX, and one D flip-flop are the main blocks of the adaptive hold logic circuit.

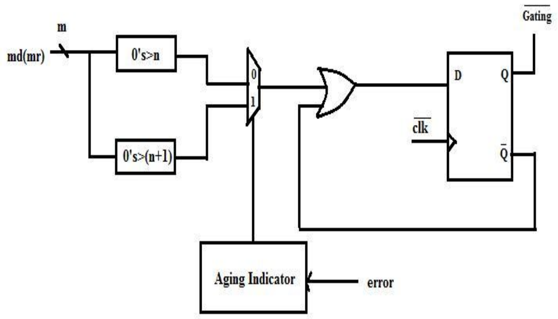

Fig. 9. AHL circuit.

When the aging increased a particular threshold value then the indicator will indicates that. The aging indicator is implemented using a simple counter that counts the number of errors over a certain amount of operations. When the operations are completed then it reset to zero. The column or row bypass multiplier operations are not completed if the cycle period is too short. It cause timing violations and the complete operation fails. The Razor flip-flops will generate error signals by catching all the timing violations. If that error exceeds a particular threshold value that means aging effect is significant and at that time aging indicator will output 1. Otherwise, the circuit has not significant timing degradation due to aging effect then the indicator will outputs to 0.

In the column-bypass multiplier the number of zeros in the multiplicand is greater than n, then the first judging block in the AHL will output 1 and if it is larger than n+1 then the second judging block will output 1. Similarly in the case of row-bypass multiplier instead of multiplicand we check multiplicator. Both the judging blocks will decide that the input pattern requires one or two cycles and at a time only one can be chosen. At the beginning aging indicator will output 0 because there is no significant aging effect and so the first judging block can be used. After a period of time second judging block can be used because the aging effect is significant.

The operation of AHL circuit is given bellow as detail. When an input pattern arrives both judging block will decide whether it requires one cycle or two cycles to complete the operation. Based on the output of aging indicator the multiplexer select either one result. Between the result of the multiplexer and the input of the D flip-flop, OR operation can be performed. When the output of multiplexer is 1 then !(gating) signal will become 1 and in next cycle the flip-flop will receive new data. On the other hand when the output is 0 which means the input patterns requires two cycles and the OR gate will output 0 to the D flip-flop. If the input pattern requires two cycles then, AHL will output 0 to disable the clock signal of the flip-flop. Otherwise normal operation can be performed. The result of the column/row bypass multiplier is passed to razor flip-flop and check for path delay timing violations. When the result of the multiplier is incorrect because of timing violations that means the cycle period is not long enough for the correct operation. Thus the razor flip-flop will output an error signal and it will force the system to re-execute using two cycles. The proposed multiplier will accurately predict whether the input pattern requires one cycle or two cycles to complete operations.

IV. EXPERIMENTAL RESULTS

Here a hardware description language such as VHDL is used as the coding language and it is an electronic design automation to describe digital mixed signal systems such as integrated circuits and field programmable gate arrays. It is developed by the U.S Department of Defense and can be used as a general purpose parallel programming language. Here in the proposed work simulation can be done using modelsim and comparison can be done using Xilinx. Modelsim is a multi-language HDL simulation environment used for simulation of much hardware description languages such as verilog, VHDL and system C and is developed by Mentor Graphics. Fig. 10 shows the simulation result of the proposed adaptive FIR filter using aging-aware multiplier.

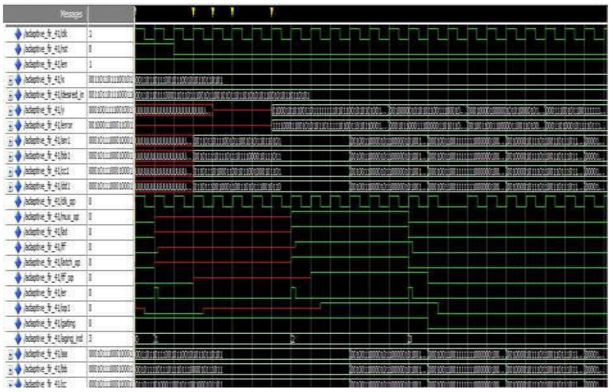

Fig. 10. Simulation result of adaptive FIR filter.

The simulation result of column bypassing multiplier is shown in Fig. 11. The advantages of the proposed column bypass multiplier are that it uses variable latency design in which the path from input to output can be varied. The existing column bypass multiplier uses fixed latency design, which will hold the clock signal for one clock cycle when there is an error, and hence there by increasing the delay. Whereas in the proposed method the error exceeds a particular threshold value, it holds the clock signal for one clock cycle and hence the delay can be minimized here.

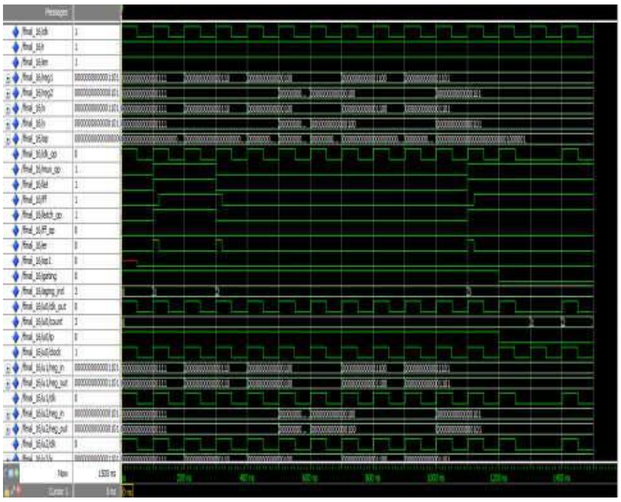

Fig. 11. Simulation result for column by passing multiplier.

The Fig. 12 shows simulation result for row bypassing multiplier. It also uses variable latency design as that of column bypassing multiplier.

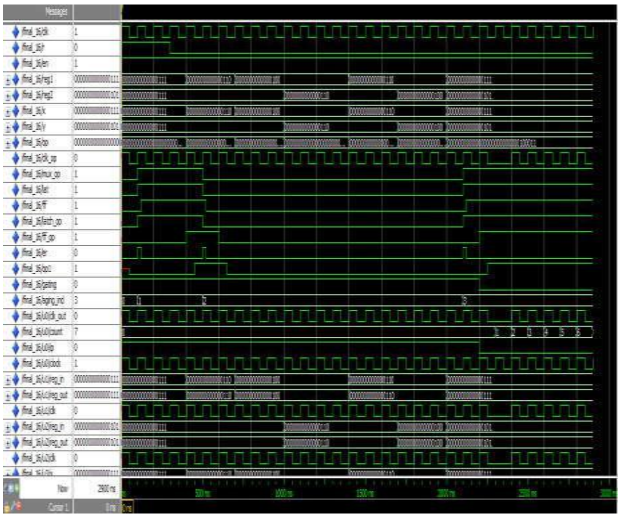

Fig. 12. Simulation result for row by passing multiplier.

V. CONCLUSION

The proposed adaptive FIR filter is designed using aging-aware reliable multiplier and the experimental result shows that proposed multiplier will provide better performance through the variable latency and adjust the AHL to mitigate the aging degradation. The delay of the transistors increased due to aging effect and interconnects also have aging issue, which is called electro migration. It occurs when the current density is high enough to cause the drift of metal ions along the direction of electron flow. Proposed adaptive FIR filter using variable latency multiplier can be used under the influence of both the BTI effect and electro migration.

REFERENCES

[1] M. Olivieri, "*Design of synchronous and asynchronous variable-latency pipelined multipliers*,"IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 9, no. 4, pp. 365–376, Aug. 2001.

[2] J. Ohban, V. G. Moshnyaga, and K. Inoue, "*Multiplier energy reduction through bypassing of partial products*," in Proc. APCCAS, 2002, pp. 13-17.

[3] D. Ernst et al., "*Razor: A low-power pipeline based on circuit-level timing speculation*," inProc. 36th Annu.IEEE/ACM MICRO, Dec. 2003, pp. 7–18.

[4] R. Vattikonda, W. Wang, and Y. Cao, "*Modeling and miimization of pMOS NBTI effect for robust naometer design*," inProc. ACM/IEEE DAC, Jun. 2004, pp. 1047–1052.

[5] M.-C. Wen, S.-J. Wang, and Y.-N. Lin, "*Low power parallel multiplier with column bypassing*," in Proc. IEEE ISCAS, May 2005, pp. 1638-1641.

[6] S. Zafar, A. Kumar, E. Gusev, and E. Cartier, "*Threshold voltage instabilities in high-k gate dielectric stacks*," IEEE Trans. Device Mater. Rel., vol. 5, no. 1, pp. 45–64, Mar. 2005.

[7] B. C. Paul, K. Kang, H. Kufluoglu, M. A. Alam, and K. Roy, "*Impact of NBTI on the temporal performance degradation of digital circuits*," IEEE Electron Device Lett., vol. 26, no. 8, pp. 560–562, Aug. 2005.

[8] R. Vattikonda, W. Wang, and Y. Cao, "*Modeling and minimization of pMOS NBTI effect for robust nanometer design*," in Proc. 43rd ACM/IEEE DAC, Aug. 2006, pp. 1047–1052.

[9] S. Zafar et al., "*A comparative study of NBTI and PBTI (charge trapping) in SiO2/HfO2 stacks with FUSI, TiN, Re gates*," in Proc. IEEE Symp. VLSI Technol. Dig. Tech. Papers, 2006, pp. 23–25.

[10] B. C. Paul, K. Kang, H. Kufluoglu, M. A. Alam, and K. Roy, "*Negative bias temperature instability: Estimation and design for improved reliability of nanoscale circui*t,"IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst., vol. 26, no. 4, pp. 743–751, Apr. 2007.

[11] Y. Chen, H. Li, J. Li, and C.-K. Koh, "*Variable-latency adder (VL-Adder): New arithmetic circuit design practice to overcome NBTI*," in Proc. ACM/IEEE ISLPED, Aug. 2007, pp. 195–200.