ISSN 2348 – 8034
Impact Factor- 3.155

GLOBAL JOURNAL OF ENGINEERING SCIENCE AND RESEARCHES

A NOVEL MULTIPLIER DESIGN USING ADAPTIVE HOLD LOGIC TO MITIGATE BTI EFFECT

PG Student A.Gopivignesh*[1] and Associate Professor P.N.Sundararajan[2]
*[1,2]PSNA College of Engineering and Technology, Tamil Nadu, India

ABSTRACT

The overall performance of a system depends on the performance of the multipliers, thus digital multipliers are among the most critical arithmetic functional units; but their performance is affected by negative bias temperature instability effect and positive bias temperature instability effect. The negative bias temperature instability effect occurs when a pMOS transistor is under negative bias (Vgs =−Vdd), increasing the threshold voltage of the pMOS transistor, and reducing multiplier speed. A similar phenomenon, positive bias temperature instability, occurs when an nMOS transistor is under positive bias. Both these effects degrade transistor speed, and in the long term, the system may fail due to timing violations. Therefore this paper proposes an aging-aware multiplier design with a new adaptive hold logic (AHL) circuit. To mitigate performance degradation due to the aging effect, this architecture can be applied to a column- bypassing multiplier or row-bypassing multiplier.

Keywords- *Adaptive hold logic (AHL), negative bias temperature instability (NBTI), positive bias temperature instability (PBTI), reliable multiplier, variable latency.*

I. INTRODUCTION

Digital multipliers are among the most critical arithmetic functional units in many applications, such as the Fourier transform, discrete cosine transforms, and digital filtering. The through put of these applications depends on multipliers, and if the multipliers are too slow, the performance of entire circuits will be reduced.

Furthermore, negative bias temperature instability (NBTI) occurs when a pMOS transistor is under negative bias (Vgs = −Vdd). In this situation, the interaction between inversion layer holes and hydrogen-passivated Si atoms breaks the Si–H bond generated during the oxidation process, generating H or H2 molecules. When these molecules diffuse away, interface traps are left. The accumulated interface traps between silicon and the gate oxide interface result in increased threshold voltage (Vth), reducing the circuit switching speed. When the biased voltage is removed, the reverse reaction occurs, reducing the NBTI effect. However, the reverse reaction does not eliminate all the interface traps generated during the stress phase, and Vth is increased in the long term. Hence, it is important to design a reliable high-performance multiplier. The corresponding effect on an nMOS transistor is positive bias temperature instability (PBTI), which occurs when an nMOS transistor is under positive bias. Compared with the NBTI effect, the PBTI effect is much smaller on oxide/polygate transistors, and therefore is usually ignored.

II. RELATED WORK

Threshold voltage instabilities in high-k gate dielectric stacks

Author: S. Zafar, A. Kuma

Year: Mar. 2005

Over recent years, there has been increasing research and development efforts to replace SiO_2 with high dielectric constant (high-κ) materials such as HfO_2, HfSiO, and Al_2O_3. An important transistor reliability issue is the threshold voltage stability under prolonged stressing. In these materials, threshold voltage is observed to shift with stressing time and conditions, thereby giving rise to threshold voltage instabilities. In this paper, we review various causes of threshold voltage instability: charge trapping under positive bias stressing, positive charge creation under negative bias stressing (NBTI), hot-carrier stressing, de-trapping and transient charge trapping effects in high-κ gate dielectric stacks.

Impacts of NBTI/PBTI on timing control circuits and degradation tolerant design in nanoscale CMOS SRAM

Author: H.-I. Yang, S.-C. Yang

Year: Jun. 2011

Negative-bias temperature instability (NBTI) and positive-bias temperature instability (PBTI) weaken PFET and NFET over the lifetime of usage, leading to performance and reliability degradation of nanoscale CMOS

SRAM. In addition, most of the state-of-the-art SRAM designs employ replica timing control circuit to mitigate the effects of leakage and process variation, optimize the performance, and reduce power consumption. NBTI and PBTI also degrade the timing control circuits and may render them ineffective. In this paper, we provide comprehensive analyses on the impacts of NBTI and PBTI on a two-port 8T SRAM design, including the stability and Write margin of the cell, Read/Write access paths, and replica timing control circuits. We show, for the first time, that because the Read/Write replica timing control circuits are activated in every Read/Write cycle, they exhibit distinctively different degradation behavior from the normal array access paths, resulting in degradation of timing control and performance.

Design techniques for NBTI-tolerant power-gating architecture

Author: A. Calimera, E. Macii

Year: Apr. 2012 While negative bias temperature instability (NBTI) effects on logic gates are of major concern for the reliability of digital circuits, they become even more critical when considering the components for which even minimal parametric variations impact the lifetime of the overall circuit. pMOS header transistors used in power-gated architectures are one relevant example of such components. For these types of devices, an NBTI-induced current capability degradation translates into a larger -drop effect on the virtual- rail, which unconditionally affects the performance and, thus, the reliability of all power-gated cells. In this brief, we address the problem of designing NBTI-tolerant power-gating architectures. We propose a set of efficient NBTI-aware circuit design solutions, including both static and dynamic strategies, that aim at improving the lifetime stability of power-gated circuits by means of oversizing, body biasing, and stress-probability reduction while minimizing the design overheads.

Performance optimization using variable-latency design style

Author: Y.-S. Su, D.-C. Wang

Year: Oct. 2011 In many designs, the worst-case delay of a critical path may be activated infrequently. Traditional optimization approaches assume the worst-case conditions, which could lead to an inefficient resource usage. It is possible to improve the throughput of such designs by introducing variable latency. One existing realization of the variable-latency design style is based on telescopic units. The design of the hold logic in telescopic units influences the circuit's throughput. In this paper, we show that the traditionally designed hold logic may be inaccurate. We use the short path activation conditions to obtain more accurate hold logic and improve the efficiency of telescopic units. To reduce the overhead for large circuits, we propose an efficient heuristic methodology of constructing non-exact hold logic.

Negative bias temperature instability: Estimation and design for improved reliability of nanoscale circuit

Author: B. C. Paul, K. Kang

Year: Apr. 2007 Negative bias temperature instability (NBTI) has become one of the major causes for temporal reliability degradation of nanoscale circuits. In this paper, we analyze the temporal delay degradation of logic circuits due to NBTI. We show that knowing the threshold-voltage degradation of a single transistor due to NBTI, one can predict the performance degradation of a circuit with a reasonable degree of accuracy. We also propose a sizing algorithm, taking the NBTI-affected performance degradation into account to ensure the reliability of nanoscale circuits for a given period of time.

III. PROPOSED FRAMEWORK

Proposed aging-aware multiplier architecture, which includes two m-bit inputs (m is a positive number), one 2m-bit output, one column- or row-bypassing multiplier, 2m 1-bit Razor flip-flops, and an AHL circuit, as shown in fig.1.

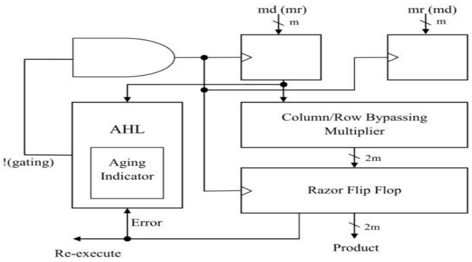


Fig. 1: Proposed architecture

3.1 Column-Bypassing Multiplier: Column-Bypassing Multiplier a column-bypassing multiplier is an improvement on the normal array multiplier (AM). The AM is a fast parallel AM and is shown in Fig. 2. The multiplier array consists of (n−1) rows of carry save adder (CSA), in which each row contains (n − 1) full adder (FA) cells. Each FA in the CSA array has two outputs: 1) the sum bit goes down and 2) the carry bit goes to the lower left FA. The last row is a ripple adder for carry propagation. The FAs in the AM are always active regardless of input states.

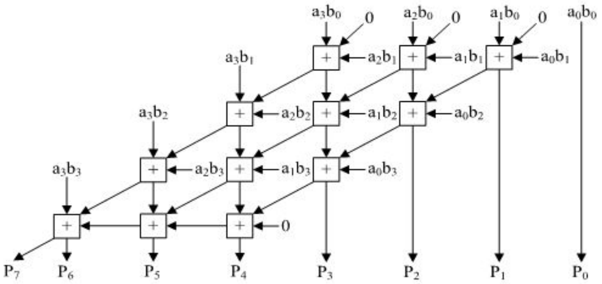


Fig. 2: 4×4 Normal Array Multiplier

A low-power column-bypassing multiplier design is proposed in which the FA operations are disabled if the corresponding bit in the multiplicand is 0. Fig. 3 shows a 4×4 column-bypassing multiplier.

The multiplicand bit a can be used as the selector of the multiplexer to decide the output of the FA, and ai can also be used as the selector of the tri-state gate to turn off the input path of the FA. If ai is 0, the inputs of FA are disabled, and the sum bit of the current FA is equal to the sum bit from its upper FA, thus reducing the power consumption of the multiplier. If ai is 1, the normal sum result is selected.

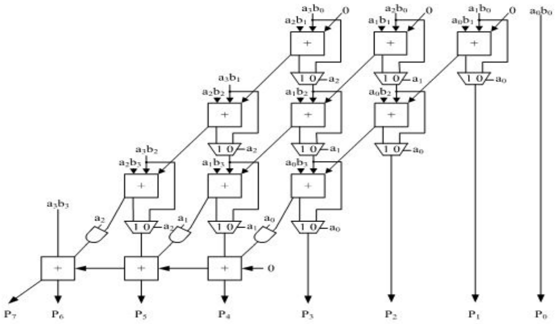


Fig. 3: 4×4 Column-Bypassing Multiplier

3.2 Row-Bypassing Multiplier The operation of the low-power row-bypassing multiplier is similar to that of the low-power column-bypassing multiplier, but the selector of the multiplexers and the tri-state gates use the multiplicator. Fig. 4 shows a 4×4 row-bypassing multiplier.

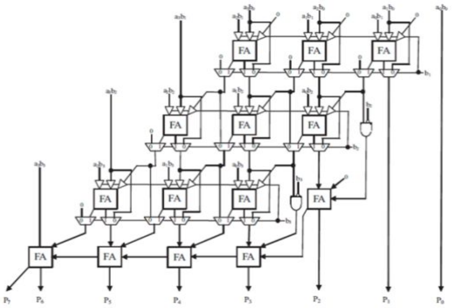


Fig. 4: 4×4 Row-Bypassing Multiplier

Here, the two aging-aware multipliers can be implemented using similar architecture, and the difference between the two bypassing multipliers lies in the input signals of the AHL. According to the bypassing selection in the column or row bypassing multiplier, the input signal of the AHL in the architecture with the column-bypassing multiplier is the multiplicand, whereas of the row-bypassing multiplier is the multiplicator.

3.3 Razor flip-flops

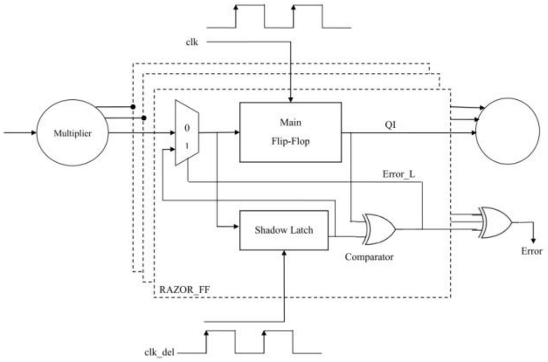


Fig. 5: Razor flip-flop

Razor flip-flops can be used to detect whether timing violations occur before the next input pattern arrives. A 1-bit Razor flip-flop contains a main flip-flop, shadow latch, XOR gate, and mux, as shown in fig. 5. The main flip-flop catches the execution result for the combination circuit using a normal clock signal, and the shadow latch catches the execution result using a delayed clock signal, which is slower than the normal clock signal. If the latched bit of the shadow latch is different from that of the main flip-flop, this means the path delay of the current operation exceeds the cycle period, and the main flip-flop catches an incorrect result. If errors occur, the Razor flip-flop will set the error signal to 1 to notify the system to re-execute the operation and notify the AHL circuit that an error has occurred. We use Razor flip-flops to detect whether an operation that is considered to be a one-cycle pattern can really finish in a cycle. If not, the operation is re-executed with two cycles. Although the re-execution may seem costly, the overall cost is low because the re-execution frequency is low.

3.4 AHL circuit The AHL circuit is the key component in the aging-ware variable-latency multiplier. Fig. 6 shows the details of the AHL circuit. The AHL circuit contains an aging indicator, two judging blocks, one mux, and one D flip-flop. The aging indicator indicates whether the circuit has suffered significant performance degradation due to the aging effect. The aging indicator is implemented in a simple counter that counts the number of errors over a certain amount of operations and is reset to zero at the end of those operations. If the cycle period is too short, the column- or row-bypassing multiplier is not able to complete these operations successfully, causing timing violations. These timing violations will be caught by the Razor flip-flops, which generate error signals. If errors happen frequently and exceed a predefined threshold, it means the circuit has suffered significant timing degradation due to the aging effect, and the aging indicator will produce output signal 1; otherwise, it will output 0 to indicate the aging effect is still not significant, and no actions are needed.

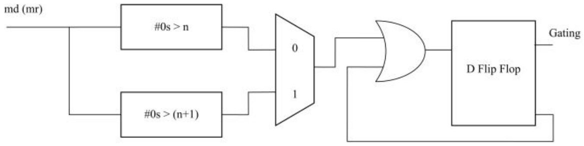


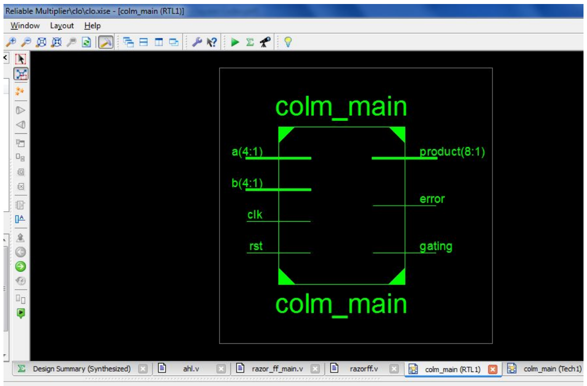
Fig. 6: AHL circuit

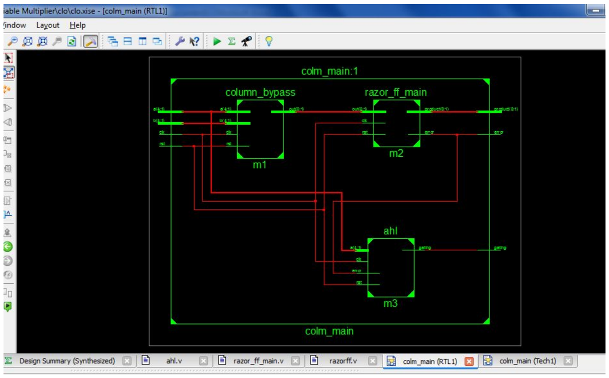
The details of the operation of the AHL circuit are as follows: when an input pattern arrives, both judging blocks will decide whether the pattern requires one cycle or two cycles to complete and pass both results to the multiplexer. The multiplexer selects one of either result based on the output of the aging indicator. Then an OR operation is performed between the result of the multiplexer, and the Q_bar signal is used to determine the input of the D flip-flop. When the pattern requires one cycle, the output of the multiplexer is 1. The !(gating) signal will become 1, and the input flip flops will latch new data in the next cycle. On the other hand, when the output of the multiplexer is 0, which means the input pattern requires two cycles to complete, the OR gate will produce output 0 to the D flip-flop. Therefore, the (gating) signal will be 0 to disable the clock signal of the input flip-flops in the next cycle. Note that only a cycle of the input flip-flop will be disabled because the D flip-flop will latch 1 in the next cycle.

IV. SIMULATION RESULTS

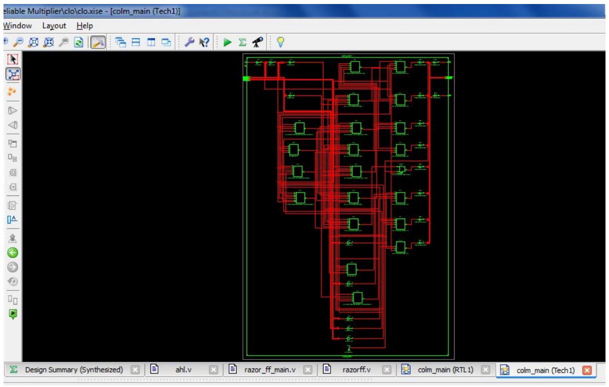
For Column-Bypassing Multiplier

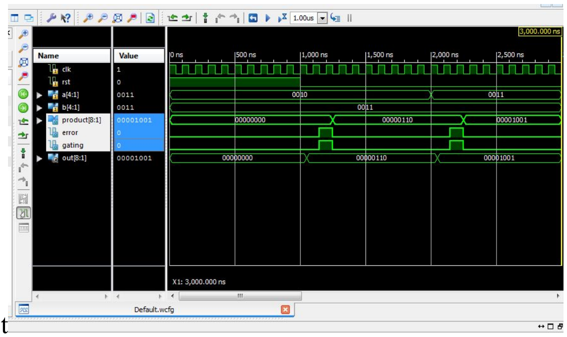
RTL Schematic diagram





Technology schematic



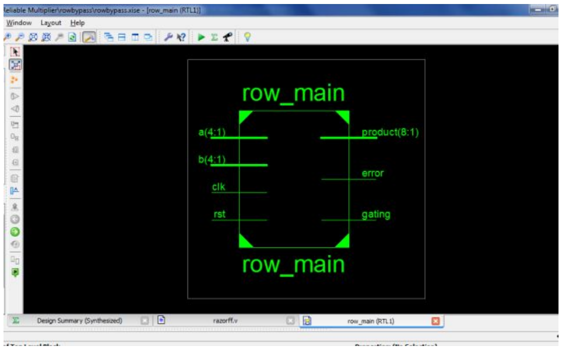


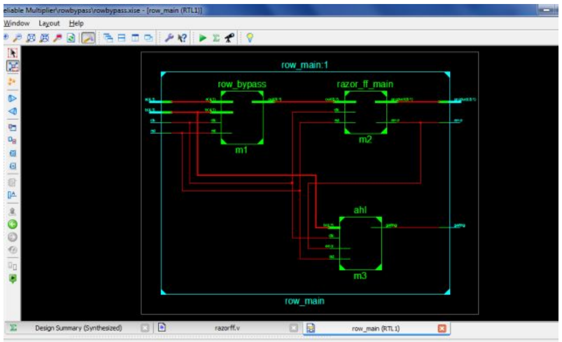
Name	Value
clk	1
rst	0
a[4:1]	0011
b[4:1]	0011
product[8:1]	00001001
error	0
gating	0
out[8:1]	00001001

Screenshot

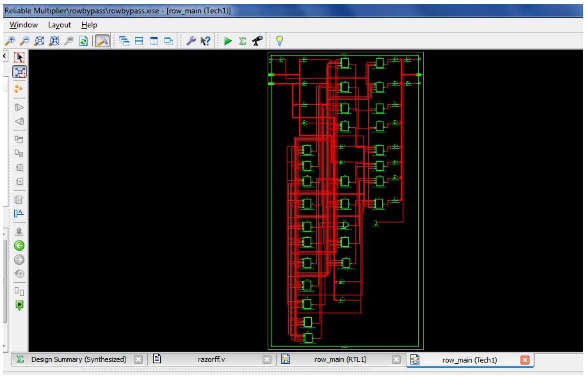
For Row-Bypassing Multiplier

RTL Schematic diagram

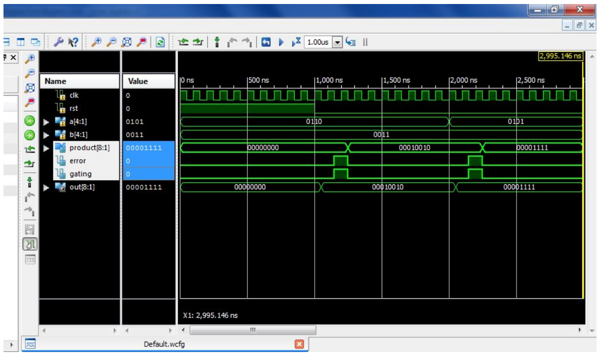




Technology schematic



Screenshot



clk 0
rst 0
a[4:1] 0101
b[4:1] 0011
product[8:1] 00001111
error 0
gating 0
out[8:1] 00001111

V. CONCLUSION

This paper proposed an aging-aware variable latency multiplier design with the AHL. The multiplier is able to adjust the AHL to mitigate performance degradation due to increased delay. Our proposed variable latency multipliers have less performance degradation because variable latency multipliers have less timing waste, but traditional multipliers need to consider the degradation caused by both the BTI effect and electro migration and use the worst case delay as the cycle period. The experimental results show that our proposed architecture with 4x4 multiplication with CLA as last stage instead of Normal RCA adder it will decrease the delay and improve the performance compared with previous designs.

REFERENCES

1. *Y. Cao. (2013). Predictive Technology Model (PTM) and NBTI Model [Online]. Available: http://www.eas.asu.edu/ ~ptm*
2. *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.*
3. *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.*
4. *H.-I. Yang, S.-C. Yang, W. Hwang, and C.-T. Chuang, "Impacts of NBTI/PBTI on timing control circuits and degradation tolerant design in nanoscale CMOS SRAM," IEEE Trans. Circuit Syst., vol. 58, no. 6, pp. 1239–1251, Jun. 2011.*
5. *R. Vattikonda, W. Wang, and Y. Cao, "Modeling and miimization of pMOS NBTI effect for robust naometer design," in Proc. ACM/IEEE DAC, Jun. 2004, pp. 1047–1052.*
6. *H. Abrishami, S. Hatami, B. Amelifard, and M. Pedram, "NBTI-aware flip-flop characterization and design," in Proc. 44th ACM GLSVLSI, 2008, pp. 29–34*

7. *S.V.Kumar,C.H.Kim,andS.S.Sapatnekar,"NBTI-awaresynthesis of digital circuits," in Proc. ACM/IEEE DAC, Jun. 2007, pp. 370–375.*

8. *A. Calimera, E. Macii, and M. Poncino, "Design techniqures for NBTI- tolerant power-gating architecture," IEEE Trans. Circuits Syst., Exp. Briefs, vol. 59, no. 4, pp. 249–253, Apr. 2012.*

9. *K.-C. Wu and D. Marculescu, "Joint logic restructuring and pin reordering against NBTI-induced performance degradation," in Proc. DATE, 2009, pp. 75–80.*

10. *Y. Lee and T. Kim, "A fine-grained technique of NBTI-aware voltage scaling and body biasing for standard cell based designs," in Proc. ASP-DAC, 2011, pp. 603–608.*