ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Aging-Aware Reliable Multiplier Design with Adaptive Hold Logic

J.Nitesh Kumar
Department of Electronics & Communication Engineering,
Hyderabad Institute of Technology & Management,
Hyderabad, Telangana-502 401, India.

K.Anil Kumar
Department of Electronics & Communication Engineering,
Hyderabad Institute of Technology & Management,
Hyderabad, Telangana-502 401, India.

Abstract:

Digital multipliers are among the most critical arithmetic functional units. The overall performance of these systems depends on the throughput of the multiplier. Meanwhile, 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 effects degrade transistor speed, and in the long term, the system may fail due to timing violations. Therefore, it is important to design reliable high-performance multipliers. In this paper, we propose an aging-aware multiplier design with a novel adaptive hold logic (AHL) circuit. The multiplier is able to provide higher throughput through the variable latency and can adjust the AHL circuit to mitigate performance degradation that is due to the aging effect. Moreover, the proposed architecture can be applied to a column- 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. The overall performance of these systems depends on the throughput of the multiplier. Meanwhile, 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 effects degrade transistor speed, and in the long term, the system may fail due to timing violations. Therefore, it is important to design reliable high-performance multipliers. In this paper, we propose an aging-aware multiplier design with a novel adaptive hold logic (AHL) circuit. The multiplier is able to provide higher throughput through the variable latency and can adjust the AHL circuit to mitigate performance degradation that is due to the aging effect. Moreover, the proposed architecture can be applied to a column- or row-bypassing multiplier.

The experimental results show that our proposed architecture with 16 ×16 and 32 ×32 column-bypassing multipliers can attain up to 62.88% and 76.28% performance improvement, respectively, compared with 16×16 and 32×32 fixed-latency column-bypassing multipliers. Furthermore, our proposed architecture with 16 × 16 and 32 × 32 row-bypassing multipliers can achieve up to 80.17% and 69.40% performance improvement as compared with 16×16 and 32 × 32 fixed-latency row-bypassing multipliers. 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 throughput of these applications depends on multipliers, and if the multipliers are too slow, the performance of entire circuits will be reduced.

Cite this article as: J.Nitesh Kumar & K.Anil Kumar, "Aging-Aware Reliable Multiplier Design with Adaptive Hold Logic", International Journal & Magazine of Engineering, Technology, Management and Research, Volume 5, Issue 1, 2018, Page 85-89.

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. 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. However, for high-k/metal-gate nMOS transistors with significant charge trapping, the PBTI effect can no longer be ignored. In fact, it has been shown that the PBTI effect is more significant than the NBTI effect on 32-nm high-k/metal-gate processes [1]–[3].

A traditional method to mitigate the aging effect is overdesign including such things as guard-banding and gate over sizing; however, this approach can be very pessimistic and area and power inefficient. To avoid this problem, many NBTI-aware methodologies have been proposed. An NBTI-aware technology mapping technique was guarantee the performance of the circuit during its lifetime. An NBTI-aware sleep transistor was designed to reduce the aging effects on pMOS sleep-transistors, and the lifetime stability of the power-gated circuits under consideration was improved. Wu and Marculescu [4] proposed a joint logic restructuring and pin reordering method, which is based on detecting functional symmetries and transistor stacking effects. They also proposed an NBTI optimization method that considered path sensitization. Dynamic voltage scaling and body-basing techniques were proposed to reduce power or extend circuit life. These techniques, however, require circuit modification or do not provide optimization of specific circuits.

The variable-latency design divides the circuit into two parts: 1) shorter paths and 2) longer paths. Shorter paths can execute correctly in one cycle, whereas longer paths need two cycles to execute. When shorter paths are activated frequently, the average latency of variable-latency designs is better than that of traditional designs. For example, several variable-latency adders were proposed using the speculation technique with error detection and recovery [5]–[7]. A short path activation function algorithm was proposed in [8] to improve the accuracy of the hold logic and to optimize the performance of the variable-latency circuit. An instruction scheduling algorithm was proposed in [17] to schedule the operations on nonuniform latency functional units and improve the performance of Very Long Instruction Word processors.

In, a variable-latency pipelined multiplier architecture with a Booth algorithm was proposed. In process-variation tolerant architecture for arithmetic units was proposed, where the effect of process-variation is considered to increase the circuit yield. In addition, the critical paths are divided into two shorter paths that could be unequal and the clock cycle is set to the delay of the longer one. These research designs were able to reduce the timing waste of traditional circuits to improve performance, but they did not consider the aging effect and could not adjust themselves during the runtime. A variable-latency adder design that considers the aging effect. However, no variable-latency multiplier design that considers the aging effect and can adjust dynamically has been done.

II. PROPOSED AGING-AWARE MULTIPLIER:

This section details the proposed aging-aware reliable multiplier design. It introduces the overall architecture and the functions of each component and also describes how to design AHL that adjusts the circuit when significant aging occurs. A. Proposed Architecture Fig. 8 shows our 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. In the proposed architecture, the column- and row-bypassing multipliers can be examined by the number of zeros in either the multiplicand or multiplicator to predict whether the operation requires one cycle or two cycles to complete. When input patterns are random, the number of zeros and ones in the multiplicator and multiplicand follows a normal distribution, as shown in Figs. 9 and 10. Therefore, using the number of zeros or ones as the judging criteria results in similar outcomes.

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.

1. Adaptive Hold Logic:

The AHL circuit is the key component in the aging-ware variable-latency multiplier. Fig. 12 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. Compared with the first judging block, the second judging block allows a smaller number of patterns to become one-cycle patterns because it requires more zeros in the multiplicand (multiplicator).

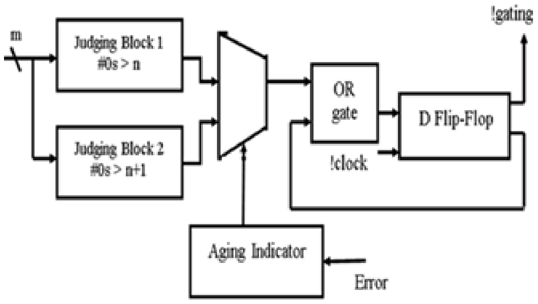

Fig.4: Diagram of AHL (md means multiplicand; mr means multiplicator)

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 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 output 0 to the D flip-flop.

2. Razor Flip-Flop:

Razor Flip-flop is a circuit-level timing speculation technique based on dynamic detection and correction of speed-path failures in digital designs. In Razor, input vectors are speculatively executed under the assumption that they would meet the setup and hold time requirements for a given clock cycle.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

A timing mis-speculation leads to a delay error which is detected by comparing the speculative execution output against worst-case assumptions. In such an event, suitable recovery mechanisms are engaged to achieve correct state. Thus, computational correctness in Razor is achieved not through worstcase safety margins but rather through in situ detection and recovery mechanisms in the presence of errors.

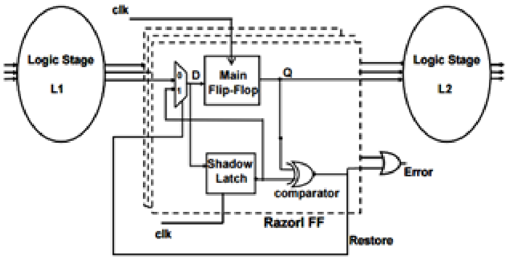

Fig.5: Abstract view of the RazorI flip-flop. The speculative data in the master-slave flip-flop is compared with the correct data in the positive level-sensitive shadow latch

3.Concept of Razor Error Detection and Recovery:

The RazorI flip- flop (henceforth referred to as the R1FF) is constructed out of a standard positive edge-triggered D Flip-Flop (DFF), augmented with a shadow latch which samples at the negative clock edge. Thus, the input data is given additional time, equal to the duration of the positive clock phase, to settle down to its correct state before being sampled by the shadow latch. In order to ensure that the shadow latch always captures the correct data, the minimum allowable supply voltage needs to be constrained during design time such that the setup time at the shadow latch is never violated, even under worst-case conditions. A comparator flags a timing error when it detects a discrepancy between the speculative data sampled at the main flip-flop and the correct data sampled at the shadow latch. Error signals of individual R1FFs are OR-ed together to generate the pipeline restore signal which overwrites the shadow latch data into the main flip-flop, thereby restoring correct state in the cycle following the erroneous cycle.

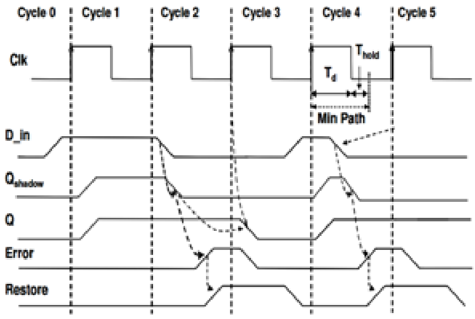

Fig. 6: Conceptual timing diagrams showing the operation of the RazorI flip- flop. In Cycle 2, a setup violation causes Error to be flagged whereas in Cycle 4, a hold violation causes error to be asserted

Thus, both the main flip-flop and the shadow latch will latch the correct data. In this case, the error signal at the output of the comparator remains low and the operation of the pipeline is unaltered. In cycle 1, we show an example of the operation when the combinational logic exceeds the intended delay due to sub-critical voltage scaling. In this case, the data is not latched correctly by the main flip-flop, but since the shadow-latch samples at the negative edge of the clock, it successfully latches the data half-way through cycle 2.

By comparing the valid data of the shadow latch with the data in the main flip-flop, an error signal is then generated in cycle 2. Error signals of individual R1FFs are OR-ed together to generate the pipeline restore signal which overwrites the shadow latch data into the main flip-flop, thereby restoring correct state at the positive edge of the subsequent cycle, cycle 4. Using suitable clock chopping techniques, the duration of the positive phase of the propagated clock can be configured as required so as to exploit the above trade-off.

IV. RESULTS:

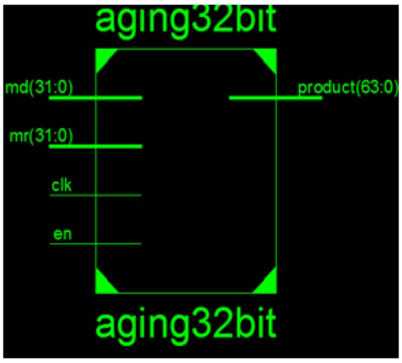

Fig.7: RTL Schematic top module

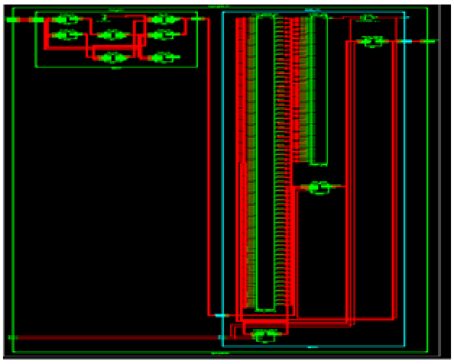

Fig.8: RTL Schematic detailed architecture for proposed system

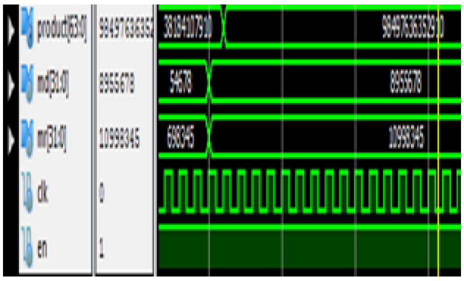

Fig.9: Simulation Results for proposed system

V. CONCLUSION:

The proposed approach is 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. The experimental results show that our proposed architecture with 16×16 and 32×32 column-bypassing multipliers can attain up to 62.88% and 76.28% performance improvement compared with the 16 × 16 and 32 × 32 FLCB multipliers, respectively. Furthermore, our proposed architecture with the 16×16 and 32×32 row-bypassing multipliers can achieve up to 80.17% and 69.40% performance improvement compared with the 16 × 16 and 32 × 32 FLRB multipliers.

REFERENCES:

[1] M. Basoglu, M. Orshansky, and M. Erez, " NBTI-aware DVFS: A newapproach to saving energy and increasing processor lifetime," in Proc.ACM/IEEE ISLPED, Aug. 2010, pp. 253– 258.

[2] K.-C. Wu and D. Marculescu, " Aging-aware timing analysis and optimizationconsidering path sensitization," in Proc. DATE, 2011, pp. 1– 6.

[3] S. V. Kumar, C. H. Kim, and S. S. Sapatnekar, "NBTI-aware synthesis of digital circuits," in Proc. ACM/IEEE DAC, Jun. 2007, pp. 370–375.

[4] 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.

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

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

[7] K. Du, P. Varman, and K. Mohanram, " High performance reliablevariable latency carry select addition," in Proc. DATE, 2012,pp. 1257– 1262.

[8] 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.