IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 2, February 2017

Design of Reliable Multiplier with an Adaptive Hold Logic

Pradnya Pagar[1] Prof. M. Vanjale[2]

Student, E & TC Dept, AISSMS Institute of Information Technology, Pune, India [1]

Asst. Professor, E & TC Dept, AISSMS Institute of Information Technology, Pune, India [2]

Abstract: Digital multipliers are one of the most critical arithmetic functional units. The overall performance of this system depends on the throughput of the multiplier. Meanwhile, the NBTI effect occurs when PMOS transistor is under negative bias (Vgs = - Vdd), increasing the threshold voltage of the PMOS transistor and reducing transistor speed. A similar phenomenon, positive bias temperature instability occurs when a PMOS transistor is under positive bias. Both temperature effects degrade transistor speed, and in the long term, the system may fail due to timing violations. Thus, it is important to design reliable high performance multipliers. In this paper, we propose an aging-aware multiplier design with the novel adaptive hold logic (AHL) circuit. The multiplier is able to improve higher throughput through the variable latency and can adjust the AHL circuits to mitigate performance degradation that is due to the aging effect. The proposed design can be allied to the column bypass 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 Fourier transform, discrete cosine transforms, and digital filtering. The throughput of this application depends on multipliers, if the multipliers are too slow, the performance of the entire circuits will be reduced. Furthermore, negative bias temperature instability (NBTI) occurs when the PMOS transistor is under negative bias (Vgs = - Vdd). In this situation, the instruction between inversion layer holes and hydrogen-passivated Si atoms breaks the Si-H bond generated during the oxidation processes, generating H or H2 molecules. When these molecules diffuse away, interface traps are left. The accumulated interface traps between silicon and gate oxide interface results in increased threshold voltage (VTh), reducing the circuit switching speed. When the biased voltage is removed, the reverse reaction occurs, reducing the NBTI effect. During the stress phase, the Vth increased in the long term. Hence it is important to design a reliable high-performance multiplier. The corresponding effect of the NMOS transistor is positive bias temperature instability (PBTI), which occurs when the 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. 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 into guarantee the performance of the circuit during its lifetime. In an NBTI-aware sleep transistor was designed to reduce the aging effect on PMOS sleep transistor, and the lifetime stability of the power-gated circuits under consideration was improved. Wu and Marculescu proposed a point logic restructuring and pin recording method, which is biased on detecting functional symmetries and transistor staking effects. They also proposed an NBTI optimization method that considered path sensitization. In dynamic voltage scaling and body biasing techniques were proposed to reduce power or extend circuit life. These techniques, however, require circuit modification or do not provide optimization of specific circuits. Traditional circuits use critical clock delay as overall circuit clock cycle in order to perform correctly. However, the probability that the critical paths are activated is low. In most cases, the path delay is shorter than the critical path. For these noncritical paths, using the critical path delay, as the overall clock period will result in significant timing waste. Hence, the variable latency design was proposed to reduce the timing west of traditional Circuits.

The variable-latency designs divide the circuit in two parts:
1) shorter paths and 2) longer paths. Shorted path can execute correctly in one cycle, whereas longer path need two cycle to execute. When shorter path is activated frequently, the average latency of variable-latency design is better than that the traditional designs. For example,

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 2, February 2017

several variable-latency adders were proposed using the speculation technique with error detection and recovery. A shorter path activation function algorithm was proposed into improve the accuracy of the hold logic and to optimize the performance of the variable-latency circuit. An instruction scheduling algorithm was proposed into schedule the operations on non-uniform latency functional unites and improves the performance of very long instruction word processor. In variable-latency pipelined multiplier architecture with booth algorithm was proposed, where the effect of process-variation is considered to increase the circuit yield. In addition, the critical paths are divided into two smaller paths that could be unequal and the clock cycle is set to the delay of longer one. These research designs were able to reduce the timing west 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 consider the aging effect was proposed. However, no variable-latency multiplier design that consider the aging effect and can adjust dynamically has been done.

This paper has been organized in the following way; we propose an aging-aware reliable multiplier design with novel adaptive hold logic (AHL) circuit. The multiplier is based on the variable-latency technique and can adjust the AHL circuit to achieve reliable operation under the influence of NBTI and PBTI effect. To be specific, the contributions of this paper are summarized as follows:

1) Novel variable latency multiplier architecture with an AHL circuits. The AHL circuits can decide whether the input patterns require one or two cycles and can adjust the judging criteria to ensure that there is minimum performance degradation after considerable aging occurs;
2)comprehensive analysis and comparison of the multiplier's performance under different cycle periods to show the effectiveness of our proposed architecture;
3) An aging-aware reliable multiplier design method that is suitable for large multipliers. Although the experiment is performed in 16- and 32-bit multipliers. Our proposed architecture can be easily extended to large designs;
4) The experimental result shows that our proposed architecture with 16 x 16 and 32 x 32 column bypassing multiplier can attain up to 62.88% and 76.28% performance improvement compared with the 16 x 16 and 32 x 32 fixed-latency column-bypassing (FLCB) multipliers. In addition, our proposed architecture with 16 x 16 and 32 x 32 row by passing multipliers can achieve up to 80.17% and 69.40% performance improvement as compared with 16 x 16 and 32 x 32 fixed-latency row-bypassing multipliers.

II. LITERATURE SURVEY

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 [2]–[4].

A traditional method to mitigate the aging effect is overdesign [5], [6], including such things as guard-banding and gate oversizing; 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 proposed in [7] to guarantee the performance of the circuit during its lifetime.

In [8], 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.

Author Wu and Marculescu [9] 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 [12].

In Paper [10] and [11], author, 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.

III. RELATED WORK

Traditional circuits use critical path delay as the overall circuit clock cycle in order to perform correctly. However, the probability that the critical paths are activated is low. In most cases, the path delay is shorter than the critical path. For these noncritical paths, using the critical path delay as the overall cycle period will result in significant timing waste. Hence, the variable-latency design was proposed to reduce the timing waste of traditional 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.

A. 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. 1. 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. a low-power column bypassing multiplier design is proposed in which the FA operations are disabled if the corresponding bit in

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 2, February 2017

the multiplicand is 0. Fig. 2 shows a 4×4 column-bypassing multiplier. Supposing the inputs are 10102* 11112, it can be seen that for the FAs in the first and third diagonals, two of the three input bits are 0: the carry bit from its upper right FA and the partial product aibi. Therefore, the output of the adders in both diagonals is 0, and the output sum bit is simply equal to the third bit, which is the sum output of its upper FA.

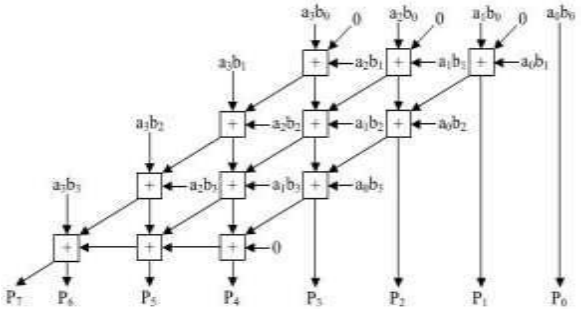

Fig. 1. 4 x 4 normal AM

Hence, the FA is modified to add two tristate gates and one multiplexer. The multiplicand bit ai 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 tristate 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.

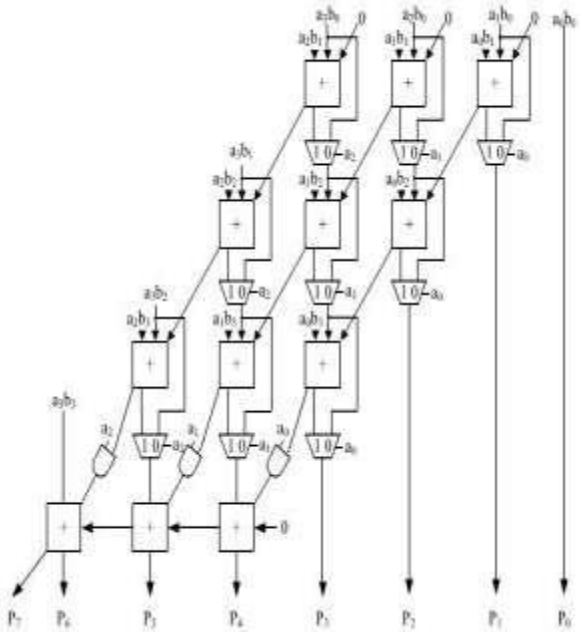

Fig. 2. 4 x 4 Column bypassing Multiplier

B. .Row-Bypassing Multiplier

A low-power row-bypassing multiplier is also proposed to reduce the activity power of the AM. 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 tristate gates use the multiplication. Fig. 3 is a 4×4 rowbypassing multiplier. Each input is connected to an FA through a tristate gate. When the inputs are 11112 * 10012, the two inputs in the first and second rows are 0 for FAs. Because b1 is 0, the multiplexers in the first row select aib0 as the sum bit and select 0 as the carry bit. The inputs are bypassed to FAs in the second rows, and the tristate gates turn off the input paths to the FAs. Therefore, no switching activities occur in the first-row FAs; in return, power consumption is reduced. Similarly, becauseb2 is 0, no switching activities will occur in the second-row FAs. However, the FAs must be active in the third row because the b3 is not zero.

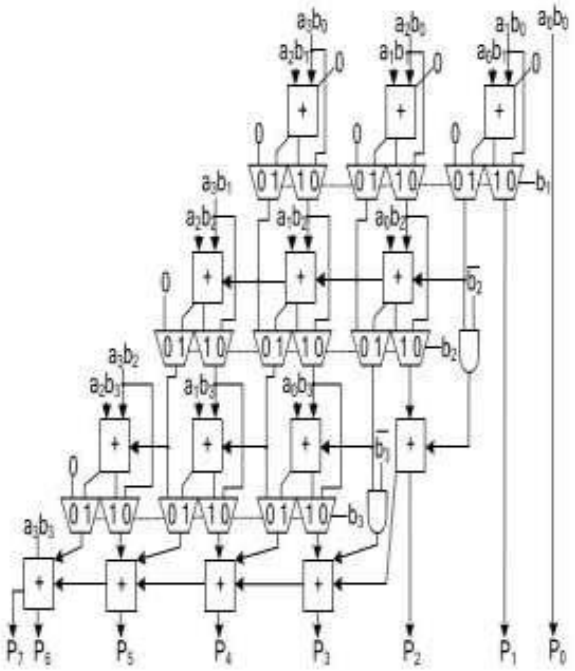

Fig. 3. 4 x 4 Row bypassing multiplier

IV. PROPOSED SYSTEM

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. 4 shows our proposed aging-aware multiplier architecture, which includes two m-bit inputs (miss a positive number), one 2m-bit output, one column- or row by passing multiplier, 2m1-bit Razor flip-flops, and an AHL circuit. The inputs of the row-bypassing multiplier are the symbols in the parentheses. 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.

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 2, February 2017

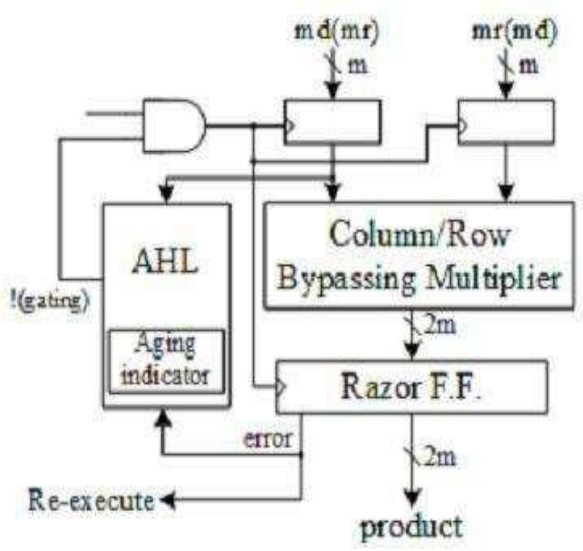

Fig. 4. Proposed Architecture

Hence, 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 that of the rowbypassing multiplier is the multiplicator. Razor flip-flops can be used to detect whether timing violations occur before the next input pattern arrives.

Fig. 6 shows the details of Razor flip-flops. A 1- bit Razor flip-flop contains a main flip-flop, shadow latch, XOR gate, and mux. 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.

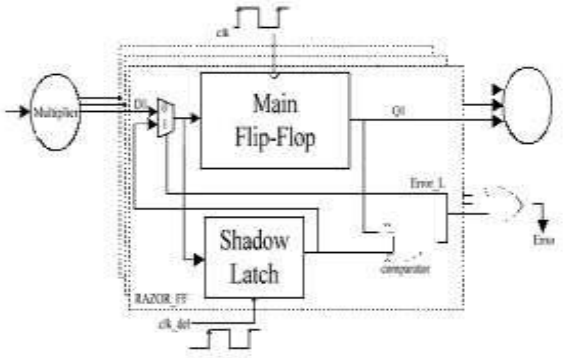

Fig. 5. Razor Flip Flops

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.

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 output signal 1; otherwise, it will output 0 to indicate the aging effect is still not significant, and no actions are needed.

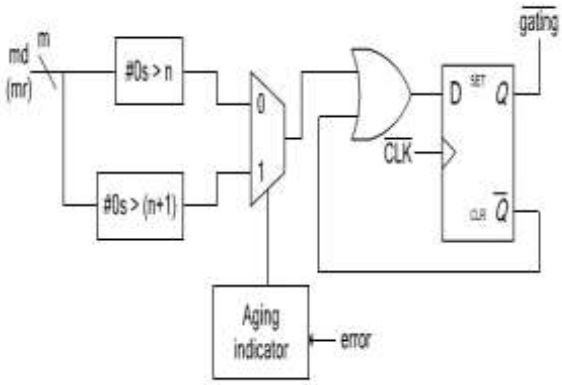

Fig. : 6. Diagram of AHL

The first judging block in the AHL circuit will output 1 if the number of zeros in the multiplicand (multiplication for the row-bypassing multiplier) is larger than n, and the second judging block in the AHL circuit will output 1 if the number of zeros in the multiplicand (multiplicator) is larger than n+1. They are both employed to decide whether an input pattern requires one or two cycles, but only one of them will be chosen at a time. In the beginning, the aging effect is not significant, and the aging indicator produces 0, so the first judging block is used. After a period of time when the aging effect becomes significant, the second judging block is chosen. 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). The details of the operation of the AHL circuit are as follows: when an input pattern arrives, both

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 2, February 2017

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 $\bar{}$ 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. 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.

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] 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.
[2] 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.
[3] 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.
[4] R. Vattikonda, W. Wang, and Y. Cao, "Modeling and minimization of pMOS NBTI effect for robust nanometre design," in Proc. ACM/IEEE DAC, Jun. 2004, pp. 1047–1052.
[5] 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.
[6] 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 circuit," IEEE Trans. Computer.-Aided Des. Integr. Circuits Syst., vol. 26, no. 4, pp. 743–751, Apr. 2007.
[7] 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. 104