International Journal of Research
Available at https://edupediapublications.org/journals
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 03 Issue 17
November 2016

High Speed Reliable Multiplier Design with Adaptive Hold Logic

Cheerla Indraneel
PG Scholar
E-mail:cheerlaindraneel@gmail.com
Department of ECE (VLSI)
Quba College of engineering and technology

K.Rajyalakshmi
Assistant professor (M.tech)
E-mail:kottamasurajyalakshmi@gmail.com
Department of ECE
Quba College of engineering and technology

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 ($V_{gs} = -V_{dd}$), 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 column 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 but here we newly implemented column multiplier in place of bypass multipliers.

Index Terms—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. 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-nmhigh-k/metal-gate processes.

A traditional method to mitigate the aging effect is overdesign, 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 to guarantee the performance of the circuit during its lifetime. In 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 proposed a point 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. In 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.

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. For example, several variable-latency adders were proposed using the speculation technique with error detection and recovery. A short path activation function algorithm was proposed in 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 to schedule the operations on non-uniform latency functional units and improve the performance of Very Long Instruction Word processors. In a variable-latency pipelined multiplier architecture with a Column 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 was proposed.

However, no variable-latency multiplier design that considers the aging effect and can adjust dynamically has been done.

II.PAPER CONTRIBUTION

In this paper, 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 effects. To be specific, the contributions of this paper are summarized as follows: 1) novel variable-latency multiplier architecture with an AHL circuit. The AHL circuit 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 results show that our proposed architecture with the 16 × 16and 32 × 32 column-bypassing multipliers can attain up to 62.88% and 76.28% performance improvement compared with the 16 × 16 and 32 × 32 fixed-latency column-bypassing (FLCB) multipliers. In addition, 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.

III.PRELIMINARIES

Row-Bypassing Multiplier

A low-power row-bypassing multiplier [23] 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

multiplicator.

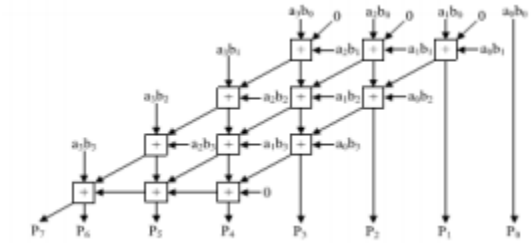

Fig. 1. 4 × 4 normal AM.

Fig. 2 is a 4 ×4 row-bypassing multiplier. Each input is connected to an FA through a tristate gate. When the inputs are 1111 * 1001, 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, becauseb2is 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.

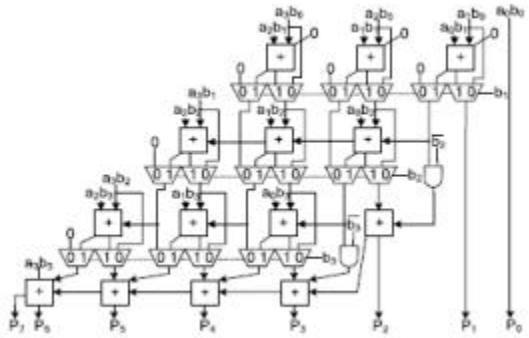

Fig. 2. 4X4 Row by passing multiplier

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. More details for the column-bypassing multiplier can be found.

Column Bypassing Multiplier:

For the better improvement, the multiplier can be design which produces less partial products. And one such multiplier is Column Multiplier (CM); the three bits are scanned at a time to reduce the number of partial products. The Algorithm is shown in *Figure 2.1* several steps of multiplication are performed at once to speed up the operation of Column's encoding. The addition (or) subtraction operation can be skipped if the consecutive bits in the multiplicand are same. In many cases the delay associated with Column Multiplication is less than the Array multiplier. Anyhow, the delay of Column multiplier is data dependant. It reduces the numbers of adders and therefore the delay required to generate the partial sums are obtain by examining three bits at a time. However the high performance of Column Multiplier gives more power consumption.

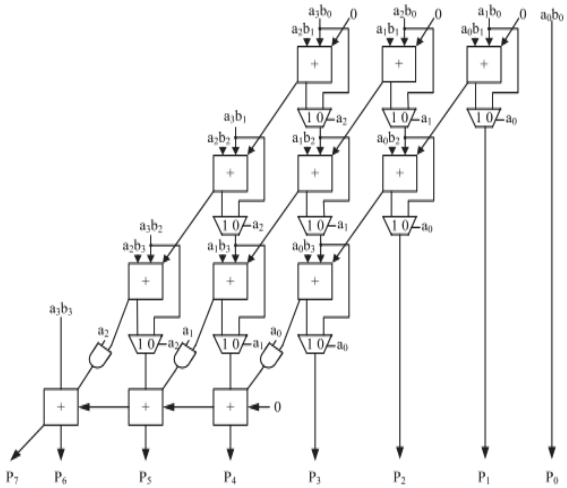

Fig 2.1 column by pass multiplier

IV.PROPOSED AGING AWARE MULTIPLIER

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

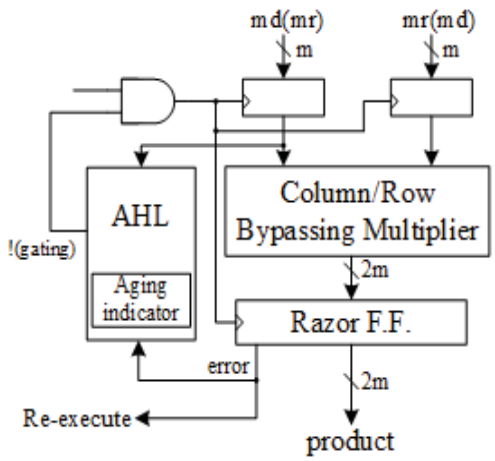

Fig 3.Proposed architecture (md means multiplicand; mr means multiplicator).

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. Newly we introduced column multiplier in place of bypass multipliers.

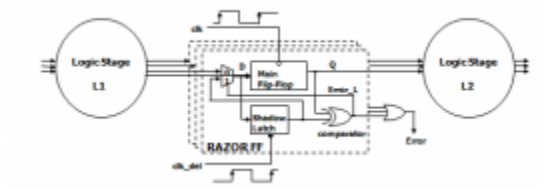

Fig 4.Razor flip flop

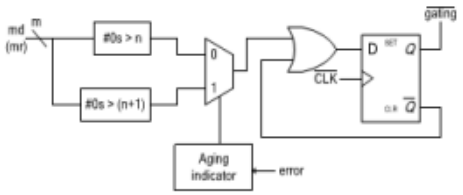

Fig 5.Diagram of AHL (md means multiplicand; mr means multiplicator).

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. 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 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. More details for the Razor flip-flop can be found. The AHL circuit is the key component in the aging-ware variable-latency multiplier. Fig5. 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 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.

The first judging block in the AHL circuit will output 1 if the number of zeros in the multiplicand is larger than n (n is a positive number, which will be discussed in SectionIV), and the second judging block in the AHL circuit will output 1 if the number of zeros in the multiplicand 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 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 is1. 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, that means the input pattern requires two cycles to complete, the OR gate will output 0to 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. The overall flow of our proposed architecture is as follows: when input patterns arrive, the column multiplier,and the AHL circuit execute simultaneously. According to the number of zeros in the multiplicand, the AHL circuit decides if the input patterns require one or two cycles. If the input pattern requires two cycles to complete, the AHL will output 0 to disable the clock signal of the flip-flops. Otherwise, the AHL will output 1 for normal operations. When the column multiplier finishes the operation, the result will be passed to the Razor flip-flops. The Razor flip-flops check whether there is the path delay timing violation. If timing violations occur, it means the cycle period is not long enough for the current operation to complete and that the execution result of the multiplier is incorrect. Thus, the Razor flip-flops will output an error to inform the system that the current operation needs to be reexecuted using two cycles to ensure the operation is correct. In this situation, the extra reexecution cycles caused by timing violation incurs a penalty to overall average latency. However, our proposed AHL circuit can accurately predict whether the input patterns require one or two cycles in most cases. Only a few input patterns may cause a timing variation when the AHL circuit judges incorrectly. In this case, the extra reexecution cycles did not produce significant timing degradation. In summary, our proposed multiplier design has three key features. First, it is a variable-latency design that minimizes the timing waste of the noncritical paths. Second, it can provide reliable operations even after the aging effect occurs. The Razor flip-flopsdetect the timing violations and reexecute the operations using two cycles. Finally, our architecture can adjust the percentage of one-cycle patterns to minimize performance degradation due to the aging effect. When the circuit is aged, and many errors occur, the AHL circuit uses the second judging block to decide if an input is one cycle or two cycles.

V.SIMULATION RESULTS

Fig 6. Simulation output waveform

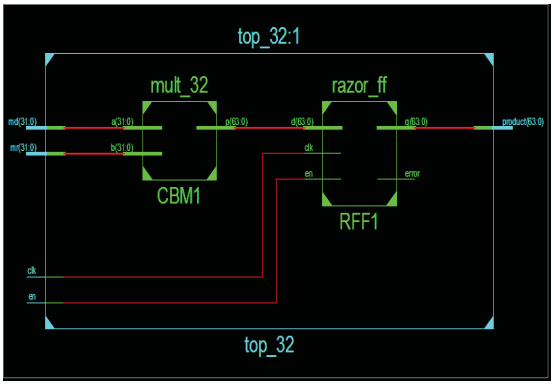

Fig 7.Block diagram

	32X32column_by passingm	16x16column-	8x8column-
D	8	43.589ns	24.957ns
Numberof	3	9	2
Numberof 4-	3	8	2
Numberof Bonded IOB's	5 6 %	2 7 %	1 3 %

Fig 8: Performance Table

VI.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. The experimental results show that our proposed architecture with 4x4 multiplication with Column 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 designin 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/IEEEDAC, 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, and S. S. Sapatnekar, "NBTI-aware synthesis of digital circuits," in Proc. ACM/IEEE DAC, Jun. 2007, pp. 370–375.
[8] 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.
[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. ASPDAC,2011, pp. 603–608.

Student details:

Mr.CHEERLA INDRANEEL was born in India.He is pursuing M.Tech degree in VLSI in ECE Department in Quba College Of Engineering and Technology, SPSR Nellore District, Andhra Pradesh State ,India.

Mail id: cheerlaindraneel@gmail.com

Guide details:

Mrs.K RAJYALAKSHMI (M.tech) was born in India. She is working as a Assistant Professor in Department of ECE at Quba College Of Engineering and Technology, SPSR Nellore District, Andhra Pradesh State ,India.

Email id: kottamasurajyalakshmi@gmail.com