International Journal of Research
Available at https://edupediapublications.org/journals
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 04 Issue 05
April 2017

A Novel Design Of Reliable Multiplier Using Adaptive Hold Logic

Suvarnaganti Rajasekhar, PG Scholar , Dept. Of ECE ,
Email-Id: sekhar0454@gmail.com
Gaddam Sekhar Reddy, M.Tech [Asst. Professor Dept. Of ECE]
Email-Id: gaddamsekharreddy@gmail.com
Sannikanti Kishore Babu , M.Tech ,[Ph.D] ,[Associate Professor & HOD Dept. Of ECE] , Email-Id: kish.fr@gmail.com
Vikas Group Of Institutions , Nunna (v) , Vijayawada(M) -521212, Krishna Dt.

Abstract : Digital multipliers are the most critical arithmetic functional units. The overall system performance depends upon the throughput of this multiplier. The positive bias temperature instability, occurs when an nmos transistor is under positive bias. The negative bias temperature instability effect occurs when a pmos transistor is under negative bias , increasing the threshold voltage of the pmos transistor, and reducing multiplier speed. 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.

Key Terms:- Adaptive Hold Logic(AHL), 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 performance of the systems depends upon multipliers, and if the multipliers are too slow, the throughput of entire circuits will be reduced.

A conventional method to overcome the aging effect is overdesign [5], [6], including the things such as guard-banding and gate oversizing; however, this approach can be very pessimistic and efficient in terms of power. To overcome this effect, 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.

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

International Journal of Research
Available at https://edupediapublications.org/journals
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 04 Issue 05
April 2017

traditional designs. For example, several variable-latency adders were proposed using the speculation technique with error detection and recovery [4]–[5].

II. LITERATURE SURVEY

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.

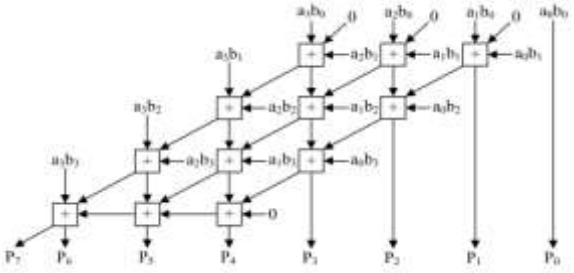

Fig 1 : A Normal Multiplier

In [8], 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. 2 shows a 4× 4 column-bypassing multiplier. Supposing the inputs are 0110_2 * 1001_2, 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.

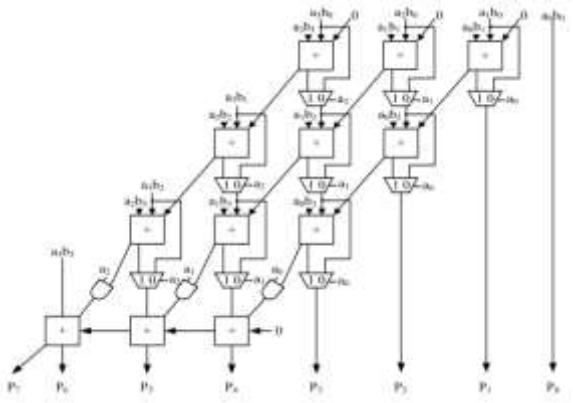

Fig 2 : Column – bypass Mutiplier.

B. Row-Bypassing Multiplier

A low-power row-bypassing multiplier [7] 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 multiplier. Fig. 3 is a 4 × 4 row-bypassing multiplier.

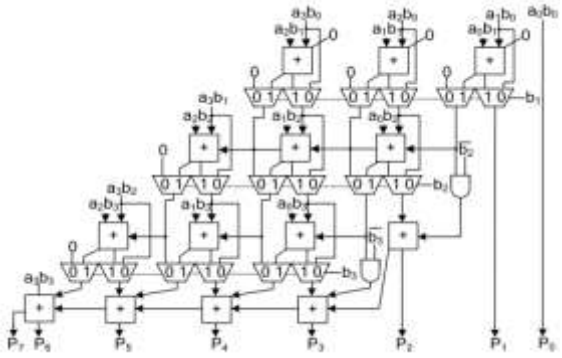

Fig 3 : Row – Bypass Multiplier.

Each input is connected to an FA through a tristate gate. When the inputs are 0111_2 * 1011_2, 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.

International Journal of Research
Available at https://edupediapublications.org/journals
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 04 Issue 05
April 2017

C. Variable-Latency Design

The variable-latency design was proposed to reduce the timing waste occurring in traditional circuits that use the critical path cycle as an execution cycle period. The basic concept is to execute a shorter path using a shorter cycle and longer path using two cycles. Since most paths execute in a cycle period that is much smaller than the critical path delay, the variable-latency design has smaller average latency.

For example, Fig. 4 is an 8-bit variable-latency ripple carry adder (RCA). A_8–A_1, B_8–B_1 are 8-bit inputs, and S_8–S_1 are the outputs. Supposing the delay for each FA is one, and the maximum delay for the adder is 8. Through simulation, it can be determined that the possibility of the carry propagation delay being longer than 5 is low. Hence, the cycle period is set to 5, and hold logic is added to notify the system whether the adder can complete the operation within a cycle period.

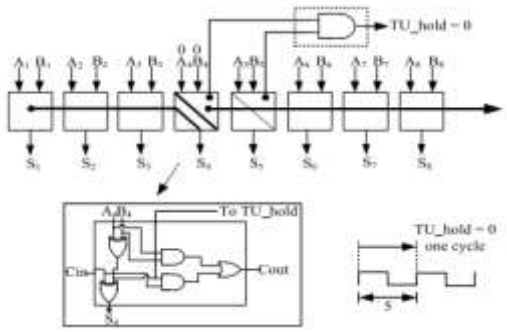

Fig 4: Hold Logic Circuit using RCA.

D. Aging Model

The NBTI (PBTI) effect occurs when a pMOS (nMOS) transistor is under negative (positive) bias voltage, resulting in V_{th} drift. When the bias voltage is removed, the recovery process occurs, reducing the V_{th} drift.

If a pMOS (nMOS) transistor is under constant stress, this is referred to as static NBTI (PBTI). If both stress and recovery phases exist, it is referred to as dynamic NBTI (PBTI). The V_{th} drift of pMOS (nMOS) transistor due to the static NBTI (PBTI) effect can be described by dc reaction-diffusion (RD) framework.

III. 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. 5 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 [7], and an AHL circuit.

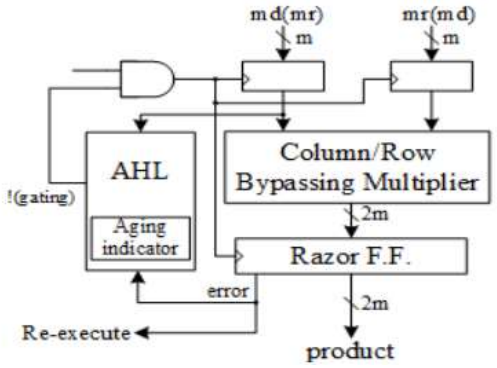

Fig 5: Proposed Multiplier architecture Using AHL Logic (md means multiplicand; mr means multiplier).

In the proposed architecture, the column- and row-bypassing multipliers can be examined by the number of zeros in either the

multiplicand or multiplier to predict whether the operation requires one cycle or two cycles to complete.

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 row-bypassing multiplier is the multiplier.

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 based flip-flop having a main flip-flop, shadow latch, XOR gate, and multiplexer. 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.

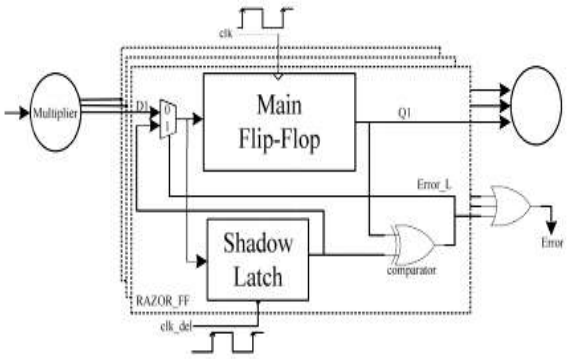

Fig 6 : Razor Flip – flop.

The AHL circuit is the key component in the aging-ware variable-latency multiplier. Fig. 7 shows the details of the AHL circuit. The AHL circuit contains an aging indicator, two judging blocks, one multiplexer, 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.

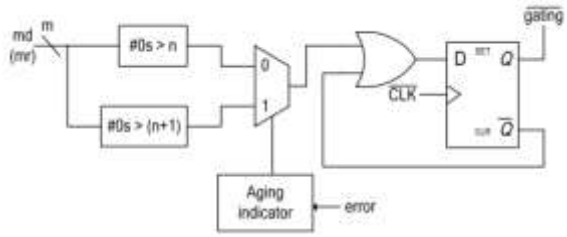

Fig 7: diagram of AHL (md means multiplicand; mr means multiplier).

The first judging block in the AHL circuit will output 1 if the number of zeros in the multiplicand (multiplier for the row-bypassing multiplier) is larger than n (n is a positive number, which will be discussed in Section IV),

and the second judging block in the AHL circuit will output 1 if the number of zeros in the multiplicand (multiplier) 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.

The implementation of AHL circuit is 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. 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.

The operation of our Proposed architecture is as follows : when input patterns arrive, the column- or row-bypassing multiplier, and the AHL circuit execute simultaneously. According to the number of zeros in the multiplicand (multiplier), 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- or row-bypassing multiplier finishes the operation, the result will be passed to the Razor flip-flops.

The extra re-execution 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 re-execution cycles did not produce significant timing degradation. In summary, our proposed multiplier design has two 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-flops detect the timing violations and re-execute the operations using two cycles.

The RTL schematic of novel design of reliable multiplier using AHL is shown in fig 8.

The Technology schematic of novel design of reliable multiplier using AHL is shown in fig 9.

The simulation output waveform of 16x16 column bypass multiplier using AHL is shown in fig 10.

IV. SIMULATION RESULTS

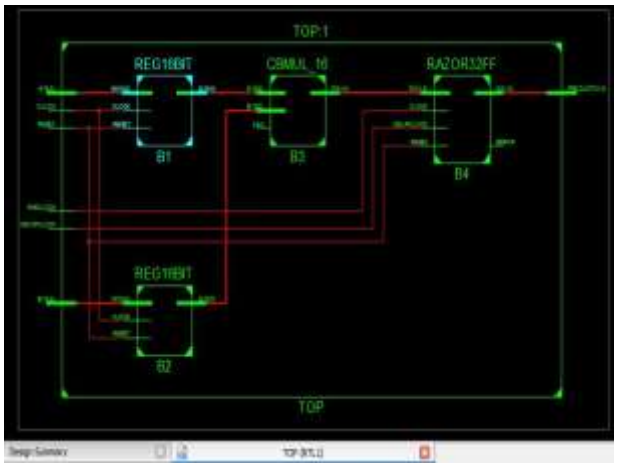

Fig 8: RTL schematic.

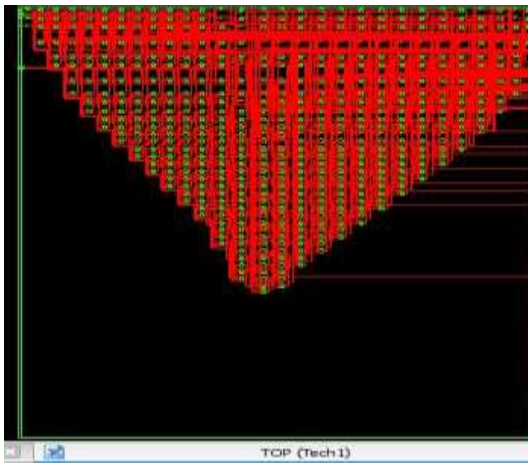

Fig 9: Technology schematic.


Fig 10: Simulation output waveform of 16x16column bypass multiplier using AHL.

V.CONCLUSION

This paper proposed a novel design of reliable multiplier design using AHL. The multiplier is able to adjust the AHL circuit to mitigate performance degradation due to increase delay. The experimental results show that our proposed architecture with 8x8 and 16×16 column-bypassing multipliers can attain better performance improvement. Furthermore, our proposed architecture with the 8x8 and 16×16 row-bypassing multipliers can achieve better performance improvement. In addition,the variable-latency bypassing multipliers exhibited the lowest average EDP and achieved. Aging effect is reduced using AHL circuit. Performance improvement is achieved due to the latency design. Overall delay is reduced in the multiplier design caused due to aging.

The BTI effect that increases transistor delay, interconnect also has its aging issue, which is called electro migration. Electro migration occurs when the current density is high enough to cause the drift of metal ions along the direction of electron flow. The metal atoms will be gradually displaced after a period of time, and the geometry of the wires will change.

Extension Work:-

As an Extension for this we are implementing the 16x16 Aging aware multiplier with AHL logic.

Future scope:-

In future the same technique can be applied to 32,64,128 bit multipliers and adders. We can try to implement the same ideas in our future work using backend tools like cadence or tanner EDA tools.

V.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, 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