



Realization of Aging Aware Reliable Multiplier Design Using Verilog

Vijayalaxmi Melamatti¹, B.B.Tigadi²

PG Student, Department of DECS, Maratha Mandal Engineering College Belgaum, Karnataka, India¹

Assistant Prof, Department of ECE, Maratha Mandal Engineering College Belgaum, Karnataka, India²

ABSTRACT: High speed and low power consumption is one of the most important design objectives in integrated circuits. As multipliers are widely used in such circuits, the multipliers must be designed efficiently. In this paper, we propose the novel adaptive hold logic circuit, which is simple and efficient approach to reduce the power consumption and delay. The proposed architecture is applied to column bypassing multiplier. The result analysis shows that the reliable multiplier has better performance in power consumption and delay.

KEYWORDS: Adaptive Hold Logic, Razor flip-flop, Reliable multiplier.

I. INTRODUCTION

The digital multipliers are most vital arithmetic functional units in various applications, like Fourier transform, Discrete cosine transform, Microprocessor and Digital filters etc. The throughput of these applications is relies on upon the multipliers. So if the execution of multipliers is moderate then the execution of complete circuit will be abatement. When nMOS transistor is under negative bias ,then negative bias temperature instability (NBTI) takes place. Similarly when pMOS transistor is under positive bias, positive bias temperature instability (PBTI) takes place. As a result, at the time of oxidation process the dissociation of Si-H bond was generated, which produce the H or H₂ molecules. When these molecules are diffused away, interface traps are left between silicon and gate oxide. Due to these interface traps increased threshold voltage takes place, which reduce the circuit switching speed.

When bias voltage is detached, reverse reaction occurs, which reduce the NBTI (PBTI) effect. But the reverse reaction does not able to remove all the interface traps, which significantly increase the threshold voltage. For this reason, it is essential to design a high performance multiplier circuit.

A typical way to reduce the aging effects is overdesigning, which includes the techniques like guard banding and gate over sizing. However this method has area and power inefficient and also may expensive in [1]. To overcome from this problem, an NBTI aware technology mapping technique was proposed in [2], which assurance the performance of the circuit throughout its lifetime. Another technique was an NBTI aware sleep transistor in [3], where it improves the lifetime constancy of the power gated circuits under considerations. Dynamic voltage scaling and body biasing methods were proposed in [4] and [5] to decrease the power or increase the circuit life. These methods required circuit alteration or do not provide optimization of exact circuits.

Every gate in any VLSI circuit has its own delay which reduces the performance of the chip. Conventional circuits use essential path delay as the overall circuit clock cycle in order to perform properly. However in many worst case designs, the probability that the critical path delay is activated is less. In such cases, the strategy of minimizing the worst case situations may lead to inefficient designs. For non critical path, using the critical path delay as the overall cycle period will result in considerable timing waste. Hence the variable latency design was proposed in [8] to reduce the timing waste of conventional circuits. The variable latency design divides the circuit into two parts: 1) Shorter path and 2) Longer path.

Shorter path require one cycle to execute properly, where as longer path takes two cycles to execute appropriately. That is, for a given timing constraint T, a path is a long path if the delay of the path is longer than or equal to T. Otherwise a path is a short path. Also a gate is a critical gate if the gate is in a long path. The basic conception of

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 5, May 2016

variable latency design is to execute a shorter path using one cycle and longer path using two cycles. As most paths execute in cycle period that is much smaller than the critical path delay, the variable latency design has a smaller average latency.

II. PAPER CONTRIBUTION

In this paper, a reliable multiplier design with an adaptive hold logic (AHL) circuit is proposed. Adaptive hold logic (AHL) circuit can decide whether the input pattern requires one or two cycles and can adjust the judging criteria to ensure that there is minimum error detection and re-execution of clock cycle.

The paper is organized as follows. Section III introduces the background of the column-bypassing multiplier, row multiplier, razor flip-flop and AHL circuit. Section IV explains the proposed architecture design. Section V results and discussion. Section VI conclusion.

III. BACKGROUND AND RELATED WORK

These are the modules used to achieve reliable high performance multiplier even after aging occurs. These are the modules in aging aware reliable high performance multiplier.

3.1 COLUMN BYPASS MULTIPLIER:

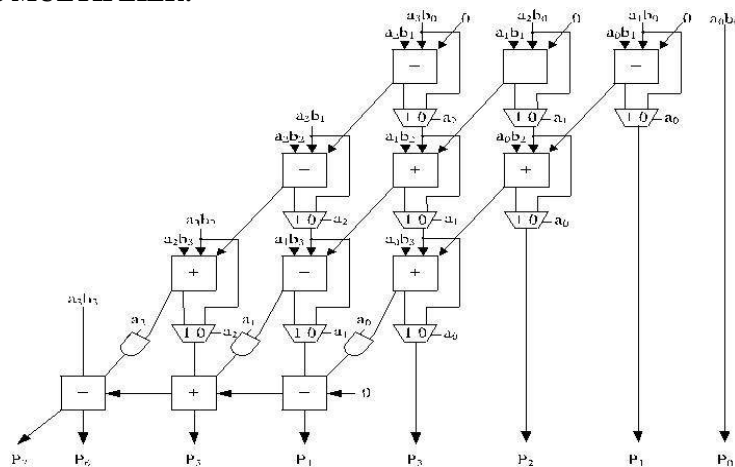


Fig 1. Column Bypassing multiplier.

A column-bypassing multiplier is an improvement on the normal array multiplier (AM). The AM is a fast parallel AM. 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. In 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.1 shows a 4×4 column bypassing multiplier. Supposing the inputs are $1010_2 * 1111_2$, 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 $ai bi$. 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. 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.

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 5, May 2016

3.2 RAZOR FLIP-FLOP

Razor relies on a combination of architectural and circuit level techniques for efficient error detection and correction of delay path failures. The concept of razor is illustrated in Figure for a pipeline stage. Each flip-flop in the design is augmented with a so called shadow latch which is controlled by a delayed clock. And illustrate the operation of razor flip-flop in Fig. 2. In clock cycle1, the combinational logic L1 meets the setup time by the rising edge of the clock and 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 XOR gate remains low and the operation of the pipeline is unaltered.

To guarantee that the shadow latch will always latch the input data correctly, the allowable operating voltage is constrained at design time such that under worst-case conditions, the logic delay does not exceed the setup time of the shadow latch [9]. 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 3 and in the subsequent cycle, cycle 4, the valid data in the shadow latch is restored into the main flip-flop and becomes available to the next pipeline stage L2.

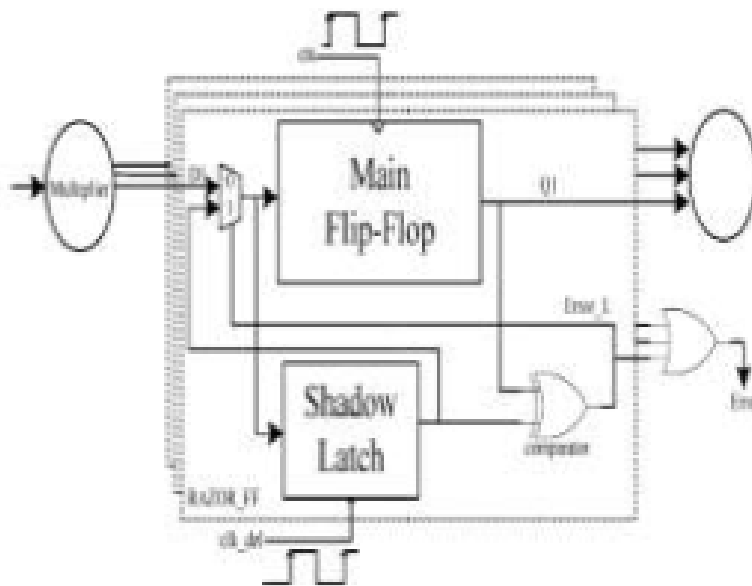


Fig 2. Razor flip-flop

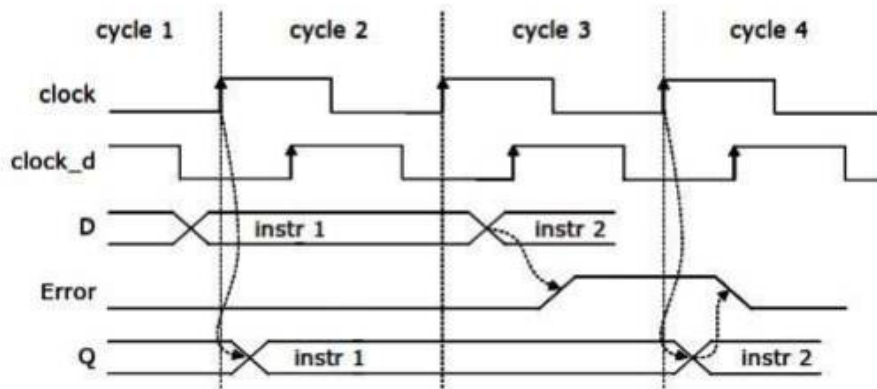


Fig 3. Operation of razor flip-flop.

3.3 AHL CIRCUIT

The AHL circuit is the key component in the aging-aware variable-latency multiplier. Fig. 4 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. These timing violations will be caught by the Razor flip-flops, which generate error signals. If errors happen frequently and exceed a pre-defined 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 . If the number of zeros in the md 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, these judging block allows a smaller number of patterns to become one cycle patterns because it requires more zeros in the multiplicand.

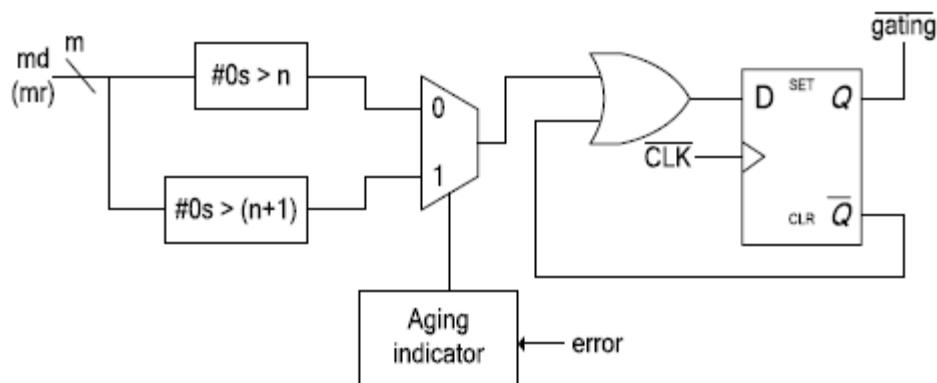


Fig 4. AHL circuit.

IV. PROPOSED METHOD

Aging Aware Reliable Multiplier's Operation

When input patterns arrive, the column bypassing multiplier and the AHL circuit execute simultaneously. According to the number of 0's in the md , the AHL circuit decides if the input patterns require 1 or 2 cycles. If the input pattern requires two cycles to complete, AHL will output 0 to disable the clock signal of the flip-flops. Otherwise, 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-flop. 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 re-executed using two cycles to ensure the operation is correct. In this situation, the extra re-execution cycles caused by timing violation incurs 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 summary, our proposed multiplier design has some key features. First, it is a variable latency design that minimizes the timing waste of the non critical 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. When the circuit is aged, and many errors occur, the AHL circuit uses these judging block to decide if an input is one cycle or two cycles.

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 5, May 2016

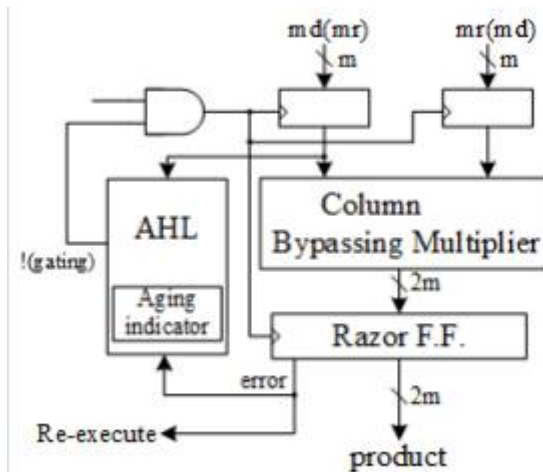


Fig 5. Aging aware reliable multiplier with AHL

V. SIMULATIONS AND RESULTS

A simulation result for column bypassing multiplier is simulated in a Xilinx ISE 14.1. These tools will help to analyze its performance and calculate the power, delay and area .Fig below shows the simulation result of 4*4 column bypassing multiplier with AHL circuit.

First simulation wave shows ,when number of zeros in multiplicand are greater than n, and second simulation wave shows the waveform when number of zeros in multiplicand is greater than (n+1). As column bypass multiplier bypasses the partial product, depends on the number of zeros in multiplicand. So the first simulation wave took 3 cycles as there are no zeros in multiplicand (it took 2cycle).

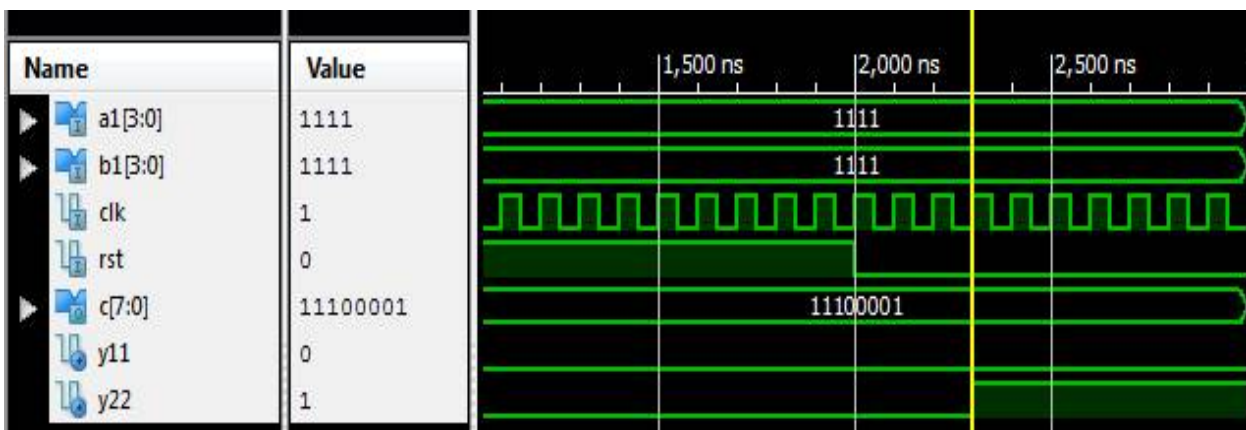


Fig 6. Simulation wave of 0's \geq n

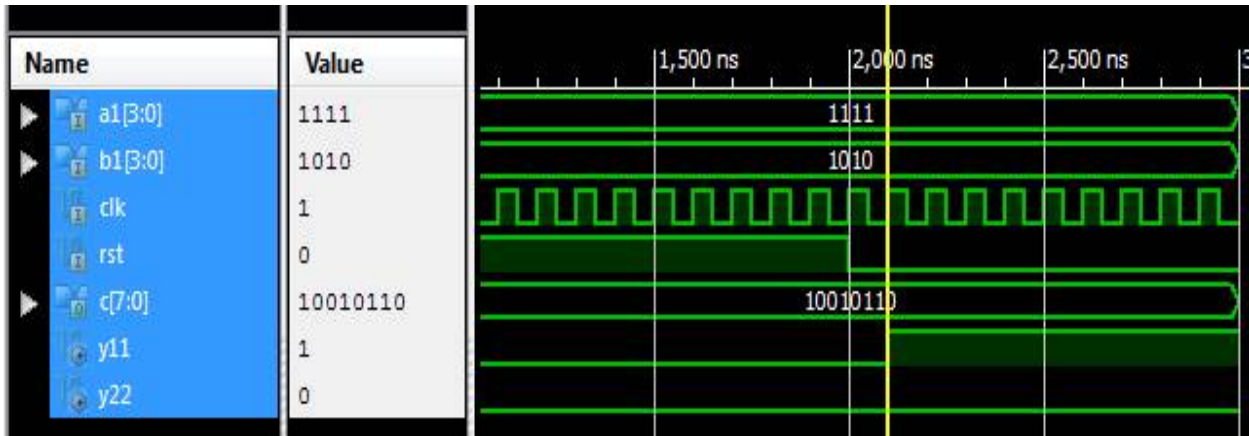


Fig 7. Simulation wave of 0's $\geq (n+1)$

WORD SIZE	AREA(GATE COUNT)	POWER(mW)	DELAY(ns)
4*4 (Fixed latency)	585	40	11.169
4*4 (Variable latency)	585	40	6.68
16*16 (Fixed latency)	12826	85	12.22
16*16 (Variable latency)	8199	53	6.5

Table.1 Comparison Result of 4*4 and 16*16 column bypassing multiplier

In the above Table.1 the fixed latency and variable latency are compared for area, power and delay. In the fixed latency the more number of clock cycles are required and due to which the area, power and delay are increased. By using proposed adaptive hold logic i.e., in the variable latency the less number of clock cycles are used and due to which the error is reduced so that the area, power and delay are reduced in variable latency.

VI. CONCLUSION

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 adjust the AHL circuit to achieve reliable operation under the influence of NBTI and PBTI effects. The multiplier is able to adjust the AHL to mitigate performance degradation due to increased delay. Note that in addition to the NBTI and PBTI effect that increases transistor delay, interconnect also has its aging issue, which is called electro migration. If the aging effects caused by the BTI effect and electro migration are considered together, the delay and performance degradation will be more significant. And error occurred due to timing violations are reduced by proposed reliable multiplier using adaptive hold logic.

REFERENCES

- [1] R. Vattikonda, W. Wang, and Y. Cao, "Modeling and minimization of, Jun. 2004, pp. 1047–1052. S. V. Kumar Sapatnekar, "NBTI-aware synthesis of digital circuits," in *Proc. ACM/IEEE DAC*, Jun. 2007, pp. 370–375. C.H. Kim, and S. S.
- [2] A. Calimera, E. Macii, and M. Poncino, "Design techniques for NBTI-tolerant power-gating architecture," *IEEE Trans. Circuits Syst., Exp. Briefs*, vol. 59, no. 4, pp. 249–253, Apr. 2012. *IEEE Trans. Circuits*
- [3] 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. *cell based designs," in*



International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 5, May 2016

- [4] M. Basoglu, M. Orshansky, and M. Erez, “NBTI-aware DVFS: A new approach to saving energy and increasing processor lifetime,” in *Proc. ACM/IEEE ISLPED*, Aug. 2010, pp. 253–258.
- [4] K.-C. Wu and D. Marculescu, “Aging-aware timing analysis and optimization considering path sensitization,” in *Proc. DATE*, 2011, pp. 1–6.
- [5] Y.-S. Su, D.-C. Wang, S.-C. Chang, and M. Marek-Sadowska, “Performance” optimization using variable-latency design style,” *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 19, no. 10, pp. 1874–1883, Oct. 2011.
- [6] 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 nano scale CMOS SRAM,” *IEEE Trans. Circuit Syst.*, vol. 58, no. 6, pp. 1239–1251, Jun. 2011.
- [7] H. Abrishami, S. Hatami, B. Amelifard, and M. Pedram, “NBTI-aware flip-flop characterization and design,” in *Proc. 44th ACMGLS VLSI*, 2008, pp. 29–34.
- [8] N. V. Mujadiya, “Instruction scheduling on variable latency functional units of VLIW processors,” in *Proc. ACM/IEEE ISED*, Dec. 2011, pp. 307–312.
- [9] D. Ernst *et al.*, “Razor: A low-power pipeline based on circuit-level timing speculation,” in *Proc. 36th Annu. IEEE/ACM MICRO*, Dec. 2003, pp. 7–18.