A Novel Paradigm to Eliminate Timing Violations using AHL

R. Navaneethakrishnan1*, S. Rekha2 and S. Bhavani3

1Department of ECE, Kumaraguru College of Technology, Coimbatore – 641049, Tamil Nadu, India; navaneethakrishnan.ece@kct.ac.in
2Karpagam Academy of Higher Education, Coimbatore - 641021, Tamil Nadu, India; rekharupa27@gmail.com,
3Department of ECE, Karpagam Academy of Higher Education, Coimbatore - 641021, Tamil Nadu, India;

Abstract

Objectives: This Paper presents a novel low power design approach for multipliers to eliminate timing violations. There are two major issues which are concentrated in this paper are positive bias temperature instability and negative bias temperature instability. Both the things affect the speed of transistor and leads to timing violations, which intern leads to the failure of an entire system. Methods: In this work, bypassing multiplier is used with adaptive hold logic. The implementation is done in 180 nm deep submicron CMOS technology. Findings: power consumption and error rate is studied after employing the AHL. Improvements: The experimental result shows that the performance of multipliers with AHL improved 72.8% when compared to existing methods and consumes less power.

Keywords: AHL, Bypass Multipliers, Low Power Design, Instability, Timing Violations

1. Introduction

Digital multipliers are the most important building blocks of many digital systems. As an example nothing can be done in signal processing units without multipliers. In most cases the multiplication will be implemented as successive addition. Timing violation of a multiplier is a serious issue that has to be addressed well. The timing violations of the multipliers are mostly because of two things which are positive bias temperature instability and negative bias temperature instability. VLSI domain will no longer consider about the size, the moore’s law exceeded already. So the power consumption is the ground where most of the researchers are playing. Low power design becomes challenging task. In older days cost and performance is considered primarily and power consumption is considered secondarily but now a days the scenario changed in opposite way. Power dissipation due to that is represented as $P_{\text{shortcircuit}}$. Power consumption plays an important role because most of electronic devises are battery operated; they are small and have more functionality. Normally in most of the systems like signal processing units the performance will depend on multiplier blocks, since the multiplication is fundamental function for digital signal processing. The performance of multiplier not only affects the throughput but also the power consumption of the system. This issue is not only concerned for digital signals processors but also general purpose processors (GPP).

Power dissipation of a system can be given as,

$$P_{\text{total}} = P_{\text{switching}} + P_{\text{shortcircuit}} + P_{\text{static}} + P_{\text{leakage}} \quad (1)$$

Here the $P_{\text{switching}}$ is power dissipated due to switching activity. It plays a vital role in these calculations so it is the dominating factor. Ideally either PMOS or NMOS will be in ON state at any instance, but in practical cases both PMOS and NMOS will be in ON state for some time. Power dissipation due to that is represented as $P_{\text{shortcircuit}}$. Static power is represented here as $P_{\text{static}}$ it depends on number of transistors used in a circuit. Finally charging and discharging of load capacitance determines the dynamic power consumption of a circuit.
In this work, we have investigated the problem of aging due to PBTI and NBTI. There are several literatures for the same issue. Overdesign is the method used in previous works to mitigate the aging effect, but it includes overdesign and guard banding. Efficiency in area and power are not addressed well. Aging effect of PMOS sleep transistors were reduced using NTBI aware sleep transistors. Variable latency design is another approach used in, in this the longer paths need two cycles to complete the execution. Negative bias temperature instability occurs when the transistor is in negative bias condition.

2. Methodology

2.1 Multiplication

Multiplication is an arithmetic operation; that performs addition of the same operand for number of times. Consider the multiplication of two unsigned n-bit numbers, where $X=x_{n-1}, x_{n-2}, \ldots, x_0$ is multiplicand and $Y=y_{n-1}, y_{n-2}, \ldots, y_0$ is multiplier so the product $p$ can be as,

$$p = AB = \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} (a_i b_j) 2^{i+j}$$

Equation (2)

Here $i$ and $j$ are the number of bits in the multiplicand and multiplier. The product can be obtained by summing the PPs.

The multiplication operation consists of two functions,

i. Partial product generation

ii. Accumulation

Partial product generation employs a wide range of techniques like shift/add, array multiplication, booth algorithms, modified booths algorithm, etc. Adders will be employed in accumulation. Parallel adders will give an instantaneous parallel output and rippling problem can be solved.

2.2 Bypass Multipliers

Switching part of a power plays an important role in power consumption, so to reduce power consumption we need to reduce the switching power. There are various low power techniques available to make the unused part of the multiplier shut down.

In multipliers by shutting down the idle part of the multiplier, we can avoid unwanted switching activity so we can reduce maximum of switching power and hence it reflects in total power consumption. To do this, we are going for bypassing techniques. In bypassing, we skip addition when the partial product of a column or row is zero. In this work, we use two bypassing multipliers namely row bypassing multipliers (RBM) and column bypassing multipliers (CBM).

3. Proposed Design

In most of the multipliers, the timing violation plays a dominant role in reducing throughput of the entire system and also higher power consumption. In this paper, we propose Adaptive Hold Logic to mitigate the aging issue. Through this, the timing violation will be controlled and hence the throughput and power consumption will be improved. The proposed architecture of Adaptive Hold Logic (AHL) is shown in Figure 1,2. Here we have two decision-making blocks, those are used to check whether the number of zeros is greater than 0 or less than 0. The decision-making block 1 will produce high output when the number of zeros encountered are greater than n. Likewise, the output of block 2 will be high when the number of zeros encountered are less than n. Here the n is the length of the sequence and that is user-defined.

Aging detector is a kind of counter, it counts the number of errors occurred in the block. Output of Aging detector will be high whenever there is more number of errors in the block. So after the threshold level the Aging detector will produce high at the output. Initially, there will be a less number of errors in the block so the output of Aging detector will be logic low. Logic high output of Aging detector also indicates that there are more errors in the block so the block is seriously affected by timing issue.

The decision making blocks will be connected with multiplexer. The MUX switches the output with respect to the Aging detector. The MUX output will be high whenever there is more number of errors in the block. If the sequence requires only one cycle then the MUX output will be 1 and that will be driven to DFF through OR gate, for this case the output of DFF will be 1, that will be given to gating. If the sequence requires two cycles then MUX output will be 0, then for this case the output of DFF will be 0, so the gating will become 0 for two cycles. So the sequence

4. Results and Discussion

The design and implementation of AHL architecture have been done in Tanner EDA. Here the number of zeros in
input x is 1 and y is 0 or higher. Decision making block and aging identifier have been implemented in AND logic.

\[
\begin{array}{cccc}
  x_3 & x_2 & x_1 & x_0 \\
  y_3 & y_2 & y_1 & y_0 \\
\end{array}
\]

\[
\begin{array}{cccccccc}
  x_3 y_0 & x_2 y_0 & x_1 y_0 & x_0 y_0 \\
  x_3 y_1 & x_2 y_1 & x_1 y_1 & x_0 y_1 \\
  x_3 y_2 & x_2 y_2 & x_1 y_2 & x_0 y_2 \\
  x_3 y_3 & x_2 y_3 & x_1 y_3 & x_0 y_3 \\
\end{array}
\]

| P7 | P6 | P5 | P4 | P3 | P2 | P1 | P0 |

**Figure 1.** Process of Multiplication.

Figure 5 is the condition where the error is detected in Aging detector. In that case the input to the DFF will made as 0 so that the gating will also be 0 for some instance.

**Figure 2.** Block Diagram of Proposed AHL Architecture.

Figure 3 is the schematic of AHL. If there is no error in the function then the next sequence will be processed and the gating will made to 1. If error rate is high then the input to DFF will be made as 0 and hence the gating also will be 0.

**Figure 3.** Circuit of AHL.

Figure 4 shows the results of AHL block when there is no error.

**Figure 4.** Result of AHL circuit without error signal.

**Figure 5.** Circuit of AHL with an error signal.

5. Conclusion

Adaptive Hold Logic have been presented in this paper to mitigate the timing violations. The simulation result shows that the timing violation have been addressed well using this design approach. This can be utilized in multipliers of MAC unit used in Digital Signal Processors.

6. References

1. Linn IC, Cho YH, Yang YM. Aging Aware Reliable Multiplier Design With Adaptive Hold Logic. IEEE Transaction on Very Large Scale Integration (VLSI) System. 2014; 23(3):544–56.
2. Kaur R, Noor A. Strategies and Methodologies For Low Power VLSI Designs: A Review. International Journal of Advances in Engineering and Technology. 2011; 1(2):159–65.
3. Rakesh V, Wang W, Cao Y. Modelling and minimization of pMOS NBTI effect for robust naometer design. Proceeding of ACM/IEEE DAC. 2004; 1047–52.
4. Abrishami H, Hatami S, Amelifard B, Pedram M. NBTI-aware flip-flop characterization and design. Proceeding of 44th ACM ACM Great Lakes symposium on VLSI, Orlando, Florida, USA. 2008; 29–34.

5. Calimera A, Macii E, Poncino M. Design techniques for NBTI tolerant power-gating architecture. IEEE Transaction on Circuits System II: Express Briefs. 2012; 59(4):249–53.

6. Baneres D, Cortadella J, Kishinevsky M. Variable-latency design by function speculation. Proceedings of the Conference on Design, Automation and Test in Europe, Belgium. 2009. p. 1704–09.

7. Su YS, Wang DC, Chang SC, Sadowska MM. Performance optimization using variable-latency design style. IEEE Transaction on Very Large Scale Integration System. 2009; 19(10):1874–83.

8. Yan J, Chen Z. Low-Power Multiplier Design with Row and Column Bypassing. Proceeding of IEEE International SOC Conference, Taiwan. 2009. p. 227–30.

9. Hwang YT, Lin JF, Sheu MH, Sheu CJ. Low Power Multiplier Designs Based on Improved Column Bypassing Schemes. Proc IEEE Asia Pacific Conference on Circuits and Systems, Taiwan. 2006.

10. Keating M, Flynn D, Aitken R, Shi AK. Low Power Methodology Manual for System on Chip Design, Springer Publications: New York, 2007.

11. Kumar SV, Kim CH, Sapatnekar NS. NBTI-aware synthesis of digital circuits. Proceedings of ACM/IEEE Annual Design Automation Conference, USA. 2007. p. 370–75.