ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Issue 3, March 2016

Analysis of an Efficient Multiplier Architecture Using Adaptive Hold Logic

S.VISHU KUMAR[1], Mrs. Priya Dharshini[2], Mr. S.SARAVANAN[3]

[1]PG Student, M.E (VLSI), Jansons Institute of Technology, Coimbatore-641659

[2]PG Student, M.E (VLSI), Jansons Institute of Technology, Coimbatore-641659

[3]Assistant Professor/Department of ECE, Jansons Institute of Technology, Coimbatore-641659

Abstract: High speed, Low power consumption and Long Life Cycle are one of the most important design objectives in integrated circuits. As digital multipliers are among the most critical arithmetic functional units and are the most widely used components in such circuits, the multipliers must be designed efficiently. The two major constraints for delay in any VLSI circuits are latency and throughput. The bias temperature instability effect occurs when a transistor is under bias, increasing the threshold voltage of the transistor, and reducing multiplier speed. This effect degrades 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 analysis such a reliable, simple and efficient multiplier which has reduced delay and increased lifecycle. We propose a multiplier design with the idea of razor flip flop and 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 moderate the performance degradation that is due to the aging effect. In the fixed latency usage of clock cycles are increased. Thus the re-execution of clock cycles is reduced by using variable latency. The result analysis shows that the reliable multiplier has better performance in power consumption and delay than contemporary architectures.

Keywords: Adaptive hold logic (AHL), negative bias temperature instability (NBTI), positive bias temperature instability (PBTI), reliable multiplier, variable latency.

I. INTRODUCTION

Traditional digital circuit uses critical path delay as the overall circuit clock cycle in order to perform correctly. Hence, the variable-latency design was proposed to reduce the timing waste of traditional circuits and thus reduce the latency of the circuit and to increase the speed of the multiplier architecture.

A. *The variable-latency design*

The variable-latency design divides the circuit into two parts; they are the shorter paths and the longer paths. Shorter path can execute correctly in one cycle, whereas a longer path needs two cycles to execute. The column bypass multiplier and row bypass multiplier design methodology inserts more number of zeros in the multiplicand and multiplier thereby reducing the number of delay. The delay reduction depends on the input bit coefficient. This means if the input bit coefficient is zero, corresponding row or column of adders need not be activated. Also as the individual multipliers are used as per the need, therefore the overall lifecycle also gets moderated. [4]

B. *The CMOS unreliability effects*

A CMOS circuit's unreliability effect can be broadly classified as spatial unreliability and temporal unreliability which can be further classified as shown in figure 1. The former are visible right after production and can be random or systematic. The latter become a potential problem during the operational lifetime of the circuit and present themselves as an aging effect or a transient effect

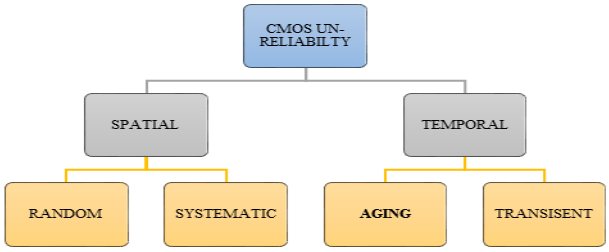


Figure.1 A CMOS circuit spatial or temporal unreliability effects.

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Issue 3, March 2016

Spatial unreliability effects can be random or systematic. The effects depend on the circuit layout, the neighboring environment, process conditions and the impact of the geometry and structure of the circuit and can lead to yield loss. This yield loss can be functional or parametric, i.e. resulting in malfunctioning circuits or circuits with degraded performance respectively.

Temporal unreliability effects, on the other hand, are time varying and change depending on operating conditions such as the operating voltage, temperature, switching activity, presence and activity of neighboring circuits.

C. Aging effects

Integrated circuit aging phenomena were first observed during the seventies and the eighties. After the turn of the century, the introduction of new materials to further scale CMOS technologies introduced additional failure mechanisms and made existing aging effects more severe. Due to aggressive scaling of the device geometries and increasing electric fields, circuit aging has become an issue. Measurements on individual transistors were used to determine circuit design margins in order to guarantee reliability. Integrated-circuit aging phenomena observed in sub-90nm CMOS technologies are as follows.

a) Hot Carrier Injection (HCI).
b) Time-Dependent Dielectric Breakdown (TDDB).
c) Bias Temperature Instability (BTI).
d) Electro-migration (EM).

D. Bias temperature instability (BTI)

Bias Temperature Instability (BTI) has gained a lot of attention due to its increasingly adverse impact in nano-meter CMOS technologies. It is a threshold voltage (V^{TH}) shift after a bias voltage has been applied to a MOS gate at elevated temperature. It causes threshold voltage (V^{TH}) increment to the MOS transistors.

Threshold voltage V^{TH} increment in a pMOS transistor that occurs under the negative gate stress is referred to as Negative Bias Temperature Instability (NBTI) and the one that occur in an nMOS transistor under positive gate stress is known as Positive Bias Temperature Instability (PBTI).

The NBTI or PBTI impact can become more significant depending on the dielectric type. These two phases differ by the gate biasing (i.e. V_{DD} or – V_{DD}) of the MOS transistors. For a MOS transistor, there are two BTI phases.

a) Stress phase.
b) Relaxation phase.

These effect are a significant reliability threat in both older SiO_2 and SiON technologies and is still a problem in newer HKMG technologies [8]. The PBTI effect affects nMOS transistors and results in a similar wear out behaviour as NBTI, but has only been observed in HKMG nMOS devices.

Currently, there still is no consensus about the microscopical origins of both BTI phenomena. Most authors argue that the NBTI effect results from a combination of hole trapping in oxide defects and generation of interface states at the channel oxide interface [6].

PBTI is believed to come from electron trapping in pre-existant oxide traps, combined with a trap generation process [7] Further, initial research on next generation CMOS structures such as multi-gate devices (MuGFETs, FinFETs, etc.) indicates that BTI remains a problem in future CMOS technologies [8].

II. Impact Of Nbti On The Performance Degradation Of Digital Circuits

Negative-bias temperature instability (NBTI) is a major side effect on the lifetime reliability of integrated circuits. With the continuous scaling of transistor dimensions, the reliability degradation of circuits has become an important issue. Due to an increasing electric field across the thin oxide, the generation of interface traps under negative bias temperature instability (NBTI) in pMOS transistors has become one of the most critical reliability issues that determine the lifetime of CMOS devices. Due to NBTI, the threshold voltage of the transistor increases with time resulting in the reduction in drive current, which in turn results in temporal performance degradation of circuits.

V^{TH} degradation Model NBTI is the result of trap generation at Si/SiO interface in negatively based PMOS transistors at elevated temperatures. The interaction of inversion layer holes with hydrogen passivized Si atoms can break the SiH bonds, creating

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Issue 3, March 2016

an interface trap and one H atom that can diffuse away from the interface or can anneal an existing trap. Delay of gate depends on threshold voltage value. Therefore, by monitoring the threshold voltage degradation, the change in gate delay can be easily estimated with a high degree of accuracy. From the above analysis, it is clear that circuit delay depends on threshold voltage variation and hence performance degradation will occur. Various techniques used to mitigate this effect is V_{DD} tuning, PMOS sizing, Tuning of gate length and Tuning of switching frequency. By this techniques we can reduce NBTI to a greater extend but area, power inefficiency is a major problem.

III. METHODOLOGY

A. *Bypassing Technique*

Bypassing multipliers are modification of normal array multipliers. Dynamic power consumption can be reduced by bypassing method when the multiplier has more zeros in input data. The path delay for an operation is strongly tied to the number of zeros in the multiplicands in the column-bypassing multiplication and in multiplier in row bypass multiplication.

B. *Aging effect*

Aging problem of transistors has a significant effect on performance of the systems and in long term, the system may fail due to timing violations. Aging effect can be reduced by using over-design approaches, but these leads to area, power inefficiency. Hence to reduce the maximum power consumption and delay, variable latency multiplier with adaptive hold logic is used. 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.

C. *Array multiplier*

The array multiplier is a fast parallel multiplier and is shown in Figure 2 and it consists of (n-1) rows of carry save adder, in which each row contains (n-1) full adders. Each full adder in the carry save adder array has two outputs they are the sum bit goes down and the carry bit goes to the lower left full adder. The last row is a ripple adder for carry propagation.

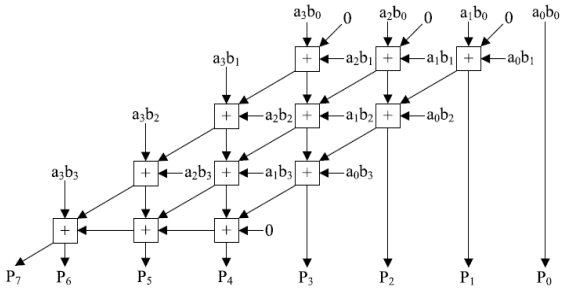


Figure 2 Parallel Array Multiplier

D. *Column bypassing multiplier*

A column-bypassing multiplier is an improvement of the array multiplier and is shown in Figure 3. A low-power column-bypassing multiplier design is proposed in which the full adder operations are disabled if the corresponding bit in the multiplicand is zero. Supposing the inputs are 1010 * 1111, it can be seen that for the full adders in the first and third diagonals, two of the three input bits are 0 and the carry bit from its upper right full adder and the partial product a_ib_i . The multiplicand bit a_i can be used as the selector of the multiplexer to decide the output of the full adder, and a_i can also be used as the selector of the tri-state gate to turn off the input path of the full adder. If a_i is 0, the inputs of full adder are disabled, and the sum bit of the current full adder is equal to the sum bit from its upper full adder, thus reducing the power consumption of the multiplier. If a_i is 1, the normal sum result is selected.

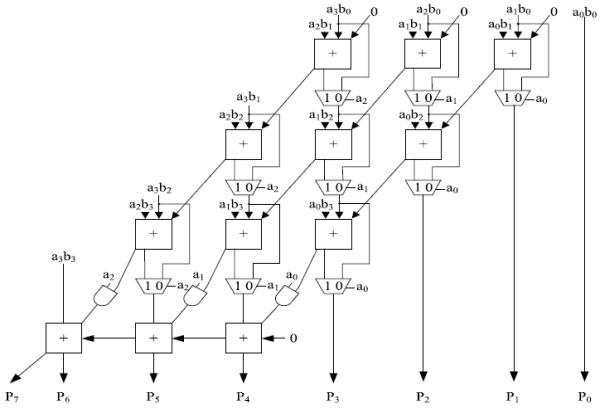


Figure 3 Column By-Pass Multiplier

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Issue 3, March 2016

E. Row bypassing multiplier

A low-power row-bypassing multiplier as shown in Figure 3.3 is also proposed to reduce the power consumption and use of more clock cycles. 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. The design of 4*4 row bypassing multiplier is shown in Figure 4. 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. Each input is connected to full adder through a tristate gate. When the inputs are 1111 * 1001, the two inputs in the first and second rows are 0 for full adders. 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 full adders in the second rows, and the tristate gates turn off the input paths to the full adders.

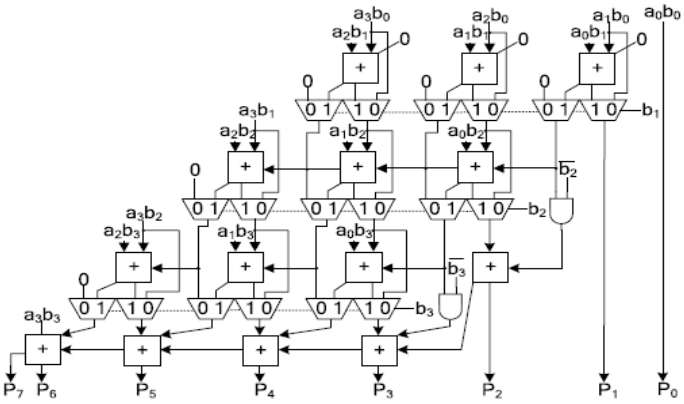


Figure 4 Row By-Pass Multiplier

F. Variable latency design

The variable-latency design is proposed to reduce the timing waste occurring in traditional circuits that use the critical path cycle as an execution cycle period cycles as shown in Figure 5. The basic concept is to execute a shorter path using a shorter cycle and longer path using two cycles.

Most of arithmetic operations, e.g. multiplication and division are implemented using several add/subtract steps. Thus, improving the speed of addition will improve the speed of all other arithmetic operations. Accordingly, reducing the carry propagation delay of adders is of great importance. Different logic design approaches have been employed to overcome the carry propagation problem. One widely used approach employs the principle of carry look-ahead solves this problem by calculating the carry signals in advance, based on the input signals.

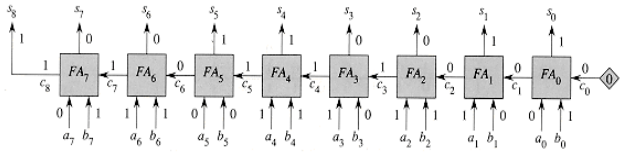


Figure 5 Carry Look Ahead adder

This type of adder circuit is called as carry look-ahead adder (CLA adder).It is based on the fact that a carry signal will be generated in two cases:

(1) When both bits A_i and B_i are 1, or
(2) When one of the two bits is 1 and the carry-in (carry of the previous stage) is 1

The Boolean expression of the carry outputs of various stages can be written as in equation (3.1 and 3.2):

$$C_1 = G_0 + P_0 \cdot C_0$$
$$C_2 = G_1 + G_0 \cdot P_1 + C_0 \cdot P_0 \cdot P_1$$

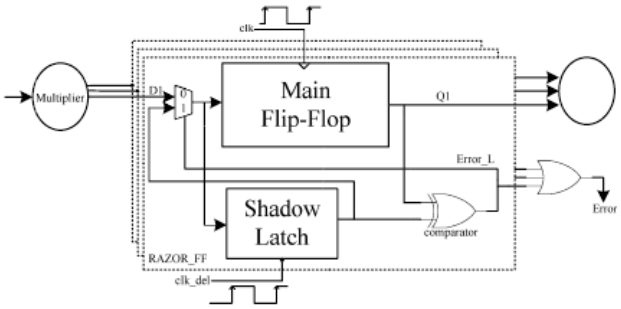
(3.1)

In general, the i^{th} carry output is expressed in the for

$$\mathbf{C_i = F_i(P's, G's, C_0).}$$

(3.2)

In other words, each carry signal is expressed as a direct



SOP function of C_0 rather than its preceding carry signal. Since the Boolean expression for each output

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Issue 3, March 2016

carry is expressed in SOP form, it can be implemented in two-level circuits.

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. Figure 6 is an 8-bit variable-latency ripple carry adder (RCA). *A*7–*A*0, *B*7–*B*0 is 8-bit inputs, and *S*8–*S*0 are the outputs. Supposing the delay for each full adder 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.

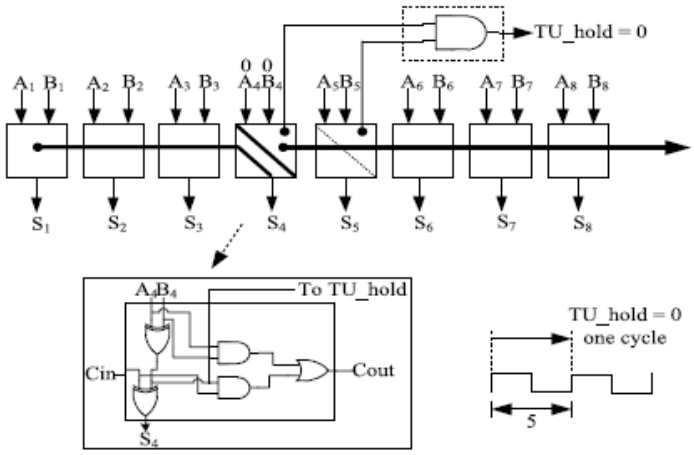


Figure 6 An 8-bit RCA with a hold logic circuit.

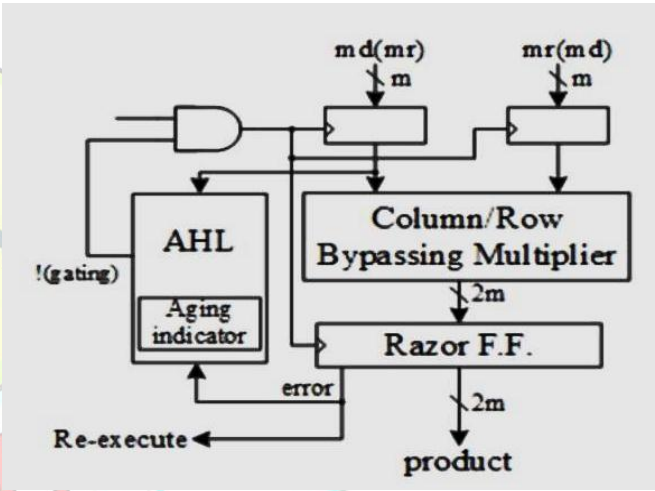
G. Razor flip flop

A 1-bit Razor flip-flop contains a main flip-flop, shadow latch, XOR gate, and mux as shown in Figure 7. 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. Although the re-execution may seem costly, the overall cost is low because the re-execution frequency is low.

Figure 7 A 1-bit Razor flip-flop

H. *Adaptive hold logic (AHL)*

The Adaptive Hold Logic (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.



The Adaptive Hold Logic (AHL) circuit is as shown in Figure 8. Assume the AHL circuit has a m bit input. The Adaptive Hold Logic (AHL) circuit consists of the following blocks.

a. Judging Blocks

b. D Flip-Flop

c. One Multiplexer

d. Aging Indicator

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Issue 3, March 2016

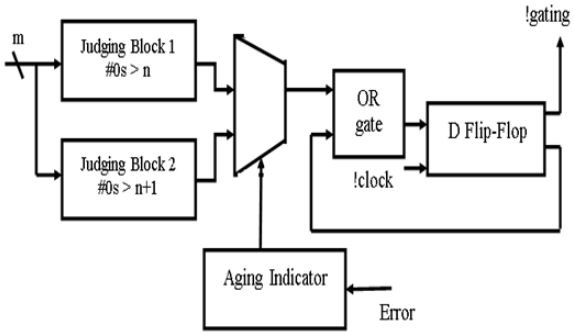


Figure 8 Adaptive Hold Logic (AHL) circuit

I. Multiplier design with adaptive hold logic

A reliable multiplier design with a adaptive hold logic (AHL) circuit is shown below. The multiplier is based on the variable-latency technique and can adjust the AHL circuit to achieve reliable operation to reduce the error and re-execution of clock cycle. The adaptive hold logic (AHL) circuit can decide whether the input patters 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. That is to find the longest delay path and shortest delay path we propose to use adaptive hold logic technique (AHL).

The 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 and an AHL circuit as shown in Figure 9. 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. When input patterns are random, the number of zeros and ones in the multiplicator and multiplicand follows a normal distribution.

Therefore using the number of zeros or ones as the judging criteria results in similar outcomes. Hence, the two 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 multiplicator.

Figure 9 Multiplier design with Adaptive Hold Logic

IV. RESULTS AND DISCUSSION

A. Basic array multiplier

The 4x4 basic array multiplier is simulated using Xilinx Web ISE Version 14.2 and verified for possible inputs given below. The simulation result and RTL diagram is shown in figure 10.

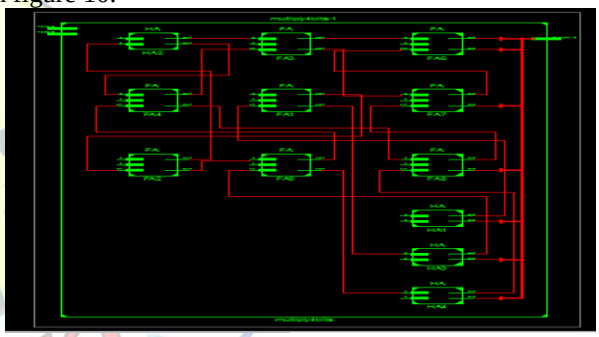


Figure 10 RTL of Basic 4*4 multiplier

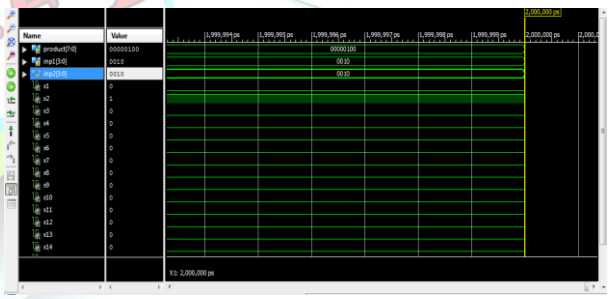


Figure 11 Simulation output waveform of a 4*4 Basic Array Multiplier

The inputs for the 4*4 Array multiplier is a [0:3] and b [0:3]. The output is product [0:7].

If the inputs are given as a_0=0, a_1=0, a_2=1, a_3=0, and b_0=0, b_1=0,b_2=1, b_3=0.

Then the simulated output is Product [00000100].

The above stated multiplier design has a maximum combinational path delay of 9.982 Nano seconds.

A. Row bypass array multiplier

The 4x4 row bypass array multiplier is simulated using Xilinx Web ISE Version 14.2 and verified for possible

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Issue 3, March 2016

inputs given below. The simulation result and RTL diagram is shown in figure 12.

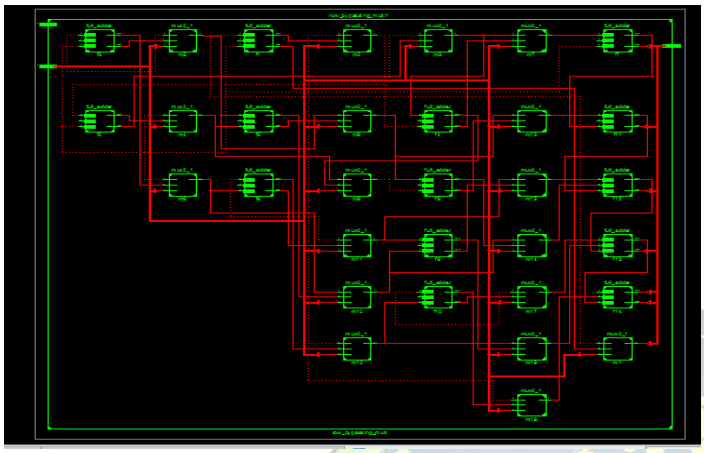


Figure 12 RTL of Row 4*4 Bypass multiplier

Figure 13 Simulation output waveform of a 4*4 Row Bypass Array Multiplier

The inputs for the 4*4 Array multiplier is a [0:3] and b [0:3]. The output is product [0:7].
If the inputs are given as a_0=0, a_1=0, a_2=0, a_3=1, and b_0=1, b_1=1,b_2=1, b_3=1.
Then the simulated output is Product [00001111].
The above stated multiplier design has a maximum combinational path delay of 10.125 Nano seconds. This shows an increased delay period.

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Issue 3, March 2016

B. Column bypass array multiplier

The 4x4 column array multiplier is simulated using Xilinx Web ISE Version 14.2 and verified for possible inputs given below. The simulation result and RTL diagram is shown in figure 14.

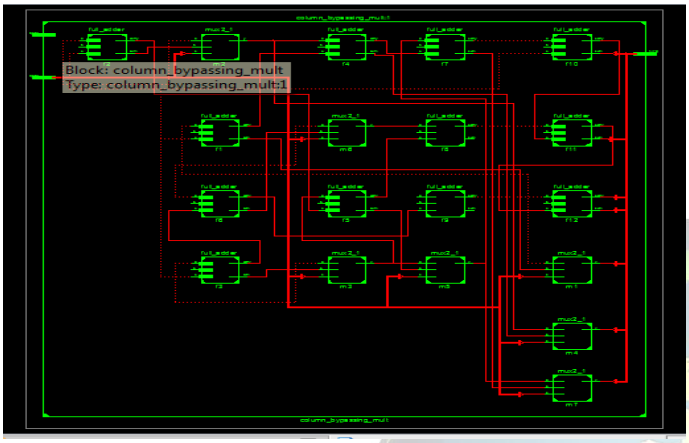


Figure 14 RTL of Column 4*4 Bypass multiplier

Figure 15 Simulation output waveform of a 4*4 Column Bypass Array Multiplier

The inputs for the 4*4 Array multiplier is a [0:3] and b [0:3]. The output is product [0:7].
If the inputs are given as a_0=1, a_1=1, a_2=1, a_3=1, and b_0=0, b_1=0,b_2=0, b_3=1.
Then the simulated output is Product [00001111].
The above stated multiplier design has a maximum combinational path delay of 6.497 Nano seconds. This shows a decrease in delay compared to conventional architecture. Thus we choose to use column bypass multiplier.

*C. 8*8 column/row bye pass array multiplier with ahl and ageing aware circuit*

The RTL generated in Xilinx ISE for the 8x8 Reliable multiplier is shown in Figure 16.

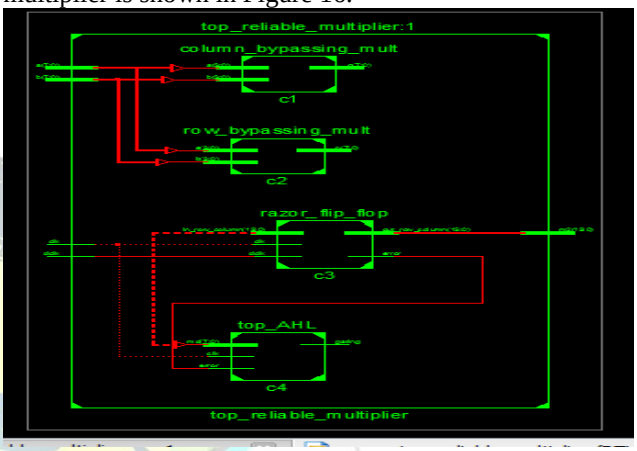


Figure 16 RTL Schematic of 8x8 Reliable Multiplier

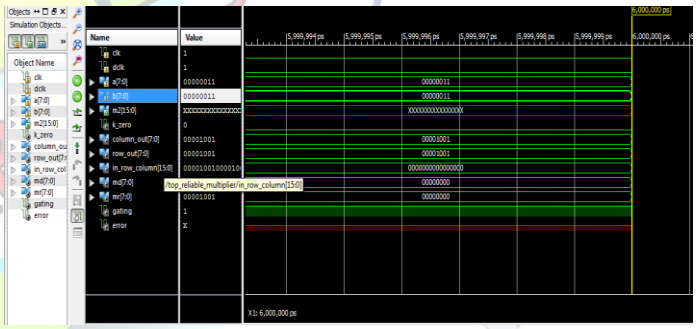


Figure 17 Column/Row Bypass Multiplier With AHL and Ageing aware Circuit

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Issue 3, March 2016

Figure 18 Column Bypass Array Multiplier

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Issue 3, March 2016

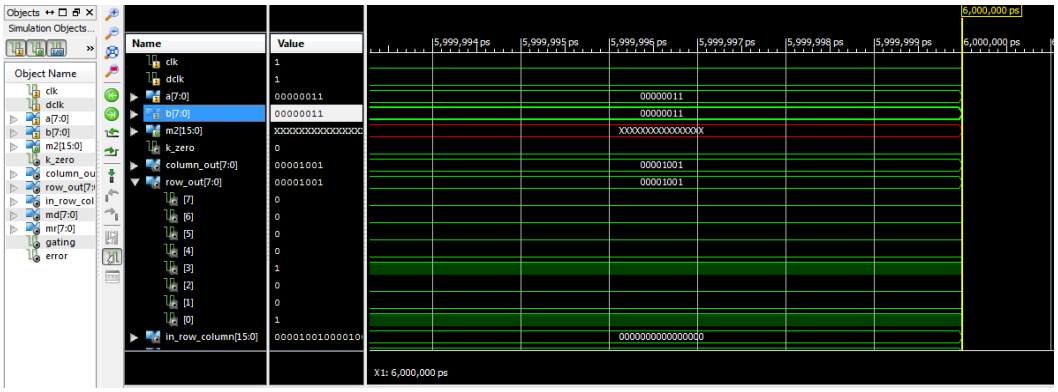


Figure 19 Row Bypass Array Multiplier

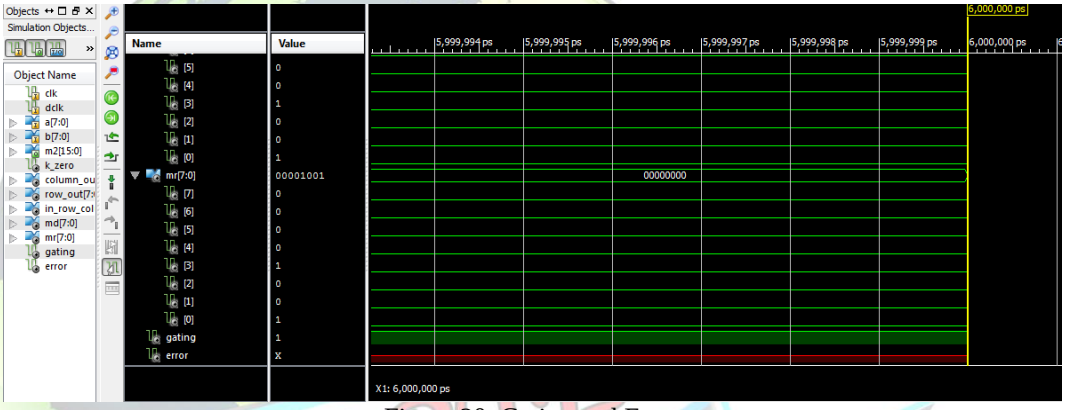


Figure 20 Gating and Error

The inputs for the 8*8 Array multiplier is a [0:7] and b [0:7].
The output is product [0:15].
If the inputs are given as a_0=0, a_1=0, a_2=0, a_3=0, a_4=0, a_5=0, a_6=1, a_7=1, and b_0=0, b_1=0,b_2=0, b_3=, b_4=0, b_5=0,b_6=1, b_7=1.
Then the simulated output is Product [00000000 00001001].

D. Comparison

The 4x4 Column/Row Bypass array multiplier along with AHL and Razor circuit is designed. The Bypass multiplier is compared with the existing multiplier architecture in term of delay. The results obtained are tabulated in Table 4.1.

Table 1 Delay Comparison of Different Multipliers

Multiple Architectures	Maximum Combinational Path Delay(ns)
Basic array multiplier	9.982
Column by pass multiplier	10.125
Row by pass multiplier	6.497

From the Table 1, it is evident that there is a reduction in delay an Increase in Life Cycle. The delay for the existing multiplier architecture is 9.982 ns. The delay for the column bypass multiplier architecture is 6.497ns and also as we use variable latency instead of fixed latency method the life cycle gets increased. Thus, it is clear that the column bypass multiplier with AHL and Razor circuit is more efficient than the existing architecture. The Column Bypass Multiplier can be used to develop a high speed complex number multiplier with reduced delay and Long life cycle.

The experimental results also shows that the proposed architecture with 16×16 and 32×32 column-bypassing multipliers can attain up to 62.88% and 76.28% performance improvement compared with the 16 × 16 and 32 × 32 FLCB multipliers, respectively. Furthermore, the proposed architecture with the 16×16 and 32×32 row-bypassing multipliers can achieve up to 80.17% and 69.40% performance improvement

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Issue 3, March 2016

compared with the 16 × 16 and 32 × 32 FLRB multipliers. [1].

V. CONCLUSION

This paper presents a novel way of realizing a highly reliable multiplier using AHL and Ageing aware circuit. The designs of 4x4 and 8x8 bits multiplier have been implemented using Xilinx Web ISE tool. This aging-aware variable latency multiplier design with the AHL is able to adjust the AHL to mitigate performance degradation due to increased delay. The 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 the architecture with 4x4 and 8x8 multiplication with CLA as last stage instead of Normal RCA adder; it will decrease the delay and improve the performance compared with previous designs.

The row bypass multiplier gives 6.497ns delay which is less when compared to the existing multiplier architecture. It is therefore seen that the row bypass multipliers with AHL and ageing aware circuit are much faster than the conventional multipliers and have longer life throughput. This gives us method for hierarchical multiplier design. So the design complexity gets reduced for inputs of large no of bits and modularity gets increased.

REFERENCE

[1] Ing-Chao Lin, Member, IEEE, Yu-Hung Cho, and Yi- Ming Yang, ”Aging-Aware Reliable Multiplier Design With Adaptive Hold Logic” IEEE Transactions On Very Large Scale Integration (VLSI) Systems, Vol.75,Feb 2014.

[2] Prabhu E, Mangalam H, Saranya K, “ Design of Low Power Digital FIR Filter based on Bypassing Multiplier”, International Journal of Computer Applications (0975 – 8887) Volume 70– No.9, May 2013

[3] J.Sudha Rani, D.Ramadevi, B.Santhosh Kmar,Jeevan Reddy K, “Design of Low Power Col-umn bypass Multiplier using FPGA”, IOSR Journal of VLSI and Signal Processing (IOSR-JVSP),Vol.1, Issue 3 (Nov. - Dec. 2012).

[4] Yu-Shih Su, Da-Chung Wang, Shih-Chieh Chang, Member, IEEE, and Malgorzata Marek-Sadowska, ”Performance Optimization Using Variable-Latency Design Style”, IEEE Trans-actions On Very Large Scale Integration (VLSI) Systems, Vol. 19, No. 10, October 2011.

[5] N.Ravi, Dr.T.S.Rao, Dr.T.J.Prasad,” Performance Evaluation of Bypassing Array Multiplier with Optimized Design,” International Journal of Computer Applications, Vol. 28, No.5, 2011.

[6] Dieter K. Schroder, Jeff A. Babcock; JOURNAL OF APPLIED PHYSICS, VOLUME 94, NUMBER 1 1 JULY 2003

[7] F. Crupi et al.”Positive Bias Temperature Instability in nMOSFETs with ultra-thin Hf-silicate gate dielectrics”, Microelectronic Engineering 80 (2005) 130–133

[8] Maricau,E; Glelen.C; “Analog IC reliability in nanometer CMOS; 2013,XVI” 198p,hardcover,ISBN:978-1-4614-6162-3

S.VISHNU KUMAR received his B.E bachelor degree in ECE from Jansons Institute of Technology under Anna University,Chennai in 2014. Currently he is pursuing M.E in VLSI Design from Jansons Institute of Technology under Anna University,Chennai. His areas of interest in research are VLSI and embedded systems.

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Issue 3, March 2016

.**K.R.PRIYA DHARSHINI** received his B.E bachelor degree in ECE from Erode sengunanthutar Engineering college under Anna University,Coimbatore in 2014. Currently he is pursuing M.E in VLSI Design from Jansons Institute of technology His areas of interest in research are VLSI and embedded systems

S. SARAVANAN received his B.E bachelor degree in ECE from Bannari Amman Institute of Technology under Anna University,Chennai and received his master's degree M.E in VLSI Design from Anna University Regional centre,Coimbatore. He is working as Assistant Professor in Department of ECE , Jansons Institute of Technology. His areas of interest are VLSI and embedded Systems.