IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 9, September 2017

"Aging-aware reliable multiplier design with Adaptive Hold Logic"

Sushant J. Sawant[1], Prof. Sanjay S. Badhe[2]

M.E. Student (VLSI & Embedded System), E&TC Department, Dr.D.Y.Patil College of Engineering, Pune, India[1]

Department of E&TC Engg, Dr.D.Y.Patil College of Engineering, Ambi, Pune,India[2]

Abstract: High speed and low power consumption is one of the most important design objectives in integrated circuits. Digital multipliers are most critical functional units. The overall performance of this system depends on the throughput of multiplier design. Aging problem of transistors has a significant effect on performance of these 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. The proposed architecture can be applied to image multiplication. Based on the idea of razor flip flop and adaptive hold logic the timing violations are reduced. In the fixed latency usage of clock cycles is increased. The re-execution of clock cycles is reduced by using variable latency.

Index Terms: Adaptive Hold Logic (AHL), Negative Bias Temperature Instability (NBTI), Positive Bias Temperature Instability (PBTI), reliable multiplier, variable latency.

1. 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 throughput 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 (V gs = V dd). In this situation, the interaction between inversion layer holes and hydrogen-passivated Si atoms breaks he SiH bond generated during the oxidation process, generating H or H 2 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 (V th), 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 V th is increased in the long term. Hence, it is important to design a reliable high-performance multiplier.
In CMOS circuits, NBTI effects occur in p-type transistors when a logic 0 is applied to the gate terminal (gate-to-source voltage V gs = V dd , i.e., negative bias). Under this condition, called the stress state, the magnitude of the threshold voltage (V th) increases over time, resulting in a degradation of the drive current. In contrast, when a logic 1 is applied to the gate terminal (V gs = 0), NBTI stress is actually removed. The latter condition, called the recovery state, induces a progressive yet partial recovery of the V th.

The impact of NBTI in random logic manifests itself as an increase of the propagation delay. Increased V th reduces in fact the drive current of individual gates, which thus require more time to propagate the input signals, and might then no longer meet the timing constraints [5]. To overcome this issue, a number of recent works proposed various design techniques to compensate and/or tolerate such NBTI-induced effects. The basic idea behind these approaches is to identify the gates which are responsible for the overall performance degradation (i.e., critical cells) and selectively apply gate/transistor resizing to guarantee larger design margin and/or circuit transformation (like NBTI-aware synthesis and technology mapping) to maximize the use of the standard cells that show smaller NBTI sensitivity. Different from these static design-time solutions, the authors of propose the use of adaptive techniques to dynamically compensate for NBTI effects during the lifetime of the circuit. Such solutions are effective because the total delay degradation of a circuit is usually significantly smaller than that of individual devices, on the order of a few percent per year of operation. This is mainly due to the fact that the propagation delay of a circuit consists of the sum of rising and falling transitions, and NBTI only affects rising transitions. Moreover, the stress time of some cell may be extremely low due to the logical structure of the circuit, thus masking the aging effect of 0 values.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 9, September 2017

Nevertheless, there exist special devices for which the stress time is dominant, which thus represent a concentrated source of timing failure. For these components, even a minimal variation on the electrical parameters drastically impacts the performance of the overall circuit. The sleep transistors used in power-gated architectures are a significant example of such devices: Due to accumulated active periods (i.e., when the circuit is not idle), NBTI effects induce a progressive increase of the channel resistance of the sleep transistors, which, in turn, causes a larger IR drop between the real V dd and the virtual V dd (at which the gated cells are connected). As a final result, the propagation delay of all the gated cells suffers from a sensible increase, no matter if it is a rising or a falling transition, independently of the stress time of the local input signals.

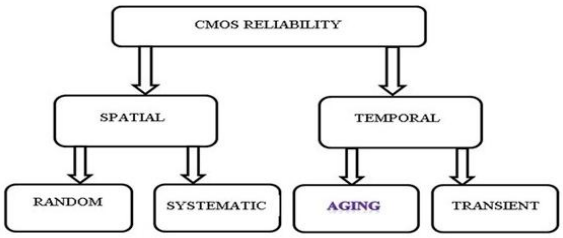

Figure A: CMOS Reliability

Multiplication is an essential arithmetic operation for common DSP applications, such as filtering and fast Fourier transform (FFT). To achieve high execution speed, parallel array multipliers are widely used. These multipliers tend to consume most of the power in DSP computations, and thus power-efficient multipliers are very important for the design of low power DSP systems. If the multipliers are too slow, the performance of entire circuits will be reduced.

2. LITERATURE SURVEY

` Multiplication is an basic arithmetic operation for common DSP applications, for example, filtering and fast Fourier transform (FFT). To accomplish high execution speed, parallel array multipliers are broadly utilized. These multipliers tend to devour most of the power in DSP calculations, and along these power-efficient multipliers are very important for the design of low-power DSP systems. If the multipliers are too slow, the execution of whole circuits will be minimize. Besides, negative bias temperature instability (NBTI) occurs when a pMOS transistor is under negative bias (V gs = V dd). In this condition, the interaction between inversion layer holes and hydrogen-passivated Si atoms breaks he SiH bond created during the oxidation process and producing H or H 2 molecules. At the point when these molecules diffuse away, interface traps are cleared out. The accumulated interface traps between silicon and the gate oxide interface result in increased expanded limit voltage (V th) and reducing the circuit switching speed. At the point When the biased voltage is removed, the reverse reaction happens, decreasing the NBTI impact.

a. Literature review presents several techniques for the reliability of digital circuits, A. Calimera, E. Macii, and M. Poncino, examines While in negative bias temperature instability (NBTI) effects on logic gates are of major concern for the reliability of digital circuits, they become even more critical when considering the components for which even minimal parametric variations impact the lifetime of the overall circuit. pMOS header transistors used in power-gated architectures are one relevant example of such components. For these types of devices, an NBTI-induced current capability degradation translates into a larger IR-drop effect on the virtual Vdd rail, which unconditionally affects the performance and, thus, the reliability of all power-gated cells. In this brief, they address the problem of designing NBTI-tolerant power-gating architectures. It propose a set of efficient NBTI-aware circuit design solutions, including both static and dynamic strategies, that aim at improving the lifetime stability of power-gated circuits by means of over sizing, body biasing, and stress-probability reduction while minimizing the design overheads. Experimental results prove the effectiveness of such techniques when applied to a suite of benchmarks mapped onto a 45-nm industrial CMOS technology library. In particular, we prove that it is possible to achieve more than ten times of lifetime extension with respect to a traditional power-gating approach.

b. In this paper, we show that the traditionally designed hold logic may be inaccurate. They use the short path activation conditions to obtain more accurate hold logic and improve the efficiency of telescopic units, Y.-S. Su, D.-C. Wang, S.-C. Chang, and M. Marek-Sadowska. In many designs, the worst-case delay of a critical path may be activated infrequently. Traditional optimization approaches assume the worst-case conditions, which could lead to an inefficient resource usage. It is possible to improve the throughput of such designs by introducing variable latency. One existing realization of the variable-latency design style is based on telescopic units. The design of the hold logic in telescopic units influences the circuits throughput.. In this paper, we show that the traditionally designed hold logic may be inaccurate. They use the short path activation conditions to obtain more accurate hold logic and improve the efficiency

of telescopic units. To reduce the overhead for large circuits, we propose an efficient heuristic methodology of constructing non-exact hold logic. We also discuss how to choose the telescopic units timing constraint. On average, our approach achieves the performance gain of 21.67/100 compared to 13.99/100, reported in the previous work.

c. A comparative study of NBTI and PBTI (charge trapping) in SiO2/HfO2 stacks with FUSI, TiN, Re gates,S. Zafar et al,shows Threshold voltage (Vt) of a field effect transistor (FET) is observed to shift with stressing time and this stress induced V t shift is an important transistor reliability issue. Vt shifts that occur under negative gate bias is referred as NBTI and those that occur under positive bias is referred as PBTI or charge trapping. In this paper, we present a comparative study of NBTI and PBTI for a variety of FETs with different dielectric stacks and gate materials. The study has two parts. In part I, NBTI and PBTI measurements are performed for FUSI NiSi gated FETs with SiO2 SiO2/HfO2 and SiO2/HfSiO as gate dielectric stacks and the results are compared with those for conventional SiON/poly-Si FETs. The main results are: (i) NBTI for SiO 2/NiSi and SiO2/HfO2/NiSi are same as those conventional SiON/polySi FETs; (ii) PBTI significantly increases as the Hf content in the high K layer is increased; and (iii) PBTI is a greater reliability issue than NBTI for HfO2/NiSi FETs. In part II of the study, NBTI and PBTI measurements are performed for SiO2/HfO2 devices with TiN and Re as gates and the results are compared with those for NiSi gated FETs. The main results are: (i) NBTI for SiO 2/HfO2/TiN and SiO2/HfO2/Re pFETs are similar with those observed for NiSi gated pFETs; and (ii) PBTI in TiN and Re gated HfO2 devices is much smaller than those observed for SiO2/HfO2/NiSi. In summary for SiO2/HfO2 stacks, NBTI is observed to be independent of gate material whereas PBTI is significantly worse for FUSI gated devices. Consequently, HfO2 FETs with TiN and Re gates exhibit over all superior transistor reliability characteristics in comparison to HfO2/FUSI FET.

d. To identify the Impacts of NBTI/PBTI on Timing Control Circuits and Degradation Tolerant Design in Nanoscale CMOS SRAM, -I. Yang, ShyhChyi Yang, Wei Hwang, and Ching-Te Chuang, They investigate the impacts of NBTI and PBTI on the stability and WM of a two-port 8T SRAM design. They then provide comprehensive analyses on the degradations of SRAM Read access and Write performance with hierarchical bit-line and Read/Write replica timing control circuits (Fig. 1). They show that because the Read/Write replica timing control circuits are activated in every Read- /Write cycle, they exhibit distinctively different degradation behavior from the normal array access paths, resulting in degradation of timing control and performance. We also show that Read access degradation dominates the SRAM performance degradation, and Write Half-Selectstability degradation dominates the stability of 8T SRAM cell. Further, we show that raising the Standby Virtual Ground voltage of the 8T cells Read buffer mitigates the Read performance degradation, and data-retention power-gating techniques and dual- 8T cell can be used tomitigate the stability degradation of bit cells. Hierarchical Read/Write scheme can improve the efficiency of these techniques.

e. For a low-power row-bypassing multiplier the addition operations in the j-th row can be disabled to reduce the power dissipation, J. Ohban, V. G. Moshnyaga, and K. Inoue, a low-power row-bypassing multiplier the addition operations in the j-th row can be disabled to reduce the power dissipation if the bit bj in the multiplier is 0. In the multiplier design, each modified FA in the CSA array is attached by three tri-state buffers and two 2-to-1 multiplexers. Because the addition operations of the rightmost FAs in the CSA rows are able to be bypassed, the extra correcting circuits must be added to correct the multiplication result.

f. Razor: A low-power pipeline based on circuit-level timing speculation,D. Ernst et al., have designed a combination of circuit and architectural techniques for low cost in-situ error detection and correction of delay failures. At the circuit level, each delay-critical flipflop is augmented with a so-called shadow latch which is controlled using a delayed clock. The operating voltage is constrained such that the worst-case delay is guaranteed to meet the shadow latch setup time, even though the main flipflop could fail. By comparing the values latched by the flipflop and the shadow latch, a delay error in the main flip-flop is detected. The value in the shadow latch, which is guaranteed to be correct, is then utilized to correct the delay failure. We present several architectural solutions for error correction, ranging from simple clock gating to more sophisticated mechanisms that augment the existing mispeculation recovery infrastructure. The proposed Razor technique was implemented in a prototype 64-bit Alpha processor design. This prototype implementation was used to obtain a realistic prediction of the power overhead for in-situ error correction and detection. We also studied the error-rate trends for data path components using both circuit-level simulation as well as silicon measurements of a full-custom multiplier block. Architectural simulations were then performed to analyze the overall throughput and power characteristics of Razor based DVS for different benchmark test programs. We demonstrate that on average, Razor reduced simulated power consumption by more than 40, compared to traditional design-time DVS and delay chain based approaches.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 9, September 2017

3. PROPOSED METHODOLOGY

The overall flow of our proposed architecture is as follows:

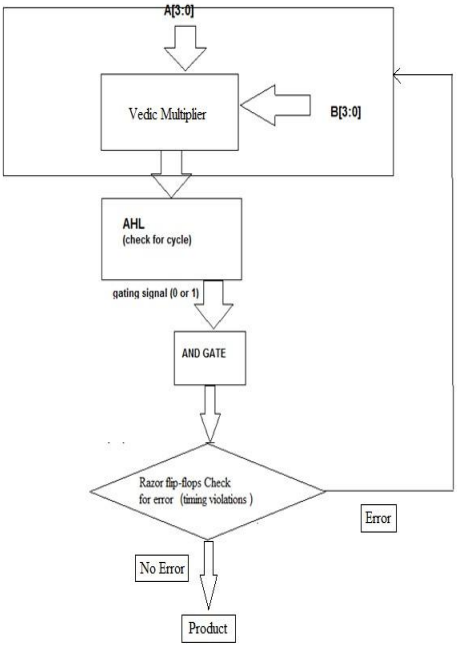

Figure B: Overall flow of Aging-Aware Multiplier

When input patterns arrive, the Vedic multiplier, and the AHL circuit execute simultaneously. According to the number of zeros in the multiplicand (multiplicator), 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 Vedic 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 re executed using two cycles to ensure the operation is correct.

In this situation, 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.

Design and Implementation of Proposed System

Low Power Vedic Multiplier with Adaptive Hold- logic:

The basic block diagram of low-power Vedic multiplier with Adaptive Hold Logic (AHL) is shown in Figure 3.1. It consists of one Vedic multiplier, 1-bit Razor flip-flop and an Adaptive Hold Logic (AHL) circuit.

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.

Razor flip flop:

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 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,

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 9, September 2017

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 reexecute 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 reexecuted with two cycles. Although the reexecution may seem costly, the overall cost is low because the reexecution frequency is low. More details for the Razor flip-flop can be found in.

Error detection and correction:

In order to detect an error at the circuit level, each flip-flop is augmented by a shadow flip-flop, which is clocked by a delayed clock. If the combinational logic meets the setup time of the main flip-flop, then the main and delayed flip-flops will latch the same value. In this case, the error signal remains low. If the setup time of the main flip-flop is not met, then the main flip-flop will latch a value that is different from the shadow flip-flop. To guarantee that the shadow flip-flop always latches the input data correctly, the input voltage is constrained such that under the worst-case condition, the logic delay does not exceed the shadow flip-flops setup time. The circuit is shown in Figure C.

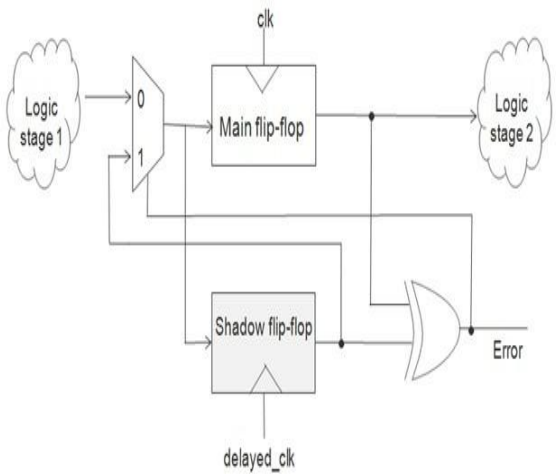

Fig C. Razor Flip Flop for Error detection and correction

The operation of the error detection circuitry is illustrated in the timing diagram in Figure. In the first clock cycle, input data D1 meets the setup time of the main flip-flop and shadow flip-flop, thus both flip-flops latch D1. But in the second clock cycle, the input data D2 does not satisfy the setup time requirement of the main flip-flop. It latches the old data D1 while the shadow latch latches D2. This is detected by the XOR circuitry and the error signal is asserted.

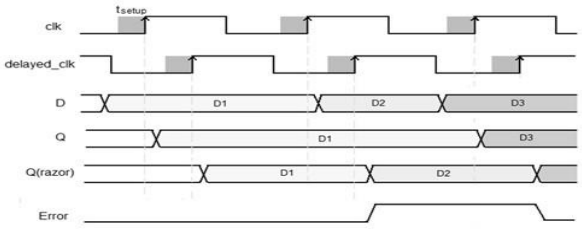

Fig D. Razor Flip flop operation

The asserted error signal is used in error correction, where it triggers the correct output value from the shadow latch to be restored to the main flip-flop in the subsequent cycle. This value is then available to the next pipeline stage.

Circuit-level implementation issues:

The power overhead of the error detection and correction circuitry must be kept minimal in order to reap the rewards of the Razor approach. Otherwise the power gains obtained by reducing supply voltage would be cancelled out by the power overheads. One way to reduce the power overhead is by selectively replacing the flip-flops with Razor flip-flops. If the maximum delay at the input of a flip-flop is guaranteed to meet the required time, then that flip-flop need not be replaced.

While the circuit is running at subcritical voltage, there is the danger of metastability. For instance, the input of the main latch may transition at the same time as the rising clock edge. Since the minimum critical voltage is constrained

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 9, September 2017

such that the setup time of the shadow flip-flop is always met, the shadow flip-flop will be stable. Since the main flip-flop feeds the XOR gate to generate the error signal, this signal would not be reliable in case of metastability.

4. SOFTWARE SPECIFICATIONS: TECHNOLOGY AND ASSOCIATED PLATFORM

Xilinx ISE 14.7 :

Xilinx ISE (Integrated Synthesis Environment) is a software tool produced by Xilinx for synthesis and analysis of HDL designs, enabling the developer to synthesize (”compile”) their designs, perform timing analysis, examine RTL diagrams, simulate a design’s reaction to different stimuli, and configure the target device with the programmer.
Xilinx ISE is a design environment for FPGA products from Xilinx, and is tightly-coupled to the architecture of such chips, and cannot be used with FPGA products from other vendors.[3] The Xilinx ISE is primarily used for circuit synthesis and design, while ISIM or the ModelSim logic simulator is used for system-level testing. Other components shipped with the Xilinx ISE include the Embedded Development Kit (EDK), a Software Development Kit (SDK) and ChipScope Pro.

User Interface:

The primary user interface of the ISE is the Project Navigator, which includes the design hierarchy (Sources), a source code editor (Workplace), an output console (Transcript), and a processes tree (Processes). The Design hierarchy consists of design files (modules), whose dependencies are interpreted by the ISE and displayed as a tree structure. For single-chip designs there may be one main module, with other modules included by the main module, similar to the main() subroutine in C++ programs. Design constraints are specified in modules, which include pin configuration and mapping. The Processes hierarchy describes the operations that the ISE will perform on the currently active module. The hierarchy includes compilation functions, their dependency functions, and other utilities. The window also denotes issues or errors that arise with each function. The Transcript window provides status of currently running operations, and informs engineers on design issues. Such issues may be filtered to show Warnings, Errors, or both.

Simulation:

System-level testing may be performed with ISIM or the ModelSim logic simulator, and such test programs must also be written in HDL languages. Testbench programs may include simulated input signal waveforms, or monitors which observe and verify the outputs of the device under test.

ModelSim or ISIM may be used to perform the following types of simulations:

Logical verification, to ensure the module produces expected results.

Behavioral verification, to verify logical and timing issues.

Post-place route simulation, to verify behavior after placement of the module within the reconfigurable logic of the FPGA.

Synthesis:

Xilinx’s patented algorithms for synthesis allow designs to run up to 30Also, due to the increasing complexity of FPGA fabric, including memory blocks and I/O blocks, more complex synthesis algorithms were developed that separate unrelated modules into slices, reducing post-placement errors.

IP Cores are offered by Xilinx and other third-party vendors, to implement system-level functions such as digital signal processing (DSP), bus interfaces, networking protocols, image processing, embedded processors, and peripherals. Xilinx has been instrumental in shifting designs from ASIC-based implementation to FPGA-based implementation.

Architecture Support and Requirements:

Operating Systems:
Xilinx only supports the following operating systems on x86 and x86-64 processor architectures.

Microsoft Windows Support:

1. Windows XP Professional (32-bit and 64-bit), English/Japanese.
2. Windows 7 Professional (32-bit and 64-bit), English/Japanese
3. Windows Server 2008 (64-bit)

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 9, September 2017

Linux Support:

1. Red Hat Enterprise Workstation 5 (32-bit and 64-bit),
2. Red Hat Enterprise Workstation 6 (32-bit and 64-bit),
3. SUSE Linux Enterprise 11 (32-bit and 64-bit)

5. HARDWARE SPECIFICATIONS

FPGA Spartan 3

The Spartan-3 family of Field-Programmable Gate Arrays is specifically designed to meet the needs of high volume, cost-sensitive consumer electronic applications. The eight-member family offers densities ranging from 50,000 to 5,000,000 system gates, The Spartan-3 family builds on the success of the earlier Spartan-IIE family by increasing the amount of logic resources, the capacity of internal RAM, the total number of I/Os, and the overall level of performance as well as by improving clock management functions. Numerous enhancements derive from the Virtex-II platform technology. These Spartan-3 FPGA enhancements, combined with advanced process technology, deliver more functionality and bandwidth per dollar than was previously possible, setting new standards in the programmable logic industry. Because of their exceptionally low cost, Spartan-3 FPGAs are ideally suited to a wide range of consumer electronics applications; including broadband access, home networking, display/projection and digital television equipment. The Spartan-3 family is a superior alternative to mask programmed ASICs. FPGAs avoid the high initial cost, the lengthy development cycles, and the inherent inflexibility of conventional ASICs. Also, FPGA programmability permits design upgrades in the field with no hardware replacement necessary, an impossibility with ASICs.

Features:

Low-cost, high-performance logic solution for high-volume, consumer-oriented applications,

1. Densities up to 74,880 logic cells
2. Up to 633 I/O pins,
3. 622+ Mb/s data transfer rate per I/O
4. 3. 18 single-ended signal standards,
5. 8 differential I/O standards including LVDS, RSDS,
6. Termination by Digitally Controlled Impedance,
7. Signal swing ranging from 1.14V to 3.465V,
8. Double Data Rate (DDR) support,
9. DDR, DDR2 SDRAM support up to 333 Mb/s.

Logic resources:

1. Abundant logic cells with shift register capability,
2. Wide, fast multiplexers,
3. Fast look-ahead carry logic,
4. Dedicated 18 x 18 multipliers,
5. JTAG logic compatible with IEEE 1149.1/1532.

Select RAM hierarchical memory :

1. Up to 1,872 Kbits of total block RAM,
2. Up to 520 Kbits of total distributed RAM,

Digital Clock Manager (up to four DCMs) :

1. Clock skew elimination,
2. Frequency synthesis,
3. High resolution phase shifting.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 9, September 2017

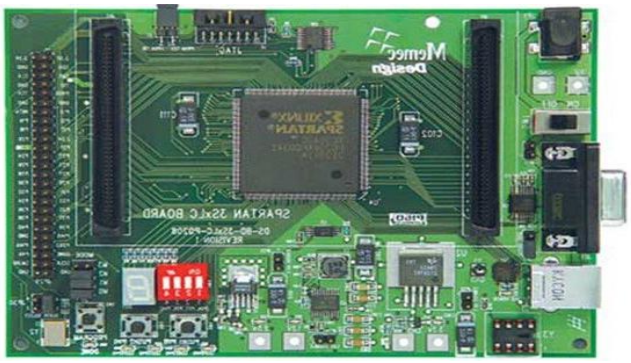

Fig. E. FPGA Sparten 3

6. CONCLUSION

This paper is a review on 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 proposed variable latency multipliers can be used under the influence of both the BTI effect and electromigration. In addition, it has 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 electromigration and use the worst case delay as the cycle period. We will analyze the behavioral simulation of this proposed multiplier by using Xilinx ISE simulator using Verilog HDL language.

This paper proposes aging aware circuit design which increases the reliability of the circuit. This aging aware was designed by adding additional circuit like Razor Flip-Flop and the AHL (Adaptive Hold Logic Circuit). In this paper we implemented the Vedic Multiplier architecture with aging aware circuit which occupies less area and less delay when we compared with the existing Multiplier.

REFERENCES

[1] 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. 249253, Apr. 2012.
[2] Y.-S. Su, D.-C. Wang, S.-C. Chang, and M. Marek-Sadowska, Performance optimization using variable-latency design style, IEEETrans. Very Large Scale Integr. (VLSI) Syst., vol. 19, no. 10, pp. 18741883, Oct. 2011.;
[3] 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. 2325.;
[4] Hao-I. Yang, Shyh-Chyi Yang, Wei Hwang, and Ching-Te Chuang Impacts of NBTI/PBTI on Timing Control Circuits and Degradation Tolerant Design in Nanoscale CMOS SRAM,IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMSI: REGULAR PAPERS, VOL. 58, NO. 6, JUNE 2011 1239.;
[5] J. Ohban, V. G. Moshnyaga, and K. Inoue, Multiplier energy reduction through bypassing of partial products, in Proc. APCCAS, 2002, pp. 1317.;
[6] 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. 718.;
[7] D. Mohapatra, G. Karakonstantis, and K. Roy, Low-power processvariation tolerant arithmetic units using input-based elastic clocking, in Proc. ACM/IEEE ISLPED, Aug. 2007, pp. 7479.;
[8] Y. Chen, H. Li, J. Li, and C.-K. Koh, Variable-latency adder (VLAdder): New arithmetic circuit design practice to overcome NBTI, in Proc. ACM/IEEE ISLPED, Aug. 2007, pp. 195200.;
[9] Y. Chen et al., Variable-latency adder (VL-Adder) designs for low power and NBTI tolerance, IEEE Trans. Very Large Scale Integr. (VLSI)Syst., vol. 18, no. 11, pp. 16211624, Nov. 2010.;
[10] M.-C. Wen, S.-J. Wang, and Y.-N. Lin, Low power parallel multiplier with Vedic, in Proc. IEEE ISCAS, May 2005, pp. 16381641.;