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, Special Issue 22, April 2016

A LOW POWER HYBRID MULTIPLIER DESIGN WITH HOLD LOGIC BY USING RAZOR LATCHES

POORNIMADEVI.P

P.G Scholar, ECE, AVS Engineering College, Salem, Tamil Nadu, India.

Poorniece04@gmail.com

***Abstract*— Digital multiplication is one of the most basic functions in a wide range of algorithms. The ubiquity of this operation in computing has given rise to a large number of multiplier implementations, each with different specifications and goals. Digital multiplication is used as opposed to analog when high precision is an issue; it is fairly straightforward to make digital multipliers as accurate as the application requires. Precision required for multiplication varies by function. The multiplier is a fairly large block of a computing system. Not only is the multiplier a high delay block, but it can be a significant source of power dissipation. Based on the argument delineated above, that several multipliers should be present on-chip as more DSP compute power is needed, the power dissipation involved in multiplication will become more dominant. Therefore, digital multipliers have become one of the prime circuits targeted for power reduction. Main aim of this proposal a low power hybrid multiplier design with hold logic by using Razor latches. 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 and to achieve reliable operation. Under the influence of NBTI and PBTI effects. Moreover, the proposed architecture can be applied to a column- or row-bypassing multiplier.**

***Index Terms*— Adaptive hold logic (AHL), negative bias temperature instability (NBTI), positive bias temperature instability (PBTI), reliable multiplier, variable latency**

I. INTRODUCTION

Digital multipliers are among the most critical arithmetic functional units in many applications, such as the Fourier transform, discrete cosine transforms, and digital filtering. The 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 (Vgs =−Vdd). In this situation, the interaction between inversion layer holes and hydrogen passivity Si atoms breaks the Si–H bond generated during the oxidation process, generating H or H2 molecules. When these molecules diffuse away, interface traps are left. The accumulated interface traps between silicon and the gate oxide interface

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, Special Issue 22, April 2016

result in increased threshold voltage (Vth), reducing the circuit switching speed. When the biased voltage is removed, the reverse reaction occurs, reducing the NBTI effect[1]-[2].

However, the reverse reaction does not eliminate all the interface traps generated during the stress phase, and Vth is increased in the long term. Hence, it is important to design a reliable high-performance multiplier. Christo Ananth et al. [5] proposed a system which can achieve a higher throughput and higher energy efficiency. The S-BOX is designed by using Advanced Encryption Standard (AES). The AES is a symmetric key standard for encryption and decryption of blocks of data. In encryption, the AES accepts a plaintext input, which is limited to 128 bits, and a key that can be specified to be 128 bits to generate the Cipher text. In decryption, the cipher text is converted to original one. By using this AES technique the original text is highly secured and the information is not broken by the intruder. From that, the design of S-BOX is used to protect the message and also achieve a high throughput, high energy efficiency and occupy less area.

A.Negative bias temperature instability and positive bias temperature instability

Negative-bias temperature instability (NBTI) is a key reliability issue in MOSFETs. NBTI manifests as an increase in the threshold voltage and consequent decrease in drain current and transconductance of a MOSFET. The degradation exhibits logarithmic dependence on time. It is of immediate concern in p-channel MOS devices, since they almost always operate with negative gate-to-source voltage; however, the very same mechanism also affects nMOS transistors when biased in the accumulation regime, i.e. with a negative bias applied to the gate.

In sub-micrometer devices nitrogen is incorporated into the silicon gate oxide to reduce the gate leakage current density and prevent boron penetration. However, incorporating nitrogen enhances NBTI.For new technologies (32 nm and shorter nominal channel lengths), high-K metal gate stacks are used as an alternative to improve the gate current density for a given equivalent oxide thickness (EOT). Even with the introduction of new materials like hafnium oxides, NBTI remains. It is possible that the interfacial layer composed of nitride silicon dioxide is responsible for those instabilities. This interfacial layer results from the spontaneous oxidation of the silicon substrate when the high-K dielectric is deposited. To limit this oxidation, the silicon interface is saturated with N resulting in a very thin and nitride oxide layer[3].

It is commonly accepted that two kinds of trap contribute to NBTI:

- ✓ First, interface traps are generated. Those traps cannot be recovered over a reasonable time of operation. Some authors refer to them as permanent traps. Those traps are the same as the one created by channel hot carrier. In the case of NBTI, it is believed that the electric field is able to break Si–H bonds located at the Silicon-oxide interface. H is released in the substrate where it migrates. The remaining dangling bond Si- (Pb center) contributes to the threshold voltage degradation.

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, Special Issue 22, April 2016

- ✓ On top of the interface states generation some preexisting traps located in the bulk of the dielectric (and supposedly nitrogen related), are filled with holes coming from the channel of pMOS.
- ✓ Those traps can be emptied when the stress voltage is removed. This Vth degradation can be recovered over time.

The existence of two coexisting mechanisms created a large controversy, with the main controversial point being about the recoverable aspect of interface traps. Some authors suggest that only interface traps are generated and recovered; today this hypothesis is ruled out. The situation is clearer but not completely solved. Some authors suggest that interface traps generation is responsible for whole trapping in the bulk of dielectrics. A tight coupling between two mechanisms may exist but nothing is demonstrated clearly. With the introduction of high K metal gates, a new degradation mechanism appeared. The PBTI for positive bias temperature instabilities affects nMOS transistor when positively biased. In this particular case, no interface states are generated and 100% of the Vth degradation may be recovered. Those results suggest that there is no need to have interface state generation to trapped carrier in the bulk of the dielectric.

B. Latency

Latency is a time interval between the stimulation and response, or, from a more general point of view, as a time delay between the cause and the effect of some physical change in the system being observed. Latency is physically a consequence of the limited velocity with which any physical interaction can propagate. This velocity is always lower than or equal to the speed of light. Therefore every physical system that has spatial dimensions different from zero will experience some sort of latency, regardless of the nature of stimulation that it has been exposed to. The precise definition of latency depends on the system being observed and the nature of stimulation.[4]. In communications, the lower limit of latency is determined by the medium being used for communications. In reliable two-way communication systems, latency limits the maximum rate that information can be transmitted, as there is often a limit on the amount of information that is "in-flight" at any one moment. In the field of human–machine interaction, perceptible latency has a strong effect on user satisfaction and usability.

C. Multiplier

A binary multiplier is an electronic circuit used in digital electronics, such as a computer, to multiply two binary numbers. It is built using binary adders. A variety of computer arithmetic techniques can be used to implement a digital multiplier. Most techniques involve computing a set of partial products, and then summing the partial products together. This process is similar to the method taught to primary schoolchildren for conducting long multiplication on base-10 integers, but has been modified here for application to a base-2 (binary) numeral system.

The corresponding effect on an nMOS transistor is positive bias temperature instability (PBTI), which occurs when an nMOS transistor is under positive bias.

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, Special Issue 22, April 2016

Compared with the NBTI effect, the PBTI effect is much smaller on oxide/polygate transistors, and therefore is usually ignored. However, for high-k/metal-gate nMOS transistors with significant charge trapping, the PBTI effect can no longer be ignored. In fact, it has been shown that the PBTI effect is more significant than the NBTI effect on 32-nm high-k/metal-gate processes. A traditional method to mitigate the aging effect is overdesign, including such things as guard-banding and gate over sizing; however, this approach can be very pessimistic and area and power inefficient. To avoid this problem, many NBTI-aware methodologies have been proposed.

An NBTI-aware technology mapping technique was proposed to guarantee the performance of the circuit during its lifetime. And, an NBTI-aware sleep transistor was designed to reduce the aging effects on pMOS sleep-transistors, and the lifetime stability of the power-gated circuits under consideration was improved. Wu and Marculescu proposed a joint logic restructuring and pin reordering method, which is based on detecting functional symmetries and transistor stacking effects.

They also proposed an NBTI optimization method that considered path sensitization. In dynamic voltage scaling and body-basing techniques were proposed to reduce power or extend circuit life. These techniques, however, require circuit modification or do not provide optimization of specific circuits. Traditional circuits use critical path delay as the overall circuit clock cycle in order to perform correctly. However, the probability that the critical paths are activated is low. In most cases, the path delay is shorter than the critical path. For these noncritical paths, using the critical path delay as the overall cycle period will result in significant timing waste[6].

Hence, the variable-latency design was proposed to reduce the timing waste of traditional circuits. The variable-latency design divides the circuit into two parts: 1) shorter paths and 2) longer paths. Shorter paths can execute correctly in one cycle, whereas longer paths need two cycles to execute. When shorter paths are activated frequently, the average latency of variable-latency designs is better than that of traditional designs. For example, several variable-latency adders were proposed using the speculation technique with error detection and recovery. A short path activation function algorithm was proposed to improve the accuracy of the hold logic and to optimize the performance of the variable-latency circuit. An instruction scheduling algorithm was proposed to schedule the operations on non uniform latency functional units and improve the performance of Very Long Instruction Word processors.

In a variable-latency pipelined multiplier architecture with a Booth algorithm was proposed.[7],[8]In process-variation tolerant architecture for arithmetic units was proposed, where the effect of process-variation is considered to increase the circuit yield. In addition, the critical paths are divided into two shorter paths that could be unequal and the clock cycle is set to the delay of the longer one.

These research designs were able to reduce the timing waste of traditional circuits to improve performance, but they did not consider the aging effect and could not adjust themselves during the runtime. A variable-latency adder design that considers

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, Special Issue 22, April 2016

the aging effect was proposed. However, no variable-latency multiplier design that considers the aging effect and can adjust dynamically has been done.

II. PRELIMINARIES

A. Normal array multiplier

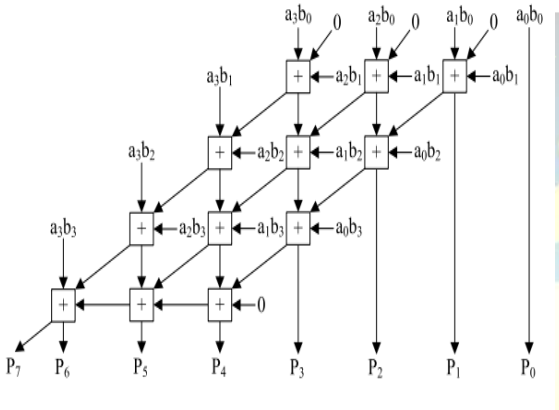

Fig. 1. 4 × 4 normal AM.

The AM is a fast parallel AM and is shown in Fig. 1. The multiplier array consists of (n−1) rows of carry save adder (CSA), in which each row contains (n−1) full adder (FA) cells. Each FA in the CSA array has two outputs: 1) the sum bit goes down and 2) the carry bit goes to the lower left FA. The last row is a ripple adder for carry propagation. 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.

B. Column-Bypassing Multiplier

A column-bypassing multiplier is an improvement on the normal array multiplier (AM). Fig. 2 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 a_ib_i. Therefore, the output of the adders in both diagonals is 0, and the output sum bit is simply equal to the third bit, which is the sum output of its upper FA.

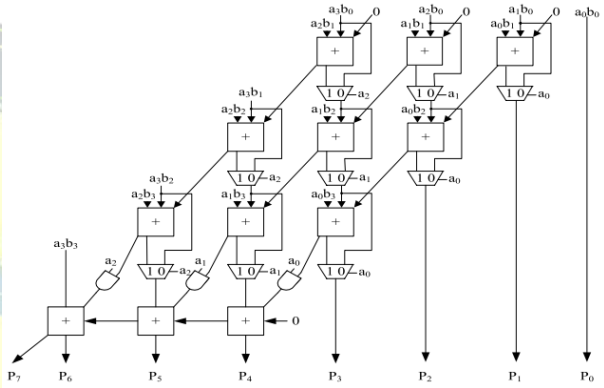

Fig. 2. 4 × 4 column – by passing multiplier.

Hence, the FA is modified to add two tristate gates and one multiplexer. The multiplicand bit a_i can be used as the selector of the multiplexer to decide the output of the FA, and a_i can also be used as the selector of the tristate gate to turn off the input path of the FA. If a_i 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 a_i is 1, the normal sum result is selected.

C. Row-Bypassing Multiplier

A low-power row-bypassing multiplier is also proposed to reduce the activity power of the AM. The operation of the low-power row-bypassing multiplier is similar to that of the low-power column-bypassing multiplier,

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, Special Issue 22, April 2016

but the selector of the multiplexers and the tristate gates use the multiplicator[9].

Fig. 3 is a 4 × 4 row-bypassing multiplier. Each input is connected to an FA through a tristate gate. When the inputs are 1111_2 * 1001_2, the two inputs in the first and second rows are 0 for FAs. Because b_1 is 0, the multiplexers in the first row select a_ib_0 as the sum bit and select 0 as the carry bit.

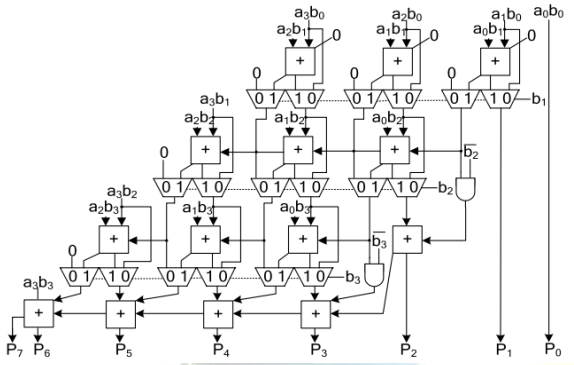

Fig. 3. 4 × 4 row – bypassing multiplier.

The inputs are bypassed to FAs in the second rows, and the tristate gates turn off the input paths to the FAs. Therefore, no switching activities occur in the first-row FAs; in return, power consumption is reduced. Similarly, because b_2 is 0, no switching activities will occur in the second-row FAs. However, the FAs must be active in the third row because the b_3 is not zero.

III. PROPOSED ARCHITECTURE

A.Proposed architecture

This section details the proposed aging-aware reliable multiplier design. It introduces the overall architecture and thefunctions of each component and also describes how to design AHL that adjusts the circuit when significant aging occurs.

The proposed aging-aware multiplier architecture in Fig. 4., which includes two m-bit inputs (m is a positive number), one 2m-bit output, hybrid multiplier [which includes either row multiplication or column multiplier], 2m1-bit Razor latch, and a modified AHL circuit.

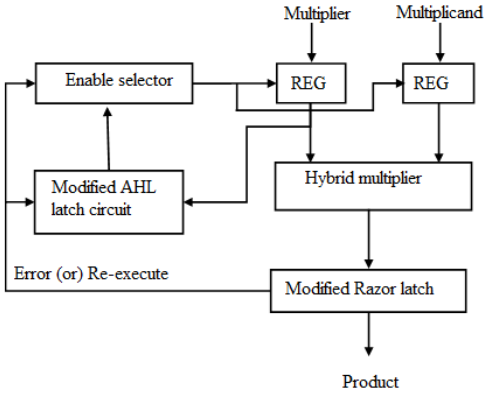

Fig. 4. Proposed architecture

In the proposed architecture, the hybrid 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. [10]Therefore, using the number of zero's or one's as the judging criteria results in similar outcomes.

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, Special Issue 22, April 2016

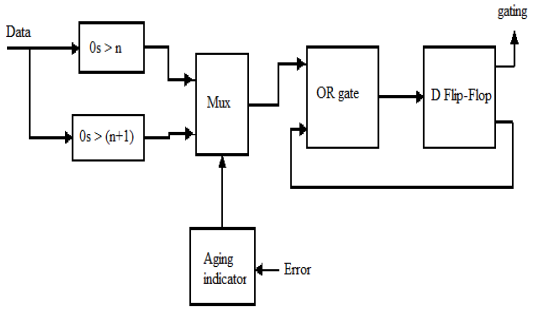

Fig. 5. Diagram of AHL

According to the bypassing selection in the hybrid bypassing multiplier, the input signal of the modified AHL in the architecture in Fig. 5. Shows that the column-bypassing multiplier is the multiplicand, whereas that of the row-bypassing multiplier is the multiplicator. Razor latch can be used to detect whether timing violations occur before the next input pattern arrives. If errors occur, the Razor latch 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[11][12].

B. Adaptive Hold Logic

The overall flow of proposed architecture is as follows: when input patterns arrive, the hybrid multiplier, and in Fig. 5.shows that 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 latch enable.

Otherwise, the AHL will output 1 for normal operations. When the hybrid bypassing multiplier finishes the operation, the result will be passed to the Razor latch. The Razor latch 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 latch 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, this proposed AHL circuit can accurately predict whether the input patterns require one or two cycles in most cases[13]-[14].

C. Razor latches

Fig. 6. shows the details of Razor latches. A 1-bit Razor latch contains a main latch , shadow latch, XOR gate, and mux. The main latch 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.

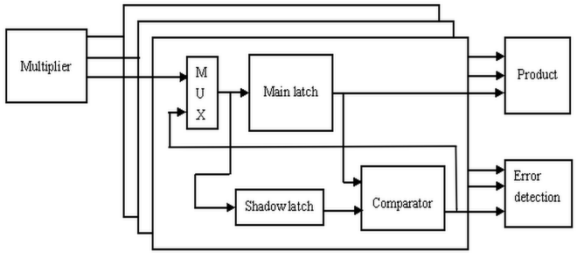

Fig. 6. Razor latches.

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, Special Issue 22, April 2016

If the latched bit of the shadow latch is different from that of the main latch , this means the path delay of the current operation exceeds the cycle period,[15] and the main flip-flop catches an incorrect result. If errors occur, the Razor latch 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.

Razor latches 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.

In existing system they use razor flip-flop instead of razor latches.But in proposed system :

- ✓ For modification using latch memory instead of flip flop memory. Because flip flop may consume more clock power ,when gating clock pulse increased. So, by using latch circuit.
- ✓ This can reduce overall circuit delay.
- ✓ As well as total power consumption.

IV. EXPERIMENTAL RESULT

Our experiments are conducted with the tools requires :

- ✓ Modelsim6.4c
- ✓ Xilinx 9.1/13.2

VHDL is a large and verbose language with many complex constructs that have complex semantic meaning. It is possible to quickly understand a subset of VHDL, which is both simple and easy to use. These are certain major capabilities that the languages provide along with the features that differentiate from other hardware description language.

Xilinx ISE 7.1i & Xilinx Platform Studio 7.1 are the programming tools for the system. This is used for generate RTL & FPGA schematic. This is used for create logic design like micro blaze design XPS includes a graphical user interface (GUI), along with a set of tools that aid in project design.

From the XPS GUI, you can design a complete embedded processor system for implementation within a Xilinx FPGA device. The XPS main window is shown in the figure below.

Note that the XPS main window is divided into three areas:

- ✓ The Project Information Panel
- ✓ The System Assembly Panel
- ✓ The Connectivity Panel

Under the tabular column shows the result of power consumption of multipliers :

Table I
Total estimated power consumption of normal 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, Special Issue 22, April 2016

Power summary:	I(mA)	P(mW)
Total estimated power consumption:		111
Vccint 1.20V:	41	49
Vccaux 2.50V:	20	50
Vcco25 2.50V:	5	12
Clocks:	1	1
Inputs:	1	1
Logic:	1	2
Outputs:		
Vcco25	5	12
Signals:	3	3
Quiescent Vccint 1.20V:	35	42
Quiescent Vccaux 2.50V:	20	50

Table II
Total estimated power consumption of row bypassing multiplier

Power summary:	I(mA)	P(mW)
Total estimated power consumption:		110
Vccint 1.20V:	40	48
Vccaux 2.50V:	20	50
Vcco25 2.50V:	5	12
Clocks:	1	1
Inputs:	1	1
Logic:	1	1
Outputs:		
Vcco25	5	12
Signals:	2	3
Quiescent Vccint 1.20V:	35	42
Quiescent Vccaux 2.50V:	20	50

Power summary:	I(mA)	P(mW)
Total estimated power consumption:		110
Vccint 1.20V:	40	48
Vccaux 2.50V:	20	50
Vcco25 2.50V:	5	12
Clocks:	1	1
Inputs:	1	1
Logic:	1	1
Outputs:		
Vcco25	5	12
Signals:	2	3
Quiescent Vccint 1.20V:	35	42
Quiescent Vccaux 2.50V:	20	50

By using the Xilinx tool for power consumption it shows in table I the total estimated power consumption of normal array multiplier is 111mW and in table II shows that the total estimated power consumption of row bypassing multiplier is 110mW and in table III shows that the total estimated power consumption of proposed architecture is 110mW .Comparing with existing multiplier system ,proposed multiplier system in proposed system consumed 10% of power is less than the existing systems.

Table III
Total estimated power consumption of proposed architecture

A working knowledge of the language in which your design and/or testbench is written (i.e., VHDL, Verilog, SystemC, etc.). Although ModelSim™ is an excellent tool to use while learning HDL concepts and practices, this document is not written to support that goal ModelSim creates a directory called *work* and writes a specially-formatted file named *_info* into that directory. The *_info* file must remain in the directory to distinguish it as a ModelSim library.

Do not edit the folder contents from your operating system; all changes should be made from within ModelSim. ModelSim also adds the library to the list in the Workspace and records the library mapping for future reference in the ModelSim initialization file (*modelsim.ini*).The results shows in the proposed architecture as follows:

The Wave window is one of several windows available for debugging. To see a listof the other debuggin windows, select the **View** menu. You may need to move or resize the windows to your liking. Window

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, Special Issue 22, April 2016

panes within the Main window can be zoomed to occupy the entire Main window or undocked to stand alone

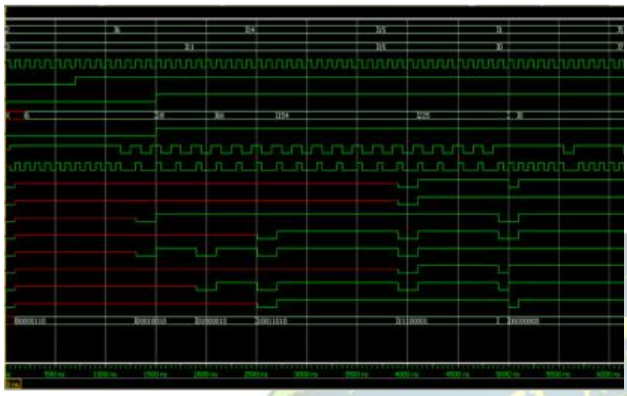

Fig.7.Wave window for proposed architecture.

V. CONCLUSION

Proposed variable latency multipliers have less performance degradation because variable latency multipliers have less timing waste and multiplier is able to adjust the AHL to mitigate performance degradation due to increased delay. By using latch memory instead of flip flop memory. Because flip flop may consume more clock power ,when gating clock pulse increased. So, by using latch circuit. This can reduce overall circuit delay. As well as total power consumption.The proposed architecture The multiplier is able to adjust the AHL to mitigate performance degradation due to increase delay.Improve performance, increase the speed of multipliers.,less timing waste,.reduce overall circuit delay and low power consumption of 10% less the existing multiplier systems .

REFERENCES

[1] K.-C. Wu and D. Marculescu, "Aging-aware timing analysis and optimization considering path sensitization," inProc. DATE, 2011, pp.

[2] D. Baneres, J. Cortadella, and M. Kishinevsky, "Variable-latency design by function speculation," inProc. DATE, 2009, pp. 1704–1709.

[3] K. Verma, P. Brisk, and P. Ienne, "Variable latency speculative addition: A new paradigm for arithmetic circuit design," inProc. DATE, 2008, pp.

[4] K. Du, P. Varman, and K. Mohanram, "High performance reliable variable latency carry select addition," in *Proc. DATE*, 2012,pp. 1257–1262

[5] Christo Ananth, H. Anusuya Baby, "S-Box using AES Technique", International Journal of Engineering Research & Technology (IJERT), Vol. 3 Issue 3, March – 2014, pp 285-290

[6] M. Basoglu, M. Orshansky, and M. Erez, "NBTI-aware DVFS: A new

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, Special Issue 22, April 2016

approach to saving energy and increasing processor lifetime," in *Proc.ACM/IEEE ISLPED*, Aug. 2010, pp. 253–258.

[7] A.Calimera, E. Macii, and M. Poncino, "Design techniqures for NBTItolerant power-gating architecture," *IEEE Trans. Circuits Syst., Exp.Briefs*,vol. 59, no. 4, 249–253, Apr. 2012.

[8] Y. Cao. (2013). *Predictive Technology Model (PTM)and NBTI Model* [Online]. Available:http://www.eas.asu.edu/□ptm.

[9] 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.

[10]N. V. Mujadiya, "Instruction scheduling on variable latency functional units of VLIW processors," in *Proc.ACM/IEEE ISED*, Dec. 2011, pp. 307–312.

[11] Y. Chen et al., (2010) "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. 1621–1624 .

[12] M.Basoglu,M.Orshansky,andM. Erez, (2011) "Impacts of NBTI/PBTI on timing control circuits and degradation tolerant design in nanoscale CMOS SRAM," IEEE Trans. Circuit Syst., vol. 58, no. 6, pp.1239–1251, Jun.

[13] D. Mohapatra, G. Karakonstantis, and K. Roy, (2007) "Low-power processvariation tolerant arithmetic units using input-based elastic clocking," in Proc. ACM/IEEE ISLPED, pp. 74–79.

[14] H. Abrishami, S. Hatami, B. Amelifard, and M. Pedram, (2008) "NBTI-aware flip-flop characterization and design," in Proc. 44th ACM GLSVLSI, , pp. 29–34.

[15] S. V. Kumar, C. H. Kim, and S. S.

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, Special Issue 22, April 2016

Sapatnekar,(2007) “NBTI-aware

synthesis of digital circuits”.