INTERNATIONAL JOURNAL
FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Column Bypassing Multiplier Implementation on FPGA

Arun A V[1], Adam Asif S[2], Aravindh R[3]

[1, 2, 3]Electronics and Communication Engineering, JEPPIAAR Engineering College, ANNA University: Chennai

Abstract: It is very important for modern DSP systems to develop low-power multipliers to reduce the power dissipation. This paper presents a new multiplier design in which switching activities are reduced through architecture optimization. The power consumption can be reduced if one can reduce the switching activity of a given logic circuit without changing its function. An obvious method to reduce the switching activity is to shut down the idle part of the circuit, Which is not in operating condition. In this the low powers Column bypasses multiplier design methodology that inserts more number of zeros in the multiplicand thereby reducing the number of switching activities as well as power consumption. The switching activity of the component used in the design 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. If multiplicand contains more zeros, higher power reduction can be achieved. To reduce the switching activity is to shut down the idle part of the circuit, which is not in operating condition.

I. NTRODUCTION

A. Introduction

The multipliers consume most Zof the power in DSP computations. Hence, it is very important for modern DSP systems to develop low-power multipliers to reduce the power dissipation. The basic idea of this work is to use new multiplier design in which switching activities are reduced through architecture optimization.

B. Need For Low Power Design

Increasing demand for portable electronics for computing and communication, as well as other applications, has necessitated longer battery life, lower weight, and lower power consumption. In order to satisfy these requirements, the project focus on low power/low voltage design techniques are underway. Since 'power' is now one of the design decision variables, the expanded design space required for low power has further increased the complexity of an already non-trivial task. Low power design basically involves two concomitant tasks: power estimation and analysis and power minimization. These tasks need to be carried out at each of the levels in the design hierarchy, namely, the behavioral, architectural, logic, circuit and physical levels. In this survey of the current state of the field, many of the salient power estimation and minimization techniques proposed for low power VLSI design are reviewed. The intel 4004 microprocessor, developed in 1971, had 2300 transistors, dissipated about 1 watts of power and clocked at 1 MHz. Then come the pentium in 2001, with 42 million transistors, dissipating around 65 watts of power and clocked at 2.40 GHz.

While the power dissipation increases linearly as the years go by, the power density increases exponentially, because of the ever-shrinking size of the integrated circuits. If this exponential rise in the power density were to increase continuously, microprocessor designed a few years later, would have the same power as that of the nuclear reactor. Such high power density introduces reliability concerns such as electro-migration, thermal stresses and hot carrier induced device degradation, resulting in the loss of performance. Another factor that fuels the need for low power chips is the increased market demand for portable consumer electronics powered by batteries. The craving for smaller, lighter and more durable electronic products indirectly translates to low power requirements. Battery life is becoming a product differentiator in many portable systems. Being the heaviest and biggest component in many portable systems, batteries have not experienced the similar rapid density growth compared to the electronic circuits. The main source of power dissipation in these high performance battery-portable digital systems running on batteries such as note-book computers, cellular phones and personal digital assistants are gaining prominence.

C. Power Optimization

The power dissipation in CMOS circuit has several components that are usually estimated on the device parameters of the technology used. The total power in the circuit is given by the following equation

$$P_{total} = P_{switching} + P_{shortcircuit} + P_{static} + P_{leakage}$$

where P switching is switching component of the power and it is a dominating component in these calculations short-circuits is the power dissipated due to the fact that during the circuit operation PMOS and NMOS transistors of CMOS gate become simultaneously during the transition at the input level, static consumption is from the leakage current for static power dissipation, the consumption is proportional to the number of the used transistors. For dynamic power dissipation, the consumption is obtained from the charging and discharging of load capacitance.

The average dynamic dissipation of a CMOS gate is given by average dynamic dissipation of a CMOS gate is given by

$$P_{avg} = 0.5*C_L*V_{dd}^2*F_b*N$$

Where CL is the load capacitance, fp is the clock frequency, VDD is the power supply voltage and N is the number of switching activity in a clock cycle.

D. Language And Tools Used

Xilinx designs, develops and markets programmable logic products including integrated circuits (ICs), software design tools, predefined system functions delivered as intellectual property (IP) cores, and technical support. Xilinx sells both FPGAs and CPLDs programmable logic devices for electronic equipment manufacturers in end markets such as communications, industrial, and data processing. The ISE design suite is the central electronic design automation (EDA) product family sold by xilinx.

Multipliers are among the main contributors of area and power consumption in a DSP system and, more importantly, they are usually placed in the critical paths of such systems. The ISE design suite features include design entry and synthesis supporting Verilog or VHDL, place-and-route (PAR), completed verification and debug using chip scope pro tools, and creation of the bit files that are used to configure the chip. This project uses XILINX ISE v 12.1 for our programming. Model synthesis map report all features in xilinx helped us a lot. . This project uses xilinx's XPower Estimator (XPE) tool in order to calculate power consumed in any arithmetic circuit. For calculation of power using xilinx's XPE we need to generate the map report file in XILINX which will be saved in the same directory at the extension of mrp.

II. LITERATURE SURVEY

A. Design of Low Power and High Speed 4X4 Multiplier using

Modified Column Bypassing Scheme for DSP Applications Author: E.Srinivas Year:2019

A low power and high speed 4X4 multiplier is designed using CMOS technology. The important factors in VLSI design are power, area, speed and design time. Now-a-days, power and speed has become a crucial factor in Digital Signal Processor (DSP) applications. However, different optimization techniques are available in the digital electronic world. The proposed approach a low power and high speed multiplier design based on modified column bypassing technique mainly used to reduce the switching power activity. While this technique offers great dynamic power savings, due to their interconnection. In this work, a low power and high speed multiplier with hybridization scheme is presented.

B. Fast and Power Efficient 16×16 Array of Array Multiplier using Vedic Multiplication

Author: Dr. K.S. Gurumurthy Year: 2018

"Array of Array" multiplier which is a derivative of braun array multiplier. Braun array are much suitable for VLSI implementation because of its less space complexity though it shows larger time complexity, on the other hand tree multipliers have time complexity of (log n) but are less suitable for VLSI implementation since, being less regular. They require larger total routing length, which leads to performance degradation. They show higher space complexity. The proposed algorithm is useful for math coprocessors in the field of computers. Algorithm is implemented on SPARTAN-3E FPGA.

C. A Design Approach for Mac Unit Using Vedic Multiplier

Author: Aditi Chhabra Year: 2020

Machine learning problems have been efficiently solved by using Artificial Neural Networks (ANNs). The realization of neural networks on hardware have been shown to provide more significant advantages. In digital neural networks, the weight-input multiplication is an important step. In this paper, a comparative study between different configurations of vedic multipliers and traditional array multipliers has been performed and further, the hardware implementation of the MAC unit has been performed using VHDL. MAC unit of ANN requires repetitive use of adders and multipliers. The aim behind the comparison is to obtain an alternative approach for the realization of the MAC unit of the neural network.

D. Low power multiplier design with Improved column by Passing Scheme

Author: Pankaj Kumar Year: 2019

Power, speed and area are prime design constraints for portable electronics devices and signal processing applications. Multiplier plays an important role in DSP applications. In this paper, a low power and high speed multiplier with improved column bypassing scheme is presented. Primary power reduction is obtained by disabling the supply voltage of non-functional blocks when the operands of the multiplicands are zero. Power reduction is achieved by both architecture and circuit level modifications. The proposed multiplier consists of new adder architecture which is also responsible for reducing the power consumption and propagation delay.

E. A Review of Different Type of Multipliers and Multiplier Accumulator Unit

Author: Suresh Kumar Year: 2020

High speed and low power MAC unit is utmost requirement of today's VLSI systems and digital signal processing applications like FFT, finite impulse response filters, convolution etc. This project discusses different types of multipliers like booth multiplier, combinational multiplier, wallace tree multiplier, array multiplier and sequential multiplier. Each multiplier has its own advantages and disadvantages. Different types of techniques are presented for improving the speed and low power consumption like pipelined booth multiplication technique in which pipelining is used in booth multiplier to reduce the delay of each stage.

F. Modified bypassing Multiplier for Power Efficient fir Filter

Author: Anita Danie Year: 2019

Low power consumption and smaller area are the most important criteria for the fabrication of DSP systems. Optimizing speed and power of the multiplier is a major design issue. However, speed and power are usual constraints conflicting to each other, so that increasing speed results in larger areas. Parallel multipliers like braun's multiplier are preferred over serial multipliers as they consume more power. In this paper, designed a low power FIR filter, by simplification of addition operation in a bypassing multiplier. The same has been implemented and the power dissipation is calculated. The effectiveness of the proposed technique is also proved by comparing the obtained results with the existing low power FIR filter design.

G. Design of Low Power Column bypass Multiplier using FPGA

Author: J. Sudha Rani Year: 2018

It is well known that multipliers consume most of the power in DSP computations. Hence, it is very important for modern DSP systems to develop low-power multipliers to reduce the power dissipation.. In this paper, presentslow power column bypass multiplier design methodology that inserts more number of zeros in the multiplicand thereby reducing the number of switching activities as well as power consumption. The switching activity of the componentused in the design 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. If multiplicand contains more zeros, higher power reduction can be achieved

H. Design of Bypassing – based Multipliers using   Ultra Low Power Technique

Author: U. Pradeep Kumar Year: 2020

Recently building low-power VLSI systems has emerged as highly in demand because of the fast growing technologies in mobile communication and signal processing. The battery technology does not advance at the same rate as the microelectronics technology. So designers are faced with more constraints: high speed, high throughput, small silicon area, and at the same time, low-power consumption. There are various types of multipliers available depending upon the application in which they are used. This paper uses a new CMOS logic style called gate diffusion input is used to address the issue of dynamic power dissipation which enables us to design complex functions with fewer gates thereby, reducing the power consumption of the circuit.

I. Modified Low Power and High Speed Row and Column Bypass Multiplier using FPGA

Author: Amala Maria Alex Year: 2019

The multiplier is the most critical arithmetic unit in many DSP applications such as digital filtering, fast fourier transform and discrete cosine transform. There is a need to design multipliers that reduce power dissipation and signal propagation delay. DSP applications like filters require faster calculations for updating their coefficients. Multipliers and their associated circuits (adders and accumulators) along with registers consume a significant portion of power for most of the DSP applications.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VI June 2022- Available at www.ijraset.com

J.   Design of Bypassing Multipier with Different Adders

Author: ManchalAhuja Year: 2018

Multiplication is one of the essential operations in Digital Signal Processing (DSP) applications like Fast Fourier Transform (FFT), digital filters etc. Multiplier is designed, considering the trade-offs between low power and high speed. The bypassing multiplier is an improvement, over braun multiplier which is one of the parallel array multiplier. The trade-offs dynamic power and delay of the bypassing multipliers can be reduced by using different adders. This paper presents a comparative study of 1-dimensional and 2-dimensional bypassing multipliers using different adders on basis of delay, area and power and for 4x4, 8x8 and 16x16 bits in FPGA spartan – 3E using xilinx 12.4 ISE and synopsis respectively.

III. PROPOSED SYSTEM

A.   Existing System

A simple and straightforward approach is to design a low-power FA to reduce the power dissipation of an array multiplier. The other approach is proposed to reduce the power dissipation of functional units by interchanging dynamic operands or using partially guarded computation. On the other hand, the reduction of the power dissipation can also be achieved through the architectural modification via row bypassing. The main idea of this design is based on the observation that most modern multipliers produce a large number of signal transitions while adding zero partial products. The design uses another way to transition activity optimization and that is hardware bypassing. Since adding zero partial products generate a large number of signal transitions in the carry-adder array without affecting the results and the additions are bypassed by disabling the adders. The transition activity of the multiplier is correlated with b, if the j-th bit of b is 0, the j-th row of adders does not need to be activated, and the corresponding partial product is 0. Shows the internal structure of the row-bypassing adder cell (RA).

The design included $(n-1)\times(n-1)$ full adders, $2\times(n-1)\times(n-1)$ multiplexers, and $3\times(n-1)\times(n-1)$ three state gates. When the corresponding partial product is zero, the RA disabled unnecessary transitions and bypassed the inputs to outputs. Two multiplexers augmented to the outputs of the adder transmit the input-carry bit and the input-sum bit of the previous addition to the outputs. The tri-state buffers placed at the input of the adder cells disable signal transitions in the adders which are bypassed, and then the input-carry bit and input-sum bit are passed to downwards. Thus, the power consumption can be reduced if one can reduce the switching activity of a given logic circuit without changing its function.
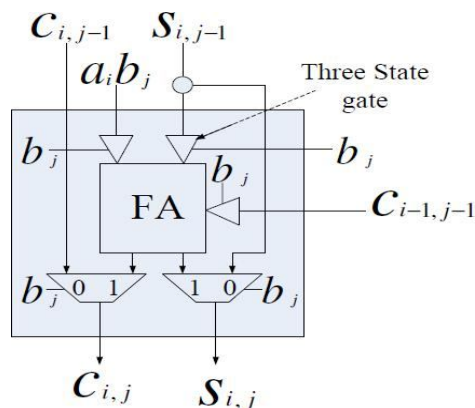

Figure 3.1 Row bypass adder cell

An obvious method to reduce the switching activity is to shut down the idle part of the circuit which is not in operating condition. As a result, the addition operations in the j-th row of CSAs is shown in figure 3.1. can be bypassed and the outputs from the (j-1)-th row of CSAs can be directly fed to the (j+1)-th row of CSAs without affecting the multiplication result. In the multiplier design, each modified FA in the CSA array is attached by 3 tri-state buffers and two 2- to-1 multiplexers.

1)   Disadvantage of Existing System

a)   The switching activity in row bypassing multiplier is less than that of braun multiplier.

b)   Extra circuitry than braun multiplier

B. Proposed System

Column-bypassing multiplier

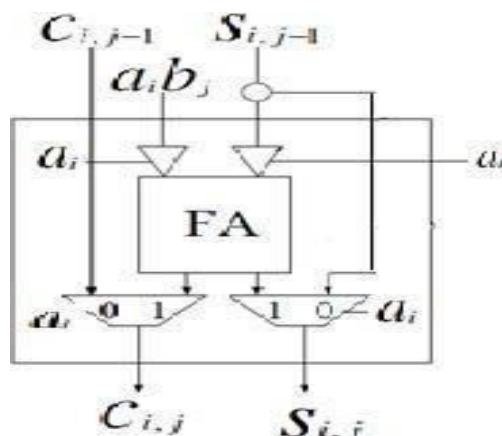Addition operation in a column is disabled. Addition operation in a row and column is disabled
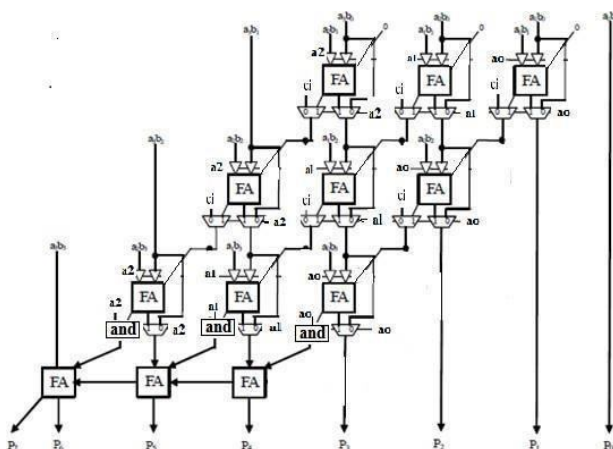

Figure 3.2 Column bypass adder cell

Figure 3.3 4*4 Braun multiplier with row bypassing

C. Low-Power Multiplier with Column by Passing

For a low-power column-bypassing multiplier, the addition operations in the (i+1)-th column can be bypassed if the bit, a_i, in the multiplicand is 0, i. e., all partial products $a_i b_j$, $0 \leq j \leq n-1$, are zero. In the multiplier design, the modified FA is simpler than that in the row bypassing multiplier. Each modified FA in the CSA array is only attached by two tri-state buffers and 2-to- 1multiplexer.

As the bit, a_i, in the multiplicand is 0, their inputs in the (i+1)-th column will be disabled and the carry output in the column must be set to be 0 to produce the correct output. Hence, the protecting process can be done b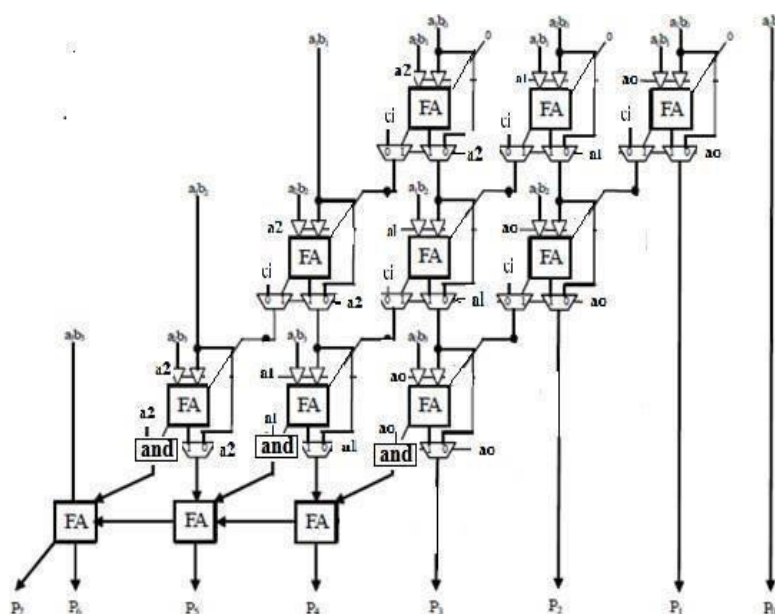y adding an AND gate at the outputs of the last row of CSAs. However, the multiplier design does not need the extra correcting circuit as illustrated in figure.3.4 In figure. 3.4, a 4X4 Braun multiplier with column by passing can be illustrate. figure 3.4 shows the internal structure of the Column-bypassing adder cell (RA), the tri-state buffers placed at the input of the adder cells, disable signal transitions in the adders which are bypassed, and then the input-sum bit are passed to downwards. When the corresponding partial product is zero, the CA disabled unnecessary transitions and bypassed the inputs to outputs. In other words, there are two bits to be added, and the output-carry bit must be zero, and the output-sum bit is equal to input-sum bit.

The operations in column i can be ignored and the adders can be disabled since the output are known. The design included (n-1)*(n-1) full adders (n-1)*(n-1) multiplexers, and 2*(n-1)*(n-1) three state gates. In comparison to row-bypassing multiplier, with the column-bypassing multiplier. Multiplier needs only two tri- state gate and two multiplexer, because the two input bit are disabled, and then three inputs are fixed, and the output-carry bit would not be changed. The column-bypassing multiplier reduced lthree state gates than the row-bypassing multiplier.

The demerit of row bypassing technique is that it needs extra correction circuitry shown in triangle. Structure of the full adder is complex as well. The Braun multiplier removes the extra correction circuitry needed. Also, number of adders is less. But, the limitation of this technique is that it cannot stop the switching activity even if the bit coefficient is zero that ultimately results in unnecessary power dissipation. In the multiplier design, the modified FA is simpler than that in the row bypassing multiplier. Each modified FA in the CSA array is only attached by two tri-state buffers and one 2-to-1 multiplexer. As the bit, a_i, in the multiplicand is 0,their inputs in the (i+1)-th column will be disabled and carry output in the column must be set to be multiplexer to produce correct output. Hence, protecting process can be done by adding an AND gate at the outputs of the last row of CSAs.
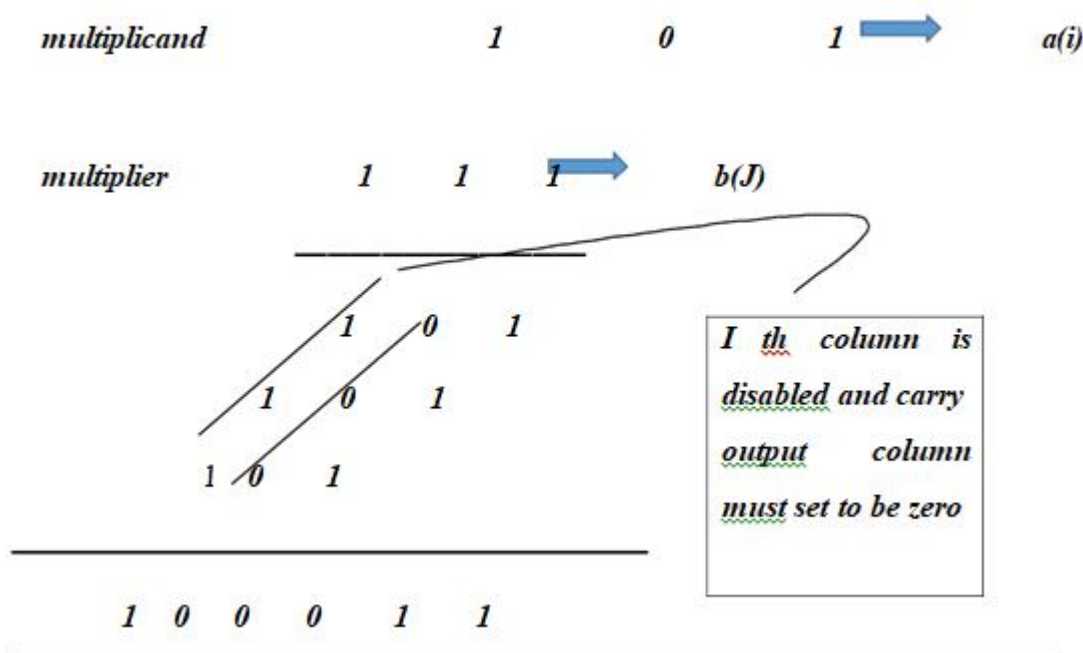

Figure 3.4 4*4 Braun multiplier with column bypassing

D. TRI –State BUFFER

Devices used in digital circuits operate by receiving and producing two logic state signals, a logic 1 and a logic 0 one type of devices that is used in digital circuits receives the two logic states but produces three different types of output signals. Because it generates three different signals, it is called a three- state buffer or tri-state buffer. A tri-state buffer has one or more inputs, one output, and a control input line, as shown below. The function of the control line is to enable the buffer to operate when an input signal is applied, or to be disabled when the input signal received.

Figure 3.5 Tristate-Buffer

E. Booth Recoding Unit

Since multiplicand contains no zero, all columns will get switched and consume more power. Higher power reduction can be achieved if the multiplicand contains more number of 0's than 1's. In this approach we propose booth recoding unit which will force multiplicand to have number of zeros, if does not have a single zero. The advantage here is that if multiplicand contains more successive number of one's then booth-recoding unit converts these ones in zeros. Booth's algorithm can be implemented by repeatedly adding (with ordinary unsigned binary addition) one of two predetermined values A and S to a product P, then performing a rightward arithmetic shift on P. Let **m** and **r** be the multiplicand and multiplier, respectively; and let x and y represent the number of bits in m and r.

1) Determine the values of A and S, and the initial value of P. All of these numbers should have a length equal to $(x + y + 1)$.
a) A: Fill the most significant (leftmost) bits with the value of m. Fill the remaining $(y + 1)$ bits with zeros.
b) S: Fill the most significant bits with the value of $(-m)$ in two's complement notation. Fill the remaining $(y + 1)$ bits with zeros.
c) P: Fill the most significant x bits with zeros. To the right of this, append the value of **r**. Fill the least significant (rightmost) bit with a zero.

2) Determine the two least significant (rightmost) bits of P.
a) If they are 01, find the value of $P + A$. Ignore any overflow.
b) If they are 10, find the value of $P + S$. Ignore any overflow.
c) If they are 00, do nothing. Use P directly in the next step.
d) If they are 11, do nothing. Use P directly in the next step.

3) Arithmetically shift the value obtained in the 2nd step by a single place to the right. Let P now equal this new value.
4) Repeat steps 2 and 3 until they have been done y times.
5) Drop the least significant (rightmost) bit from P. This is the product of m and r.
6) This unit chooses force the multiplicand to have greater number of zeros in case multiplicand does not have zero using booth table.

Table 3.1 Booth recoding table

Multiplicand Bit	Bit i-1	Version of multiplier Selected by bit i
0	0	0 X M
0	1	+1XM
1	0	-1X M
1	1	0 X M

F. Introduction To Verilog HDL

Verilog HDL is one of the two most common Hardware Description Languages (HDL) used by Integrated Circuit(IC) designers. The other one is VHDL. HDL's allows the design to be simulated earlier in the design cycle in order to correct errors or experiment with different architectures. Designs described in HDL are technology-independent, easy to design and debug and are usually more readable than schematics, particularly for large circuits. Verilog can be used to describe designs at four levels of abstraction:

1) Algorithmic level (much like c code with if, case and loop statements).
2) Register transfer level (RTL uses registers connected by Boolean equations).
3) Gate level (interconnected AND, NOR etc.).
4) Switch level (the switches are MOS transistors inside gates).

The language also defines constructs that can be used to control the input and output of simulation. More recently verilog is used as an input for synthesis programs which will generate a gate-level description (a netlist) for the circuit. Some Verilog constructs are not synthesizable. Also the way the code is written will greatly affect the size and speed of the synthesized circuit. Most readers will want to synthesize their circuits, so nonsyn the sizable constructs should be used only for test benches.
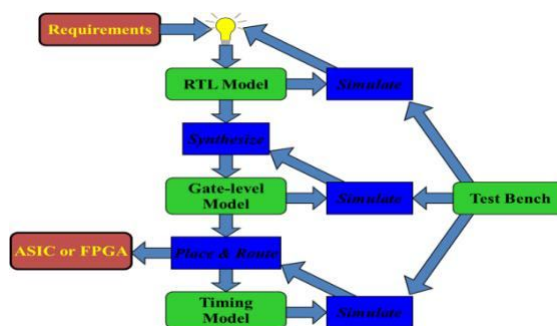
G. Basic Design Methodology

Figure:3.6 Basic Design Methodology

IV. RESULTS AND DISCUSSION

A. Simulate Environments

This project uses the following languages and tools to design and simulate our low- power multiplier design, the braun's design, the row-bypassing design, the column-by passing design, the 2-dimensional bypassing or mixed bypassing design and the proposed design are simulated in XILINX.

1) ISE Design Suite Overview

The ISE design suite is the Xilinx design environment, which allows you to take your design from design entry to xilinx device programming. With specific editions for logic, embedded processor or Digital Signal Processing (DSP) system designers, the ISE design suite provides an environment tailored to meet your specific design needs.

ISE design suite: Logic edition

This edition includes exclusive tools and technologies to help achieve optimal design results, including the following:

a) *Xilinx Synthesis Technology (XST)* - synthesizes VHDL, Verilog or mixed language designs.
b) *ISim* - enables you to perform functional and timing simulations for VHDL, Verilog and mixed VHDL/Verilog designs.
c) *PlanAhead™ software* - enables you to do advanced FPGA floor planning. The PlanAhead software includes I/O planner, an environment designed to help you to import or create the initial I/O port list, group the related ports into separate folders called "Interfaces" and assign them to package pins. I/O planner supports fully automatic pin placement or semi-automated interactive modes to allow controlled I/O port assignment.
d) *CORE Generator™ software* - provides an extensive library of xilinx LogiCORE™ IP from basic elements to complex system level IP cores.
e) *SmartGuide™ technology* - enables you to use results from a previous implementation to guide the next implementation for faster incremental implementation.

f) *Design Preservation* - enables you to use placement and routing for unchanged blocks from a previous implementation to reduce iterations in the timing closure phase.

g) *Partial Reconfiguration* - enables dynamic design modification of a configured FPGA. The ISE software uses partition technology to define and implement static and reconfigurable regions of the device. This feature requires an additional license code.

h) *XPower Analyzer* - enables you to analyze power consumption for xilinx FPGA and CPLD devices.

i) Power Optimization for Virtex®-6 devices - minimizes logic toggling to reduce dynamic power consumption.

j) *iMPACT* - enables you to directly configure xilinx FPGAs or program Xilinx CPLDs and PROMs with the xilinx cables. It also enables you to create programming files, read back and verify design configuration data, debug configuration problems, and execute SVF and XSVF files.

k) *ChipScope™ Pro tool* - assists with in-circuit verification.

2) ISIM Overview

Xilinx® ISim is a Hardware Description Language (HDL) simulator thatenables you to perform functional and timing simulations for VHDL, Verilog andmixed VHDL/Verilog designs.

The Xilinx® ISE software provides an integrated flow with the xilinx ISE simulator (ISim) that allows simulations to be launched directly from the xilinx project navigator. All simulation commands that prepare the ISim simulation are generated by ISE Project Navigator and automatically run in the background when simulating a design using this flow.

B. Software Requirements

a) ISE Web PACK™ 11, or

b) One of the ISE Design Suite 12.1 Editions (Logic, DSP, Embedded, System)

The ISim Graphical User Interface (GUI) contains the wave window, toolbars, panels and status bar. In the main window, you can view the simulation-visible parts of the design, add and view signals in the wave window, utilize ISim commands to run simulation, examine the design, and debug as necessary.

1) Virtex 5

Using the second generation ASMBL™ (Advanced Silicon Modular Block) column-based architecture, the Virtex-5 family contains five distinct platforms (sub-families), the most choice offered by any FPGA family. Each platform contains a different ratio of features to address the needs of a wide variety of advanced logic designs. In addition to the most advanced, high- performance logic fabric, virtex-5 FPGAs contain many hard-IP system level blocks, including powerful 36-K bit block RAM/FIFOs, second generation 25 x 18 DSP slices, SelectIO™ technology with built-in digitally-controlled impedance, ChipSync™ source-synchronous interface blocks. Additional platform dependant features include power-optimized high-speed serial transceiver blocks for enhanced serial connectivity, PCI Express® compliant integrated endpoint blocks, tri-mode Ethernet MACs (Media Access Controllers), and high-performance PowerPC® 440 microprocessor embedded blocks.

Figure 4.1 Virtex 5

International Journal for Research in Applied Science & Engineering Technology (IJRASET)

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VI June 2022- Available at www.ijraset.com
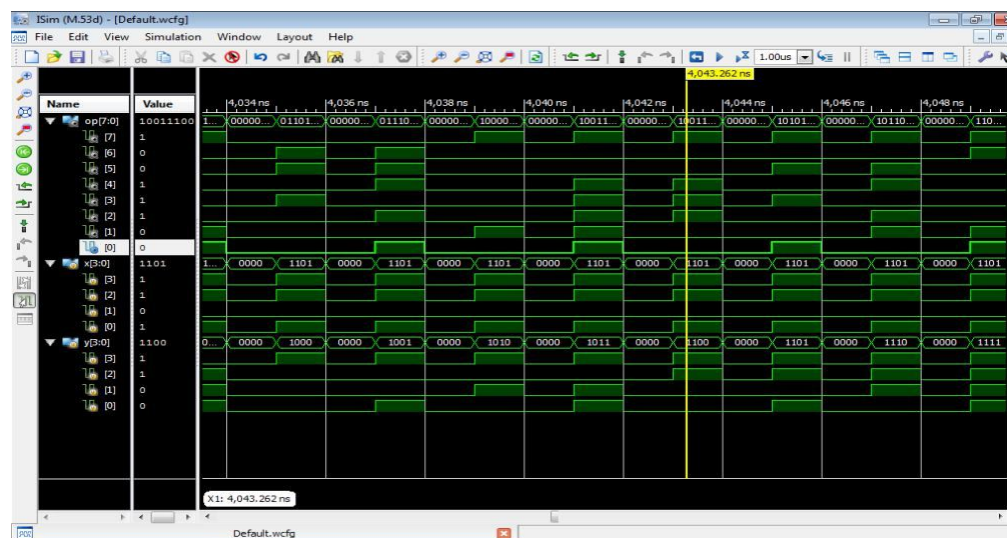
C. Simulation For Row Bypass
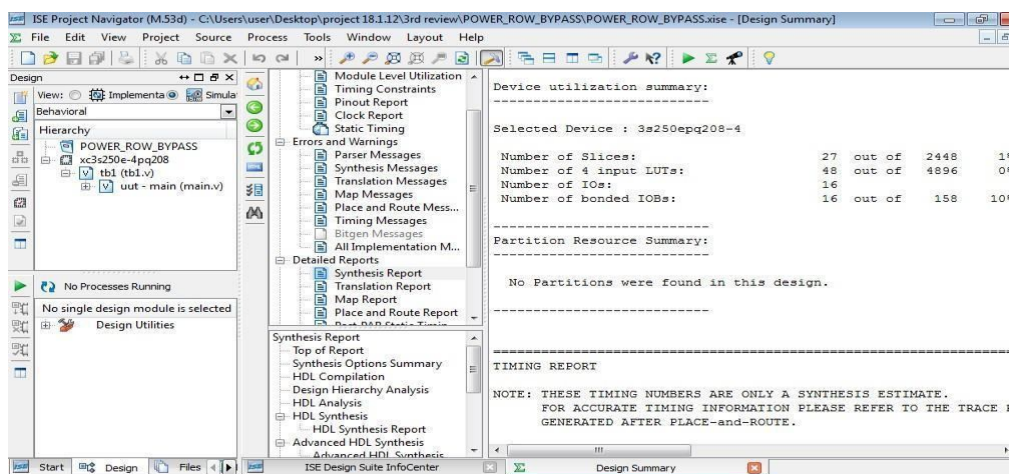

Figure 4.2 Output for row bypass
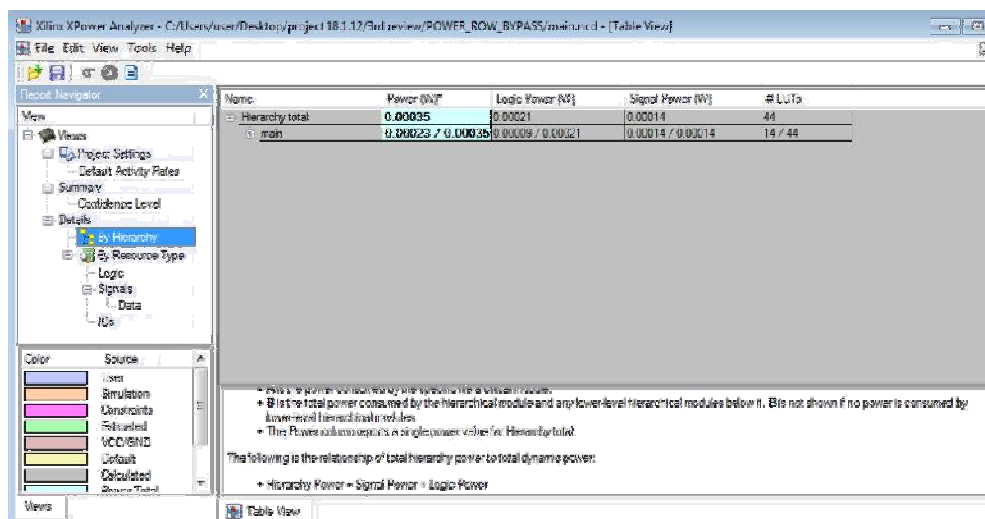

Figure 4.3 Area analysis for row bypass

Figure 4.4 Power analysis for row bypass
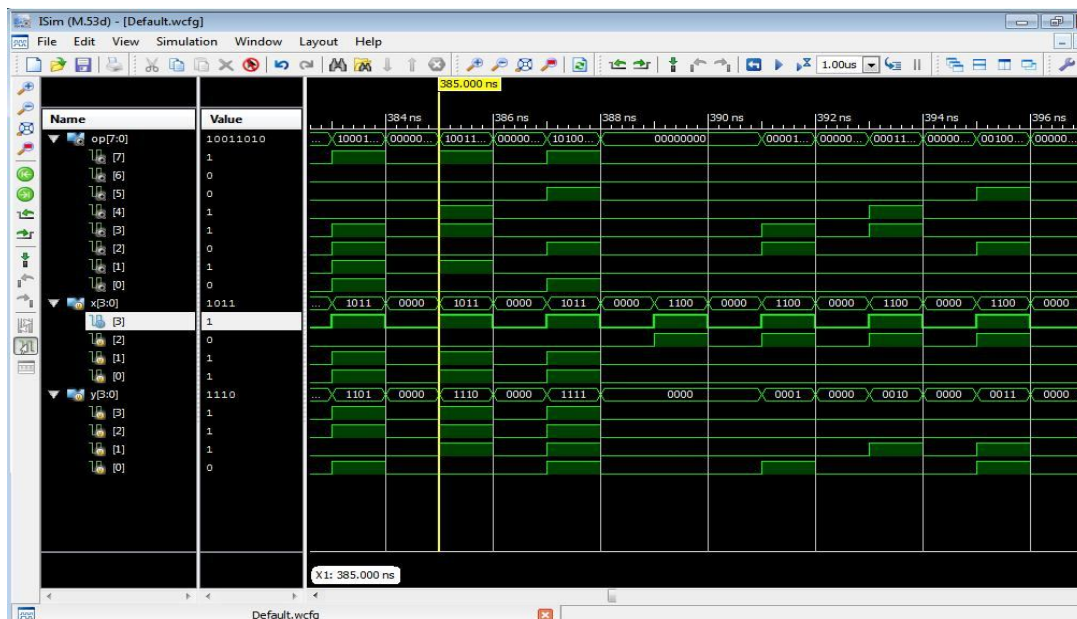
D. Simulation For N Mixed Bypass
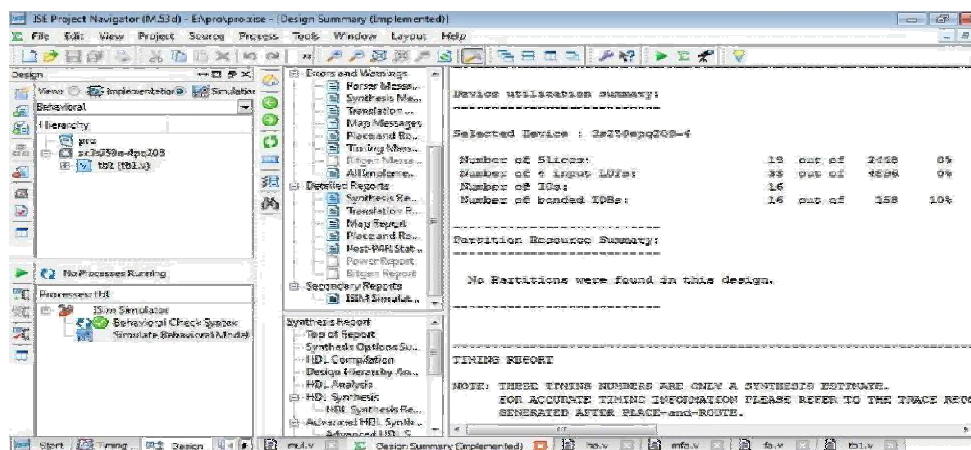

Figure 4.5 Output for N mixed bypass

Figure 4.6 Area analysis for N mixed bypass
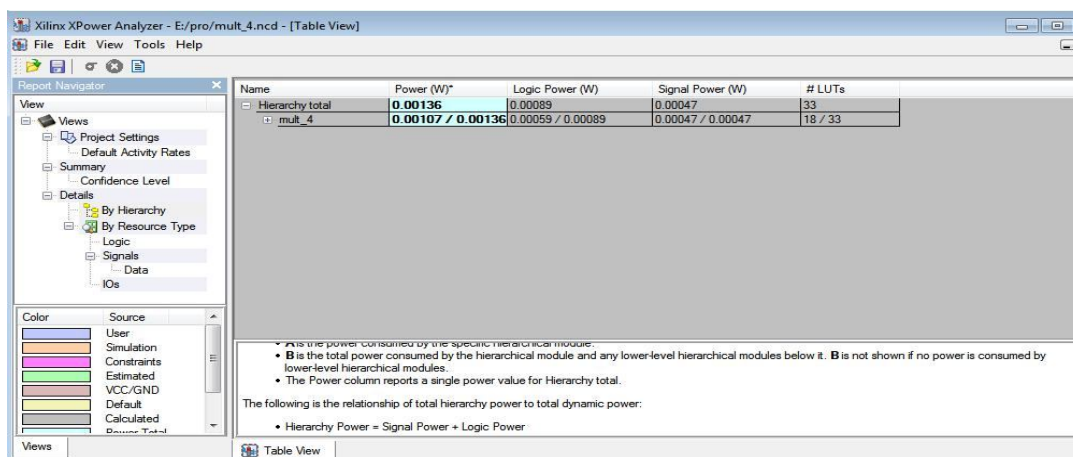

Figure 4.7 Power analysis for n mixed bypass
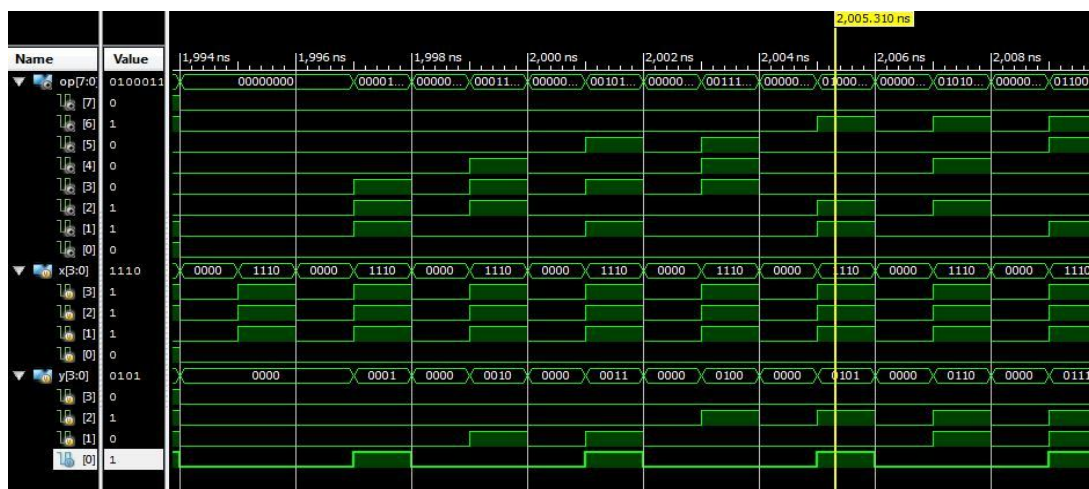
E. *Simulation For Array Multiplier*
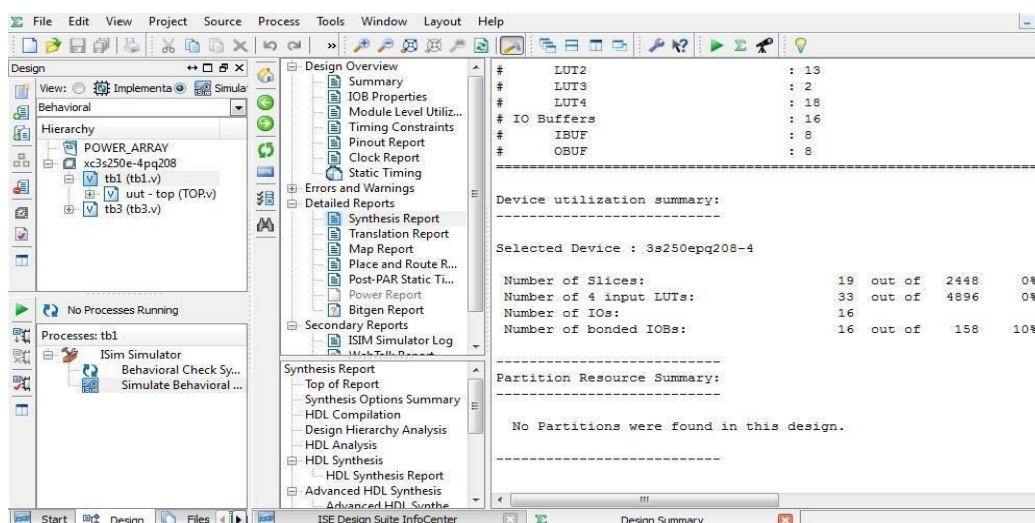

Figure 4.8 Output for array multiplier

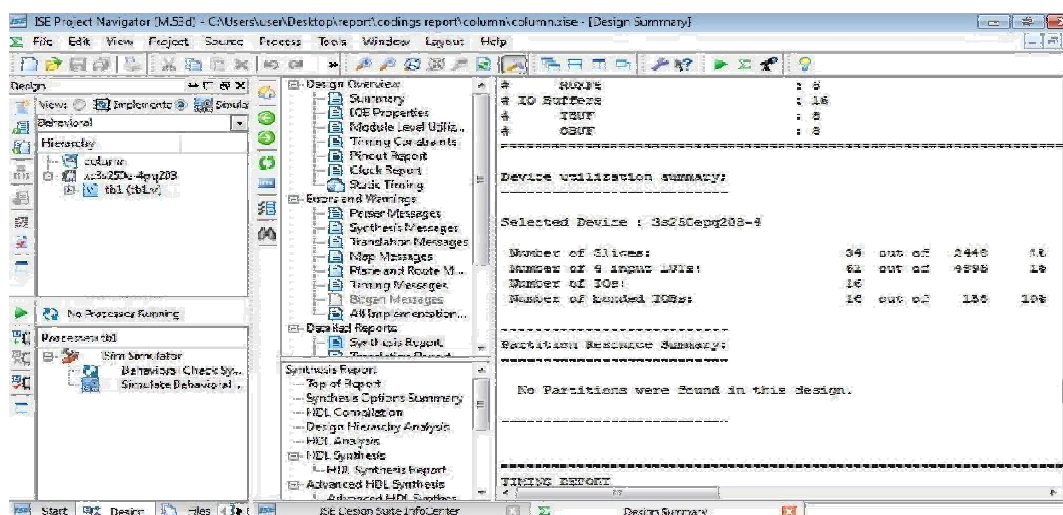

Figure 4.9 Area analysis for array multiplier

Figure 4.10 Power analysis for array multiplier

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VI June 2022- Available at www.ijraset.com
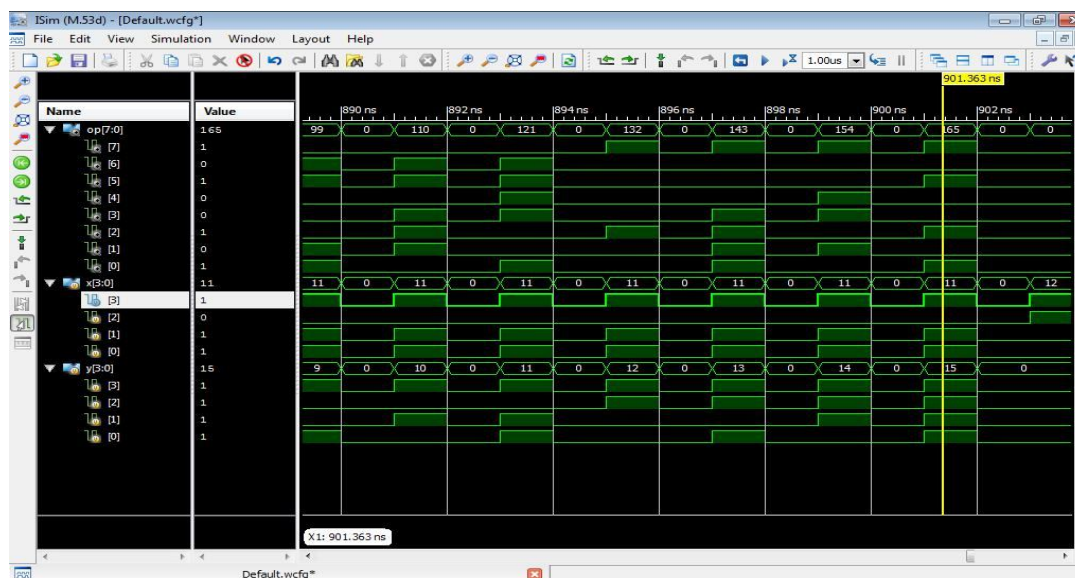
F.   Simulation For Column Bypass

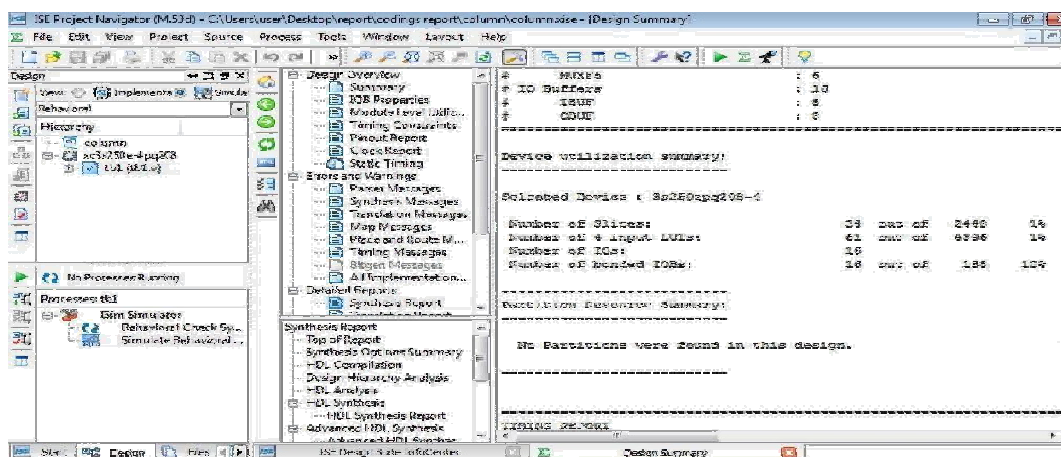Figure 4.11 Output for column bypass
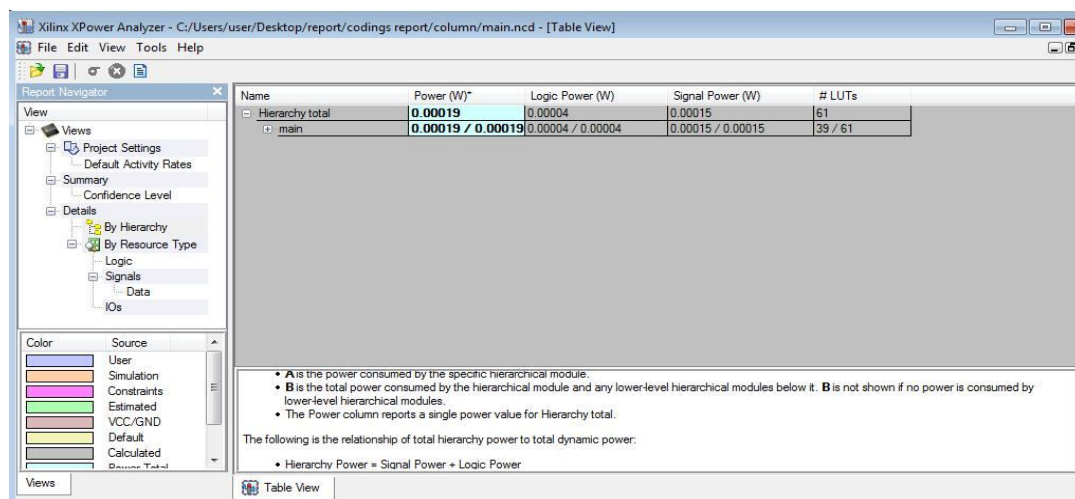

Figure 4.12 Area analysis for column bypass

Figure 4.13 Power analysis for column bypass

G. Synthesis Result On Xpower Tool

MULTIPLIER TYPE	ROW BYPASS	N MIXED	ARRAY MULTIPLIER	PROPOSED
Vendor	XILINX	XILINX	XILINX	XILINX
Deviceand Family	Virtex 5	Virtex 5	Virtex 5	Virtex 5
Logic power	0.00136	0.00020	0.00004	0.00092
Signal power	0.00053	0.00024	0.00015	0.00049
Total power(w)	0.00189	0.00044	0.00019	0.00141

Table 4.1: Power Dissipation of Multipliers

H. Synthesis Result On Xilinx Xst

MULTIPLI ER	ROW BYPASS	N MIXED	ARRAY MULTIPL IER	COLUMN BYPASS
No. of Slices LUT	19	27	33	19
	35	48	61	33

Table 4.2: Area of Multipliers

V. CONCLUSION AND FUTURE WORK

A. Conclusion

After going through all the hard work and facing problems, this project managed to complete its objectives that are to study different multiplier and learn the power, delay and time trade off among them so that can design efficient faster low power multiplier. Thus, the power consumption can be reduced if one can reduce the switching activity of a given logic circuit without changing its function. An obvious method to reduce the switching activity is to shut down the idle part of the circuit. After comparing all we came to a conclusion that Column bypass multiplier are best suited for low power dissipation. Similarly, To reduce the power dissipation, we proposed column-bypassing multiplier, mixed bypassing multiplier, low-cost low power bypass multiplier that is useful for various DSP applications, such as filtering, fast fourier transform, and wavelet transform. However, there are limitations in our work and several future work directions are possible.

B. Future Work

The project can be implemented to 64 and 128 bit column bypassing multiplier. Less switching activity can be achieved in this multiplier. Low Power consumption can be achieved. In this multiplier, a low power multiplier design column bypassing using Booth recoding is proposed. Compared with other multipliers such as row bypassing array, the results achieve higher power reduction and hardware overhead.

REFERENCES

[1] B. Parhami, Computer Arithmetic: Algorithms and HardwareDesigns, Oxford University Press, 2000.

[2] V. G. Moshnyaga and K. Tamaru, "A comparative study ofswitching activity Reduction techniques for design of lowpowermultipliers," IEEE International Symposium on Circuits and Systems, pp.1560-1563, 1995.

[3] J. Ohban, V.G. Moshnyaga and K. Inoue,"Multiplier energy reduction Through bypassing of partial products," Asia-Pacific Conf.on CircuitsandSystems.vol.2, pp.13-17. 2002.

[4] M. C. Wen, S. J. Wang and Y. N.Lin,"Low-power parallel multiplier with column bypassing," IEE Electronics Letters.Volume 41, Issue10, pp.581– 583,12 May 2005.

[5] Y. Huang, J. Lin, M. Sheu and C. Sheu. Low Power MultiplierDesigns Based on Improved Column Bypassing Schemes. IEEE Asia Pacific Conference on Circuits and Systems. 2006.

[6] J. T. Yan and Z. W. Chen, Low-power multiplier design with row and column bypassing, IEEE International SOC Conference, pp.227-230,2009

[7] A. Wu, "High performance adder cell for low powerpipelined multiplier, " IEEE International Symposium onCircuits and Systems, pp.57–60, 1996.

[8] C. Chou and K. Kuo. Low Power Multiplier with Bypassing and TreeStructures. IEEE Asia Pacific Conference on Circuits and Systems.2006.

[9] J. Yan and Z. Chen, Low-Cost Low-Power Bypassing-BasedMultiplier Design. IEEE International Symposium on Circuits andSystems. 2010.

[10] A Review of Different types of Multipliers and Multiplier-Accumulator unit 2013.

[11] Fast and power efficient 16*16 array of array multiplier using Vedic multiplication 2018.

[12] Yan J and Z, Chen, Low-Cost Low-Power Bypassing-Based Multiplier Design. IEEE International Symposium on Circuits and Systems, 2010.

[13] Design of area efficient, low power, high power and full swing hybrid multipliers 2021.

[14] Ultra low voltage GDI based hybrid full adder design for area and energy efficient computing system. 2019.

[15] A review on various design in VLSI

INTERNATIONAL JOURNAL
FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 ⓒ (24*7 Support on Whatsapp)