PROJECT REPORT ON

DESIGN OF DUAL CHANNEL MULTIPLIER

Submitted in partial fulfilment of the Requirement for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

Submitted by

	Regd. No:
N.Venkata Madhuri	19091A04Q6
S.K.Benarjee	19091A0419
N.Ravi Kumar	19091A04F5
P.Sai Kumar Naik	20095A0414

Under the Esteemed Guidance of

Dr.R.HANUMA NAIK

MTech, Ph.D

Associate Professor, RGMCET, Nandyal

(ESTD-1995)

DEPARTMENT OF

ELECTRONICS AND COMMUNICATION ENGINEERING

RAJEEV GANDHI MEMORIAL COLLEGE OF

ENGINEERING AND TECHNOLOGY

(AUTONOMOUS)

Affiliated to J.N.T.U.A - Anantapuramu, Approved by A.I.C.T.E - New Delhi,
Accredited by N.B.A - New Delhi, Accredited by NAAC with A+ Grade – New Delhi

NANDYAL –518501, Nandyal Dist. A.P.

YEAR: 2019 - 2023

RAJEEV GANDHI MEMORIAL COLLEGE OF ENGINEERING & TECHNOLOGY

AUTONOMOUS

(Approved by A.I.C.T.E - New Delhi, Affiliated to JNTUA - Anantapuramu, Accredited by NBA - New Delhi, Accredited by NAAC with 'A+' Grade - New Delhi)

NANDYAL – 518 501, A.P, India

CERTIFICATE

This is to certify that the dissertation entitled "DESIGN OF DUAL CHANNEL MULTIPLIER" is being submitted by N.Venkata Madhuri (19091A04Q6), S.K.Benarjee (19091A0419), N.Ravi kumar (19091A04F5), P.Sai Kumar Naik (20095A0414) under the guidance of Dr.R.Hanuma Naik, Associate Professor for Project of the award of B.Tech Degree in Electronics and Communication Engineering, Rajeev Gandhi Memorial College of Engineering & Technology, Nandyal (Autonomous) (Affiliated to J.N.T.U.A Anantapuramu) is a record of bonafide work carried out by them under our guidance and supervision.

Head of the Department
Dr. Kethepalli Mallikarjuna

Dr. V.N.V. SATYANARAYANA PRAKASH
B.Tech, M Tech Ph.D MISTE,MIETE,MIE.
Professor
Department of E.C.E.
R.G.M College of Engg. & Tech., (Autonomous)
NANDYAL - 518 501, Kurnool (Dist), A.P.

04/05/2023

Project Guide
Dr.R.Hanuma Naik

Signature of the External Examiner

Date of Viva-Voce:

ACKNOWLEDGEMENT

We earnestly take the responsibility to acknowledge the following distinguished personalities who graciously allowed us to carry our project work successfully.

We express deep gratitude to our guide **Dr.R.Hanuma Naik**, MTech. Ph.D, Associate Professor in ECE Department, RGMCET, Nandyal for his guidance, incessant help and encouragement throughout the course of the project work, which contributed, to the successful completion of this project.

We would like to thank **Dr. Kethepalli Mallikarjuna**, M.Tech., Ph.D, Professor & Head, Department of ECE, RGMCET, Nandyal, for this valuable suggestions and encouragement whenever we encountered difficulties during the project work.

We would like to express my deep sense of gratitude and indebtedness to **Dr. T. Jayachandra Prasad**, Principal, RGMCET, Nandyal for his sustained encouragement and moral support throughout the course.

We are thankful to our honorable Chairman **Dr. M. Santhiramudu**, RGM Group of Institutions and **Sri. M. Sivaram**, Managing Director, RGMCET, Nandyal for providing us with required facilities in the college.

We express our thanks to all the faculty and staff members of ECE Department, RGMCET, Nandyal, for their support in various aspects in completing the project.

Project Associates

N.Venkata Madhuri	19091A04Q6
S.K. Benarjee	19091A0419
N.Ravi kumar	19091A04F5
P.Sai Kumar Naik	20095A0414

TABEL OF CONTENTS

LIST OF FIGURES

ABSTRACT

In this project, A dual-channel multiplier for piecewise-polynomial function of energy efficient and power efficient is proposed for 3-D graphics applications. It Compensates the complex multipliers by using DCM where it reduces the hardware complexity. DCM scheme performs complex functions with power-efficient and area efficient approach. it reduces the hardware computational effort in the piecewise polynomial approximation with uniform or nonuniform segmentation. The performance of the evaluation process is highly dependent on the design of the multiplication and squaring structure. A novel hardware implementation for polynomial evaluation is presented. The proposed approach compensates the complex multipliers by using DCM which reduces the hardware complexity. The DCM scheme performs complex functions with power efficient and area-efficient approach. The multiplier reduces the hardware computational effort in the piecewise polynomial approximation with uniform or nonuniform segmentation. For large operand input size, a multiplier adder converter and a dedicated radix-4 squaring unit are also proposed. These units achieve the least power consumption compared to previous approaches with large input word size. The above proposed system is designed and developed using VHDL environment.

CHAPTER 1
INTRODUCTION

1.1 INTRODUCTION

Graphical processing units (GPUs) have a wide variety of applications in different fields such as compute art, engineering, science, medicine, entertainment, advertising, visualization, military, and graphical user interface. The growth in the GPUs applications has led to evolution in the hardware design to handle the needed increase in performance.

The General-purpose GPUs are used to perform intensive computations on the same hardware covering large sector of applications. GPUs are mainly composed of several unified shaders. Each unified shader contains general purpose arithmetic unit and special function unit (SFU) that is used for computing special transcendental and algebraic functions not provided by the general function unit. Functions such as sine, cosine, reciprocal, logarithm, exponential, and compound functions are computed by SFUs.

The main design constraint in the handheld devices is power dissipation. To extend the battery lifetime, low-power schemes are essential at all stages of design. SFUs handle the complex operations which are required for graphics applications. Most of the power is consumed by those heavy arithmetic functions. Three main algorithms have been adapted to evaluate various functions in hardware including direct lookup table algorithm, polynomial and rational algorithm, and table based piecewise-polynomial (PWP) algorithm.

Large lookup tables are used to store the approximated value of the function in the direct lookup algorithm. High-degree polynomial is used in this algorithm to approximate the function. As a result, large number of multiplications and additions leads to high power consumption and long

execution times. As a compromise between the direct lookup table algorithm and the polynomial algorithm, the table-based PWP is employed.

PWP has low overhead and small table size for hardware computation which makes that algorithm the most attractive for function evaluation. The approximation coefficients of the polynomial are stored in small lookup table. PWP algorithm gives attractive tradeoff between the computation error and hardware cost. Several PWP evaluation techniques are proposed in literature. The five most common ways of implement multipliers are related to the expected DCM methodology. Various numbers of bits are used for evaluation. Remember that DCM only takes cycles of N clock while the serial multiplier takes cycles of 2N clock to attain regeneration. Consequently, the time limit for an 8-bit serial multiplier is tantamount to a clock cycle multiplication by 16.

DCM is characterized by low power consumption and a small area. Together with the fully parallel multiplier, DCM saves up to 37% and 93% in area and power, respectively. On the other hand, the multiplier requires 1 clock cycle to complete the multiplication, while the DCM requires 8 clock cycles. However, the expected DCM improves PDP by at least 80%. This type of multiplier is ideal for low energy applications such as low power consumption handheld device applications.

Therefore, DCM provides a simple and energy efficient design structure, making it an excellent choice for functional evaluation of PWP. For radix-8 booth multipliers, you can multiply by 0, ± 1, ± 2, ± 3, or ± 4 instead of multiplying by 0 or 1. Therefore, extra hardware is expected to manage the Multiplying ± 3 stage of evolution, resulting in enormous energy and area consumption, even though partial product reduction results in faster productivity.

The booth multipliers radix 4 and radix-8 are fully captured using Wallace Adder Sapling to once again minimize partial products and measure end result. Serial multipliers, by comparison have different configurations, and therefore different enactments. For the serial multiplier, substantial

energy and area savings are achieved, at the cost of maximum power. The strategies suggested by DCM show better communication between area, power, and delay. The suggested DCM thus demonstrates that at least a 70 percent reduction in power requirements ranges through multiple forms of modern multipliers. A recently announced DCM will be used for significant power reduction in a secondary PWP architecture. The results of the hardware enactment for a second order PWP using DCM.

The system design will be implemented in MODELSIM 6.3 Software. A Simple and easy to use interactive simulation tool from Mentor Graphics. Includes a built-in C debugger for VHDL, Verilog, System C and other HDL emulations and simulations. Model-sim can be used with Intel Quart us II Prime, Xilinx ISE.

1.2 MOTIVATION

Multiplier is an electronic device that performs the multiplication of two uncorrelated analog signals. It can multiply two binary numbers.

The multiplier is not only used as the basic unit of analog operations such as multiplication, division, power and square extraction, but also widely used in electronic communication systems as modulation, demodulation, mixing, phase discrimination and automatic gain control; it can also be used for filtering.

1.3 OBJECTIVE OF THESIS

- Design of dual channel multiplier for evaluation of piecewise polynomial function
- By designing this dual channel multiplier we can have an efficient power , area and delay
- Design of dual channel multiplier can help in maintaining energy efficient and power efficient multiplier which help in 3-d graphics .

1.4 ORGANISATION OF CHAPTERS

In chapter 1 Introduction, objective of the thesis and organization is described.

In chapter 2 literature survey of this dual channel multiplier is described.

In chapter 3 our proposed system design of dual channel multiplier enhancement is described.

In chapter 4 software tools and code which has used in Dual channel multiplier is described.

In chapter 5 stimulation results and design is described.

In chapter 6 conclusion and Future scope is described.

CHAPTER 2
LITERATURE SURVEY

2.1 LOW-POWER DIGITSERIAL MULTIPLIERS [1]

The paper "Low-power digit-serial multipliers" by Y.-N. Chang, J. H. Satyanarayana, and K. K. Parhi was presented at the IEEE International Symposium on Circuits and Systems in June 1997.

The paper proposes a low-power implementation of digit-serial multipliers, which are commonly used in digital signal processing applications. The proposed design uses a modified booth encoding scheme and a carry-select adder to reduce power consumption.

The authors show that their proposed design achieves significant power savings compared to traditional designs while maintaining high performance. They also provide simulation results to validate their approach and show the effectiveness of their design.

Overall, the paper presents a promising approach for reducing power consumption in digital signal processing circuits through the use of optimized encoding schemes and adder structures.

The working of the low-power digit-serial multiplier proposed in the paper can be summarized as follows:

The input operands are first converted to digit-serial form. This involves breaking down each operand into smaller "digits" and processing these digits sequentially.

A modified Booth encoding scheme is used to reduce the number of partial products generated during multiplication. This is achieved by representing groups of two or more adjacent digits as a single "Booth digit" and using a lookup table to generate the corresponding partial products.

The partial products are then added together using a carry-select adder (CSA). The CSA uses multiple adder structures to compute partial sums and carries simultaneously, allowing it to perform addition quickly and efficiently.

2.2 VLSI IMPLEMENTATION OF PIECEWISE APPROXIMATED ANTILOGARITHMIC CONVERTER [2]

The paper "VLSI implementation of piecewise approximated antilogarithmic converter" by R. R. Selina was presented at the International Conference on Communications and Signal Processing in April 2013.

The paper proposes a VLSI implementation of an antilogarithmic converter, which is commonly used in digital signal processing applications. The proposed design uses a piecewise approximation approach to reduce the hardware complexity and power consumption of the converter.

The authors show that their proposed design achieves significant savings in area and power consumption compared to traditional designs while maintaining high accuracy. They also provide simulation results to validate their approach and show the effectiveness of their design.

Overall, the paper presents a promising approach for reducing the hardware complexity and power consumption of antilogarithmic converters in digital signal processing circuits through the use of piecewise approximation techniques.

The working of the piecewise approximated antilogarithmic converter proposed in the paper can be summarized as follows:

The input signal is first buffered to ensure that it is stable and has the correct voltage levels. The piecewise approximation circuit approximates the antilogarithmic function using a piecewise linear approach. This involves breaking down the input range into multiple segments and using a different linear function to approximate the antilogarithmic function within each segment.

The linear approximation functions are implemented using simple circuits, such as operational amplifiers and resistors, to minimize hardware complexity and power consumption. The output signal is then buffered to ensure that it has the correct voltage levels and can drive the load circuit.The control logic generates the control signals needed.

2.3 HIGH SPEED SPECIAL FUNCTION UNIT FOR GRAPHICS PROCESSING UNIT [4]

The paper "High speed special function unit for graphics processing unit" by A.-E. G. Qoutb, A. M. El-Gunidy, M. F. Tolba, and M. A. El-Moursy was presented at the 9th International Design and Test Symposium in December 2014.

The paper proposes a special function unit (SFU) for use in graphics processing units (GPUs) that can perform a wide range of mathematical functions, such as exponentiation, logarithm, and trigonometric functions, with high speed and accuracy. The proposed SFU is designed to overcome the limitations of traditional SFUs in GPUs, such as limited precision and high power consumption.

The authors show that their proposed SFU achieves significant improvements in speed and accuracy compared to traditional SFUs in GPUs while maintaining low power consumption. They also provide simulation results to validate their approach and show the effectiveness of their design.

Overall, the paper presents a promising approach for designing SFUs for use in high-performance GPUs that can perform a wide range of mathematical functions with high speed and accuracy while maintaining low power consumption.

The working of the proposed SFU for graphics processing units (GPUs) in the paper:

Input signal is first buffered to ensure stable voltage levels and prepare it for processing.The input signal is used to access the look-up table (LUT) that stores pre-calculated values of the special function being computed. The LUT is designed to offer high speed and accuracy.

The interpolation circuit approximates the output value by interpolating between the values in the LUT using a piecewise linear approximation approach.The control logic generates control signals to coordinate the operation of the SFU, such as selecting the appropriate segment of the LUT and interpolation circuit for a given input value.

The output signal is then buffered to ensure stable voltage levels and prepare it for further processing or transmission.

The authors note that their proposed SFU is designed to achieve high speed and accuracy while maintaining low power consumption. The LUT stores pre-calculated values, which saves computation time, and the piecewise linear approximation approach reduces the hardware complexity and power consumption of the interpolation circuit. The result is a special function unit that can perform a wide range of mathematical functions with high speed and accuracy, making it suitable for use in high-performance GPUs. The simulation results presented in the paper validate the effectiveness of the proposed design.

2.4 A 130.3 MW 16-CORE MOBILE GPU WITH POWER-AWARE PIXEL APPROXIMATION TECHNIQUES [7]

The paper "A 130.3 mW 16-core mobile GPU with power-aware pixel approximation techniques" by Y.-J. Chen et al. was published in the IEEE Journal of Solid-State Circuits in September 2015.

The paper presents a 16-core mobile graphics processing unit (GPU) that is designed to operate with low power consumption while maintaining high image quality. To achieve this goal, the authors propose a set of power-aware pixel approximation techniques that reduce the computational complexity of the GPU. The authors propose several power-aware pixel approximation techniques that reduce the computational complexity of the GPU.

These techniques include:

- Early-z rejection: This discards pixels that are hidden from view before they are processed by the pixel shader.
- Trilinear filtering approximation: This approximates the trilinear filtering operation used to apply texture to the image.
- Hierarchical depth test: This approximates the depth test operation used to determine which pixels are visible in the image.
- Reduced-precision arithmetic: This uses reduced-precision arithmetic to perform the operations in the GPU, which reduces power consumption.

The authors show that their proposed techniques achieve significant improvements in power consumption while maintaining high image quality. They also provide simulation results to validate their approach and show the effectiveness of their design.

Overall, the paper presents a promising approach for designing mobile GPUs that can operate with low power consumption while maintaining high image quality by using power-aware pixel approximation techniques.

The working of the proposed mobile GPU with power-aware pixel approximation techniques can be summarized as follows:

Input data is first buffered and sent to the vertex shader, which transforms the vertices of 3D objects into 2D images.The rasterizer then maps the vertices onto the pixels of the display to create a 2D image.The pixel shader processes the pixels of the 2D image to add color and texture.

To reduce power consumption, the proposed techniques are used to approximate certain operations that are computationally expensive. For example, the early-z rejection technique discards pixels that are hidden from view before they are processed by the pixel shader, which reduces the number of pixels that need to be processed. The trilinear filtering approximation technique approximates the trilinear filtering operation used to apply texture to the image, which reduces the number of texture samples that need to be computed. The hierarchical depth test technique approximates the depth test operation used to determine which pixels are visible in the image, which reduces the number of pixels that need to be processed. The reduced-precision arithmetic technique uses reduced-precision arithmetic to perform the operations in the GPU, which reduces power consumption. The final image is stored in the frame buffer and displayed on the screen.

The authors show that their proposed power-aware pixel approximation techniques significantly reduce power consumption while maintaining high image quality. They also provide simulation results to validate their approach and show the effectiveness of their design.

2.5 DOUBLE LOGARITHMIC ARITHMETIC TECHNIQUE FOR LOW-POWER 3-D GRAPHICS APPLICATIONS [10]

The authors show that the proposed double logarithmic arithmetic technique reduces the power consumption of 3-D graphics applications while maintaining high image quality. They also provide simulation results to validate their approach and show the effectiveness of their design.

The proposed double logarithmic arithmetic technique in this paper is aimed at reducing the power consumption of 3-D graphics applications. Here is a summary of the working of the technique:

The input data is converted from fixed-point to double logarithmic format. This conversion is done using a specialized converter circuit that takes the input data and generates the corresponding double logarithmic representation.

The double logarithmic representation is processed using double logarithmic arithmetic. This involves performing the arithmetic operations on the double logarithmic representation of the input data.

The output of the double logarithmic arithmetic is converted back to fixed-point format using a specialized converter circuit. The final output is generated by processing the fixed-point data using conventional graphics processing techniques.

SUMMARY

In this chapter we have described about the previous papers on simulation of different multipliers for low power and efficient methods for low power and delay for evaluation of piece wise polynomial function.

CHAPTER 3
DESIGN OF DUAL CHANNEL MULTIPLIER

3.1 PROPOSED SYSTEM

Low-power and low area DCM architecture is proposed based on a serial algorithm. The proposed DCM architecture is shown in Fig. 1. The DCM scheme has simple and uniform construction. x and y are the serial input and the parallel input, respectively, and p is the final product. Two serial input bits are processed each clock cycle. Even index numbers (x6, x4, x2, x0) of the serial input are directed to the upper channel while, odd index numbers (x7, x5, x3, x1) are processed in the lower channel. Concurrently, the pairs are transferred and handled on the same clock phase.

The partial product (y0x1) is added to the partial product (y1x0) and propagated to the output. Also, the partial product (y0x0) is directly propagated to the output. The DCM includes parallel-to-serial and serial-to-parallel converters to handle the input and to compute the parallel result, respectively. For large word size, additional cells can be added in the hardware implementation to perform the multiplication. Each cell includes two AND gates, two full adders, and 2-D flip flops.

Compared to different previous architectures, DCM uses less hardware. Also, the proposed DCM includes only full adders and delay units. the multiplier-adder converter (MAC) scheme is introduced and is compared with recent techniques in terms of power, area, and delay. The MAC is composed of two logarithmic converters, an adder, and an antilogarithmic converter as shown in Fig. The implementation of the logarithmic and antilogarithmic converters controls the MAC performance.

BLOCK DIAGRAM

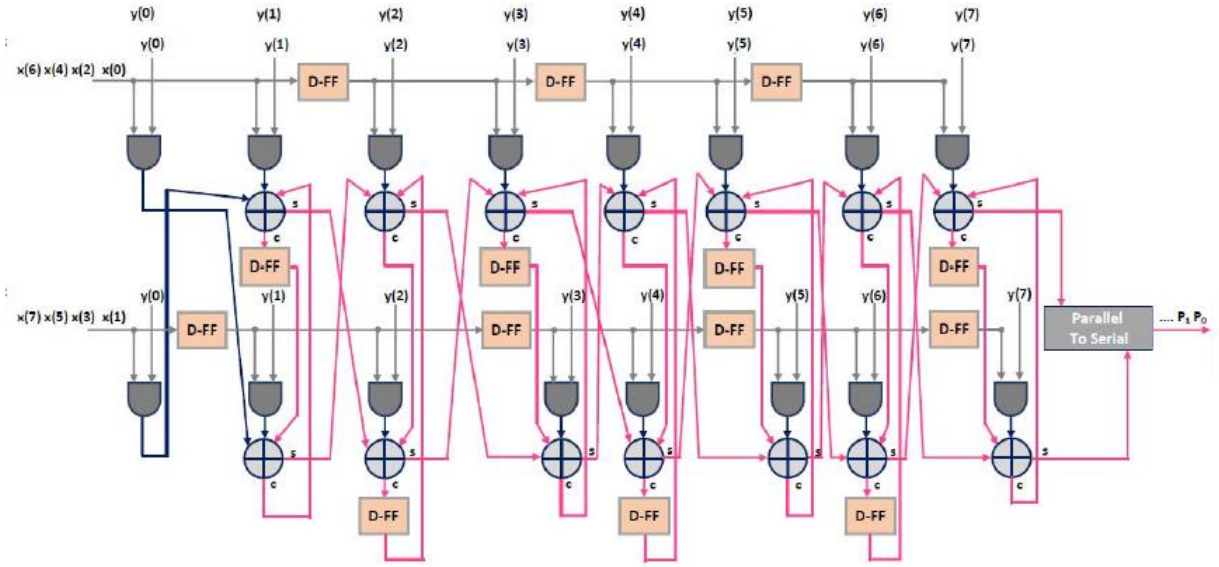


Figure 3. 1 **Block Diagram of 8 Bit DCM Design**

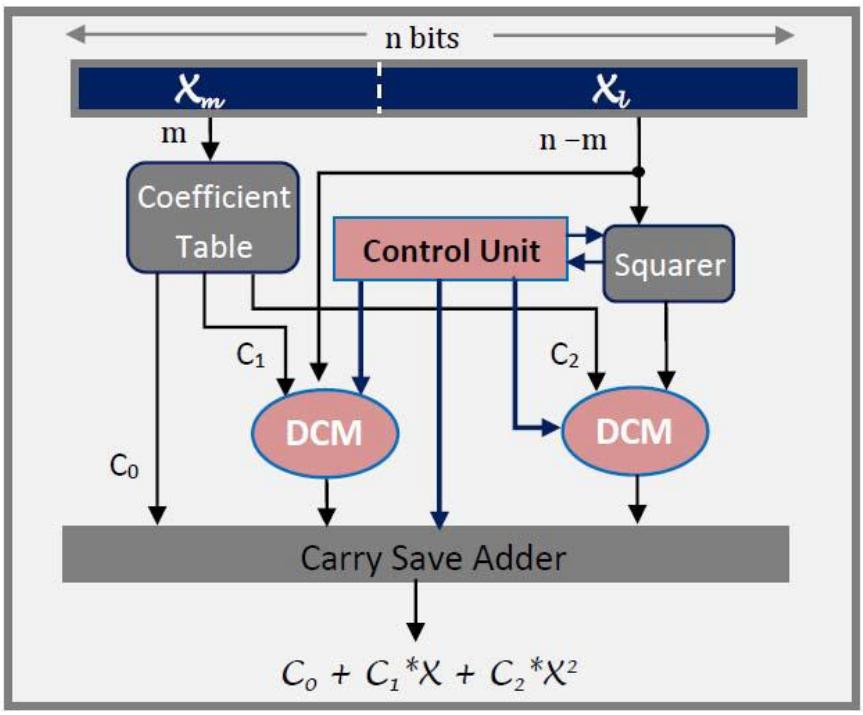


Figure 3. 2 Proposed MAC scheme

MODULES NAME:

- Registers
- DCM
- Coefficient Table
- Squarer
- Carry Save adder
- Booth Multipliers

3.2 MODULES EXPLANTIONS

REGISTERS:

A register is a memory device that can be used to store more than one bit of information. A register is usually realized as several flip-flops with common control signals that control the movement of data to and from the register.

DCM:

The proposed DCM architecture is shown in Fig. 3. The DCM scheme has simple and uniform construction. x and y are the serial input and the parallel input, respectively, and p is the final product. Two serial input bits are processed each clock cycle. Even index numbers (x6, x4, x2, x0) of the serial input are directed to the upper channel while, odd index numbers (x7, x5, x3, x1) are processed in the lower channel. Concurrently, the pairs are transferred and handled on the same clock phase. Starting with the first clock cycle, the partial products (PP0) are generated.

$$P_0 = y_0 x_0$$
$$P_1 = y_0 x_1 + y_1 x_0$$

In the next clock cycle, the serial input data have been moved one phase to the right after each delay element. Note that the generated carry from addition remains to be added for the exact final result. The current partial products which are generated for the upper channel are (y0x2),

(y1x2), (y2x0), and (y3x0) as the following equation. For the lower channel, the partial products (y0x3), (y1x1), and (y2x1) are produced as in the following equation:

$$PP_1 = \begin{cases} y_0x_2, y_1x_2, y_2x_0, y_3x_0 \\ y_0x_3, y_1x_1, y_2x_1 \end{cases}$$

Starting from the partial product (y0x2) in the upper channel, two complete additions are carried out. The partial product (y0x2) is added to (y1x1) then is added to (y2x0) and the final result becomes (P2). Three complete additions are performed starting from the partial product (y0x3) in the lower channel. The partial product (y0x3) is added to (y1x2) then is added to (y2x1) and at last is added to (y3x0).

The final result of the previous sum is (P3). The carry bits which are produced by the full adders are propagated forward and added to the next partial products. The multiplication process is repetitive

$$P_2 = y_0x_2 + y_1x_1 + y_2x_0$$
$$P_3 = y_0x_3 + y_1x_2 + y_2x_1 + y_3x_0$$

The next partial products are generated by the same procedure. After eight clock cycles, the product of 8-bit by 8-bit multiplication is achieved.

COEFFICIENT TABLE:

The approximation coefficients of the polynomial are stored in small lookup table. A small size lookup table is used to store the approximation coefficients for each sub segment. The total number of segments equals 2m. The function evaluation is approximated by the least significant bits (n − m) which are the second portion of the input operand bits. The approximation coefficients C0, C1, and C2 are determined for each segment using mathematical algorithms.

SQUARER:

A radix-4 squarer is proposed. As shown in the PWP function evaluation structure, the squarer is located in the critical path. Therefore, to speed up the overall performance, the proposed radix-4 squarer can be

realized by operating on each two bits simultaneously. The number of partial products is reduced to half. Instead of multiplying by 0 or 1, multiplying by 0, +1, +2, or +3 is performed in radix-4.

CARRY SAVE ADDER:

A carry-save adder is a type of digital adder, used in computer micro architecture to compute the sum of three or more n-bit numbers in binary. It differs from other digital adders in that it outputs two numbers of the same dimensions as the inputs, one which is a sequence of partial sum bits and another which is a sequence of carry bits.

BOOTH ENCODER SCHEME ENCODER:

Both BE schemes introduce errors and two correction terms are required. When the NB number is converted to a RB format, -1 must be added to the LSB of the RB number; when the multiplicand is multiplied by -1 or -2 during the Booth encoding, the number is inverted and +1 must be added to the LSB of the partial product. A single ECW can compensate errors from the radix-16 Booth recoding.

3.3 BOOTH'S MULTIPLICATION ALGORITHM

3.3.1 INTRODUCTION

In computing, especially digital signal processing, the multiply–accumulate operation is a common step that computes the product of two numbers and adds that product to an accumulator. The hardware unit that performs the operation is known as a multiplier–accumulator (MAC, or MAC unit); the operation itself is also often called a MAC or a MAC operation. The MAC operation modifies an accumulator a <= a + (b x c).

When done with floating point numbers, it might be performed with two roundings (typical in many DSPs), or with a single rounding. When performed with a single rounding, it is called a fused multiply–add (FMA) or fused multiply–accumulate (FMAC).

Modern computers may contain a dedicated MAC, consisting of a multiplier implemented in combinational logic followed by an adder and an accumulator register that stores the result. The output of the register is fed back to one input of the adder, so that on each clock cycle, the output of the multiplier is added to the register. Combinational multipliers require a large amount of logic, but can compute a product much more quickly than the method of shifting and adding typical of earlier computers. Percy Ludgate was the first to conceive a MAC in his Analytical Machine of 1909,[1] and the first to exploit a MAC for division (using multiplication seeded by reciprocal, via the convergent series (1+x)−1). The first modern processors to be equipped with MAC units were digital signal processors, but the technique is now also common in general-purpose processors.

3.3.2 NORMAL BOOTH MULTIPLIER

In many real-time DSP applications, high performance is a critical concern. Multiplication is the most fundamental arithmetic operation used in most of the signal processing algorithms. But multipliers usually have large area, larger delay and consume more power. However, achieving this may be done at the cost of area, on chip power consumed and delays. In the binary number system the digits, called bits, are limited to the set {0, 1}.

The result of multiplying any binary number by a binary bit is either 0, or the original number. This makes formation of the intermediate partial-products simple and efficient. Adding all these partial-products is time consuming task for a n y binary multipliers. The entire process consists of three steps partial product generation, partial product reduction and addition of partial products as shown in Fig 1. But in booth multiplication, partial product generation is done based on recoding scheme radix 2 encoding. Bits of multiplicand (Y) are grouped from left to right and corresponding operation on multiplier (X) is done in order to generate the partial product.

Table 3.1: NORMAL BOOTH MULTIPLIER

x_k	y_k	$x_k \times y_k$
-1	-1	1
-1	0	0
-1	1	-1
0	-1	0
0	0	0
0	1	0
1	-1	-1
1	0	0
1	1	1

Rule to compute the partial results

Z=X*Y

Multiplication using normal Booth‟s recoding algorithm technique based on the fact that partial product can be generated for group of consecutive 0‟s and 1‟s which is called Booth‟s recoding. This recoding algorithm is used to generate efficient partial product. These partial products always have large number of bits than the input number of bits. This increase in the width of partial product usually depends upon the radix scheme used for recoding.

3.4 SUMMARY

In this chapter the proposed Dual channel multiplier for evaluation of polynomial function and all algorithms, methods which are required to design the dual channel multiplier are described.

CHAPTER 4
SOFTWARE REQUIREMENTS

- Verification Tool
    - Modelsim 6.4
- Synthesis Tool
    - Xilinx tool

4.1 INTRODUCTION TO MODELSIM

ModelSim is a useful tool that allows you to stimulate the inputs of your modules and view both outputs and internal signals. It allows you to do both behavioral and timing simulation, however, this document will focus on behavioral simulation. Keep in mind that these simulations are based on models and thus the results are only as accurate as the constituent models. ModelSim /VHDL, ModelSim /VLOG, ModelSim /LNL, and ModelSim /PLUS are produced by Model Technology™ Incorporated. Unauthorized copying, duplication, or other reproduction is prohibited without the written consent of Model Technology.

The information in this manual is subject to change without notice and does not represent a commitment on the part of Model Technology. The program described in this manual is furnished under a license agreement and may not be used or copied except in accordance with the terms of the agreement. The online documentation provided with this product may be printed by the end-user.

The number of copies that may be printed is limited to the number of licenses purchased. ModelSim is a registered trademark of Model Technology Incorporated. Model Technology is a trademark of Mentor Graphics Corporation. PostScript is a registered trademark of Adobe Systems Incorporated. UNIX is a registered trademark of AT&T in the USA and other countries.

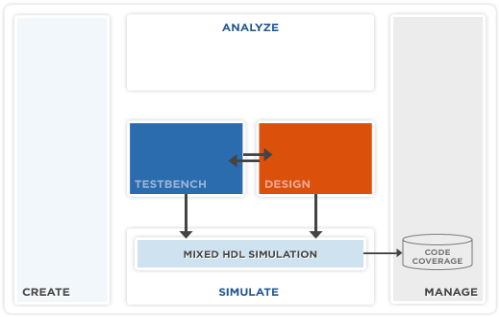
Standards Supported

ModelSim VHDL supports both the IEEE 1076-1987 and 1076-1993 VHDL, the 1164-1993 Standard Multivalve Logic System for VHDL Interoperability, and the 1076.2-1996 Standard VHDL Mathematical Packages standards. Any design developed with ModelSim will be compatible with any other VHDL system that is compliant with either IEEE Standard 1076-1987 or 1076-1993.

ModelSim Verilog is based on IEEE Std 1364-1995 and a partial implementation of 1364-2001, Standard Hardware Description Language Based on the Verilog Hardware Description Language. The Open Verilog International Verilog LRM version 2.0 is also applicable to a large extent. Both PLI (Programming Language Interface) and VCD (Value Change Dump) are supported for ModelSim PE and SE users.

MODELSIM - ADVANCED SIMULATION AND DEBUG

ASIC and FPGA design



Mentor Graphics was the first to combine single kernel simulator (SKS) technology with a unified debug environment for Verilog, VHDL, and System C. The combination of industry-leading, native SKS performance with the best integrated debug and analysis environment make "ModelSim" the simulator of choice for both ASIC and FPGA designs. The best standards and platform support in the industry make it easy to adopt in the majority of process and tool flows.

Overview

- **Unified mixed language simulation engine** for the fastest regression suite throughput
- **Native support** of Verilog, System Verilog for design, VHDL, and System C for effective verification of the most sophisticated design environments
- **Fast time-to-debug** causality tracing and multi-language debug environment
- **Advanced code coverage and analysis** tools for fast time to coverage closure

Major product features:

- Leading RTL and gate performance with ASIC sign-off
- Native support of VHDL, Verilog, System Verilog, and System C

• Powerful, intuitive GUI speeds RTL and gate debug

• Textual and graphical dataflow with automated X-tracing and source annotation

• Code coverage

• Post-simulation debug

• Customizable, open architecture with C and Tcl/Tk

• Integrated for simulation farm support

• Upgradeable to Questa for advanced verification solutions

Leader in Single Kernel, Mixed Language Technology

Mentor Graphics® was the first to combine single kernel simulator (SKS) technology with a unified debug environment for Verilog, System Verilog, VHDL, and System C. The combination of industry-leading performance and capacity with the best integrated debug and analysis environment make ModelSim the simulator of choice for both ASIC and FPGA

design. The best standards and platform support in the industry make it easy to adopt in the majority of process and tool flows.

High-Performance, Scalable Simulation Environment

ModelSim provides seamless, scalable performance and capabilities. Through the use of a single compiler and library system for all ModelSim configurations, employing the right ModelSim configuration for project needs is as simple as pointing your environment to the appropriate installation directory.

Easy-to-Use Simulation Environment

An intelligently engineered graphical user interface (GUI) efficiently displays design data for analysis and debug. The default configuration of windows and information is designed to meet the needs of most users.

However, the flexibility of the ModelSim SE GUI allows users to easily customize it to their preferences. The result is a feature-rich GUI that is easy to use and quickly mastered. A message viewer enables simulation messages to be logged to the ModelSim results file in addition to the standard transcript file.

The GUI's organizational and filtering capabilities allow design and simulation information to be quickly reduced to focus on areas of interest, such as possible causes of design bugs. ModelSim SE allows many debug and analysis capabilities to be employed post-simulation on saved results, as well as during live simulation runs. For example, the coverage viewer analyzes and annotates source code with code coverage results, including FSM state and transition, statement, expression, branch, and toggle coverage. Signal values can be annotated in the source window and viewed in the waveform viewer. Race conditions, delta, and event activity can be analyzed in the list and wave windows.

SYNTHESIS TOOL:

4.2 XILINX

Xilinx is a supplier of programmable logic devices. It is known for inventing the field programmable gate array (FPGA) and as the first semiconductor company with a fabless manufacturing model. Founded in Silicon Valley in 1984, the company is headquartered in San Jose, California, U.S.A.; Dublin, Ireland; Singapore; and Tokyo, Japan. The company has corporate offices throughout North America, Asia and Europe.

FOUNDATION

Xilinx was founded in 1984 by two semiconductor engineers, Ross Freeman and Bernard Vonderschmitt, who were both working for integrated circuit and solid-state device manufacturer Zilog Corp. While working for Zilog, Freeman wanted to create chips that acted like a blank tape, allowing users to program the technology themselves. At the time, the concept was paradigm-changing. "The concept required lots of transistors and, at that time, transistors were considered extremely precious – people thought that

Ross's idea was pretty far out", said Xilinx Fellow Bill Carter, who when hired in 1984 as the first IC designer was the company's eighth employee. Big semiconductor manufacturers were enjoying strong profits by producing massive volumes of generic circuits. Designing and manufacturing dozens of different circuits for specific markets offered lower profit margins and required greater manufacturing complexity.[4] What became known as the FPGA would allow circuits produced in quantity to be tailored by individual market segments.

Growth

As demand for programmable logic continued to grow, so did Xilinx's revenues and profits. From 1988 to 1990, the company's revenue grew each year from $30 million to $50 million to $100 million. During this time period, funding provider Monolithic Memories Inc. (MMI) was purchased by Xilinx competitor AMD. As a result of the AMD acquisition, Xilinx dissolved the deal with MMI and went public on the NASDAQ in 1989. The company also moved to a 144,000-square-foot (13,400 m^2) plant in San Jose, California in order to keep pace with demand from companies like HP, Apple Inc., IBM and Sun Microsystems who were buying large quantities from Xilinx.

Xilinx competitors emerged in the FPGA market in the mid-1990s.[4] Despite the competition, Xilinx's sales grew to $135 million in 1991, $178 million in 1992 and $250 million in 1993.The company reached $550 million in revenue in 1995, one decade after having sold its first product. According to market research firm supply, Xilinx has led the programmable logic device market share since the late 1990s. Over the years, Xilinx expanded operations to India, Asia and Europe. Xilinx's sales rose from $560 million in 1996 to nearly $2 billion by the end of its fiscal year 2010.[11] Moshe Gavrielov – an EDA and ASIC industry veteran who was appointed as president and CEO in early 2008 – introduced targeted design platforms to provide solutions that combine FPGAs with software, IP cores, boards and kits to address focused target applications.[12] These targeted design

platforms are an alternative to costly application-specific integrated circuits (ASICs) and application-specific standard products (ASSPs).

Today

The company has expanded its product portfolio since its founding. Xilinx sells a broad range of FPGAs, complex programmable logic devices (CPLD), design tools, intellectual property and reference designs. Xilinx also has a global services and training program. The company's products have been recognized by EE Times, EDN and others for innovation and market impact. Xilinx product lines (see Current Family Lines) include the Virtex, Kintex and Artix series, each including configurations and models optimized for different applications.[19] With the introduction of the Xilinx 7 Series in June, 2010, the company has moved to three major product families, the high-end Virtex , the mid-range Kintex family and the low-cost Artix family, retiring the Spartan brand, which ends with the Xilinx Series 6 FPGAs.

Technology

Xilinx designs, develops and markets programmable logic products including integrated circuits (ICs), software design tools, predefined system functions delivered as intellectual property (IP) cores, design services, customer training, field engineering and technical support. Xilinx sells both FPGAs and CPLDs programmable logic devices for electronic equipment manufacturers in end markets such as communications, industrial, consumer, automotive and data processing. Xilinx's FPGAs have been used for the ALICE (A Large Ion Collider Experiment) at the CERN European laboratory on the French-Swiss border to map and disentangle the trajectories of thousands of subatomic particles.[29]

Xilinx has also engaged in a partnership with the United States Air Force Research Laboratory's Space Vehicles Directorate to develop FPGAs to withstand the damaging effects of radiation in space for deployment in new satellites, which are 1,000 times less sensitive to space radiation than the commercial equivalent. The Virtex-II Pro, Virtex-4, Virtex-5, and Virtex-6

FPGA families are focused on system-on-chip (SoC) designers because they include up to two embedded IBM PowerPC cores. Some members of the Virtex-II Pro, Virtex-4 and Virtex-5 FPGA families contain PowerPC processor blocks.

Xilinx FPGAs can run a regular embedded OS (such as Linux or Vs Works) and can implement processor peripherals in programmable logic. Xilinx's IP cores include IP for simple functions (BCD encoders, counters, etc.), for domain specific cores (digital signal processing, FFT and FIR cores) to complex systems (multi-gigabit networking cores, Micro Blaze soft microprocessor, and the compact Picoblaze microcontroller). Xilinx also creates custom cores for a fee.

The ISE Design Suite is the central electronic design automation (EDA) product family sold by Xilinx. The ISE Design Suite features include design entry and synthesis supporting Verilog or VHDL, place-and-route (PAR), completed verification and debug using ChipScope Pro tools, and creation of the bit files that are used to configure the chip. Xilinx's Embedded Developer's Kit (EDK) supports the embedded PowerPC 405 and 440 cores (in Virtex-II Pro and some Virtex-4 and -5 chips) and the Micro blaze core. Xilinx's System Generator for DSP implements DSP designs on Xilinx FPGAs. A freeware version of its EDA software called ISE Web PACK is used with some of its non-high-performance chips.

Xilinx is the only (as of 2007) FPGA vendor to distribute a native Linux freeware synthesis tool chain. Xilinx announced the architecture for an Extensible Processing Platform, which licenses the ARM Cortex-A9 MP Core processor for embedded systems designers' familiar with the ARM platform. The Extensible Processing Platform architecture abstracts much of the hardware burden away from the embedded software developers' point of view, giving them an unprecedented level of control in the development process.

Because the system boots an OS at reset, software development can get under way quickly within familiar development and debug environments using tools such as ARM's Real View development suite and related third-party tools, Eclipse-based IDEs, GNU, the Xilinx Software Development Kit and others. The platform targets embedded designers working on market applications that require multifunctionality and real-time responsiveness, such as automotive driver assistance, intelligent video surveillance, industrial automation, aerospace and defense, and next-generation wireless.

Xilinx SRAM-based FPGAs

The basic structure of Xilinx FPGAs is array-based, meaning that each chip comprises a two dimensional array of logic blocks that can be interconnected via horizontal and vertical routing channels. An illustration of this type of architecture was shown in Figure.

Xilinx introduced the first FPGA family, called the XC2000 series, in about 1985 and now offers three more generations: XC3000, XC4000, and XC5000. Although the XC3000 devices are still widely used, we will focus on the more recent and more popular XC4000 family. We note that XC5000 is similar to XC4000, but has been engineered to offer similar features at a more attractive price, with some penalty in speed.

The XC4000 features a logic block (called a Configurable Logic Block (CLB) by Xilinx) that is based on look-up tables (LUTs). A LUT is a small one bit wide memory array, where the address lines for the memory are inputs of the logic block and the one bit output from the memory is the LUT output. A LUT with K inputs would then correspond to a 2K x 1 bit memory.

The XC4000 CLB contains three separate LUTs, in the configuration There are two 4-input LUTS that are fed by CLB inputs, and the third LUT can be used in combination with the other two. This arrangement allows the CLB to implement a wide range of logic functions of up to nine inputs, two separate functions of four inputs or other possibilities. Each CLB also contains two flip-flops.

4.3 CODE

The main code for design of dual channel multiplier is described below:

```
module Main_DCM_16 (X,Y,P);
  input [7:0]X,Y;
  output [15:0]P;
  wire
[7:0]PP0,PP1,PP2,PP3,PP4,PP5,PP6,PP7,PP8,PP9,PP10,PP11,PP12,PP13,PP
14,PP15;

  PP_Gen_16 MM0 (X,Y[0],PP0);
  PP_Gen_16 MM1 (X,Y[1],PP1);
  PP_Gen_16 MM2 (X,Y[2],PP2);
  PP_Gen_16 MM3 (X,Y[3],PP3);
  PP_Gen_16 MM4 (X,Y[4],PP4);
  PP_Gen_16 MM5 (X,Y[5],PP5);
  PP_Gen_16 MM6 (X,Y[6],PP6);
  PP_Gen_16 MM7 (X,Y[7],PP7);


  Adder_Unit_16 MM16 (PP0,PP1,PP2,PP3,PP4,PP5,PP6,PP7,P);


End module
```

SUMMARY

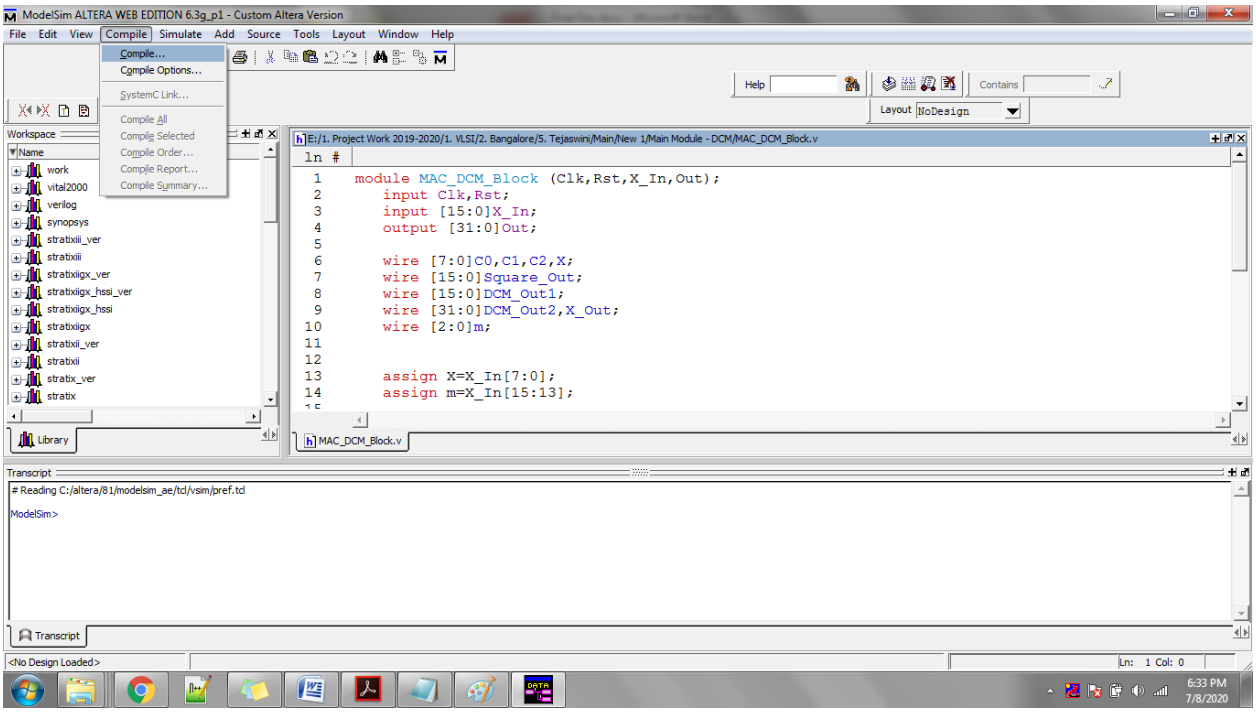
In this chapter software tool modelsim and code which is required for evaluation and stimulation of required wave forms for dual channel multiplier is described.

CHAPTER 5
SIMULATION RESULTS

The stimulation results after running the main module code in modelsim are shown below.

Simulation Process:

5.1 Compile Process



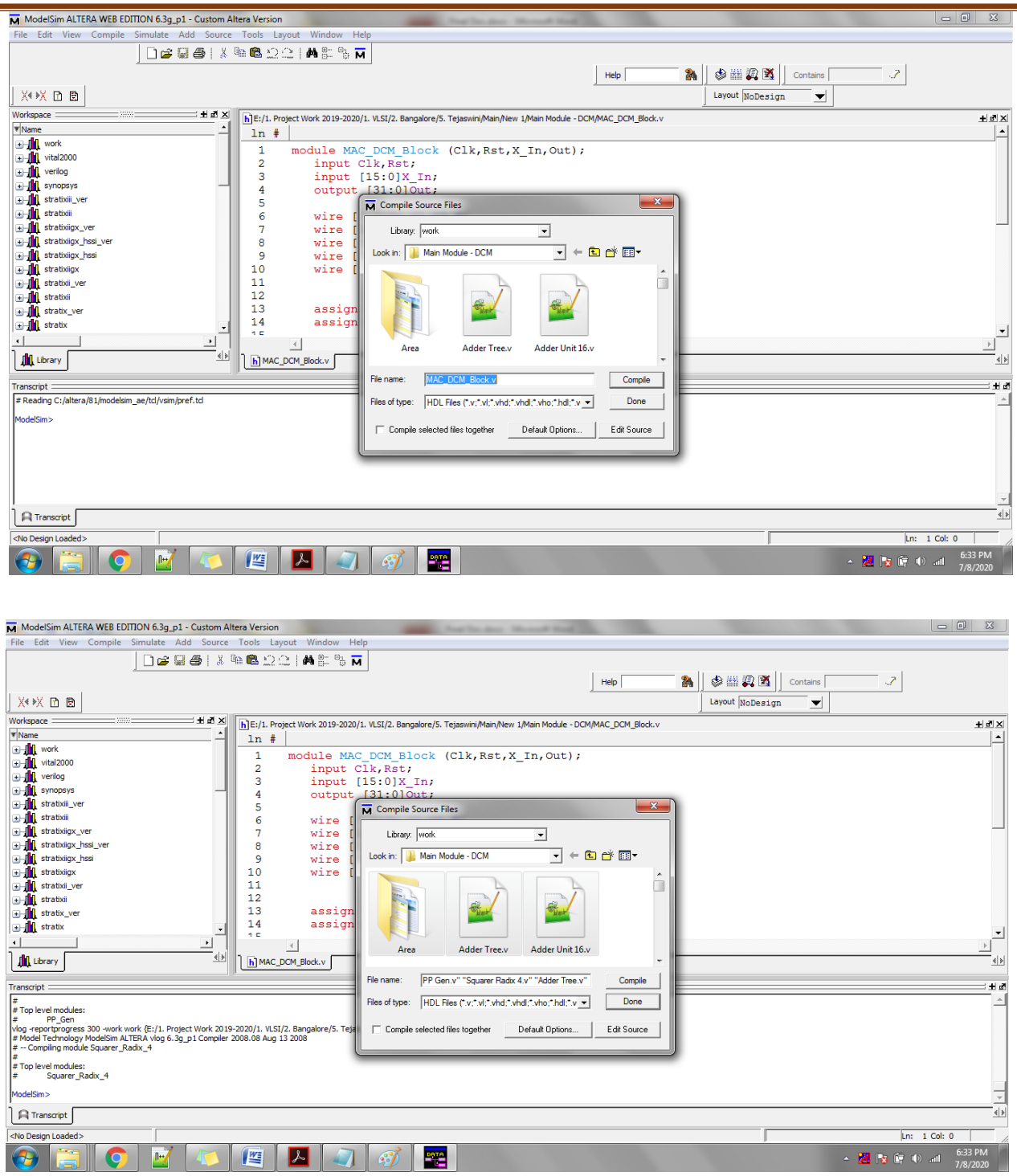


Figure 5. 1 Compilation

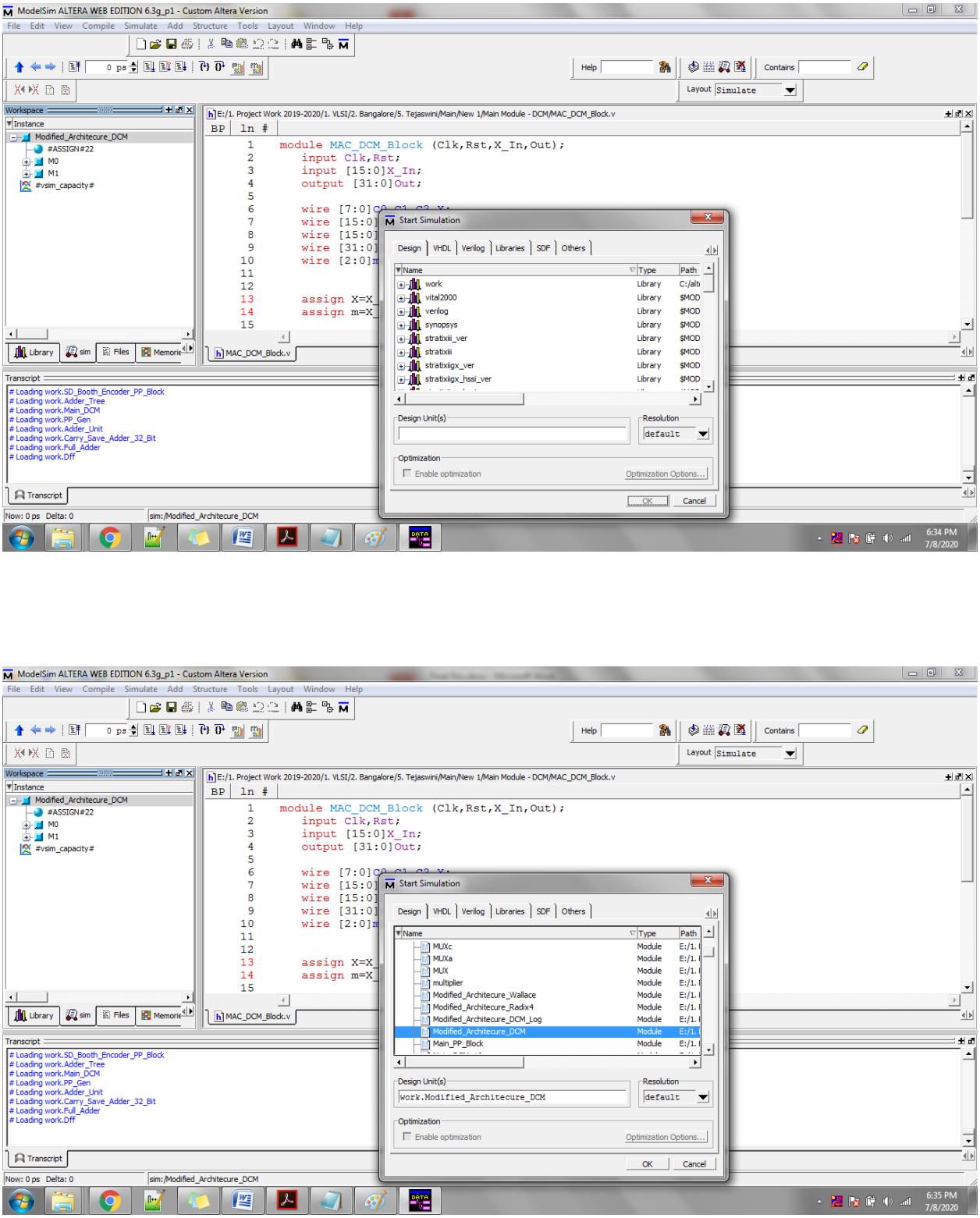
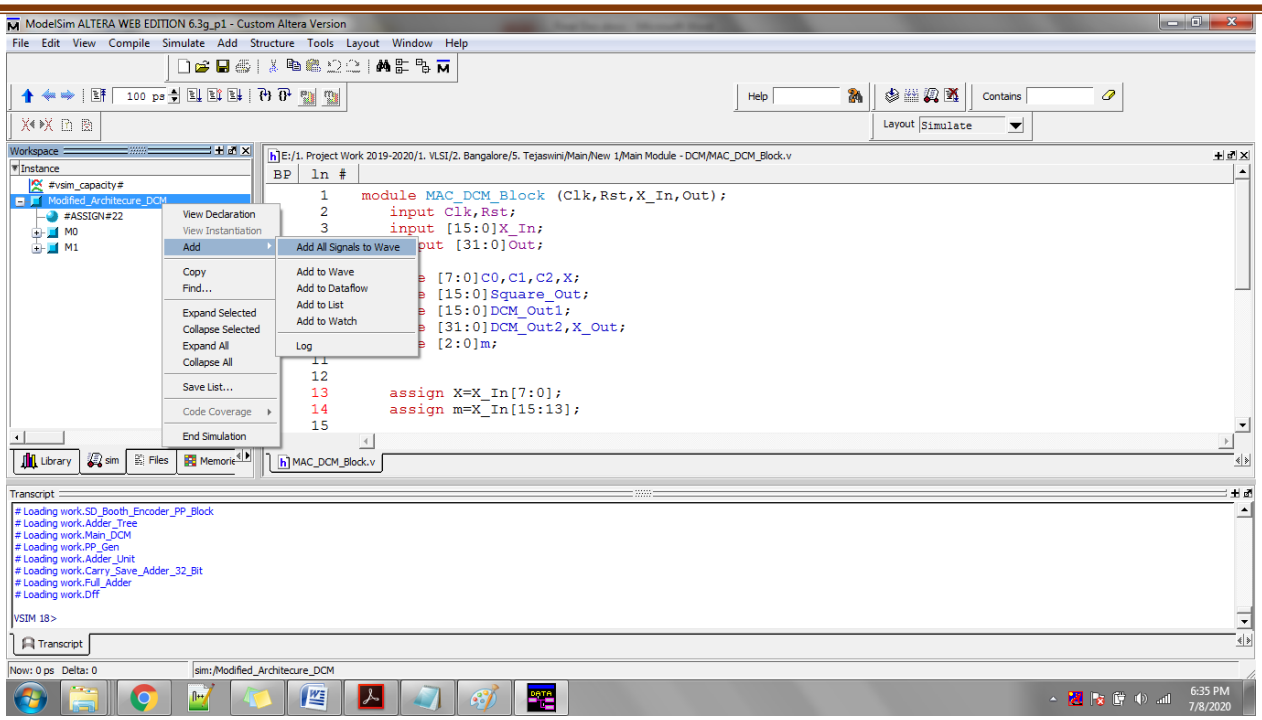


Figure 5. 2 Simulation



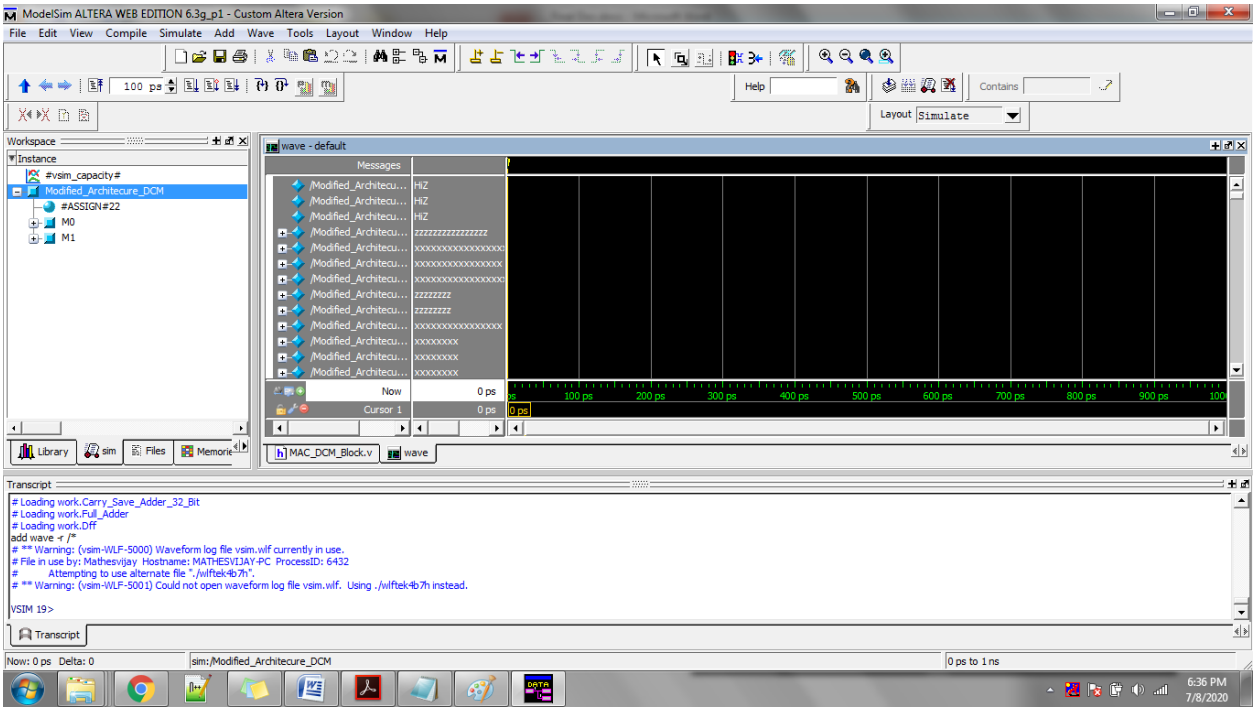
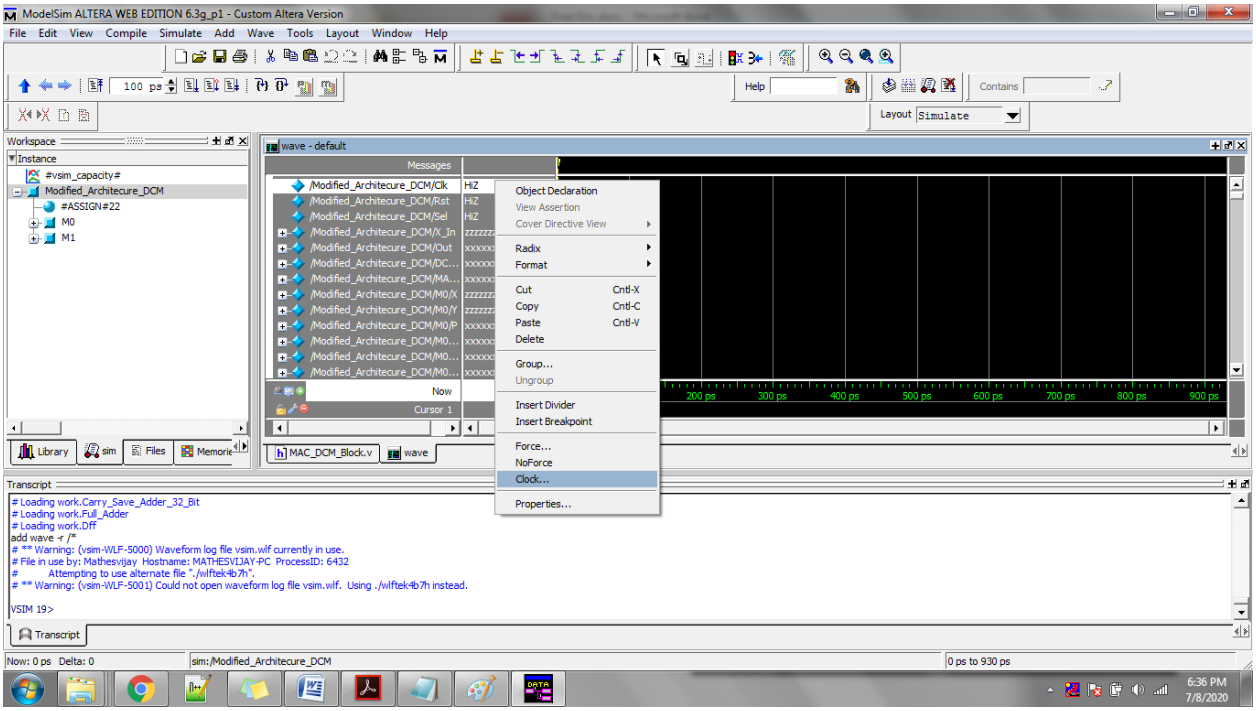


Figure 5. 3 Waveform Generation



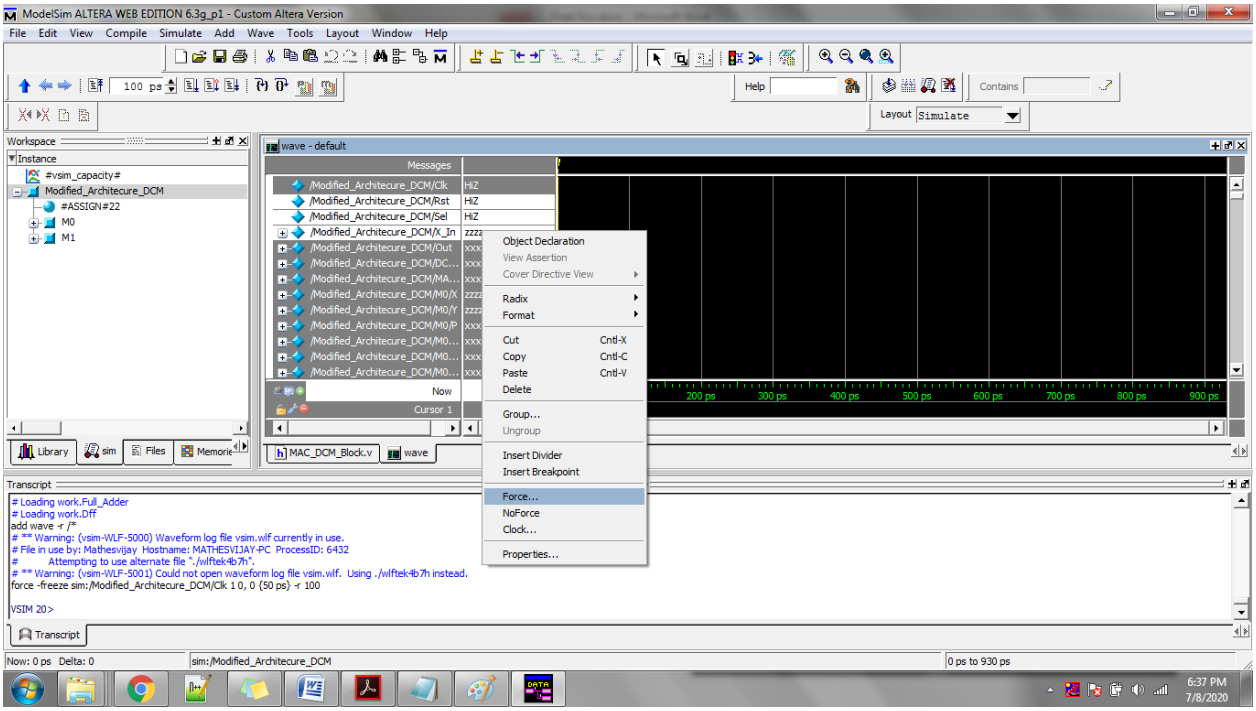


Figure 5. 4 Forcing the Inputs

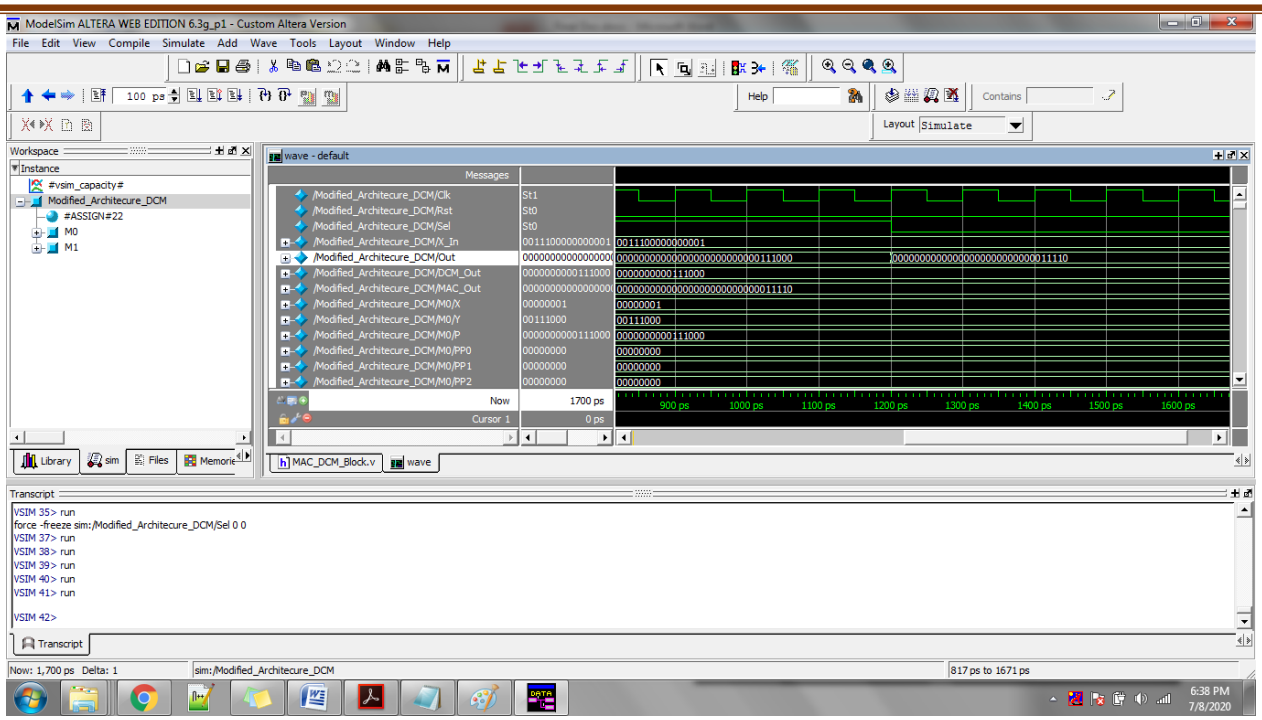


Figure 5. 5 Waveform Verification

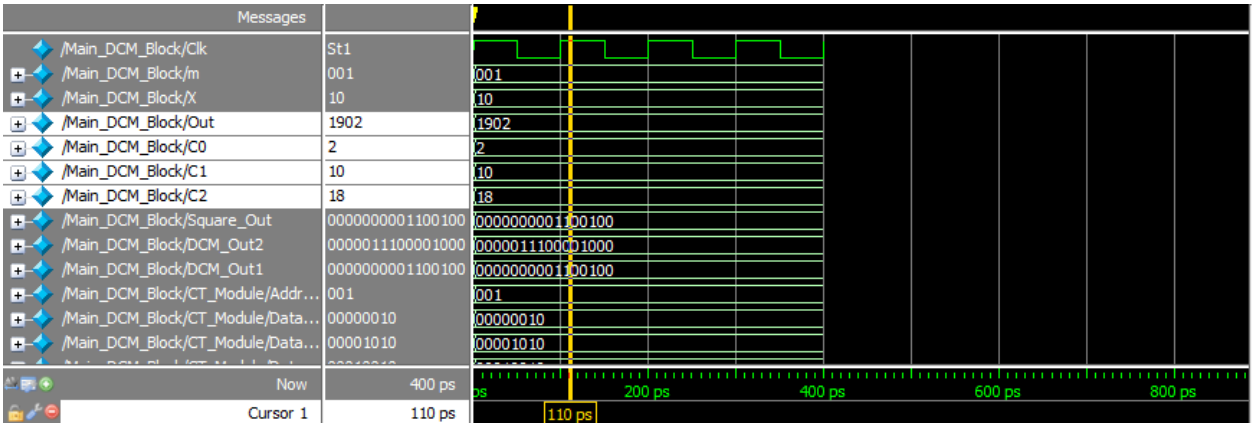


Figure 5. 6 Main Module

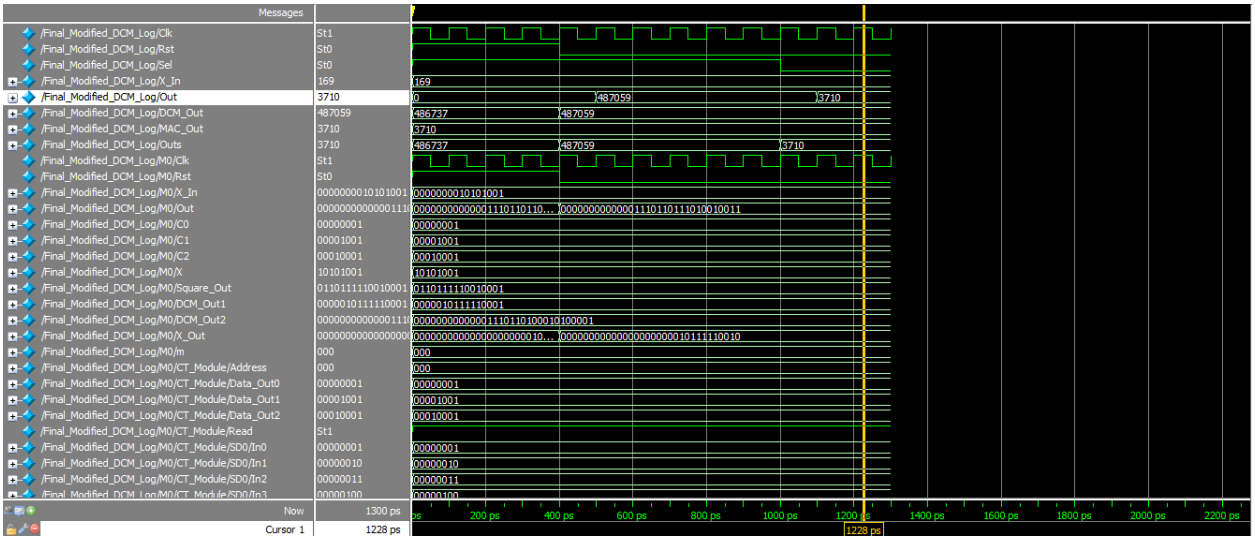


Figure 5. 7 Final DCM with log module

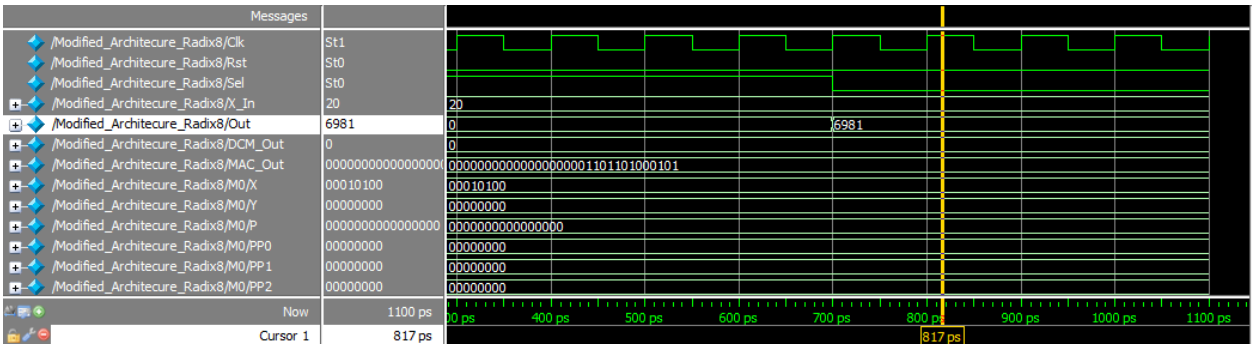


Figure 5. 8 Final DCM with Radix-8 Booth

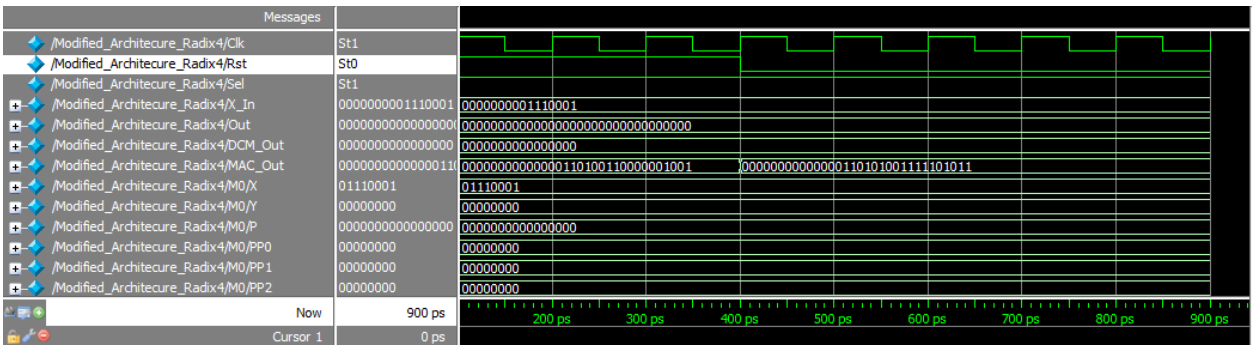


Figure 5. 9 Final DCM with Radix-4 Booth

DESIGN AND COMPARISION RESULTS

RTL FOR DCM:

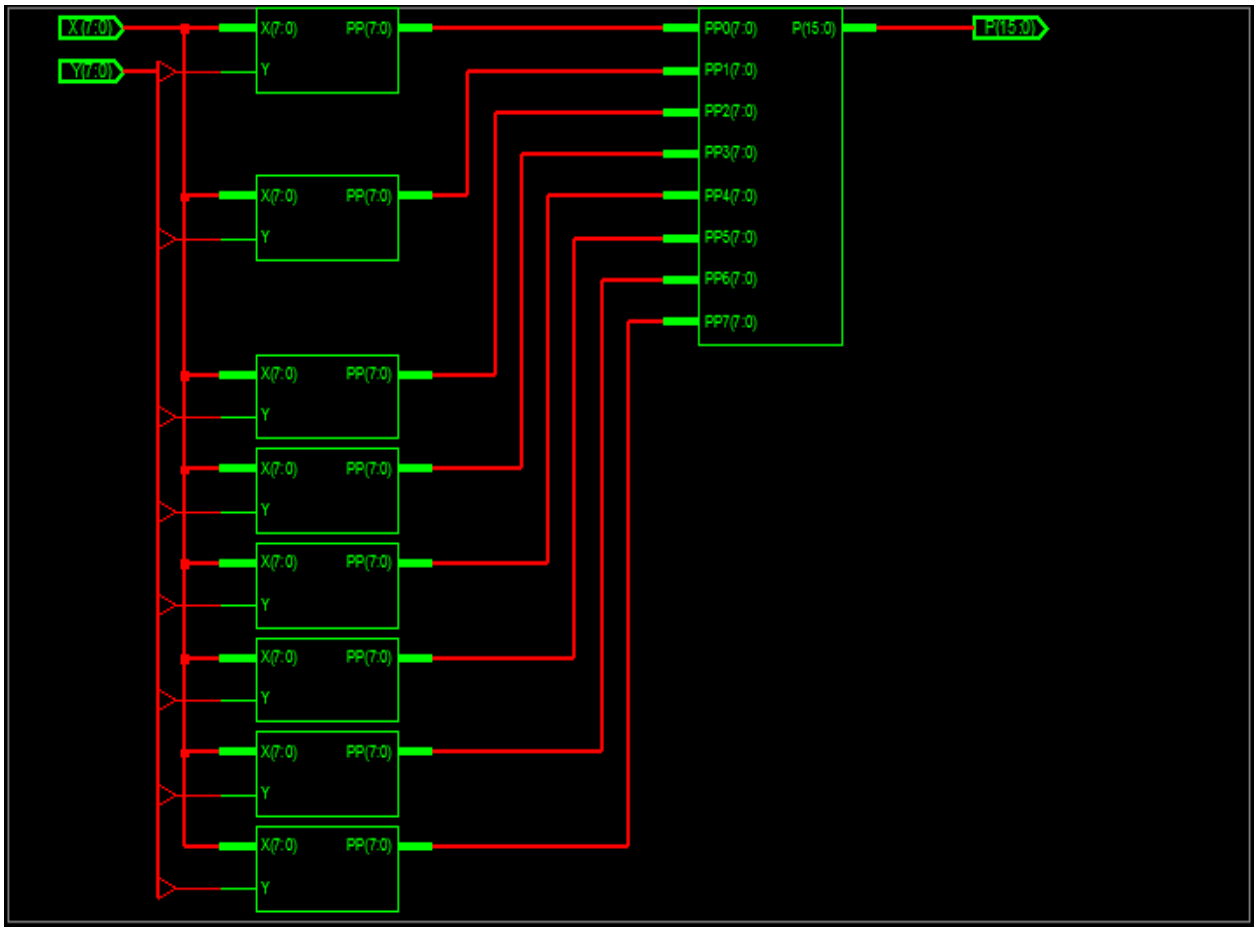


Figure 5. 10 RTL for DCM

5.2 POWER, DELAY AND AREA CALCULATION FOR DCM 16 AND DCM 32

We have calculated the power, delay and area of dual channel multiplier 16 and 32 by using Xilinx from the wave forms generated from modelsim software

POWER FOR DCM 32

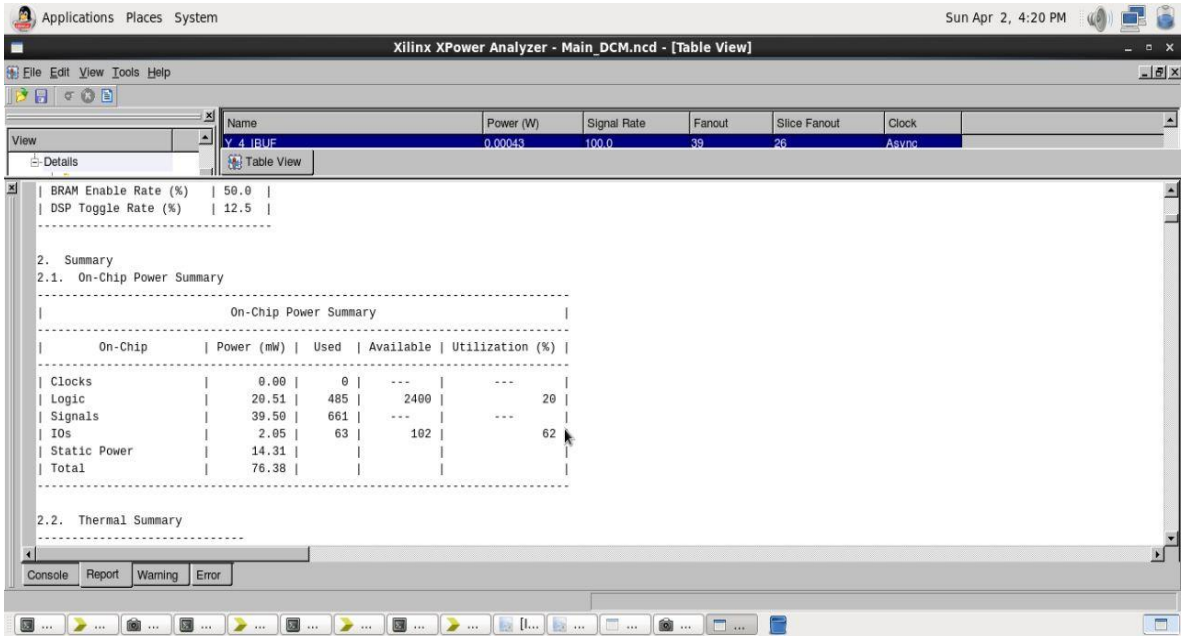


Figure 5. 11 power for dcm 32

POWER FOR DCM 16

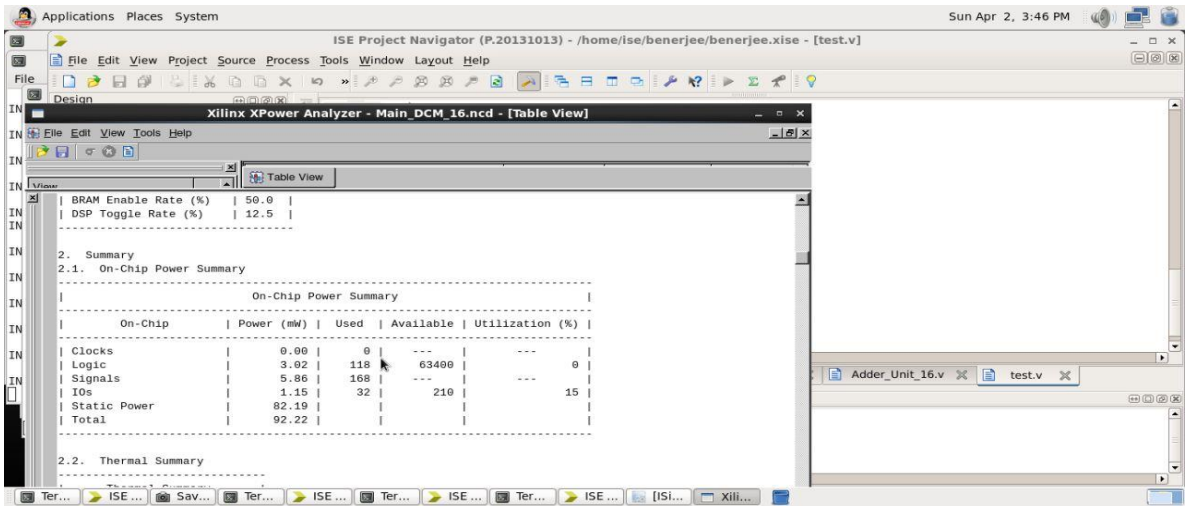


Figure 5. 12 Power for dcm 16

DELAY FOR DCM 16

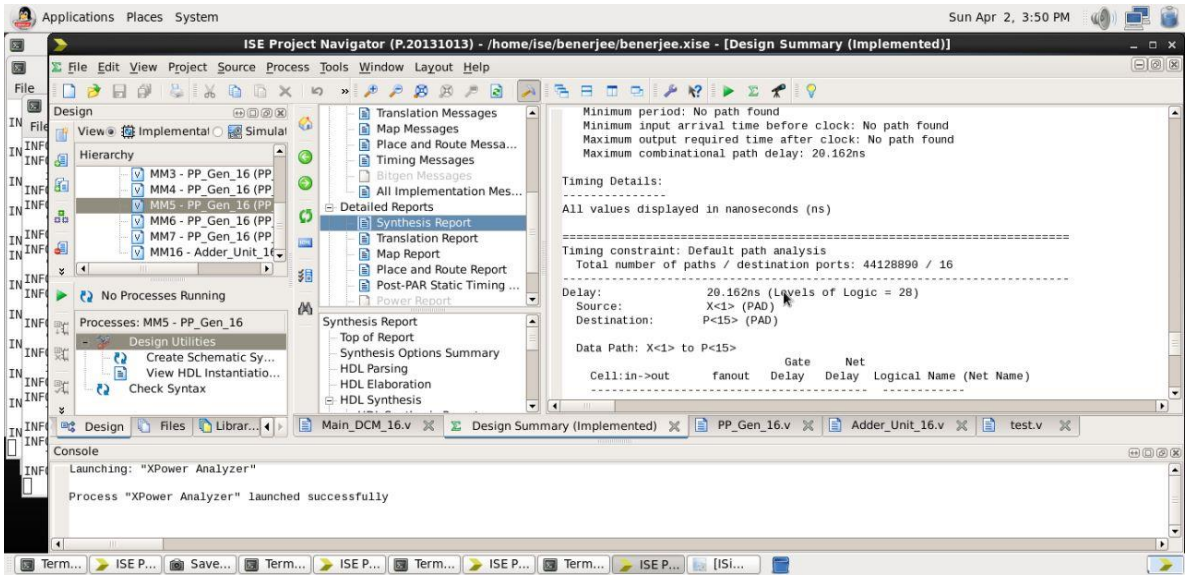


Figure 5. 13 Delay for dcm 16

DELAY FOR DCM 32

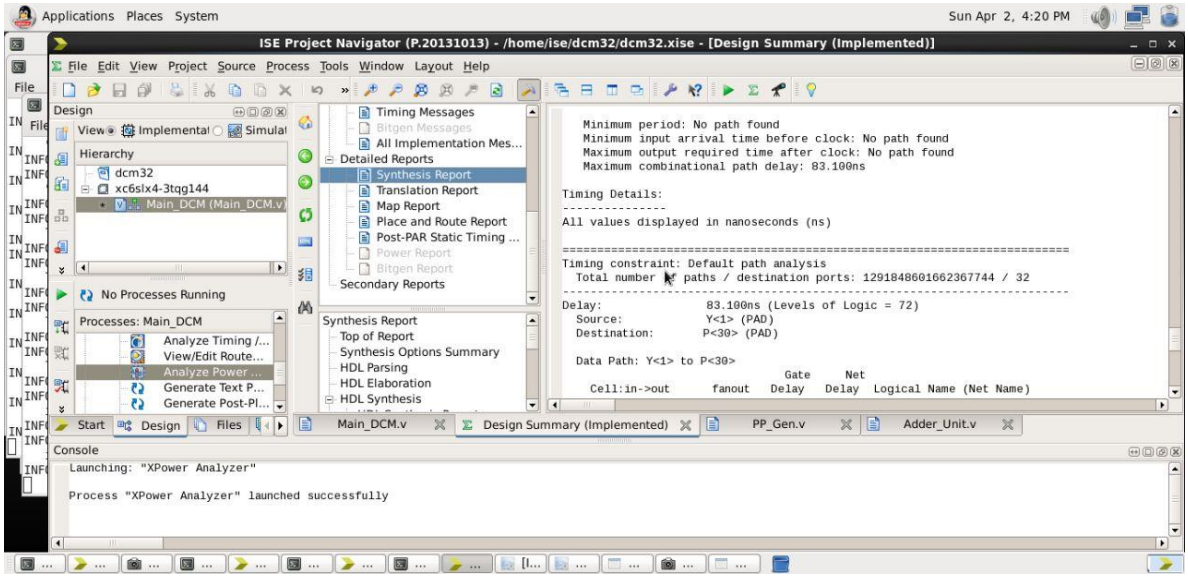


Figure 5. 14 Delay for dcm 32

POWER AND DELAY FOR BOOTH RADIX-8 MULTIPLIER

```
2.  Summary
2.1.  On-Chip Power Summary
--------------------------------------------------------------------------------
|                          On-Chip Power Summary                               |
--------------------------------------------------------------------------------
|        On-Chip        | Power (mW) |  Used  | Available | Utilization (%) |
--------------------------------------------------------------------------------
| Clocks                |       0.00 |      0 |    ---    |       ---       |
| Logic                 |      18.02 |    850 |      2400 |              35 |
| Signals               |      47.64 |   1058 |    ---    |       ---       |
| IOs                   |       2.05 |     63 |       102 |              62 |
| Static Power          |      14.37 |        |           |                 |
| Total                 |      82.09 |        |           |                 |
--------------------------------------------------------------------------------
```

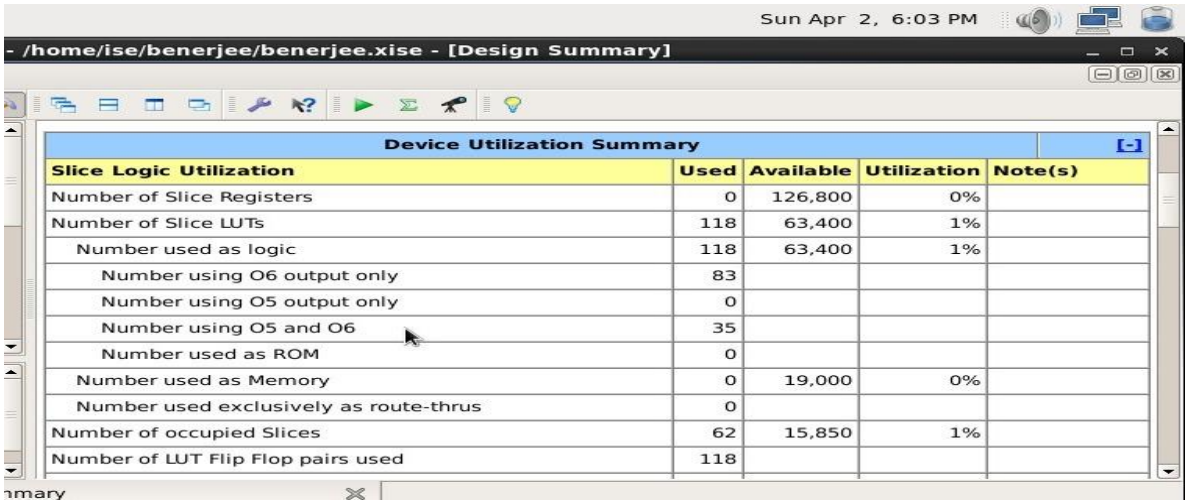
Figure 5. 15 Power for radix 8

DELAY FOR BOOTH RADIX-8 MULTIPLIER

```
Timing Details:
---------------
All values displayed in nanoseconds (ns)

================================================================================
Timing constraint: Default path analysis
  Total number of paths / destination ports: 16414711 / 32
--------------------------------------------------------------------------------
Delay:               14.004ns (Levels of Logic = 39)
  Source:            a<0> (PAD)
  Destination:       out<31> (PAD)
```

Figure 5. 16 Delay for radix 8



Device Utilization Summary				
Slice Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Number of Slice Registers	0	126,800	0%	
Number of Slice LUTs	118	63,400	1%	
Number used as logic	118	63,400	1%	
Number using O6 output only	83			
Number using O5 output only	0			
Number using O5 and O6	35			
Number used as ROM	0			
Number used as Memory	0	19,000	0%	
Number used exclusively as route-thrus	0			
Number of occupied Slices	62	15,850	1%	
Number of LUT Flip Flop pairs used	118			

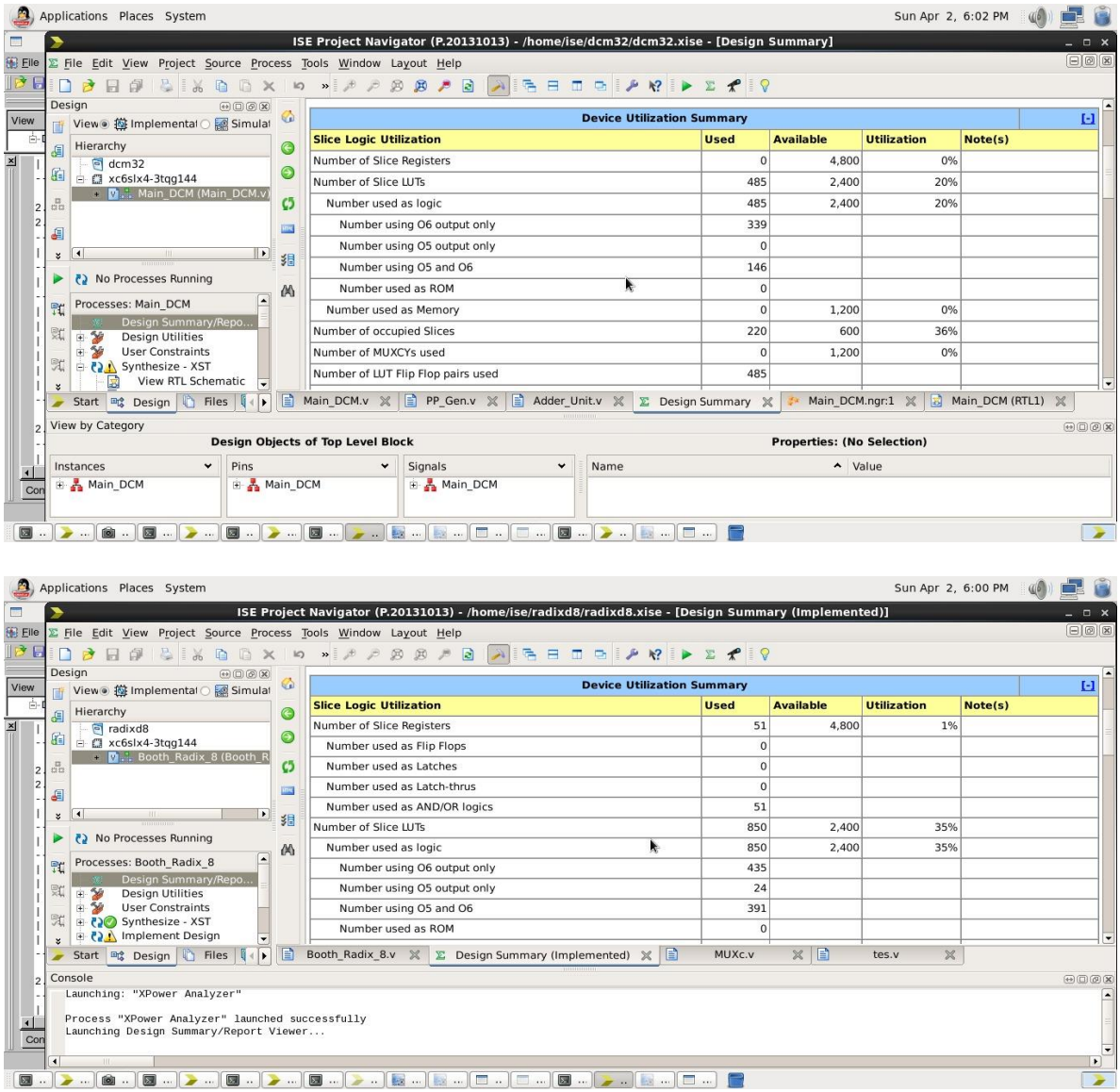


Figure 5.17 Area for dcm 16,dcm 32, radix 8

TABLE 5.1: COMPARISON DCM AND BOOTH MULTIPLIER

MULTIPLIER	Power(mw)	Delay(ns)
DCM 16	92	20.162
DCM 32	76	83.100
RADIX 8	82	14.00

CHAPTER 6
CONCLUSION AND FUTURE SCOPE

6.1 CONCLUSION

The implementation of PWP evaluation in the SFU of the GPUs can be highly improved by boosting the performance of multiplication and squaring unit which are the basic components in the evaluation process. Exploiting the serial/parallel algorithm, an energy efficient DCM is proposed in this paper. Comparisons with the well-known multiplication schemes have demonstrated savings in area, power, and energy. Moreover, for large operand input size, MAC is utilized to replace the traditional parallel multiplier scheme. MAC is proposed to perform the computation of the overall approximated polynomial without the need for multiplication in SFUs. The proposed scheme can implement different functions using simple hardware design. Also, high-speed dedicated squaring unit is proposed. It can be readily applied to any number of bits. PWP function evaluation by MAC is implemented using FPGA

6.2 FUTURE SCOPE

A dual channel multiplier is a type of electronic circuit this multiply two input signals, producing an output signal that is the product of the two inputs. The future scope of such a project largely depends on the specific application and context in which it is being used.

Here are a few potential future scopes for a dual channel multiplier project:

Digital signal processing (DSP) applications: Dual channel multipliers can be used in a wide range of DSP applications, including digital filters, mixers, and modulators. As the demand for DSP continues to grow in various industries, including telecommunications, multimedia, and consumer electronics, the demand for dual channel multipliers is also likely to increase.

Audio and music applications: Dual channel multipliers can be used in audio and music applications to control the gain and amplitude of signals. As the demand for high-quality audio processing and mixing equipment

continues to increase, the use of dual channel multipliers is also likely to grow.

Control systems: Dual channel multipliers can also be used in control systems to adjust the gain of feedback signals. As automation and robotics continue to become more prevalent in various industries, the use of control systems is also likely to increase, leading to a growing demand for dual channel multipliers.

Overall, the future scope of a dual channel multiplier project is likely to be shaped by the continued growth and development of various industries that rely on signal processing, control systems, and other applications that require the manipulation of analog signals.

REFERENCES

[1]. Y.-J. Chen et al., "A 130.3 mW 16-core mobile GPU with power-aware pixel approximation techniques," IEEE J. Solid-State Circuits, vol. 50, no. 9, pp. 2212–2223, Sep. 2015.

[2]. S. F. Hsiao, P.-H.Wu, C.-S. Wen, and P. K. Meher, "Table size reduction methods for faithfully rounded lookup-table-based multiplierless function evaluation," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 62, no. 5, pp. 466–470, May 2015.

[3]. H. Kim, B.-G. Nam, J.-H. Sohn, J.-H. Woo, and H.-J. Yoo, "A 231-MHz, 2.18-mW 32-bit logarithmic arithmetic unit for fixed-point 3-D graphics system," IEEE J. Solid-State Circuits, vol. 41, no. 11, pp. 2373–2381, Nov. 2006.

[4]. S.-F. Hsiao, C.-S. Wen, Y.-H. Chen, and K.-C. Huang, "Hierarchical multipartite function evaluation," IEEE Trans. Comput., vol. 66, no. 1, pp. 89–99, Jan. 2017.

[5]. I. Koren and O. Zinaty, "Evaluating elementary functions in a numerical coprocessor based on rational approximations," IEEE Trans. Comput., vol. 39, no. 8, pp. 1030–1037, Aug. 1990.

[6]. D. De Caro, N. Petra, and A. G. M. Strollo, "High-performance special function unit for programmable 3-D graphics processors," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 56, no. 9, pp. 1968–1978, Sep. 2009.

[7]. S. A. Tawfik and H. A. H. Fahmy, "Algorithmic truncation of minimax polynomial coefficients," in Proc. IEEE Int. Symp. Circuits Syst., May 2006, p. 2427.

[8]. G. Cao, H. Du, P. Wang, Q. Du, and J. Ding, "A piecewise cubic polynomial interpolation algorithm for approximating elementary function," in Proc. 14th Int. Conf. Comput.-Aided Design Comput. Graph., Aug. 2015, pp. 57–64.

[9]. A. G. M. Strollo, D. De Caro, and A. Petra, "Elementary functions hardware implementation using constrained piecewise-polynomial approximations," IEEE Trans. Comput., vol. 60, no. 3, pp. 418–432, Mar. 2011.

[10]. M. Sadeghian, J. E. Stine, and E. G. Walters, "Optimized linear, quadratic and cubic interpolators for elementary function hardware implementations," Electronics, vol. 5, no. 2, p. 17, Apr. 2016.