

**CERTAIN INVESTIGATIONS ON DESIGN AND
IMPLEMENTATION OF DIGITAL FIR FILTER
USING LOW POWER AND HIGH SPEED
MULTIPLIERS**

A THESIS

Submitted by

CHINNAPPARAJ S

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY



**FACULTY OF INFORMATION AND
COMMUNICATION ENGINEERING**

ANNA UNIVERSITY

CHENNAI 600 025

AUGUST 2019



CENTRE FOR RESEARCH
ANNA UNIVERSITY, CHENNAI-600 025



CERTIFICATE

This is to certify that all corrections and suggestions pointed out by the Indian /Foreign Examiner(s) are incorporated in the Thesis titled " CERTAIN INVESTIGATIONS ON DESIGN AND IMPLEMENTATION OF DIGITAL FIR FILTER USING LOW POWER AND HIGH SPEED MULTIPLIERS " submitted by Mr. Chinnapparaj.S

D. Somasul 26.08.19
Signature of the Supervisor

Place : **COIMBATORE**

Date : **26.08.2019**



CENTRE FOR RESEARCH

ANNA UNIVERSITY, CHENNAI-600 025



Proceedings of the Ph.D. Viva-Voce Examination of Mr.Chinnapparaj.S held at 10.30 AM on 26.08.2019 in Seminar Hall, SriGuru Institute of Technology, Coimbatore

The Ph.D. Viva-Voce Examination of Mr.Chinnapparaj.S (Reg. No. 71010521004) on his/her Ph.D. Thesis Entitled " CERTAIN INVESTIGATIONS ON DESIGN AND IMPLEMENTATION OF DIGITAL FIR FILTER USING LOW POWER AND HIGH SPEED MULTIPLIERS " was conducted on **26.08.2019** at 10.30 AM in the Seminar Hall, SriGuru Institute of Technology, Coimbatore.

The following Members of the Oral Examination Board were present:

- | | |
|---|-----------------|
| 1. Dr. N.P.Gopalan,Professor,Department of Computer Applications,National Institute of Technology,Tiruchirappalli - 620 015 | Indian Examiner |
| 2. Dr. G Zayaraz,Professor,Department of Computer Science and Engineering,Pondicherry Engineering College,Puducherry - 605 014 | Subject Expert |
| 3. Dr. Somasundareswari.D,Professor, Department of Electrical and Electronics Engineering, Sriguru Institute of Technology,Coimbatore | Supervisor |

The research scholar, Mr. Chinnapparaj.S presented the salient features of his/her Ph.D. work. This was followed by questions from the board members. The questions raised by the Foreign and Indian Examiners were also put to the scholar. The scholar answered the questions to the full satisfaction of the board members.

The corrections suggested by the Indian/Foreign examiner have been carried out and incorporated in the Thesis before the Oral examination.

Based on the scholars research work, his/her presentation and also the clarifications and answers by the scholar to the questions, the board recommends that Mr.Chinnapparaj.S be awarded Ph.D. degree in the **Faculty of Information and Communication Engineering.**


Indian Examiner


Subject Expert


Supervisor **26/08/19**

ANNA UNIVERSITY**CHENNAI 600 025****CERTIFICATE**

The research work embodied in the present Thesis entitled **“CERTAIN INVESTIGATIONS ON DESIGN AND IMPLEMENTATION OF DIGITAL FIR FILTER USING LOW POWER AND HIGH SPEED MULTIPLIERS”** has been carried out in the Department of Electrical and Electronics Engineering, Srirguru Institute of Technology, Coimbatore. The work reported herein is original and does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion or to any other scholar.

I understand the University’s policy on plagiarism and declare that the thesis and publications are my own work, except where specifically acknowledged and has not been copied from other sources or been previously submitted for award or assessment.



CHINNAPPARAJ S
RESEARCH SCHOLAR



Dr. D. SOMASUNDARESWARI
SUPERVISOR

Professor

Department of Electrical and
Electronics Engineering

Srirguru Institute of Technology,
Coimbatore

ABSTRACT

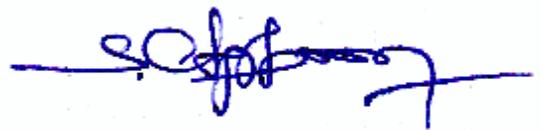
Finite Impulse Response (FIR) filter is one of the filters used in Digital Signal Processing where its impulse response duration is finite and it resolves to zero in finite time. Since FIR filter provides a finite output for finite input, it is always stable and this property is more useful in various DSP applications. FIR filter can have a huge number of coefficients for a preferred frequency response with tight constraints on the transition band, pass band and stop band. The phase of the FIR filter is a linear function of the frequency because it has the linear phase property. Here the signals of all frequencies are delayed by the same amount of time, thereby eliminating the possibility of phase distortion. This property implies that FIR can be used in audio applications. The finite I/O of FIR filter a stable property which is also very much useful in DSP.

The chief objective of this study is to design an efficient FIR filter for optimizing the power, area, delay, and speed, suitable for various DSP applications. Due the efficiencies and advantages of the FIR filter, this research work is motivated to design and implement a direct form of FIR filter using effective multipliers and adder circuits for optimizing the power, area, delay and speed in DSP. Most of the existing research works were focused on utilizing FIR filter as an important component in various communications, DSP and in portable applications. In order to do this the entire research work is focused in two different stages such as: (i). Efficient Multiplication and Accumulation (MAC) in digital FIR filter is intended to proliferate the speediness and throughput of digital FIR filter. (ii). Design and implement the direct form FIR filter by incorporating reduced full adder and half adder into Wallace Multiplier and improved Carry- Save adder for digital FIR filter.

ACKNOWLEDGEMENT

First of all, I heartily thank the great almighty for his showers of blessings and for providing me the good health and self-confidence to do our project work successfully. I wish to record my deep sense of gratitude and profound thanks to my research supervisor **Dr.D.Somasundareswari**, Principal, Sriguru Institute of Technology, Coimbatore, for his keen interest, inspiring guidance, constant encouragement with my work during all stages, to bring this thesis into fruition. Besides my advisor, I would like to thank the rest of my thesis committee members **Dr.V.Duraiswamy**, Principal, The Kavery Engineering College, Salem & **Dr.V.Manikandan**, Professor, Department of Electrical and Electronics Engineering, Coimbatore Institute of Technology, Coimbatore for their valuable advice, constructive criticism and their extensive discussions around my work.

I express my thanks to the Managing Trustee **Smt.Sarasuwathy Khannaiyann**, for providing the essential infrastructure and helping me to carry out this project. I am extremely indebted to **Dr.C.Natarajan, M.E. Phd.**, Principal, Hindusthan Institute of Technology, Coimbatore, for their valuable suggestions and support during the course of my research work. I thank each and everyone who have helped and guided me in all aspects for the completion of my project. I also thank the faculty members and non-teaching staff members of the Department of Electronics and Communication Engineering, Hindusthan Institute of Technology, Coimbatore and my family for their valuable support throughout the course of my research work.



CHINNAPPARAJ S

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	v
	LIST OF TABLES	xii
	LIST OF FIGURES	xiii
	LIST OF SYMBOLS AND ABBREVIATIONS	xvi
1	INTRODUCTION	1
1.1	OBJECTIVES	1
1.2	INTRODUCTION	1
1.3	RESEARCH PROBLEM	4
1.4	RESEARCH OBJECTIVES	5
1.5	RESEARCH METHODOLOGY	6
1.6	THESIS ORGANIZATION	7
1.7	NEED OF THE RESEARCH	8
1.8	POWER USAGE IN CMOS	10
1.8.1	Static Power Consumption	10
1.9	DIGITAL FILTERS	13
1.9.1	General Purpose FIR Filter	15
1.9.2	MAC UNIT	16
1.9.3	Frequency Response	21
1.10	FINITE IMPULSE RESPONSE	24
1.10.1	FIR vs IIR filtering	26

CHAPTER NO.	TITLE	PAGE NO.
1.10.2	Infinite impulse response (IIR) filters	26
1.10.3	Finite impulse response (FIR) filters	26
1.10.4	Examples of FIR and IIR	27
1.10.5	Crossover filter	28
1.10.6	Parametric filter	29
1.11	SUMMARY	30
2	LITERATURE SURVEY	31
2.1	A REVIEW ON FIR FILTER	31
2.1.1	Multipliers	33
2.2	A REVIEW ON FIR BASED APPLICATIONS	34
2.2.1	Modified FIR Filter	47
2.2.2	A Review on SQRT-CSLA Based FIR Filter	49
2.3	RECENT SURVEY ON FIR FILTER	53
2.4	SUMMARY	55
3	WALLACE MULTIPLIER WITH KOGGE- STONE ADDER	56
3.1	INTRODUCTION	56
3.1.1	Compressors	61
3.2	MULTIPLIER TOPOLOGIES	63

CHAPTER NO.	TITLE	PAGE NO.
3.2.1	Booth Multiplier	64
	3.2.1.1 Booth Recoding	64
	3.2.1.2 Booth example	65
3.2.2	Modified Booth Algorithm	67
3.2.3	Wallace Tree multiplier	70
3.2.4	Dadda Multiplier Architecture	73
3.2.5	Reduced complexity Wallace multiplier	74
3.3	IMPLEMENTATION OF KOGGE-STONE ADDER WITH REDUCED COMPLEXITY WALLACE MULTIPLIER	75
3.3.1	Kogge-stone Adder	75
	3.3.1.1 Preprocessing	76
	3.3.1.2 Carry look ahead network	76
	3.3.1.3 Post processing	77
3.4	RESULTS AND DISCUSSIONS	77
3.5	SUMMARY	80
4	HIGH SPEED MULTIPLICATION AND ACCUMULATION (MAC) DESIGN FOR DIGITAL FIR FILTER	81
4.1	OBJECTIVES	81
4.2	PROBLEM STATEMENT	81
4.3	EXISTING REDUCED COMPLEXITY	

CHAPTER NO.	TITLE	PAGE NO.
	WALLACE MULTIPLIER	82
4.4	MODIFIED SQRT CSLA	83
4.5	REDUCED COMPLEXITY WALLACE MULTIPLIER USING MODIFIED SQRT CSLA	84
4.6	PROPOSED DIRECT FORM FIR FILTER	86
4.7	RESULTS AND DISCUSSIONS	87
4.8	SUMMARY	90
5	INCORPORATION OF REDUCED FULL ADDER AND HALF ADDER INTO WALLACE MULTIPLIER AND IMPROVED CARRY- SAVE ADDER FOR DIGITAL FIR FILTER	91
5.1	OBJECTIVES	91
5.2	PROBLEM STATEMENT	92
5.3	REDUCED FULL ADDER AND HALF ADDER STRUCTURE	93
5.4	IMPROVED 16-BIT CARRY-SAVE ADDER	95
5.5	ENHANCED WALLACE MULTIPLIER	98
5.6	PROPOSED DIRECT FORM DIGITAL FIR FILTERS	100
5.7	RESULTS AND DISCUSSION	101
5.8	SUMMARY	103

CHAPTER NO.	TITLE	PAGE NO.
6	PERFORMANCE EVALUATION	105
6.1	OBJECTIVES	105
6.2	SQRT CSLA BASED FIR FILTER	105
6.3	PROPOSED DIRECT FORM OF FIR FILTER	108
6.4	SUMMARY	112
7	CONCLUSION AND FUTURE WORK	114
7.1	CONCLUSION	114
7.2	FUTURE WORK	116
	REFERENCES	117
	LIST OF PUBLICATIONS	122

LIST OF TABLES

TABLE NO.	TITLE	PAGE NO.
3.1	Booth Algorithm	65
3.2	Booth Example	66
3.3	Modified booth algorithms	68
3.4	Modified booth example	69
3.5	Comparison of Delay and Power of Proposed multiplier.	78
4.1	Comparison of BEC based SQR T CSLA and modified SQR T CSLA	89
4.2	Reduced complexity Wallace multiplier and Modified SQR T-CSLA	89
5.1	Comparison between conventional CSA and improved CSA	102
5.2	Comparison of conventional Wallace multiplier and modified Wallace multiplier	102
5.3	Comparison between proposed direct form FIR filter and conventional direct form FIR filter	103

LIST OF FIGURES

FIGURE NO.	TITLE	PAGE NO.
1.1	Switching power usage in CMOS	12
1.2	FIR filter diagram	14
1.3	RTL representation of FIR filter	14
1.4	General purpose FIR filter	15
1.5	MAC unit	18
1.6	Basic steps for deriving MAC unit	19
1.7	MAC unit architecture	20
1.8	The reaction of a lowpass filter to various input frequencies	24
1.9	The logical structure of FIR filter	25
1.10	Phase Response	29
1.11	Parametric Filter Response	29
1.12	Phase Shift Response	30
3.1	Generic multiplier block diagram	58
3.2	Array Multiplier Mechanisms	59
3.3	Partial product addition using tree topology	60
3.4	Generic compressor.	61
3.5	Gate level design of (3:2) compressor	62
3.6	(4:2) Compressor logic diagram	63
3.7	(4:2) Compressor using (3:2) compressor	63
3.8	$n \times n$ modified booth multiplier	70
3.9	Wallace tree examples	72

FIGURE NO.	TITLE	PAGE NO.
3.10	Dot Diagram of an 8x8-bit Dadda Multiplier	74
3.11	Reduced Complexity Wallace multiplier	75
3.12	5. 8-bit Kogge-Stone adder's carry generation stage	77
3.13	Reduced Wallace multiplier with Kogge Stone Adder (KSA)	78
3.14	Comparison chart for Delay	79
3.15	Comparison Chart for Power	79
4.1	Architecture of modified SQRT CSLA	85
4.2	Block diagram of 16-bit modified SQRT CSLA	85
4.3	Partial products generation of reduced complexity Wallace multiplier	86
4.4	Direct Form of FIR Filter	87
4.5	Performance of both BEC based SQRT CSLA and modified SQRT CSLA	88
4.6	Performance of existing and proposed reduced complexity Wallace multiplier	88
5.1	Structures of reduced Half Adder and Full Adder 5.1(A) Reduced Half Adder, 5.1(B) Reduced Full Adder	94
5.2	Architecture of enhanced 16-bit carry-save adder using modified 5-bit BEC structure and parallel processing	97
5.3	Structure of modified 5-bit binary to excess one code (BEC) converter	98
5.4	Reduced complexity Wallace multiplier	99
5.5	General structure of direct form digital FIR filter	100

FIGURE NO.	TITLE	PAGE NO.
5.6	Simulation result of proposed digital FIR filter	103
6.1	Comparison of BEC based SQRT CSLA and modified SQRT CSLA	106
6.2	Performance of both BEC based SQRT CSLA and modified SQRT CSLA	107
6.3	Comparison of both reduced complexity Wallace multiplier with help of BEC based SQRT CSLA and modified SQRT CSLA	107
6.4	Performance of existing and proposed reduced complexity Wallace multiplier	108
6.5	Comparison between conventional CSA and improved CSA	109
6.6	Comparison of conventional Wallace multiplier and modified Wallace multiplier	110
6.7	Comparison between proposed direct form FIR filter and conventional direct form FIR filter	111
6.8	Simulation result of proposed digital FIR filter	112

LIST OF SYMBOLS AND ABBREVIATIONS

CSD	-	Canonic Signed Digit
CG	-	Carry Generation
CLA	-	Carry Look Ahead
CSA	-	Carry Save Adder
CS	-	Carry Selection
CMOS	-	Complementary Metal-Oxide-Semiconductor
CCFF	-	Conditional Capture Flip-Flop
DSP	-	Digital Signal Processor
EDP	-	Energy Delay Product
FFT	-	Fast Fourier Transformation
FPGA	-	Field Programmable Gate Array
FIR	-	Finite Impulse Response
HSG	-	Half Sum Generation
ICSA	-	Improved Carry Save Adder
IIR	-	Infinite Impulse Response
KSA	-	Kogge-Stone Adder
LTI	-	Linear Time Invariant
MOS	-	Metal-Oxide-Semiconductor
MSM	-	Multiple Constant Multiplication
MAC	-	Multiply-And-Accumulate
NMOS	-	N-Type Metal-Oxide-Semiconductor
PMOS	-	P-Type Metal-Oxide-Semiconductor

- SDR - Software Defined Radio
- SQRT CSLA - Square Root Carry Select Lookahead Adder
- VHDL - Verilog Hardware Description Language
- VLSI - Very Large-Scale Integrated Circuits

CHAPTER 1

INTRODUCTION

1.1 OBJECTIVES

The key objective of this research work is to design and implement a novel architecture of a FIR filter in order to save the power, memory, delay, complexity and increase the throughput. To manage that, this chapter offers detailed information about FIR filter, applications, various designs and merits and demerits of FIR filter comparing with other filters. Also, this chapter provides the importance of the FIR filter in DSP applications. From this chapter it can be able to understand the concepts and functionalities of FIR filter.

1.2 INTRODUCTION

Digital signal processing is one of most emerging and increasing popular research which frequently used in recent days. An efficient utilization of area, power consumption and speed of the multimedia applications are increased by the extensive use of FIR filters. To change the signal behaviors within a time interval and in a frequency domain, FIR filters are mainly used in signal processing. Hence, as a basic DSP element, it is recognized. In terms of commercial processors, DSP applications are getting more prominence. Other than conventional processors, DSP processors have more features and novel architectures. To design the FIR filter-based processors, there are more



algorithms are required due to the large demand for these unique features of DSP processors. DSP processing like filtering, convolution and inner products, multiplier and multiplier-accumulator (MAC) are the important elements. In MAC unit, the sum of products is calculated, whereas it is the heart of algorithms like FIR and FFT. To obtain high performance, the capability of MAC is playing a vital role in DSP.

To implement the Multiply-And-Accumulate (MAC) blocks that constitute the central piece in FIR filters and several functions, the design methods are mainly focused in multiplier-based architectures in DSP. For different DSP applications, FIR filters are very much important building blocks. It is essential to provide a high speed and higher order programmable FIR filters for shaping, equalizing, adjusting and controlling signal frequencies in real time especially a big demand in video signal processing and transmission due to the emerging applications growth. FIR can able to do channel equalization and ghost cancellation. Hence, an effective FIR filter design is required by an efficient VLSI architecture for emerging applications. By designing a direct form FIR filter, the efficiency can be increased. By Canonic Signed Digit (CSD) representation including Multiple Constant Multiplication (MSM), it can be obtained by reducing the number of adders and multipliers.

For improving the efficiency, FIR filter architectures can be reconfigured and it is suitable for any emerging applications. Reconfiguring FIR architecture is not time and cost effective and it suits only for certain kind applications. Hence, by modifying and extending the existing research works, this research work motivated to design a novel FIR architecture. CSD, MSM, Square Root Carry Select Adder (SQRT CSLA) and Improved Carry Save Adder (ICSA) are used after modification to do that. The modification of SQRT CSLA into reduced complexity Wallace multiplier is the main objective



of this work. The circuit for SQRT CSLA is re-designed to reduce the number of gates in this design and therefore modified SQRT CSLA is incorporated into reduced complexity Wallace multiplier to offer better performance in terms of chip size, delay and power than existing reduced complexity Wallace multiplier with the help of modified carry save adder.

In signal processing, a finite impulse response (FIR) filter is a filter whose impulse response (or reply to any finite length input) is of finite duration, because it resolves to zero in finite time. Finite impulse response (FIR) filters are the most popular type of filters implemented in software. This presentation will help you to understand them both on a theoretical and a practical grade.

Filters are signal conditioners. Each function by having an input signal, blocking prespecified frequency components, and overhauling the original signal minus those components of the yield. For instance, a typical phone line works as a filter that determines the frequencies to a range considerably smaller than the range of frequencies human beings can learn. That's why listening to CD-quality music over the phone is not as pleasing to the ear as listening to it immediately. A digital filter accepts a digital input, passes a digital output, and consists of digital elements. In a typical digital filtering application, software functioning on a digital signal processor (DSP) reads input samples from an A/D converter, performs the mathematical manipulations dictated by theory for the required filter type, and outputs the solution via a D/A converter. An analog filter, by contrast, runs like a shot on the analog inputs and is established entirely with analog components, such as resistors, condensers, and inductors. There are many filter types, but the most common are lowpass, highpass, bandpass, and bandstop. A low pass filter allows only low frequency signals (below some specified cut off) through to its output, then it can be applied to get rid of high frequencies. A low pass



filter is handy, in that respect, for setting the uppermost range of frequencies in an audio signal; it's the type of filter that a phone line resembles. A high pass filter does only the opposite, by rejecting only frequency components below some threshold. An example helps application is burning out the audible 60Hz AC power "hum", which can be plucked up as noise accompanying almost any signal in the U.S.

The architect of a cell telephone or whatever other form of wireless transmitter would typically put an analog bandpass filter on its output RF stage, to secure that only output signals within its narrow, government-authorized range of the frequency spectrum are transmitted. Engineers can use band-stop filters, which pass both low and high frequencies, to block a predefined range of frequencies in the center.

1.3 RESEARCH PROBLEM

It is noticed that FIR filter is the most important component in communication systems and digital signal processing including several portable applications from the above discussion. Also, by focusing on multipliers and adder's configuration in FIR filter architectures, the efficiency can be increased. Using efficient multiplier and adder circuits for an optimized area, power, delay and increase in speed in digital signal processing (DSP), this problem is considered and this research work focused on designing a direct-form Finite Impulse Response (FIR) digital filter.

From the earlier research works, it is found that the performance of the FIR filter mainly depends on the multipliers used from the experimental results whereas by concentrating on the multipliers and adders the performance of the FIR filters can be improved. Hence, this research work is motivated for improving the efficiency which suits for any emerging DSP



applications, this research work is focused on the multipliers and adders involved in the FIR filter architecture.

1.4 RESEARCH OBJECTIVES

The key objective of this research work is to design and implement a novel architecture of a FIR filter in order to save the power, memory, delay, complexity and increase the throughput. In order to fulfill the main objective, the following objectives are carried out one by one in order fulfill the main objective.

- A detailed study on FIR filters merits and demerits comparing with other filters in DSP.
- Learn how far FIR filters are much useful in DSP applications and better understand the concepts of FIR filter involving in DSP.
- Understand the frequency response in a signal in the pass band, transition band and stop bands.
- Implement and verify the logic structure of FIR filter in any simulation or in real time software and check length and coefficient selection of the filter.
- Learn and better understand the concepts and functionalities of MAC, SQRT, CSLA, Wallace multiplier, carry save adder-based FIR filters.
- Design and implement a high-speed MAC-based FIR filter for reducing the complexity.
- Obtain a modified FIR filter by modifying SQRT-CSLA incorporated with Wallace Multiplier.
- Incorporation of Reduced Full Adder and Half Adder into Wallace Multiplier
- Improve the Carry- Save Adder for Digital FIR Filter



1.5 RESEARCH METHODOLOGY

The entire research work is carried out into two different stages in order to fulfill the main and additional objectives of this research work. To obtain a novel design of FIR filter to optimize the area, power, complexity and increase the throughput, both stages are incorporated together. In the beginning stage of the research work, the modification of SQR T CSLA and it is incorporated into a reduced complexity Wallace multiplier is the main objective of this stage. Multiplication and accumulation unit-based FIR filter is used for increasing the speed and throughput in the initial stage of the research work whereas it is used for signal processing applications. It is well known that one of the most important key factors in signal processing is FIR filter. With help of reduced complexity Wallace multiplier, a high speed and area efficient MAC unit is designed to do that. Then, for digital FIR filter, it is modified SQR T CSLA. To reduce the chip size and delay for addition process, conventional BEC based SQR T CSLA is re-designed. To improve the performance of digital multiplication process, this modified SQR T CSLA is incorporated into reduced complexity Wallace multiplier. By re-designing the carry selection block, the best existing BEC (Binary to Excess1 conversion) based SQR T CSLA (Square Root Carry Select Adder) accumulation unit is modified. For addition process, it modified SQR T CSLA is incorporated into reduced complexity Wallace multiplier. Hence, more than other existing MAC units, the results of proposed MAC (reduced complexity Wallace multiplier with help of modified SQR T CSLA) unit provide high speed and less area. Further, to improve the filtering performance, the designed efficient MAC unit is integrated into direct form FIR filter.

Finally, the design of direct form FIR filter with efficient MAC unit has been presented in the second stage of the research work. Initially, by reducing the number of gates, full adder and half adder structures are shrunk



down. These compact full-adder and half adder structures are incorporated into Wallace Multiplier and Improved Carry-Save Adder. By splitting into four parallel phases, the proposed 16-bit Carry-Save Adder has been improved. Consequently, the delay of enhanced Carry-Save Adder is reduced. Using the number of OR gates in a sequential manner, generation of carrying output is performed. To reduce the area, delay, and power utilization, all these enhanced architectures are incorporated into the Digital FIR Filter.

FIR filter circuit must be able to drive at high sample rates, whereas in extra applications, the FIR filter architecture must be a low-power circuit operating at moderate sample rates. For digital filters, the low-power or low-area schemes developed particularly. Decrease the power utilization and area of the original filter in order to further increase the effective throughput. To digital FIR filters, parallel processing can be applied.

1.6 THESIS ORGANIZATION

In order to read and understand the entire thesis work, the thesis is organized in the following manner.

Chapter-1: This chapter presents the basic necessary information about various digital filters, functionalities, utilization, and FIR filter with MAC, SQRT CSLA based architectures. It also described about power consumptions in digital circuits.

Chapter-2: This chapter presents a detailed literature review about digital circuits and filters, power consumptions in digital circuits by using and configuring digital filters. It presents the merits and demerits of the methodologies proposed in the earlier researches.



Chapter-3: This chapter discussed about designing a high-speed multiplication and accumulation (MAC) design for digital FIR filter with experimental analysis. It gives knowledge about multipliers, adders and accumulation process.

Chapter-4: This chapter discussed about the process of incorporation of reduced full adder and half adder into Wallace multiplier and improved carry-save adder for digital FIR filter. It presents how filters can be used for emerging applications due to its ability in terms of reduced complexity and power consumption.

Chapter-5: This chapter discussed about the performance analysis of various filters and the merits while using in digital circuits.

Chapter-6: This chapter discussed about the conclusion and Future scope of the research work.

1.7 NEED OF THE RESEARCH

Various architectures and designing methodologies predict the growth rate of integrated circuits. One estimate states the rate at 2X for every eighteen months. Others claim that the device density increases ten-fold every seven years. Regardless of the exact numbers, everyone agrees that the growth rate is rapid with no signs of slowing down. New generations of processing technology are being developed while present generation devices are at very safe distance from the fundamental physical limits. A need for low power VLSI chips arises from such evolution forces of integrated circuits. The Intel 4004 microprocessor, developed in 1971, had 2300 transistors, dissipated about 1 watt of power and clocked at 1MHz. Coming to the Pentium in 2001,



with 42 million transistors, dissipating around 65 watts of power and clocked at 2.40 GHz as discussed by Chandrakasan *et al.* (1999). 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, a 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. For these systems, low power consumption is a prime concern, because it directly affects the performance by having effects on battery longevity. In this situation, low power VLSI design has assumed great importance as an active and rapidly developing field.

At the circuit design level, considerable potential for power savings exists by means of proper choice of a logic style for implementing combinational circuits. This is because all the important parameters governing power dissipation switching capacitance, transition activity, and short circuit currents are strongly influenced by the chosen logic style.



1.8 POWER USAGE IN CMOS

The power used in CMOS-circuits consists of two parts, dynamic and static power dissipation:

$$P = P_{dynamic} + P_{static} \quad (1.1)$$

The dynamic power consumption is one of the powers used as a function of activity. The static component is power consumed as a function of time.

1.8.1 Static Power Consumption

The static part describes power used even though there is no activity in the circuit. Ideally CMOS components should not have any static power consumption, since there are no direct paths from V_{dd} to ground. In practical applications this is not the case, since MOS transistors are not perfect switches. In MOS transistors, always there will be the leakage in currents and it was discussed by Rabaey & Pedram (1996). Reverse biased currents flow through the source or drain and the substrate, because parasitic diodes in the MOS transistors are one of the static leakage currents. The sub threshold leakages current run through the transistors (from source to drain), because the gate of the transistor is close to the threshold voltage, and therefore some current flow through. These currents used to be negligible, however it seems to become more prominent as transistors become smaller as discussed by Liu & Svensson (1993) and really starts to emerge at 0.13 μm as discussed by Piguet, (2007). The static power dissipation is primarily determined by fabrication technology.

The dynamic part of the power consumption in CMOS can be divided into two parts as discussed by Chandrakasan, (1992).



$$P_{dynamic} = P_{short-circuit} + P_{switching} \quad (1.2)$$

The short circuit happens when both the PMOS and the NMOS transistor is open at the same time. This happens in a switch, because the PMOS and NMOS do not switch instantly, but has a switching delay. This makes a short circuit line from V_{dd} to ground through the CMOS component. As seen in Figure 1.1, if the NMOS and PMOS transistors in the inverter are both open at the same time, a short-circuit path is available from V_{dd} to ground. This phenomenon is described in Equation 1.3, where V_{dd} is the supply voltage and I_{sc} is the current flowing through the short circuit period of the switch. As long as the inputs of the NMOS and PMOS transistors are properly balanced, this power dissipation should be less than 20% of the dynamic power dissipation as discussed by Veendrick, (1984).

$$P_{short-circuit} = V_{dd}I_{sc} \quad (1.3)$$

The power used in switching the CMOS from one state to another is largely used to charge parasitic capacitance in lines between the CMOS-cells. When the output of a gate is turned from 0 to 1, the NMOS part of the CMOS cuts off the connection to ground, and the PMOS part of the CMOS enables a connection from V_{dd} to the output.

This causes the capacitance on the output port and line to be charged, with the energy equal to:

$$Energy\ transition = CV_{dd}^2 \quad (1.4)$$

where V_{dd} is the power source. Half of this power is dissipated at once in the PMOS transistors, while the other half is stored in the capacitance. When the port is turned from '1' to '0', the line is connected to ground, and the energy stored in the capacitance is also dissipated (see Figure 1.1). Since an equal amount of energy is used to charge the circuit for each 0 to 1 transition, it is



possible to get an Equation for power used in switching. Considering the frequency f of the circuit and the probability for the 0 to 1 to the gate α , and the Equation is discussed by Rabaey, (1996).

$$P_{switching} = \alpha f C V_{dd}^2 \quad (1.5)$$

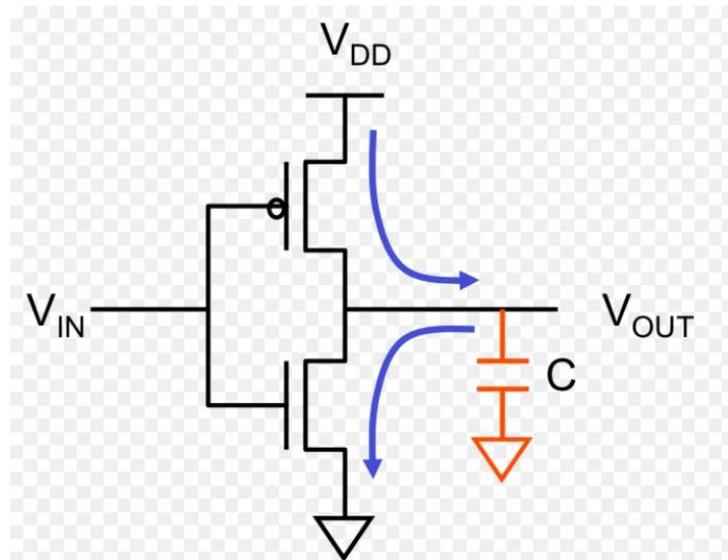


Figure 1.1 Switching power usage in CMOS

Although the other sources of power dissipation have increased their share, switching power consumption is still the largest source for power usage in CMOS today, and is therefore a prime candidate for optimization as discussed by Veendrick, (1984). As seen in the Equation, there are three elements to improve power usage: Voltage, physical capacitance and activity. Over the years, lower voltage has been employed in CMOS, causing a reduction in switching power usage. Physical capacitance is strongly correlated to the line length between transistors and the kind of technology being used (size of transistors and lines). Activity is maybe the most system-dependent factor in the Equation. By reducing the activity in the design, it is possible to reduce the amount of power used in the design.

1.9 DIGITAL FILTERS

Digital signal processing (DSP) finds innumerable applications in the fields of audio, video, and communications, among others. Such applications are generally based on LTI (linear time invariant) systems, which can be implemented with digital circuitry.

Any LTI system is represented by the following Equation:

$$\sum_{k=0}^N a_k y[n - k] = \sum_{k=0}^M b_k x[n - k] \quad (1.6)$$

where a_k and b_k are the filter coefficients, and $x[n-k]$, $y[n-k]$ are the current (for $k=0$) and earlier (for $k>0$) input and output values, respectively. To implement this expression, registers are necessary to store $x[n-k]$ and/or $y[n-k]$ (for $k>0$), besides multipliers and adders, which are well-known building blocks in the digital domain.

The impulse response of a digital filter can be divided into two categories: IIR (infinite impulse response) and FIR (finite impulse response). The former corresponds to the general case described by the Equation above, while the latter occurs when $N=0$. Only FIR filters can exhibit linear phase, so they are indispensable when linear phase is required, as in many telecom applications. With $N=0$, the Equation above becomes

$$y(n) = \sum_{k=0}^{N-1} C_k x(n - 1) \quad (1.7)$$

Where, C_k represents the filter coefficients, $x(n)$ denotes the input of the filter, $y(n)$ represents the filter output, and N is the total length of the filter. This Equation can be implemented by the system of Figure 1.2, where D (delay) represents a register (flip-flops), a triangle is a multiplier, and a circle means adder.



An equivalent RTL representation is shown in Figure 1.3. As shown, the values of x are stored in a shift register, whose outputs are connected to multipliers and then to adders. The coefficients must also be stored on chip. However, if the coefficients are always the same (that is, if it is a dedicated filter), their values can be implemented by means of logic gates rather than registers (just need to store CONSTANTS). On the other hand, if it is a general-purpose filter, then registers are required for the coefficients. In the architecture of Figure 1.3, the output vector (y) was also stored, in order to provide a clean, synchronous output.

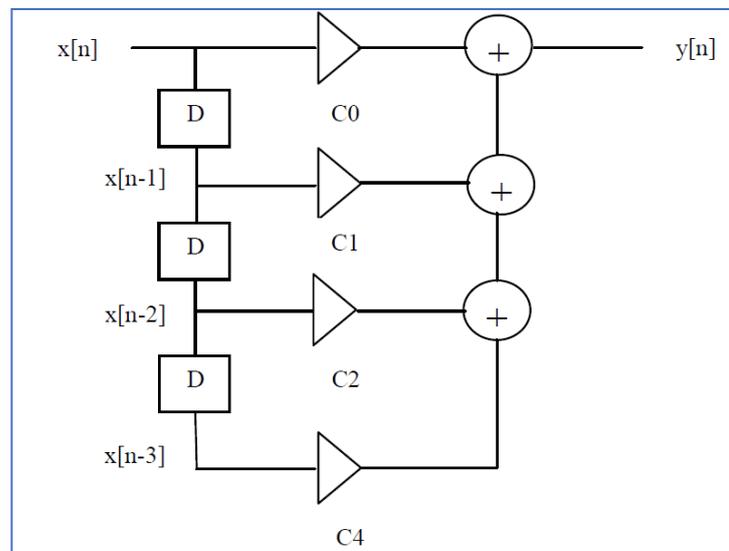


Figure 1.2 FIR filter diagram

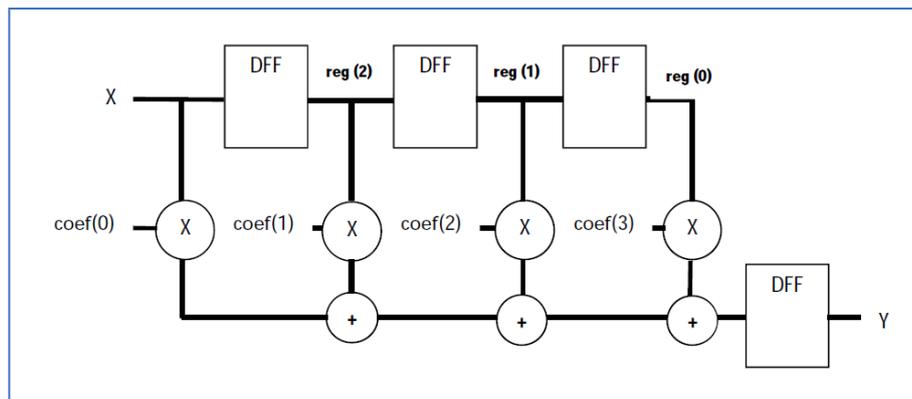


Figure 1.3 RTL representation of FIR filter

The circuit of Figure 1.3 can be constructed in several ways. However, if it is intended for future reuse or sharing, then it should be as generic as possible.

1.9.1 General Purpose FIR Filter

The design presented above contained fixed coefficients, and is therefore adequate for an ASIC with a dedicated filter. For a general-purpose implementation (that is, with programmable coefficients), the architecture of Figure 1.4 can be used instead. This structure is modular and allows several chips to be cascaded, which might be helpful in some applications, because FIR filters tend to have many taps (coefficients).

In this structure, there are two shift registers, one is for storing the inputs (x) and the other is for the coefficients. The structure is divided into n equal modules, called TAP_1, \dots, TAP_n . Each module (TAP) contains a slice of the shift registers, plus a multiplier and adders.

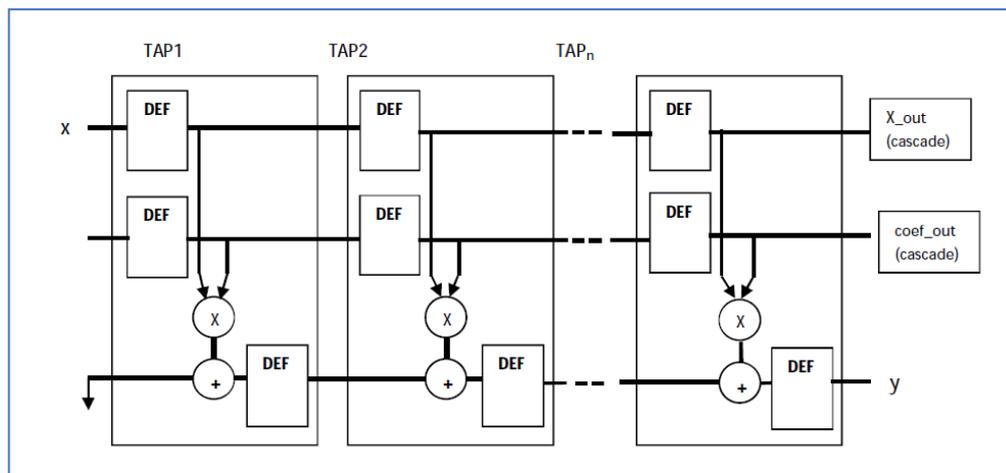


Figure 1.4 General purpose FIR filter

1.9.2 MAC UNIT

In the core of every microprocessor, DSP and data-processing ASIC is its data path. Statistics showed that more than 70% of the instructions perform additions and multiplications in the data path of RISC machines as discussed by Hsun et al (2000). At the heart of data-path and addressing units in turn are arithmetic units, such as comparators, adders, and multipliers. Digital multipliers are the most commonly used components in any digital circuit design. Multiplication based operations such as Multiply and Accumulate and inner product are among some of the frequently used Computation Intensive Arithmetic Functions, that is currently implemented in many DSP applications such as convolution, fast Fourier transform, filtering and in microprocessors in its arithmetic and logic unit. Since multiplication dominates the execution time of most DSP algorithms, there is a need of high-speed multiplier. Currently, multiplication time is still the dominant factor in determining the instruction cycle time of a DSP chip. The demand for high speed processing has been increasing as a result of expanding computer and signal processing applications. Higher throughput arithmetic operations are important to achieve the desired performance in many real-time signal and image processing applications. One of the key arithmetic operations in such applications is multiplication and the development of fast multiplier circuit has been a subject of interest over decades. Reducing the time delay and power consumption are very essential requirements for many applications.

The MAC unit determines the speed of the overall system; it always lies in the critical path. In order to improve the speed of the MAC unit, there are two major bottlenecks that need to be considered. The first one is the partial products reduction network that is used in the multiplication block and the second one is the accumulator. Both of these stages require addition of large operands that involve long paths for carry propagation. To speed up the



multiplication process it implements both the multiplication and the accumulation operations within the same functional block by merging the accumulator with the multiplication circuit using tree architectures for the partial products reduction network as discussed by Chan et al (1991). Many researchers have attempted in designing MAC architecture with high speed computational performance and low power consumption. Elguibaly (2000) proposed a fast-pipelined implementation to lower the MAC architecture's critical delay. Murakami *et al.* adopted the half array implementation to design a high-speed and area-effective MAC architecture. Raghunath (1997) made use of a carry-save multiplier that can simplify sign extension and saturation, and further applies it on MAC architecture to reduce the unit's area and power consumption. Hsun et al (2000) proposed a low-power Multiplication Accumulation Computation (MAC) unit using the radix-4 Booth algorithm, by reducing its architectural complexity and minimizing the switching activities. Kwon *et al.* developed a merged MAC unit based on fast 5:2 compressors instead of 3:2 and 4:2 compressors. Fayed et al (2002) proposed new data merging architecture for high speed multiply accumulate units. The architecture can be applied on binary trees constructed using 4:2 compressor circuits. Increasing the speed of operation is achieved by taking advantage of the available free input lines of the compressor circuits, which result from the natural parallelogram shape of the generated partial products and using the bits of the accumulated value to fill in these gaps.

The general construction of the MAC operation can be presented by this Equation

$$Z = A \times B + Z \quad (1.8)$$

where, the multiplier A and multiplicand B are assumed to have n bits each and the addend Z has (2n+1) bits. The basic MAC Unit is made up of a multiplier and an accumulator as shown in Figure 1.5.



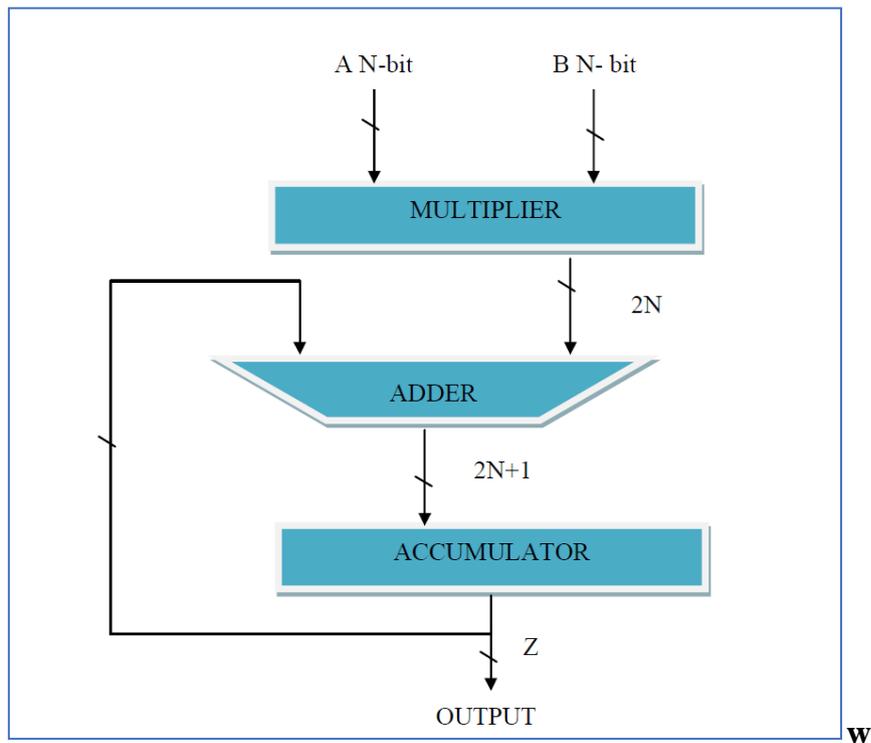


Figure 1.5 MAC unit

The multiplier can also be divided into the partial products generator, summation tree, and final adder. This construction leads to four basic blocks to implement. The summation network represents the core of the MAC unit. This block occupies most of the area and consumes most of the circuit power and delay. Several algorithms and architectures are developed in attempt to optimize the implementation of this block. It executes the multiplication operation by multiplying the input multiplier and the multiplicand. This is added to the previous multiplication result as the accumulation step. The basic MAC operation comprises of a multiplication which can be divided into three operational steps. The first is radix-2 Booth encoding in which a partial product is generated from the multiplicand and the multiplier. The second is adder array or partial product compression to add all partial products and convert them into the form of sum and carry. The last is the final addition in which the final multiplication result is produced by adding

the sum and the carry. A MAC consists of four steps, as shown in Figure 1.6, which shows the operational steps explicitly.

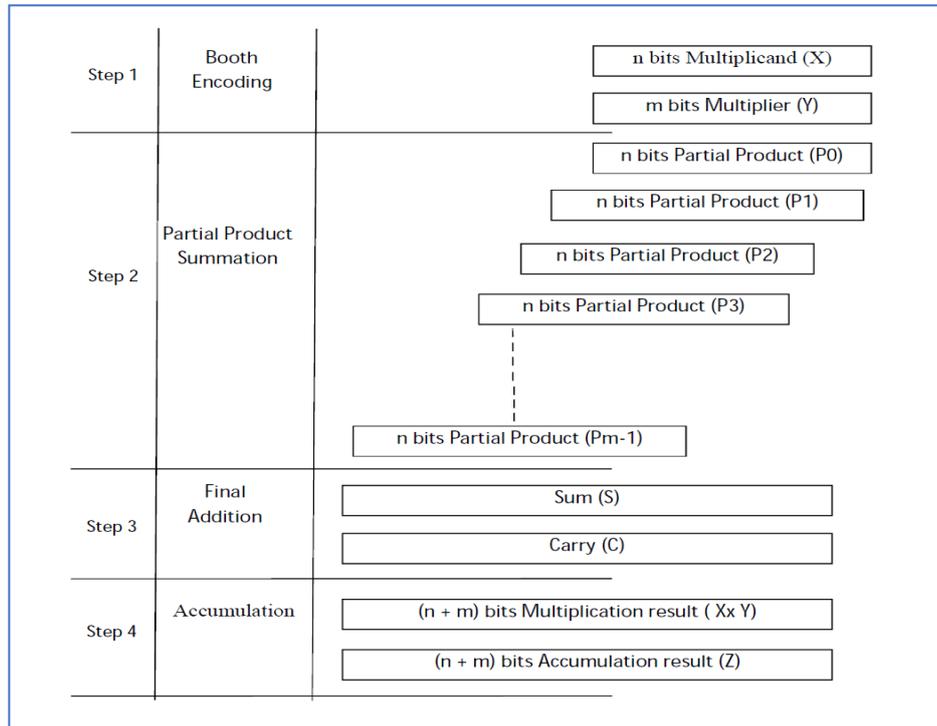


Figure 1.6 Basic steps for deriving MAC unit

In order to add them serially, the execution time is also proportional to N . The architecture of a multiplier, which is the fastest, uses radix-2 Booth encoding that generates partial products and a Wallace tree based on CSA as the adder array to add the partial products. If radix-2 Booth encoding is used, the number of partial products, i.e., the inputs to the Wallace tree, is reduced to half, resulting in the decrease in CSA tree step. In addition, the signed multiplications based on 2 's complement numbers are also possible. Due to these reasons, most current used multipliers adopt the Booth encoding. Figure 1.7 shows the MAC unit architecture. The inputs for the MAC are fetched from memory location and fed to multiplier block of the MAC, which will perform multiplication and gives the result to adder which will accumulate the result and then will store the result into a memory location. This entire process is to be achieved in a single clock cycle. The design of

MAC unit architecture in Figure 1.7 shows that the design consists of one 17-bit register, one 8-bit Wallace tree multiplier, 17-bit accumulator using carry look ahead adder (CLA) and two 18-bit accumulator register are used.

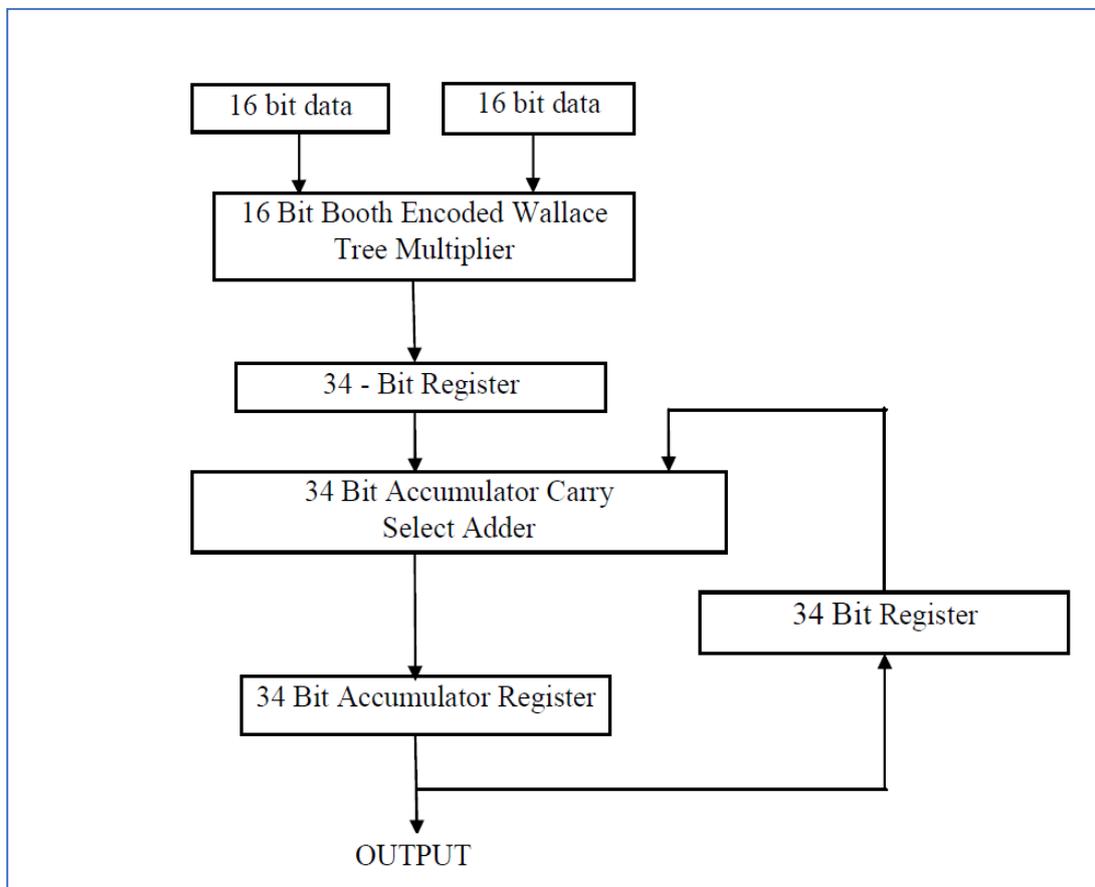


Figure 1.7 MAC unit architecture

In this project, Wallace tree multiplier and carry look adder are used for high performance MAC unit design. Wallace tree multiplier is used to multiply the values of A and B. CLA are used in accumulator and carry save adder (CSA) used in the final stage of the given multiplier for reducing power consumption of the MAC unit. The product of $A_i \times B_i$ is always fed back into the 17-bit CLA in accumulator and added again with the next product $A_i \times B_i$. This MAC unit is capable of multiplying and adding with the previous product consecutively up to as many as.

A multiplier design consists of three operational steps. The first is radix-2 Booth encoding in which a partial product is generated from the multiplicand X and the multiplier Y . The second is adder array or partial product compression to add all partial products and convert them into the form of sum and carry. The last is the final addition in which the final multiplication result is produced by adding the sum and the carry.

1.9.3 Frequency Response

Simple filters are normally specified by their reactions to the individual frequency components that establish the input signal. There are three dissimilar types of reactions. A filter's response to different frequencies is characterized as passband, transition band, or stopband. The passband response is the filter's effect on frequency components that are fallen through (mostly) unchanged. Frequencies within a filter's stopband are, by contrast, highly attenuated. The transition band represents frequencies in the center, which may receive some care but are not transferred altogether from the output signal.

In Figure 1.8, which shows the frequency response of a lowpass filter, ω_p is the passband ending frequency, ω_s is the stipend beginning frequency, and A_s is the measure of attenuation in the stopband. Frequencies between ω_p and ω_s fall within the transition band and are attenuated to some lesser level. Given these individual filter parameters, one of numerous filter design software packages can bring forth the required signal processing equations and coefficients for implementation on a DSP. Ahead we can lecture about specific implementations, however, some additional conditions need to be inserted. Ripple is usually defined as a peak-to-peak level in decibels. It depicts how little or how much the filter's amplitude varies within a circle.



Smaller amounts of ripple represent the more consistent response and are generally preferable.

Transition bandwidth describes how quickly a filter transition from a passband to a stopband, or vice versa. The more rapid this transition, the higher the transition bandwidth; and the more difficult the filter is to accomplish. Though an almost instantaneous conversion to full attenuation is typically desired, real-world filters don't often have such ideal frequency response curves. There is, nevertheless, a tradeoff between ripple and transition bandwidth, so that decreasing either will just suffice to increase the other. Finite impulse response (FIR) filters are the most popular type of filters implemented in software. This presentation will help you understand them both on a theoretical and a practical grade. Filters are signal conditioners. Each function by having an input signal, blocking prespecified frequency components, and overhauling the original signal minus those components of the yield. For instance, a typical phone line works as a filter that determines the frequencies to a range considerably smaller than the range of frequencies human beings can learn. That's why listening to CD-quality music over the phone is not as pleasing to the ear as listening to it immediately.

A digital filter accepts a digital input, passes a digital output, and consists of digital elements. In a typical digital filtering application, software functioning on a digital signal processor (DSP) reads input samples from an A/D converter, performs the mathematical manipulations dictated by theory for the required filter type, and outputs the solution via a D/A converter. An analog filter, by contrast, runs like a shot on the analog inputs and is established entirely with analog components, such as resistors, condensers, and inductors. There are many filter types, but the most common are lowpass, high-pass, bandpass, and band-stop. A low pass filter allows only low frequency signals (below some specified cutoff) through to its output, then it



can be applied to get rid of high frequencies. A low pass filter is handy, in that respect, for setting the uppermost range of frequencies in an audio signal; it's the type of filter that a phone line resembles.

A high pass filter does only the opposite, by rejecting only frequency components below some threshold. An example helps application is burning out the audible 60Hz AC power "hum", which can be plucked up as noise accompanying almost any signal in the U.S. The architect of a cell telephone or whatever other form of wireless transmitter would typically put an analog bandpass filter on its output RF stage, to secure that only output signals within its narrow, government-authorized range of the frequency spectrum are transmitted. Engineers can use band-stop filters, which pass both low and high frequencies, to block a predefined range of frequencies in the center.

Given these individual filter parameters, one of numerous filter design software packages can bring forth the required signal processing equations and coefficients for implementation on a DSP. Ahead we can lecture about specific implementations, however, some additional conditions need to be ushered in [1]. Ripple is usually defined as a peak-to-peak level in decibels. It depicts how little or how much the filter's amplitude varies within a circle. Smaller amounts of ripple represent the more consistent response and are generally preferable.



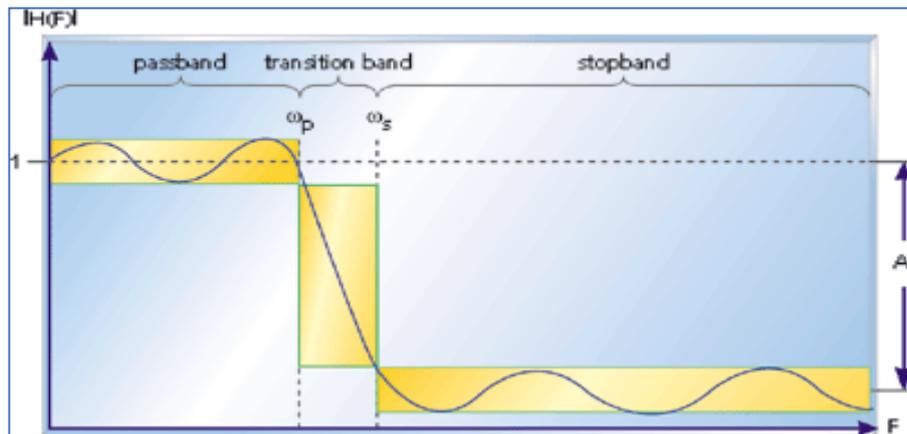


Figure 1.8 The reaction of a lowpass filter to various input frequencies

Transition bandwidth describes how quickly a filter transition from a passband to a stopband, or vice versa. The more rapid this transition, the higher the transition bandwidth; and the more difficult the filter is to accomplish. Though an almost instantaneous conversion to full attenuation is typically desired, real-world filters don't often have such ideal frequency response curves. There is, nevertheless, a trade off between ripple and transition bandwidth, so that decreasing either will just suffice to increase the other.

1.10 FINITE IMPULSE RESPONSE

Digitally, a finite impulse response (FIR) filter is a filter structure that can be used to implement almost any sort of frequency response. To create the filter's output, a FIR filter is usually implemented by using a series of delays, multipliers, and adders. Figure-1.9 shows the basic block diagram for a FIR filter of length N . The delays result in operating on prior input samples. By the appropriate coefficients, the h_k values are the coefficients used for multiplication, so that the output at time n is the summation of all the delayed samples multiplied.

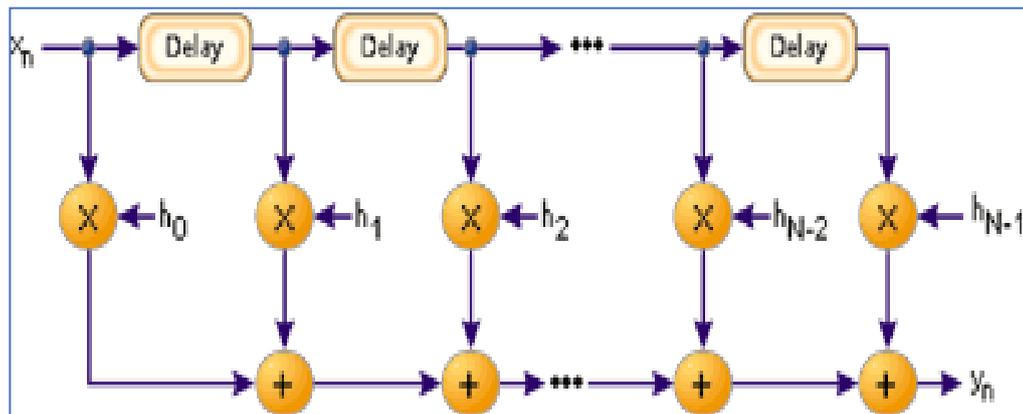


Figure 1.9 The logical structure of FIR filter

The process of selecting the filter's length and coefficients is called filter design. The goal is to set those parameters such that certain desired stopband and passband parameters will result from running the filter. To do their filter design, most engineers utilize a program such as MATLAB. Although, whatever tool is used, the results of the design effort should be the same:

- A frequency response plot, like the one shown in Figure 1.8, which verifies that the filter meets the desired specifications, including ripple and transition bandwidth.
- The filter's length and coefficients.

The longer the filter (more taps), the more finely the response can be tuned. With the length, N , and coefficients, $h[N] = \{ \dots \}$, decided upon, the implementation of the FIR filter is fairly straightforward. As it can be seen, a FIR filter simply produces a weighted average of its N most recent input samples. All of the magic is in the coefficients, which dictate the actual output for a given pattern of input samples. Other digital filter structures are possible, including infinite impulse response (IIR), which uses feedback to keep more historical information active in the calculation.

1.10.1 FIR vs IIR filtering

In this application note, we will explain the difference between FIR ("finite impulse response") and IIR ("infinite impulse response") filtering.

1.10.2 Infinite impulse response (IIR) filters

IIR filters are the most efficient type of filter to implement in DSP (digital signal processing). They are usually provided as "biquad" filters. For example, in the parametric EQ block of a mini DSP plugin, each peak/notch or shelving filter is a single biquad. In the crossover blocks, each crossover uses up to 4 biquads. Each band of a graphic EQ is a single biquad, so a full 31-band graphic EQ uses 31 biquads per channel. The amount of processing that is required to compute a biquad is relatively small. This is what enables the low-cost mini DSP products to implement a full active crossover with parametric EQ on all input and output channels. The DSP (digital signal processor) on each board can compute a certain number of biquads, and this is the primary thing that determines how many filters are available in each plugin. The mini DSP biquads can be programmed using the crossover parameters (slope and frequency), the parametric filter parameters (center frequency, gain, and Q), and so on. They can also be programmed with custom filter shapes by directly entering the biquad coefficients - five numbers that are used to compute the biquad output from its input. You can generate these coefficients by using the community-contributed [custom biquad programming spreadsheet](#).

1.10.3 Finite impulse response (FIR) filters

A FIR filter requires more computation time on the DSP and more memory. The DSP chip therefore needs to be more powerful. Mini-DSP



products that support FIR filtering include the Open-DRC and the mini-SHARC kit. FIR filters are specified using a large array of numbers. In the case of the Open-DRC, there are 6144 coefficients (or "taps") per channel. In the case of the mini-SHARC, there are a total of 10240 taps assignable to all input and output channels. Generation of this large array of numbers must be done in a separate program, such as re-phase, Accurate, and others.

FIR filtering has these advantages over IIR filtering:

1. It can implement linear-phase filtering. This means that the filter has no phase shift across the frequency band. Alternately, the phase can be corrected independently of the amplitude.
2. It can be used to correct frequency-response errors in a loudspeaker to a finer degree of precision than using IIRs.

However, FIRs can be limited in resolution at low frequencies, and the success of applying FIR filters depends greatly on the program that is used to generate the filter coefficients. Usage is generally more complicated and time-consuming than IIR filters.

1.10.4 Examples of FIR and IIR

Here we will provide some simple examples to illustrate the difference between FIR and IIR. Improvements of wireless standards and mobile computing applications have largely demanded on low power digital signal processing (DSP) architectures. One of the most important DSP operations for signal processing application is Finite Impulse Response (FIR) filter. FIR filter is a type of digital filter has linear phase and stability characteristics. A large endeavour has been implemented the direct form FIR filter to improve the performance of digital FIR filter. In this paper, design of MAC unit for FIR filter is done with a smaller number of chip size, delay and



power. The input-output relation of linear time invariant (LTI) direct form FIR filter is represented as in Equation (1).

$$Y(n) = \sum_{k=0}^{N-1} C_k x(n - k) \quad (1.9)$$

where, $x(n)$ represents the filter input, $y(n)$ represents the filter output, N is the length of filter or order of the filter and C_k denotes the filter coefficients. The filter order (N) is fixed in case of direct form FIR filter. The heart of direct form FIR filter is MAC unit. To implement MAC unit through VLSI System design environment, efficient structure of adder and multiplier required with VLSI main concerns (Low power consumption, less area and high speed).

1.10.5 Crossover filter

In a two-way crossover filter, the low pass and high pass outputs are sent to the woofer and tweeter respectively and are summed acoustically. We can simulate this behavior electrically - the Figure below shows the measured phase response of a summed fourth-order Linkwitz-Riley crossover (24 dB/octave) at 300 Hz in blue. The phase of this crossover shifts by 360 degrees from low frequencies to high frequencies. Shown in red is the measured output from a crossover with the same amplitude response curves, but implemented with a linear-phase FIR filter. The phase shift is very close to zero across the audio band.



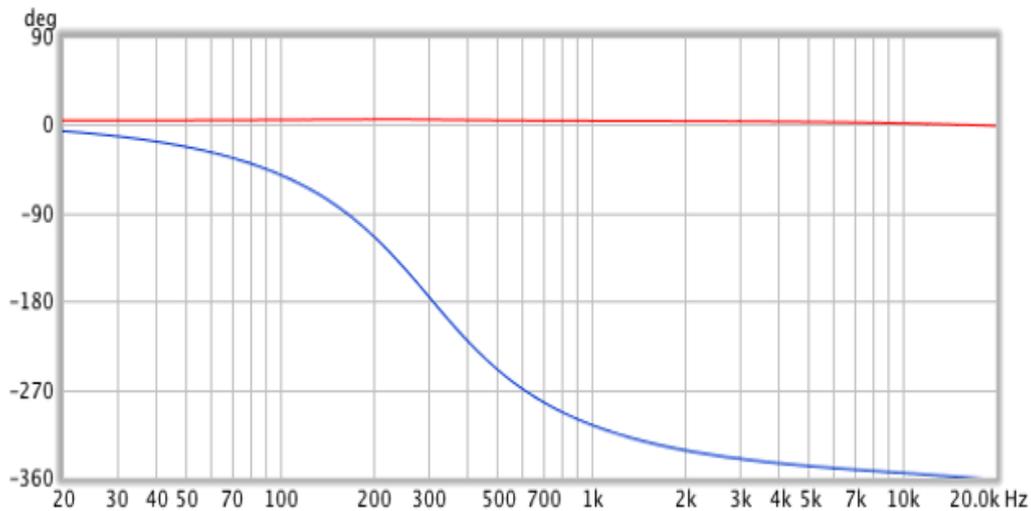


Figure 1.10 Phase Response

1.10.6 Parametric filter

Parametric filters also have a phase shift. Consider a parametric filter with this response:

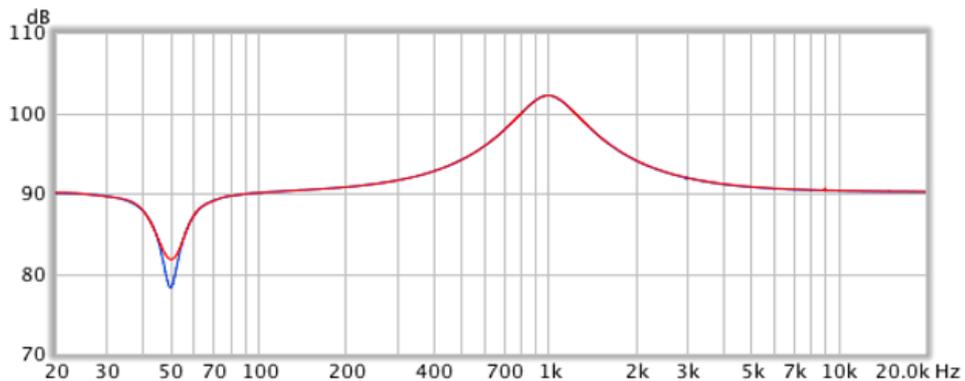


Figure 1.11 . Parametric Filter Response

Below is the measured phase shift of this filter, in blue as implemented by an IIR filter, and in red as implemented by a linear-phase FIR filter. Again, the linear-phase filter has minimal phase shift across the audio band.

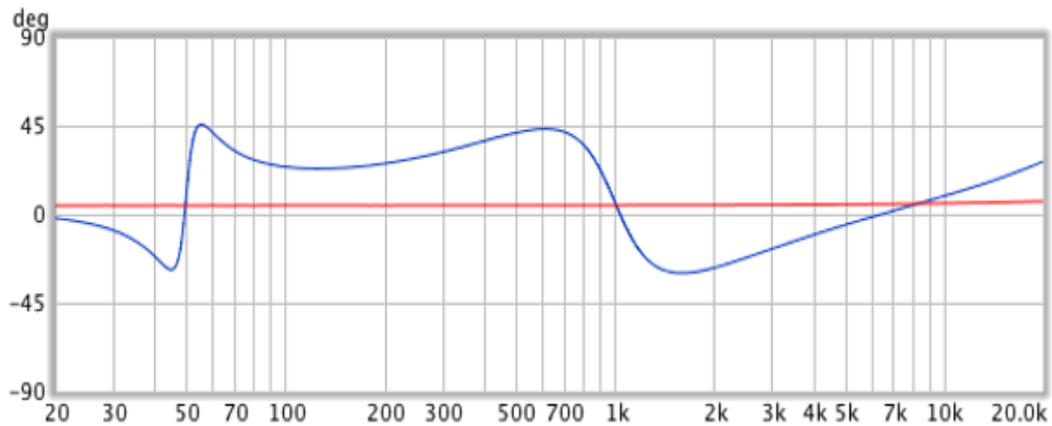


Figure 1.12 Phase Shift Response

Note that sometimes the phase shift shown above is desirable, as it acts to correct phase as well as amplitude errors in (for example) the speaker driver being corrected. FIR filters can implement this curve either with the phase shift (minimum phase) or without it (linear phase).

1.11 SUMMARY

FIR filters are more powerful than IIR filters, but also require more processing power and more work to set up the filters. They are also less easy to change "on the fly" as you can by tweaking (say) the frequency setting of a parametric (IIR) filter. However, their greater power means more flexibility and ability to finely adjust the response of your active loudspeaker. The entire research work discussed a brief study on existing research works focused on FIR filter with MAC, SQRT CSLA based architectures in order to obtain the research problem.

CHAPTER 2

LITERATURE SURVEY

This chapter presents various research works focused on FIR filters and its applications. Common functionality of FIR filter, FIR filter-based applications are also presented here. It also presents the research works focused on using FIR filter for power consumption applications. It presents FIR filter based high speed applications. From this chapter it is able to understand various methods used for configuring FIR filter and improve the efficiencies.

2.1 A REVIEW ON FIR FILTER

Filter is the essential block of most of the Digital Signal Processing systems. Hence there is need to design an effective filter in terms of area, power and time lag. Filters play important part in a wide assortment of applications originating from video processing for wireless communications. Granting to the demand of application, we need to modify filter structures. Some of the application needs to function at low power while some require to control at high velocity. Hence, in order to design the fast-operating application, then each and every element in the circumference must be quicker. As the filter is an essential constituent of most of the DSP systems, hence if we designed the filter which is hardware efficient it automatically improves the functioning of the application. Hence, lots of attempts are needed



to improve the filter operation. In that respects are basic two techniques to improve the filter performance are pipelining and parallel processing.

Pipelining reduces the critical path by incorporating the pipe, lined latches along the data path which does an increment in the latches and latency of the scheme, whereas parallel processing increases the throughput of the system by replicating hardware so that a number of inputs are simultaneously processed and output is generated at the same time stated in Parhi *et al.* (1999).

In the past decades, many attempts are asked to optimize the filter structures are discussed in Parhi *et al.* (1999), Parker and Parhi (1996), Acha (1989), Cheng and Parhi (2004), Yu-Chi Tsao and Ken Choi (2012), Tian *et al.* (2013) and Cheng and Parhi (2005). In Parker and Parhi (1996), the combination of fast FIR algorithms and coefficient quantization technique called Maximum Absolute Difference technique is used for filter optimization which is providing hardware saving of 45% as compared with the traditional FIR filter implementations. By cutting down the hardware complexity as compared with FFA a new technique is given for speeding up the signal throughput said in Acha (1989). It applies the fast FIR algorithms based on short convolutions.

In Cheng and Parhi (2004), to obtain more hardware, saving a raw advance is applied to call iterative shorter convolution based on the Fast convolution Algorithms and mixed radix algorithms. In this convolution technique the long convolution is decomposed into short convolutions and such short convolution blocks are iteratively applied to obtain longer convolutions. This proficiency is more beneficial for improvement of the filter performance in terms of time lag involved. In this manner most of the techniques are used for optimizing the filter structures. In Yu-Chi Tsao and Ken Choi (2012), and Tian *et al.* (2013), the more concentration is on



optimization of the sub filter block of the filter. They are getting the use of symmetric coefficients of the filter so that if more symmetric sub filter blocks are formed, then the number of multipliers required in sub filter blocks are scaled down to half as compared with the unsymmetrical block. It is invariably preferable to cut the number of multipliers as compared with the adder, since adders weigh less than multipliers. Hence, lots of attempts are studied in order to replace multiplier with added. Tian, *et al.* (2013) proposed a novel technique for filter design using modified Winograd algorithm which provides the significant amount of the hardware saving.

2.1.1 Multipliers

Low power use and smaller area are some of the most important touchstones of the Digital Signal Processing (DSP) systems. Hence it is invariably preferable to use optimized multiplier. One such multiplier is Wallace multiplier which requires less area as compared with the binary array multiplier. The author in Tian, *et al.* (2013), Ali *et al.* (2014), and Balasubramaniam and Bharathi (2012), shows that Linear phase FIR filter implementation using Ripple Carry Adder (RCA) and Sandhya *et al.* (2014), Gnanasekaran and Manikandan (2014) using Wallace Multiplier is efficient. This report presents the implementation of the filters using RCA and Wallace Multiplier.

Generally, FIR filters with large number of taps are necessary to obtain high spectral containment and noise reductions. The computational delay and required chip size of direct form FIR filter has raised due to inefficient adder and multiplier structures. In previous maneuver, Dempster *et al.* (1995), Canonic Signed Digit (CSD) representation is used to reduce the number of adder and multiplier. To perform constant multiplication of direct form FIR filter, Multiple Constant Multiplication (MCM) is used in,



Dash *et al.* (2014). This approach cannot be used when filter coefficients change dynamically. Hence, the effective Wallace multiplier is proposed in, M. B Kumar and S.K Patel, (2014), C.Satish *et al.* (2014), Bharti *et al.* (2013), Gowrishankar *et al.* (2013), Malini *et al.* (2014), Rao *et al.* (2012), Gahlan *et al.* (2012), Senthilkumar *et al.* (2013), Priyatharshne *et al.* (2014), Yu *et al.* (2011), by using compressors. Further to improve the multiplication process Modified Booth Algorithm, Rao *et al.* (2012), is used for design of Wallace multiplier. Further to improve the performance of Wallace multiplier, effective adder structures are used for adding the partial product results, Dash *et al.* (2014). Square Root Carry Select Adder (SQRT CSLA) is one of the best adders which provide less area and power for addition process. Generally, it consists of Half Sum Generation (HSG), Carry Generation (CG) and Full Sum Generation (FSG) as in M. B Kumar and S.K Patel, (2014). In addition, Modified Carry Save Adder is also producing better result for addition process, Senthilkumar *et al.* (2013) and Gowrishankar *et al.* (2013).

2.2 A REVIEW ON FIR BASED APPLICATIONS

Rashidi and Rashidi (2011), presents the methods to reduce dynamic power consumption of a digital Finite Impulse Response (FIR) filter these methods include low power serial multiplier and serial adder, combinational booth multiplier, shift/add multipliers, folding transformation in linear phase architecture and applied to fir filters to power consumption reduced thus reduce power consumption due to glitching is also reduced. The minimum power achieved is 110mw in fir filter based on shift/add multiplier in 100MHZ to 8staps and 8bits inputs and 8bits coefficients. The proposed FIR filters were synthesized implemented using Xilinx ISE Virtex IV FPGA and power is analized using Xilinx XPower analyzer.



Soojin Kim and Kyeongsoon Cho (2010), describes the pipeline architecture of high-speed modified Booth multipliers. The proposed multiplier circuits are grounded on the modified Booth algorithm and the pipeline technique which are the most widely used to speed up the multiplication speed. In parliamentary law to implement the optimal pipelined multipliers, many sorts of experiments have been carried. The velocity of the multipliers is greatly improved by properly determining the number of pipeline stages and the offices for the pipeline registers to be entered. We identified the proposed modified Booth multiplier circuits in Verilog HDL and synthesized the gate-level circuits using 0.13um standard cell library. The resulting multiplier circuits show better functioning than others. Since the proposed multipliers operate at GHz ranges, they can be used in the systems requiring very high functioning.

Hemalatha and Shanmugam (2011), SDR is fast becoming a crucial component of wireless technology the use of SDR technology is anticipated to replace many of the traditional methods of implementing transmitters and receivers while offering a broad scope of advantages, including adaptability, reconfigurability, and multifunctionality encompassing modes of procedure, radio frequency bands, air interfaces, and waveforms. Research in this area is primarily aimed towards improving the architecture and the computational efficiency of SDR systems. Software-defined radio (SDR) refers to wireless communication in which the transmitter modulation and the receiver demodulation are both generated through software. The primary advantage of this approach is flexibility, as the software moves on one common hardware platform for whatever case of receiver configuration. The most computationally intensive portion of the wideband receiver of a software defined radio (SDR) is the average frequency (IF) processing block. Digital filtering is the main task in IF processing. The computational complexity of finite impulse response (FIR) filters used in the IF processing block is



dominated by the number of adders (subtractions). The proposed reconfigurable synthesizes multiplier blocks offer significant savings in area over the traditional multiplier blocks for high-speed digital signal processor (DSP) systems are implemented on field programmable gate array (FPGA) hardware platforms.

In addition, software radio has recently gained much attention due to the need for integrated and reconfigurable communication systems. To this end, reconfigurability has become an important event for the future filter design. Previous research in this area has focused on minimizing multiplier block adder cost but the outcomes shown here show that this optimization goal does not minimize FPGA hardware. Minimizing multiplier block logic depth and pipeline registers is shown to possess the greatest influence in reducing FPGA area cost. Fully pipelined, full-parallel transposed-form FIR filters to reconfigurable multiplier block were generated using the novel and old algorithms, implemented on an FPGA target and the results compared. The proposed method offers average reductions of adders and full adders needed for the coefficient multipliers over conventional FIR filter implementation methods.

Dandapat (2007), presents higher order compressors which can be effectively utilized for high speed multiplications. The proposed compressors offer less delay and area. Only the Energy Delay Product (EDP) is somewhat higher than lower order compressors. The performance of 8×8 , 16×16 and 24×24 multipliers using the proposed higher order compressors has been compared with the same multipliers using lower order compressors and found that the new structures can be applied for high speed multiplications. These compressors are simulated with Cadence RTL compiler at a temperature of 25°C with the supply voltage of 1.2 V.



Shahnam Mirzaei (2006), represents a method for carrying out high speed Finite Impulse Response (FIR) filters using just registered adders and hardwired shifts. We extensively use a modified common sub expression elimination algorithm to cut down the number of adders. We target our optimizations to Xilinx Virtex II devices where we compare our implementations with those made by Xilinx Coregen™ using Distributed Arithmetic. We keep up to 50% decrease in the number of slices and up to 75% decrease in the number of LUTs for fully parallel implementations. We also observed up to 50% decrease in the total active power consumption of the filters. Our designs perform significantly faster than the MAC filters, which use embedded multipliers.

A new algorithm that synthesizes multiplier blocks with low hardware requirement suitable for implementation as part of a full-parallel finite impulse response (FIR) filters is presented in K.N. Macpherson and R.W. Stewart (2006). Although the techniques in use are applicable to implementation on application-specific integrated circuit (ASIC) and Structured ASIC technologies, analysis is performed using field programmable gate array (FPGA) hardware. Fully pipelined, full-parallel transposed-form FIR filters with multiplier block were generated using the novel and old algorithms, implemented on an FPGA target and the results compared. Previous research in this area has focused on minimizing multiplier block adder cost but the outcomes shown here show that this optimization goal does not minimize FPGA hardware. Minimizing multiplier block logic depth and pipeline registers is shown to possess the greatest influence in reducing FPGA area cost. In addition to providing lower area solutions than existing algorithms, comparisons with equivalent filters generated using the distributed arithmetic technique demonstrate further area advantages of the new algorithm.



Kousuke TARUMI *et al.* (2004), evaluated an approach for a low power digital baseband processing. In this evaluation, our target specification of the digital FIR filter is a cosine roll off filter. The tap length of the digital FIR filter is fixed. We assess the power expenditure and the circuit area of those digital FIR filter circuits. Three kinds of digital FIR filters are designed and evaluated. First one, we call it 16-bits coefficient in Table2 and Table3, is designed as an exemplar of a high accuracy digital FIR filter that all data path, bit width of being the same as 16-bits. Second one, we call it 8-bits coefficient in Table2 and Table3, is designed as an exemplar of a low accurate digital FIR filter that all data path, bit width of being the same as 8-bits.

A high-speed FIR filter architecture is implemented using, possibly pipelined, carry-save adder trees for accumulating the partial products by (Anton Blad and Oscar Gustafsson 2010). A method to detect redundant adders in the reduction tree was nominated and evaluated using multi-rate FIR filter structures for CIC decimation and interpolation transfer functions. The redundancy reduction is performed at the bit-level to further work for short word length data like as those obtained from sigma-delta modulators.

A modification to the Wallace reduction is passed that ensures that the delay is the same as for the conventional Wallace reduction, Ron S. Waters, Earl E. And Swartzlander (2010). The modified reduction method greatly reduces the number of half adders; producing implementations with 80 percent fewer half adders than standard Wallace multipliers, with a very slight gain in the number of full adders. Both the traditional Wallace and modified Wallace reductions have the same number of stages and consequently the delay is expected to be the same. It is important that both the conventional and modified Wallace second phase reductions use extra gates than the data reduction, although the penalty is less for the modified Wallace reduction. A VLSI architecture for low power MAC has been presented by, Ashish B. *et al.*



(2013). The aim of this study is to design and implementation of the Finite impulse response (FIR) filter using a low power MAC unit with clock gating and pipelining techniques to preserve power. The Power is estimated for the cubes. A 1-bit MAC unit is designed which enable to reduce the entire power consumption based on above proposed techniques. Using this block, the N-bit MAC unit is constructed and the total power consumption is computed for the MAC unit. The MAC unit designed in this study can be used in filter realizations for High speed DSP applications.

The systematic approach is proposed for Finite Impulse Response (FIR) using the rounded truncated multiplier which offers diminution in the area, delay, and power discussed in R.Ambika and S.SivaRanjani (2014). This anticipated method finally reduces the number of full adders and half adders during the tree reduction in the multiplier block. LSB and MSB are the output form of this multiplier. Deletion, reduction, truncation, rounding and final addition are the operations performed to compress the LSB part. When this scheme is followed the truncation, error does not exceed 1 up (unit of least position). So, it does not necessitate any error compensation circuits, and the final output will be precise. The proposed filter using truncated multiplier will be designed using VHDL and simulated using ISE Simulator (ISIM). It achieves the best area and power result when compared with previous FIR design approaches. This filter design can be extended by using the Montgomery multiplier

N. Kannan *et al.* (2014), presented a hardware design and carrying out of FPGA based parallel architecture for Truncated Multiplier and Wallace Tree Multipliers utilizing Verilog. The Wallace Tree Multiplier shows much more reduction in device usage as compared to truncated multiply. Furthermore, the truncated and Wallace Tree Multiplier shows that the number of occupied slices, four inputs LUTs, total equivalent gate count,



average connection Delay and maximum pin delay have been significantly shortened.

The alteration to the Wallace multiplier is performed stated in, Anju. S and M. Saravanan (2013). The comparison result shows that the modified one reduces the Figure of half adders by 80%. Only Wallace and Modified Wallace reduction use more gates for their reduction than data multiplier. CSA with BEC and CSA with D Latch is introduced in the final carry propagation path of the multipliers. From all the comparison results we can reason that the data multiplier with CSA with D Latch in the final carry propagation path is more effective. A Carry Select Adder using BEC is introduced, but it provides some speed penalty. This report offers an efficient carry select adder using D Latch.

Rakhi Thakur and Kavita Khare (2013), progressed to an overture to the carrying out of the digital filter based on field programmable gate arrays (FPGAs) which are elastic and provides performance comparable or superior to traditional approaches, low power, area-efficient re-configurable digital signal processing architecture that is sewn for the recognition of arbitrary response Finite impulse response (FIR) filters.

An architectural approach to the design of low-power reconfigurable finite impulse response (FIR) filter formulated by Seok-Jae Lee (2011). This overture is well suited when the filter order is made and not modified for particular applications, and the efficient trade-off between power savings and filter performance can be reached utilizing the proposed architecture. Mathematical analysis on power savings and filter performance degradation and its experimental results indicate that the proposed approach achieves significant power savings without seriously compromising the filter operation. The power savings are up to 41.9% with minor performance



degradation and the area overhead of the suggested system is less than 5.3% compared to the formal approach.

Sweety Kashyap and Mukesh Maheshwari (2014), goes through a high-performance FIR filter using low power adder and multiplier. The different adder and multiplier on the base of their dynamic power dissipation are analyzed. Granting to the analysis result of adders and multipliers, carry save adder and the radix4 multiplier is consuming low power among all adder and multiplier respectively. By using CSA and Radix 4 multiplier, we went through the FIR filter and broke down its power use. The performance curve of power dissipation by adders and multipliers was derived from analysis of different adders and multipliers. On the base of power consumption result of proposed and existing FIR filter. The ending comes out that proposed FIR filter consumes 10.36% power lesser than existing FIR Filter. Thus, according to result proposed FIR filter is the best for DSP system.

A simple attack is projected in this composition to melt off the expanse and power of SQR T CSLA architecture is discussed by Andamuthu and Rithanyaa (2012). The reduced number of gates of this work extends the big advantage in the reduction of area and also the full power. The comparative results show that the modified SQR T CSLA has a slightly larger delay (only 3.76%), but the area and power of the 128-bit modified SQR T CSLA are significantly scaled down by 17.4% and 15.4% respectively. The power-delay product and also the area-delay product of the proposed design show a decrease for 16, 32, 64 and 128-bit sizes which indicate the success of the method and not a mere trade off of delay for power and expanse. The modified CSLA architecture is, consequently, low area, low force, simplicity and efficient for VLSI hardware implementation.

The 0–1 ILP formalization for designing digit-serial MCM operation with the optimal area at the gate level by seeing the execution costs



of digit-serial addition, subtraction, and transformation operations are introduced by Levent Aksoy *et al.* (2013). Since there are still instances with which the exact CSE algorithm cannot cope, they also proposed an approximate GB algorithm that determines the best partial products in each iteration, which yield the optimal gate-level area in digit-serial MCM design. This research also brought out the design architectures of the digit-serial MCM operation and a CAD tool for the realization of digit-serial MCM operations and FIR filters. The experimental results suggest that the complexity of digit-serial MCM designs can be further scaled down using the high-level optimization algorithms proposed in this report. It is recorded that the realization of digit-serial FIR filters under the shift-adds architecture yields a significant area reduction when compared to the filter designs whose multiplier blocks are implemented using digit-serial constant multipliers. It is mentioned that an architect can find the tour that fits best in an application by changing the Figure size.

Dakupati. Ravi Sankar *et al.* (2013), proposed the implementation and analysis of a novel Wallace tree architecture. The response time of the existing Wallace tree multiplier which is found to be 27 has been trimmed to 15. The comparison result also indicates that a substantial reduction of power is attained. At an operating frequency of 50 MHz at 1.2V, the power is found to be 153.47mW. It is a realization of 11.6% of power reduction than the conventional Wallace tree multiplier. At 1.14V, the power consumed is found to be 147.66mW, which is a 12.03% reduction of that obtained from the existing architecture. The results show that the suggested architecture is more efficient than the conventional one in terms of Power consumption and response time.

The Performance analysis of finite impulse response (FIR) designs is presented by the concept of modified Wallace multipliers is



discussed by M. Gnanasekaran and Dr. M. Manikandan (2014). This report aims at scaling down the leakage current delays and power use of Wallace multiplier. This is accomplished by MCSA. An efficient Verilog HDL has been written, successfully simulated and synthesized in Xilinx and the answers prove that proposed design achieves the best delay and power than an existing scheme which utilizes the concept of truncated multiplier.

Gowrishankar *et al.* (2013), proposed a new approach to design a FIR filter to increase the speed of addition and decrease the power consumed by the multiplier unit. It has been easily concluded that the proposed FIR filter design has consumed less power than the formal pattern. The power delay comparison of both Wallace tree multiplier and carry select adders are shown. The comparative results prove that proposed carry select adder with binary to excess-1 converter performs faster than conventional carry select adder. With proposed multiplier unit and carry select adder unit, the designed FIR Filter consumes 55% power less than the conventional filter without a significant increase in area. The power & delay comparison is done for both existing and suggested methods of FIR filters. The pattern is implemented using 0.18 μ m technology.

Shen-Fu Hsiao *et al.* (2013), introduces the low-cost FIR filter designs by jointly considering the optimization of coefficient bit width and hardware resources in implementations. Although most prior designs are founded on the transposed form, it is noted the direct FIR structure with faithfully rounded MCMAT leads to the smallest area cost and power use of goods and services.

B. Ramkumar and Harish M Kittur (2013), utilizes a simple and efficient gate-level modification to significantly cut the expanse and power of the CSLA. Founded on this modification 8-, 16-, 32-, and 64-b square root CSLA (SQRT CSLA) architecture have been built up and compared to the



regular SQR T CSLA architecture. The proposed plan has reduced area and power as compared with the regular SQR T CSLA with only a little gain in the holdup. The modified CSLA architecture is a depressed area, low force, simplicity and efficient for VLSI hardware implementation. Therefore, the proposed CSLA structure is more beneficial than the regular SQR T CSLA.

Suresh Srinivasan *et al.* (2013), exhibits a split path FPMAC design which is 14% quicker than the fastest known silicon implementation. The goodness of the design is punctuated by the timing gains at no additional area costs. The split path design provides a instinctive means for gating opportunities and still under normal case may lead to 15-20% of lesser switching gates based on the near or far path operation.

A high performance and low power FIR filter design, which is based on computation sharing multiplier (CSHM) are given by, Jongsun Park *et al.* (2013). CSHM specifically targets computation re-utilization of vector-scalar products and is effectively practiced in our FIR filter design. Efficient circuit level techniques: a new carry select adder and conditional capture flip-flop (CCFF), are also used to further improve power and execution. The proposed FIR filter architecture was put through in 0.25 μm technology. Experimental results in a 10 tap, low pass CSHM FIR filter show speed and power improvement of 19% and 17%, respectively, with regard to a FIR filter based on a Wallace tree multiplier.

Anindita Dash *et al.* (2014), aims to optimize a Wallace Tree multiplier. The multiplier was implemented at the circuit level of design abstraction with the Virtuoso® tool in Cadence. Then searched the different topologies of compressors and implemented them at circle level. These topologies were compared and then implemented in the multiplier. The multiplier implemented was of 5x5bits which could be increased to higher order for the single precision IEEE floating point multiplier and used in it.



A Wallace tree multiplier using modified booth algorithm is proposed in Jagadeshwar Rao M and Sanjay Dubey (2012). This story points at additional reduction of latency and power consumption of the Wallace tree multiplier. This is accomplished by the use of booth algorithm, 5:2, 4:2, and 3:2 compressor adders. An efficient Verilog HDL code has been written, successfully simulated and synthesized for Xilinx FPGA vertex-6 low power (Xc6vlx75tl-1Lff484) device, using Xilinx 12.2 ISE and XST. The result shows that the proposed architecture is around 67% faster than the existing Wallace-tree multiplier. This approach may be well suited for multiplication of numbers with more than 16-bit size for high velocity applications. The ability of the proposed multiplier can be explored to implement high performance multiplier in VLSI applications.

Capri Satish *et al.* (2014), describes the Wallace tree multiplier which is considered faster than a simple array multiplier and is an effective implementation of a digital circuit which multiplies two integers. A Wallace tree multiplier is a parallel multiplier which uses the carry to save addition algorithm to reduce the response time. The Wallace tree basically multiplies two unsigned integers. The simulation and synthesis of multipliers are done in Xilinx ISM 14.2 and functionally tested in Modelsim with different test cases. The new architecture enhances the speed performance of the widely acknowledged WTM. Wallace tree multiplier using booth algorithm is very a serious technique for high speed applications, its implementation with different logics in VLSI.

The Wallace tree multipliers can be solved & analyzed using a new modified method of Wallace tree construction using compressors is discussed in Naveen Kr. Gahlan (2012). The modified tree has a slightly smaller critical path, a slightly larger wiring overhead but gives high speed. Wallace Tree CSA structures have been used to add the partial products in



shortened time. In this research Wallace tree construction is investigated and measured. In this research Wallace tree is made by traditional methods and with the help of compressor techniques such as 4:2 compressor, 5:2 compressor, 6:2 compressor, 7:2 compressor. Thus, minimizing the number of half adders used in a multiplier reduction will cut the complexity.

Deepshikha Bharti and K. Anusudha (2013), submits a high speed FIR filter design by studying the optimization of coefficient bit width and hardware resources in implementations. Although most prior designs are founded along the direct cast, we remark that the reversed form of FIR filter structure with faithfully truncated multiplier and parallel adder leads to less time lag in the calculation of output of FIR filters. Multiplication and addition are frequently involved in Digital Signal Processing. Parallel prefix adder provides a high-speed addition and the improved version of truncated multiplier also provides a sequential reduction in delay and the ingredients employed.

The implementation of digit serial FIR filter was put through with low complexity MCM architectures for digit sizes $d_o=2, 4, 8$ is discussed in Hema Malini *et al.* (2014). In parliamentary law to cut down the amount of shifting, an addition/subtraction operation, Graph Based (GB) algorithm is considered as a suggested algorithm to construct multiplier for transposed form of digit serial FIR filter design 4-tap digit serial FIR filter with digit sized= $2,4,8$. The MCM approach drastically reduces the system complexity, area, and delay. Ultimately, this FIR filter is implemented in FPGA spartan3 hardware for real-time implementation.

The resource minimization problem with the scheduling of adder-tree operations for the MCM block of transposed direct-form FIR filter and presented an MIP-based algorithm for accurate bit-level resource optimization have been identified by Yu Pan *et al.* (2014). The experimental result shows



that upwards to 15% reduction of area and 11.6% reduction of ability can be achieved on top of an already optimized ADD/SUB networks of MCM blocks. Further exploration of efficient heuristic algorithms for resource minimization of adder-trees of FIR filters could be performed in the future.

2.2.1 Modified FIR Filter

Finite impulse response digital filter is the most important component in communication systems and applications of digital signal processing. When it provides limited power and area, it is extensively used in several portable applications, Parhi and K.K (1998). The two fundamental FIR structures used for a linear phase FIR filter are transposed form and direct form. In this paper, direct form digital FIR filter is used for DSP applications. Multiplier-Accumulator (MAC) unit of FIR filter is the most important element. The efficiency of the MAC unit is affected by full adder. Full adder circuit power reduction is necessary for low power application. The heart of the processor is Arithmetic & Logic Unit (ALU), Srinivasan *et al.* (2013). It contains elements for reckoning operations. It plays a very important role in computation time of the processor. Multiplication operation is more recurrent in Digital Signal Processing (DSP) application. Sinking delay in the multiplier shrinks the overall computation time, Kharate *et al.* (2013). One of the fast multipliers is available such as Wallace multiplier. It works due to speeding up the addition process. Carry Propagating Adder has been used to sum the final two rows. A direct implementation needs a $(2N - 2)$ bit Carry Propagating Adder (CPA), where N is the number of bits of operands. Carry Propagating Adder obtains long time when the carry is required to get promulgated until the last adder, Parhami, B. (2010). In this work, a fast carry-save adder is implemented at the last stage to obtain superior performance.



Modified Carry-Save Adder consumes more delay and area due to propagation delay and sequential process, Ramkumar *et al.* (2010). Hence Improved Carry-Save Adder (ICSA) is designed in this work with parallel processing and without carry propagation delay. Our ICSA adder offers less area and higher speed than all other schemes. Regular Wallace and reduced Wallace Multipliers are designed using different high-speed adders, Gowrishankar *et al.* (2013). But it consumes more area, power and less delay, Kannan *et al.* (2014), Gnanasekaran and Manikandan (2014), Waters *et al.* (2010). So compact full adder, half adder and ICSA adder are incorporated into Wallace to improve the efficiency of our multiplier. Several previous endeavours for reducing area, delay and power consumption of digital FIR filter usually focus on the optimization of the filter coefficient while the filter order is fixed, Kashyap and Maheshwari (2014). FIR filter structures are simplified to, minimizing the number of additions/subtractions & Add and Shift operations which is the main focus of those approaches. However, one of the drawbacks encountered in those approaches is that once the filter architecture is determined, the coefficients cannot be altered, Aksoy *et al.* (2013). Consequently, those schemes are not appropriate to the FIR filter with programmable coefficients, Hsiao *et al.* (2013). Reconfigurable FIR filter with modified Amplitude Detector (AD) and control logic is introduced to reduce the area and power utilization, Lee *et al.* (2011). But it makes performance degradation. Previously described works have been focused on reducing the power consumption and improving the configuration of filter coefficients. However, all those architectures have more complexity, because of using traditional hardware structures to perform multiplication and accumulation functions. In order to reduce the hardware complexity of MAC unit, redundant logical functions are identified with the help of Boolean expressions. It is identified that half adder and full adder are used in every digital signal processing operation like MAC and ALU. Hence, the redundant Boolean logical expressions of half adder and full adder are identified to



optimize the digital signal processing operations. So, our proposed Direct FIR filter offers optimum area, delay and power compared with the all other filter techniques also without any degradation. Because Enhanced Wallace Multiplier with Improved Carry-Save adder is incorporated into proposed FIR filter.

2.2.2 A Review on Sqrt-CSLA Based FIR Filter

An Optimized Wallace Tree Multiplier using Parallel Prefix Han-Carlson Adder is proposed by Priyatharshne *et al.* T.N (2014). The response time of existing Wallace tree multiplier has been scaled down. The comparison result also indicates that a substantial reduction of power is attained at an operating frequency of 200 MHz. The proposed system offers the minimum propagation delay and reduces the number of sequential adding stages. The computation time of the Wallace tree has reached the lower bound of $O(\log_{3/2} N)$. For n-bit Wallace tree multiplier, the number of steps needed is $(\log_{3/2}(n/2) + 1)$. Wallace tree has significant complexity and timing advantages over traditional matrix multipliers. The results show that the suggested architecture is more efficient than the conventional one in terms of power consumption and latency. The simulations have been carried out using the pyxis v10.1 EDA tool.

A simple approach is proposed by P. Sreenivasulu *et al.* (2012) to reduce the area and power of Sqrt CSLA architecture. The reduced number of gates of this work offers the great advantage in the reduction of area and also the total power. Based on this modification square root CSLA (Sqrt CSLA) architecture have been developed and compared with the regular Sqrt CSLA architecture. The proposed design has reduced resource utilization and power as compared with the regular Sqrt CSLA. This work evaluates the performance of the proposed designs in terms of delay, area,



power, and their products by hand with logical effort and through programmable logic design technology. The results analysis shows that the proposed CSLA structure is better than the regular SQR CSLA.

Chu (1987), describes a Booth multiplier for multiplying the first number with a second number to produce a product as an array of adder cells arranged in a plurality of rows of adder cells and is provided with input circuitry that reduces the power consumption of the multiplier. This input circuitry includes a plurality of Booth recoding logic cells that provide the control signals to multiplexers in the adder cells in the array. The Booth recoding logic cells receive different subsets of bits of the second number and generate the Booth recoded control signals as a function of the received subset of bits. Each Booth recoding logic cell includes balanced logic circuitry for generating all of the Booth recoded control signals from that Booth recoding logic cell at the same time. The balanced logic circuitry minimizes temporary short circuit paths in the multiplexers in the adder cells. This minimization of the short-circuit paths greatly reduces the power consumption of the Booth multiplier.

Anveshkumar *et al.* (2010), describes an FFT circuit that provides the high performance (throughput), dynamic range, low power, functionality, and flexibility that can benefit future needs of wireless communications systems. The Fast Fourier Transform (FFT) is a computationally intensive digital signal processing (DSP) function widely used in applications such as imaging, software-defined radio, wireless communication, instrumentation and machine inspection. In particular, it provides for run-time FFT transform length selection, pruning to support Software defined radio (SDR). This is achieved with a simple, localized, regular circuit which minimizes overall system support costs associated with design, test, and maintenance.



A new reduced-bit multiplication algorithm based on a formula of ancient Indian Vedic mathematics has been proposed by Honey Durga Tiwari *et al.* (2008). In this research new multiplier and square architecture are proposed based on the algorithm of ancient Indian Vedic Mathematics, for low power and high-speed applications. It is based on generating all partial products and their sums in one step. The design implementation on ALTERA Cyclone –II FPGA shows that the proposed Vedic multiplier and square are faster than array multiplier and Booth multiplier. The FPGA implementation result shows that the delay and the area required in proposed design is far less than the conventional booth and array multiplier designs making them efficient for the use in various DSP applications.

A carry-select adder is implemented by using a single ripple-carry adder and an add-one circuit instead of using dual ripple-carry adders by Youngjoon Kim (2001). A multiplexer-based add-one circuit is also proposed to reduce the area with a negligible speed penalty. The proposed 64-bit carry-select adder requires 42% fewer transistors than the conventional carry-select adder.

A 64-bit square root carry-select adder with only one carry evaluation block and one modified add-one circuit instead of a dual ripple-carry adder structure are presented in Yajuan He *et al.* (2005). In this research, an area efficient square root CSL scheme based on a new first zero detection logic is proposed. The proposed CSL witnesses a notable power-delay and area-delay performance improvement by virtue of proper exploitation of logic structure and circuit technique. For 64-bit addition, the proposed CSL requires 44% fewer transistors than the conventional one. Simulation results indicate that the proposed CSL can complete 64-bit addition in 1.50 ns and dissipates only 0.35mW at 1.8V in TSMC 0.18 μm CMOS technology.



Ming-Chen *et al.* (2005), presents a low power parallel multiplier design, in which some columns in the multiplier array can be turned-off whenever their outputs are known. In this case, the columns are bypassed, and thus, the switching power will be saved. The advantage of this design is that it maintains the original array structure without introducing extra boundary cells, as did in previous designs. Experimental results show that it saves 10% of power for random input. Higher power reduction can be achieved if the operands contain more 0's than 1's. Compared with row by passing multipliers, this approach achieves higher power reduction with smaller area overhead.

A new reduced-bit multiplication algorithm based on a formula of ancient Indian Vedic mathematics has been proposed by Harpreet Singh Dhillon (2008). A multiplier architecture based on this Sutra has been developed and is seen to be similar to the popular array multiplier where an array of adders is required to arrive at the final product. Due to its structure, it suffers from a high carry propagation delay in case of multiplication of large numbers. This problem has been solved by introducing Nikhila Sutra which reduces the multiplication of two large numbers to the multiplication of two small numbers. The framework of the proposed algorithm is taken from this Sutra and is further optimized by use of some general arithmetic operations such as expansion and bit shifting to take full advantage of bit-reduction in multiplication. The computational efficiency of the algorithm has been illustrated by reducing a general 4x4-bit multiplication to a single 2x2-bit multiplication operation.

O. J. BEDRIJT (1962) describes a large, extremely fast digital adder with sum selection and multiple-radix carry. Boolean expressions for the operation are included. The amount of hardware and the logical delay for a 100-bit ripple-carry adder and a carry-select adder are compared. The adder



system described increases the speed of the addition process by reducing the carry-propagation time to the minimum commensurate with economical circuit design. The problem of carry propagation delay is overcome by independently generating multiple radix carries and using these carries to select between simultaneously generated sums. In this adder system, the addend and augend are divided into sub added and subagent sections that are added twice to produce two subsumes. One addition is done with a carry digit forced into each section, and the other addition combines the operands without the forced carry digit. The selection of the correct, or true, subsumed from each of the adder sections depends on upon whether or not there actually is a carry into that adder section.

2.3 RECENT SURVEY ON FIR FILTER

In order to determine the research problem a brief literature review is done on various existing research works focused on designing FIR filters for improving the efficiency of DSP applications. Some of them are presented here. For example, Kanchana Bhaskaran (2013) proposed a modified carry select adder for operating in lower power and provided more area efficiency. Circuit and logic level modifications are applied to reduce the number of transistors in order to diminish the area utilization and power dissipation. From the simulation it is verified that the modified CSA has more advantages than the conventional one. Srinivasan *et al.* (2013) discussed about two basic FIR structures utilized in linear phase FIR filter. One is direct form FIR filter and the other one is MAC unit-based FIR filter. The efficiency of MAC unit is changed by full adder. Full adder circuit is used for power minimization which is essential for reducing the power consumption for low power applications. Kharate *et al.* (2013), presented about computational time taken by the processor. FIR filter has elements for estimating operations. More number of multiplication operation is repeatedly carry out in DSP applications.



The delay reduction can shrink the entire computational time in DSP processors.

Kannan *et al.* (2014) designed an Improved CSA (ICSA) involving parallel processing lacking carry propagation delay. The proposed ICSA adders provide area reduction and high speed comparing with other conventional systems. But Gnanasekaran *et al.* (2014) proposed normal Wallace multipliers with various high-speed adders consumes more power, area and reduced delay suits for recent DSP applications. Kashyap *et al.* (2014) used a compact full adder, half adder with ICSA adder incorporated into Wallace multipliers for increasing the performance of the multiplier. Most of the existing research works were focused on minimizing the area, reducing the delay and power consumption in FIR filter with the help of optimization process over the filter coefficients. The optimized filter coefficients can increase the efficiency in terms of area, delay and power than the existing approaches but not up to the market. Shrividhya *et al.* (2015) discussed about the implementation of 24-tap FIR filter incorporated with Winograd algorithm for increasing the area efficiency. The author also analyzed the FIR filter performance by changing the adders like CSA, RCA and Wallace multiplier adders. Finally, the structure is optimized for making it as cost effective.

Thanuja *et al.* (2016) proposed a new distributed arithmetic method for calculating sum of product to reduce the number of multipliers and accumulators to reduce the size of the circuit. To do that the blocks are reused by using multiplexer structures for reducing the memory locations needed for operations. Sejal *et al.* (2017) presented a detailed survey and presented about various design methods carried out for implementing FIR filters with MAC operations. Also, FIR filters replaced using DA algorithms and using LUT as a part of FPGA. Finally, from the performance analysis among the various methods it is concluded that the modifying FIR filter is more effective in terms



of area, power, delay and Hardware complexity by reconfiguration. Kiran Mojesh *et al.* (2017) proposed a new design and implementation method for a micro-programmed S/P FIR filter architecture using various adders like RCA, Kogge-Stone adders and Vedic multipliers using compressors combined with them respectively. From the experimental results the author concluded that the performance of the FIR filter is mainly depend on the multipliers used whereas by concentrating on the multipliers and adders the performance of the FIR filters can be improved.

2.4 SUMMARY

From the above discussion, it is noticed that FIR filter is the most important component in communication systems and digital signal processing including several portable applications. Also, the efficiency can be increased by focusing on multipliers and adder's configuration in FIR filter architectures. This problem is considered and this research work focused on designing a direct-form Finite Impulse Response (FIR) digital filter using efficient multiplier and adder circuits for optimized area, power, delay and increase in speed in digital signal processing (DSP). Hence this research work is focused on the multipliers and adders involved in the FIR filter architecture for improving the efficiency which suits for any emerging DSP applications.



CHAPTER 3

WALLACE MULTIPLIER WITH KOGGE-STONE ADDER

A complete performance procedure performed in this study is clearly explained in this chapter. It describes the complete analysis, comparison and discussion with various types of multipliers.

3.1 INTRODUCTION

In digital signal processing various basic functions, arithmetic functions and arithmetic calculations are performed like

- Addition
- Subtraction
- Multiplication
- division, and so on.

Multiplication is one of the complex procedural methods in digital signal processing comparative to addition and subtraction. In Asadi (2007), author has discussed about distinctive characteristics of processing of a multiplier and its overall percentage is calculated as 8.72%. To implement and perform arithmetic operation in a computer, the central processing unit (CPU) commits substantial amount of processing time. In that for performing multiplication it takes a considerable amount of time. To achieve high data throughput in most of the high-performance digital signal processing systems they depend on hardware multiplication. Multiplier is one of the imperative



components in digital signal processing unit, which throw in substantially to total power consumption of the entire system. Multiplication is mandate function of multiply and accumulate (MAC) unit, so it required the need of high-speed multiplier. The main factor of Multiplication time is determined by the instruction cycle time of a DSP chip. So, the amount of multiplier circuits is directly proportional to the square of its resolution i.e. $O(n^2)$ gates – Size of multiplier of n bits.

Various innovative thoughts have been proposed for designing multipliers with improved performance. Finding innovations is increased for designing high speed processors where those processors can be used in recent multi-media and signal processing applications. In order to obtain the improved performance in terms of throughput/arithmetic operations it is essential to concentrate on adder, multiplier and other CMOS circuit designing. Reduction of time delay and power consumption is very essential for many applications. In past few decades, one of the required arithmetic operations in various applications is multiplication and proposing new multiplier circuit with high performance. To perform an M -bit by N -bit multiplication the procedure is shown in Figure 3.1. This Figure-3.1 shows that A is the multiplicand, and B is the multiplier will be product using the Equation 3.1 and 3.2. The Equation for the product is defined in Equation 3.3 as discussed by Kim (2010).



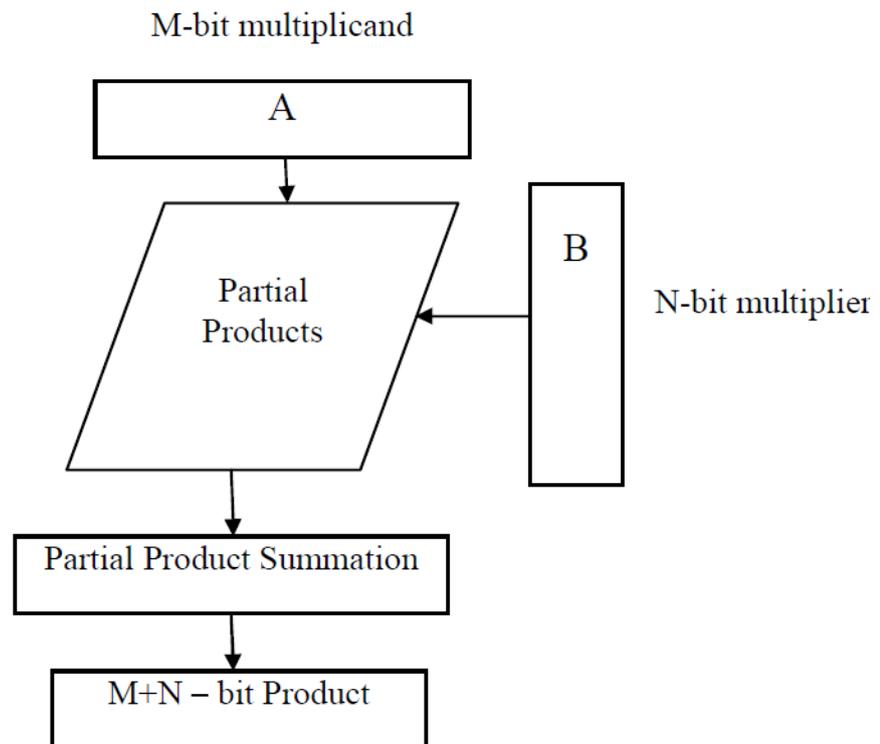


Figure 3.1 Generic multiplier block diagram

A multiplier can be divided into three Stages,

1. Partial productions generation stage
2. Partial Products addition stage
3. Final addition stage.

In Stage One, A partial product is generated bit by bit by multiplying the multiplicand and the multiplier. A second order booth encoding algorithm is normally used as an alternate for the reduction of partial products to half. In stage two, it is very important stage that determines the speed of the complete multiplier and it is most complicated stage. In stage three, look ahead adder and high-speed adder is used to generate the output. This uses two row outputs in a tree to generate output results.

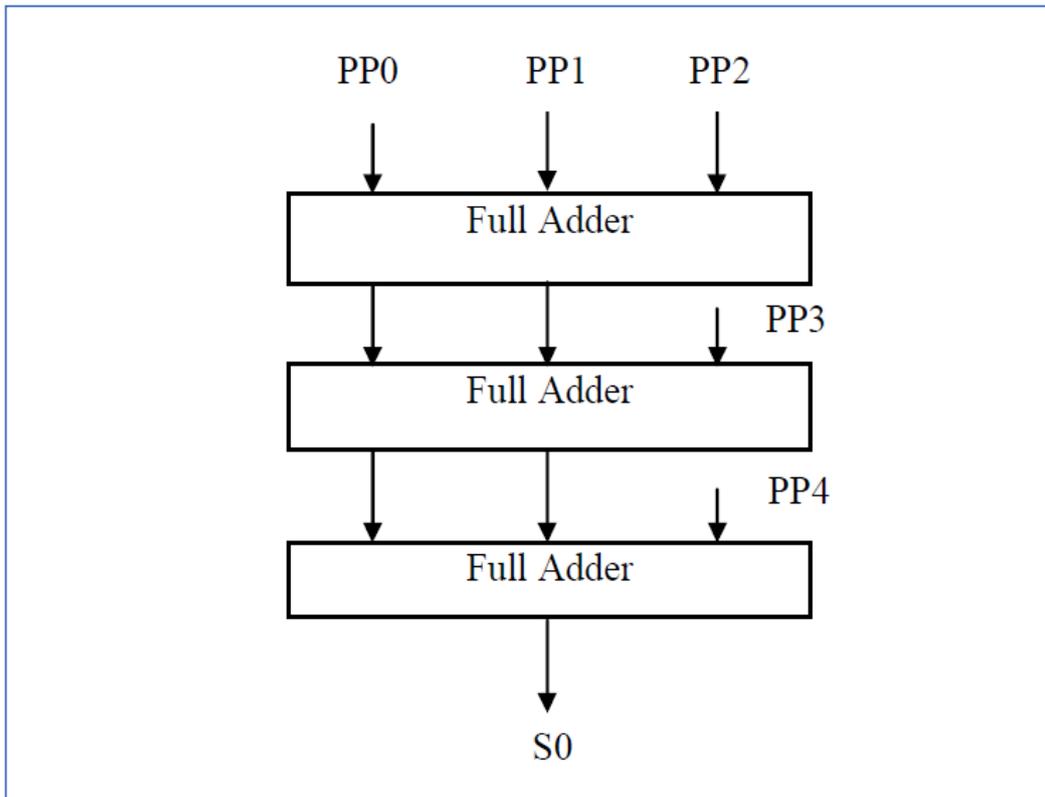


Figure 3.2 Array Multiplier Mechanisms

Mechanisms of the multiplier are classified into their structure, application usage and production of partial products and added up. Which are categorized as

1. Array Multiplier
2. Tree Multiplier

Array multiplier mechanism is described in Figure 3.2. Tree Multiplier is extreme fast in architecture for adding partial products. To perform parallel addition in tree structure it requires order $\log N$ stage. This is reduced to N partial products literally. Multiple input compressors are employed in tree multiplication algorithm to reduce the number of partial products which are accumulated in several partial products. This was clearly explained in Figure 4.3. Multiplications process of large operands can be handled in tree multiplier. Tree multiplier implements CSA tree constructed

from 1-bit full adder, which was used to achieve minimization of partial product numbers in quick and proficient way.

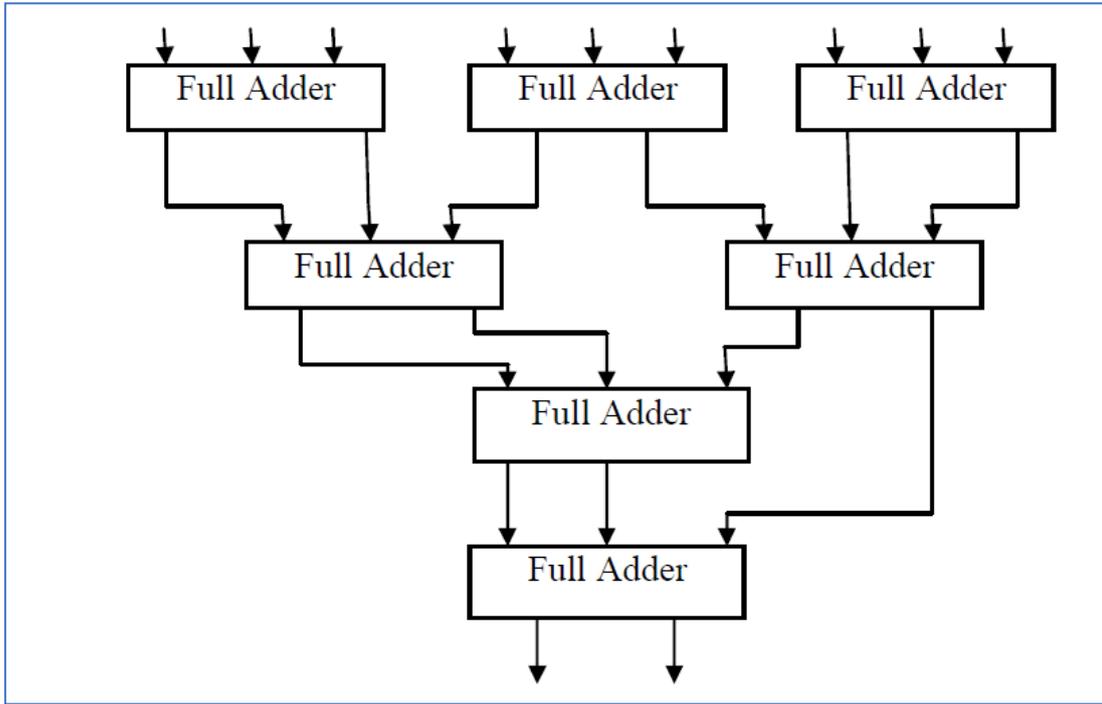


Figure 3.3 Partial product addition using tree topology

Wallace proposed the first tree structure. Wallace implemented connection of 3:2 compressors in tree topology. These compressors are connected in parallel structure in order to reduce Partial products. The standard tree structures consist of Balanced delay, binary and overturned staircase as same as 9:2 compressors.

There are various tree structures that are classified as

1. Binary Tree
2. Balanced –Delay tree
3. Overturned Staircase tree
4. Wallace tree

3.1.1 Compressors

Multiplier uses Compressors mostly to limit the operands while summing terms of partial products. A compressor is a combinational device. It compresses N input lines in the position i to 2 output lines. 2 output lines are sum and carry. In this compressors L input lines impending to the compressors to various levels j . A simple compressor methodology was described in Figure 3.4.

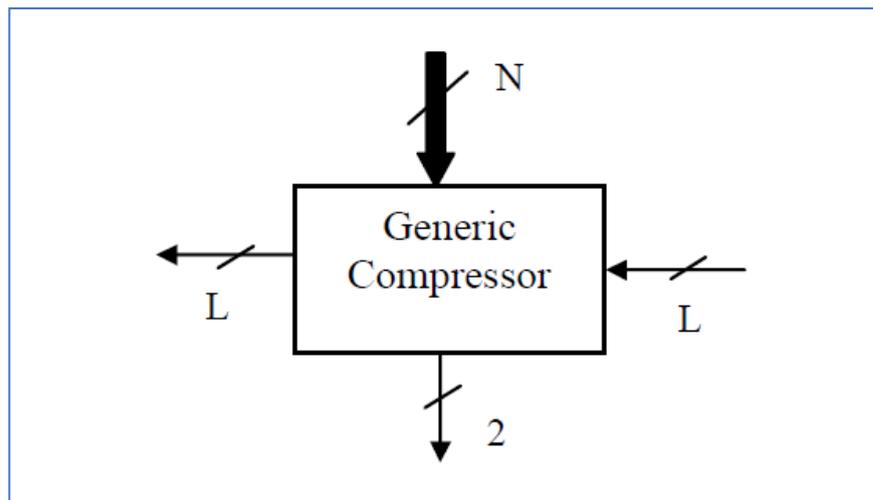


Figure 3.4 Generic compressor.

Compressor is normally a full adder. It has 3 inputs such as i_1 , i_2 , i_3 . These 3 inputs are summed up and provide 2 outputs i.e. sum and carry. Figure 3.5 describes a gate level diagram of 3.2 compressors.

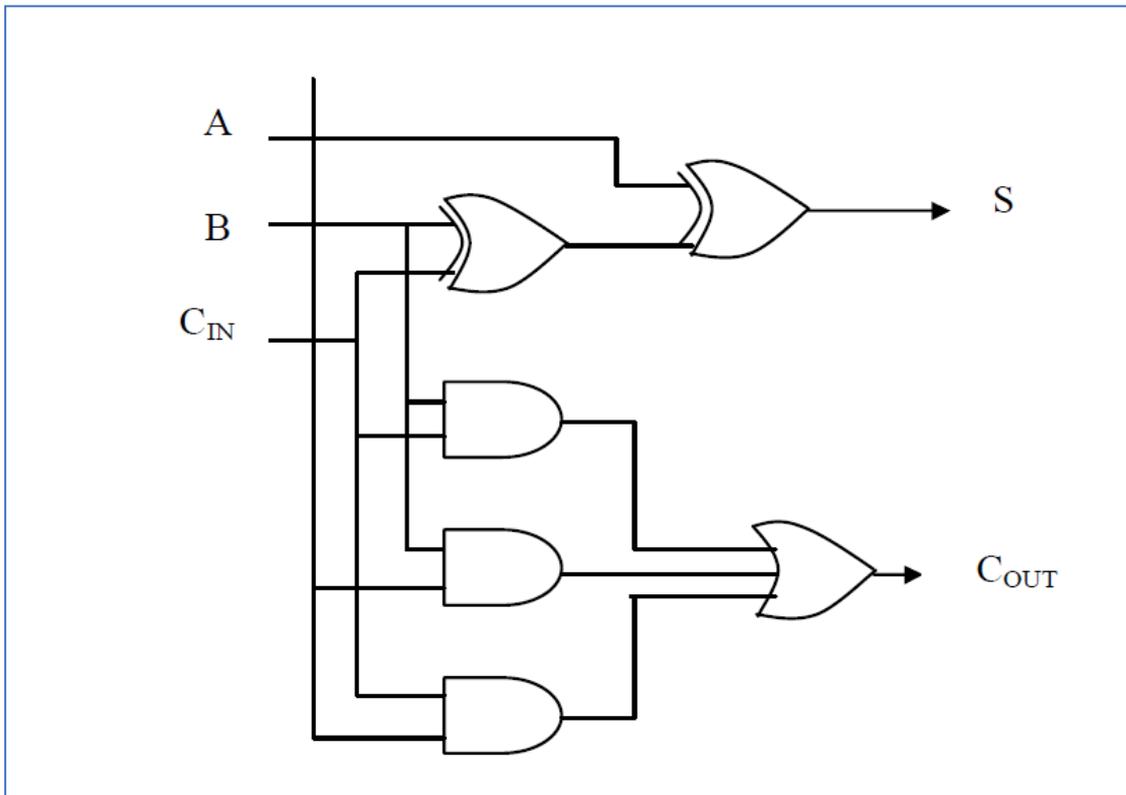


Figure 3.5 Gate level design of (3:2) compressor

Similarly, 4:2 compressors have 4 input lines such as i_1 , i_2 , i_3 and i_4 which are added up and have 2 output lines sum and carry. The other additional lines are input and put-put carries. Figure 3.6 describes gate level design of a 4:2 compressor. Figure 3.7 illustrate a 4, 2 compressor that designed using two 3:2 compressors.

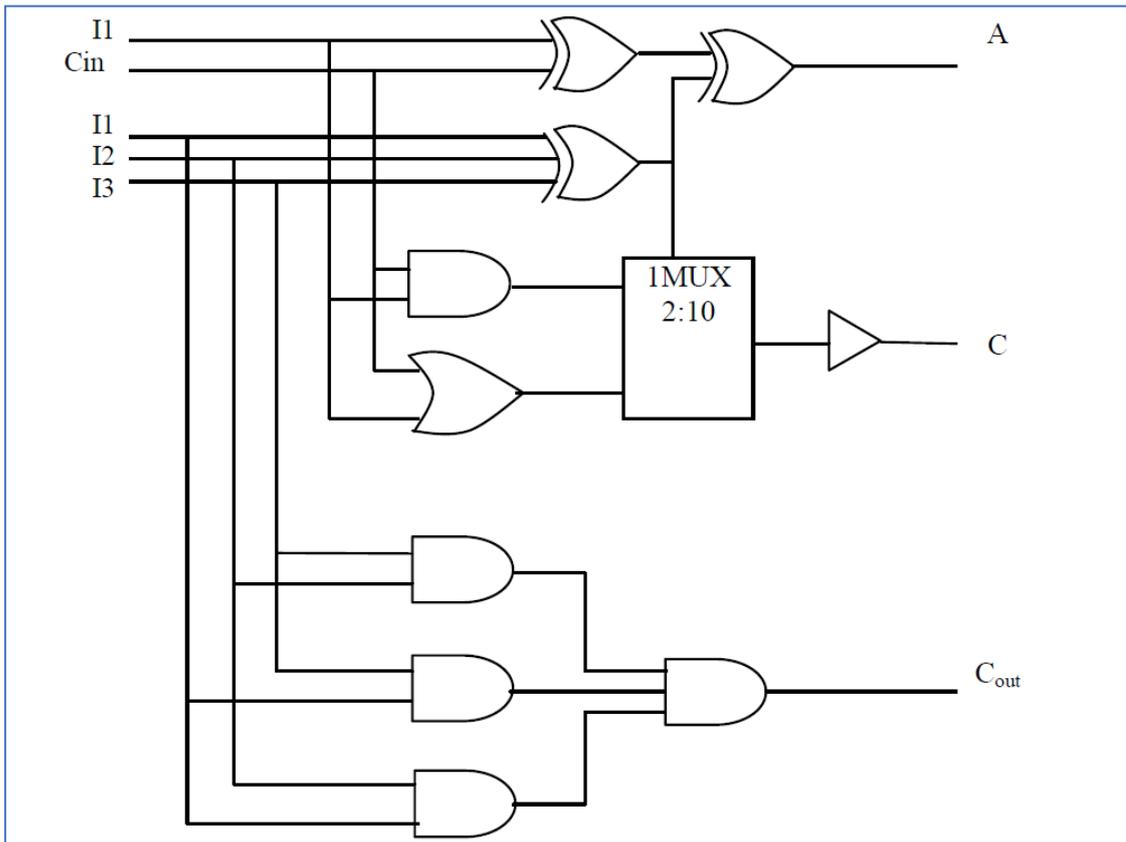


Figure 3.6 (4:2) Compressor logic diagram

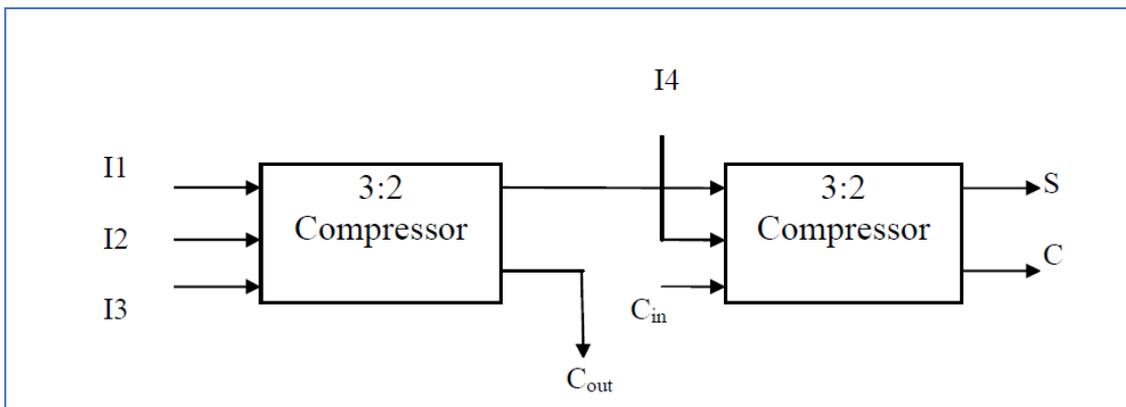


Figure 3.7 (4:2) Compressor using (3:2) compressor

3.2 MULTIPLIER TOPOLOGIES

This segment clearly explained the representation and the design structure of a multiplier topology. In this activity the multipliers structures are

organized, combined and analyzed for new MAC unit proposal. Various multiplier topologies are

- Booth multiplier
- Modified Booth Multiplier
- Booth encoded Wallace Tree Multiplier and
- Wallace tree Multiplier.

3.2.1 Booth Multiplier

Booth multipliers are conventional array multipliers. They are like Baugh Woolley multiplier and Braun Multiplier to achieve extreme performance comparatively. They require large area of silicon distinct add shift algorithms. These add shift algorithms uses limits hardware and very poor performances. Booth multiplier uses Booth encoding algorithm. By using this algorithm, it reduces the number of partial products by believing 2 bits multiplier at same time. This is to achieve speed advantage over different multipliers architectures. Both signed and unsigned numbers use the same algorithm. This algorithm accepts radix 2 computation.

3.2.1.1 Booth Recoding

Booth recoding is also known as Booth algorithm, it has been proposed by Andrew D.Booth in 1951. In this technique, the multiplication of 2's complement number is performed without the signed bit extension. The number of partial products can be decreased by subtraction in string of '1' bit occur in the multiplicand. Booth algorithm operation was explained in Table 3.1.



Table 3.1 Booth Algorithm

Xi	Xi-1	Operations	Comments	Yi
0	0	Shift only	String of zeros	0
1	0	Sub and shift	Beg of string of ones	1
1	1	Shift only	String of zeros	0
0	1	Add and shift	End of string of ones	1

Arithmetic is avoided in case of string 0's, so it is left alone. Booth old algorithm takes care on only one bit at a time whereas; in innovated booth algorithm it uses original coding by taking 2 bits multiplier in contrast. New algorithm depends on four main cases. Mainly it depends on values of 2 bits. In the first step the pair of bits are examined, that consists bit by bit to right. In the second step, the product is shifted right.

3.2.1.2 Booth example

Consider two numbers that to be multiplied, $= -34 = -(0100010)_2$ and $= 22 = (010110)_2$. This represents the operands with negation of signed 2's compliment:

22:0010110, -22:1101010

34:0100010, -34:1011110

An example of booth algorithm is showed in table 3.2. Results are stored in two registers namely (A) and (Q). Multiplicand is Register (M). 2's bits multiplier is recorded at same time to perform the actions that stipulated in table-3.4. Hereafter bits are recoded and the result is 1111010 0010100. The results are stored in 2 registers, the upper half of the results stored in register (A), and the lower half of the result will be stored in register (Q). The final product result is in signed.



$$A \times B = -00001011101100 = -(748)_{10}$$

Table 3.2 Booth Example

qiqi-1	Action	[M] [A]	0010110 0000000	[Q]	1011110	0
00	Right shift		0000000 0101111	0		
01	-A	+	1101010			
			1101010		0101111	0
	Right shift		1110101 0010111	1		
11	Right shift		1111010 1001011	1		
11	Right shift		1111101		0100101	1
11	Right shift		1111110 1010010	1		
01	+A	+	0010110			
			0010110 1010010	1		
	Right shift		0001010 0101001	0		
10	-A	+	1101010			
			1110100 0101001	0		
	Right shift		1111010 0010100	1		

Serial multiplier is usually applied in the scheme serial recoding. This method is simple and eases to implement. Booths algorithm results in minimization in number of scenarios that comes under booth algorithm. In such sequence the result is such as $01010101 \dots 01$ is stumbled upon. Where n is multiplier length in addition and subtractions. This is known as worst case standard multiplier.



3.2.2 Modified Booth Algorithm

Macsorley proposed the modified Booth algorithm (MBE) or Modified Booths algorithm (MBA) by Lio (002) as discussed. To implement large number of partial multipliers and generate the partial products this recoding method is widely used. It implements parallel encoding scheme. By enhancing parallelism in high speed multiplier reduces the number of subsequent stages. There are major drawbacks in original version of Booths algorithm (Radix -2) are:

Addition and subtraction operations and the numbers of shift operations are variable and it becomes problematic in planning parallel multiplier.

During the isolation of 1's the algorithm becomes inefficient.

These drawbacks can be overcome by modified Booths algorithm. During Recoding the MB processes 3 bits at the same time. To speed up the standard booth multiplication algorithm in powerful way recoding of higher radix multiplier is required. In every cycle the greatest number of bits can be identified and eradicated. Hereafter it limits the total number of cycles to get product is minimized. The number of bits identified in radix r is stated as $n=1+\log_2 r$.

In radix 4 algorithm, 3 bits are identified and 2 bits are deleted in each cycle.

Steps to be followed to implement radix 4 algorithm as follows:

- Add 0 to right of LSB
- If necessary extend the sign bit 1 position to make certain that n is even.
- Find partial product for the value of each vector.



Table 3.3 Modified booth algorithms

Y_{2i+1}	Y_{2i}	Y_{2i-1}	Recoded Digit	Operand Multiplication
0	0	0	0	0*Multiplicand
0	0	1	+1	+1*Multiplicand
0	1	0	+1	+1*Multiplicand
0	1	1	+2	+2*Multiplicand
1	0	0	-2	-2*Multiplicand
1	0	1	-1	-1*Multiplicand
1	1	0	-1	-1*Multiplicand
1	1	1	0	0*Multiplicand

Radix 4 minimizes the number of multipliers digits by factors of 2, which implies the multiplier digit reduces in such a way from 16 to 8. Propagating carry to subsequent stages or cycles is not followed in Booths recoding method. In this algorithm the multiplier is grouped into three consecutive digits, where the outermost digit in every group is shared with outermost digit of each adjacent group. In each group the binary digits, then corresponding to the one of the numbers the group of sets {2, 1, 0, -1, -and 1}. Recoded produces 3 bit output where the first bit is represents number 1 and the second bit represents number 2. The third and final bit represents the first and second bit is negative or not.

Modified Booth Example

Consider 2 numbers to be multiplies,

=34 and = -42

Multiplicand A = 34 = 00100010

Multiplicand B = -AxB = -1428



Table 3.4 Modified booth example

										0	0	1	0	0	0	1	0		34
										1	1	0	1	0	1	1	0		-42
						1	1	1	1	0	1	1	1	1	0	0			PP ₁
				1	1	0	0	1	0	0	0	1	0	0					PP ₂
		1	1	0	0	0	1	0	0	0	1	0							PP ₃
1	0	1	1	1	0	1	1	1	1	1	0								PP ₄
1	1	1	1	1	1	0	1	0	0	1	1	0	1	1	0	0			-1428

Modified Booth Partial products are formed from booths algorithm like PP₁, Pp₂, Pp₃, and Pp₄. Partial product indentified first is determined and appending with zero in three digits LSB multiplier. Multiplicand A is a three-digit number i.e. 100 to multiply by -2. The one bit of the product is multiplied by -2. Therefore, the first PP is 1100111100.

Partial products are nine bits length. The second partial product is determined and multiplied by 2 with next three bits. Multiplying by two states that the multiplicand value is shifted left one bit.

Therefore, the second partial product is 001000100. Likewise, the third partial product has to multiply by 1. Then the third partial product value of multiplicand is 000100010. The 4th partial product is identified by immediate 3 bits and multiplied by -1. It means the multiplicand has to convert into 2's complement value. Then the fourth partial product is 111011110. Sign bit of each previous block was stored in LSB of each bit. There are not negative products in LSB. So, least significant both of every frost block is always consider being zero. Figure 3.8 describes the Block diagram of nxn bits of Modified booths Multiplier.



It implements Booth encoder, Sign Extension bits and the multiplier array. This contains the partial product generator and single bit adders and the final stage adder. This was executed by 2-bit addition.

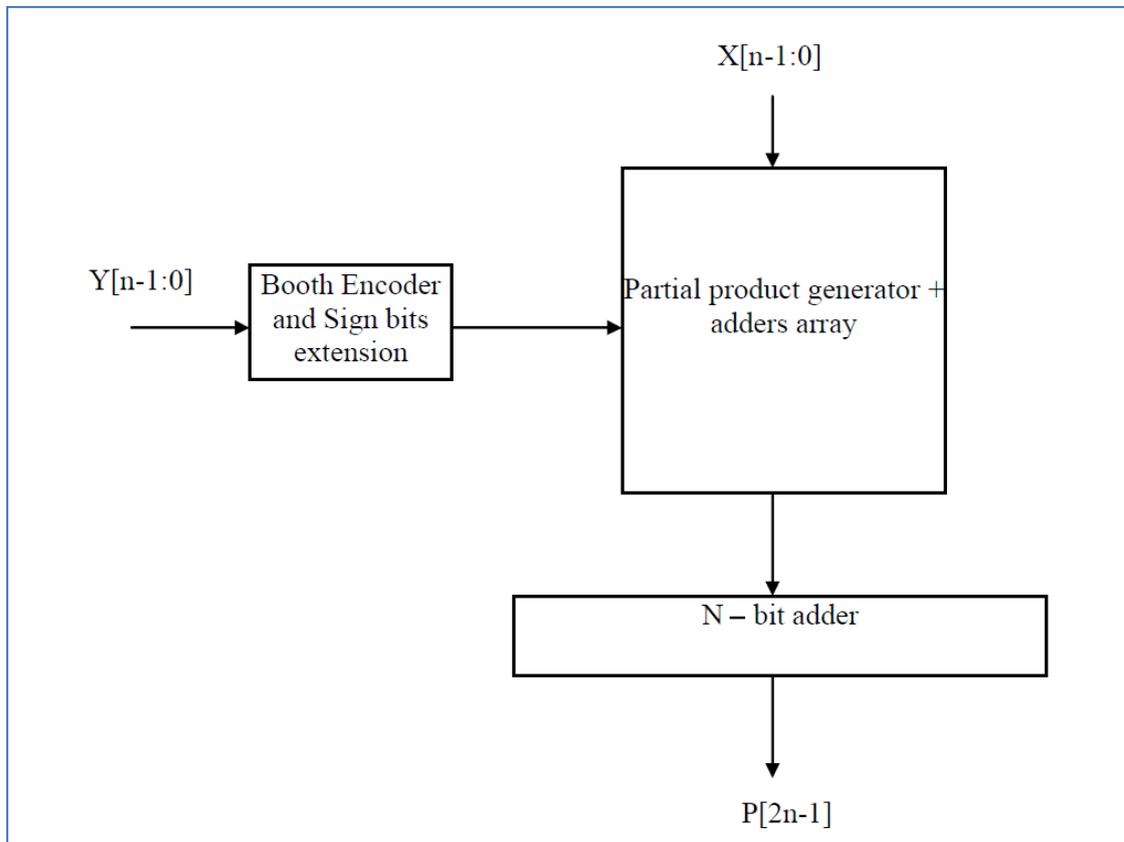


Figure 3.8 $n \times n$ modified booth multiplier

3.2.3 Wallace Tree multiplier

Wallace introduced the quickest process of multiplication among two numbers. In which the addition is performed parallel in minimum delay was observed by Wallace in 1964. Wallace proposed a new way of parallel addition of a partial product bit by using a tree of carry save adder. By in turns considered as an effective architecture of digital circuit tat multiplies 2 numbers to perform a multiplication with Wallace implementation. It minimizes partial product matrix to two-row matrix. For this purpose, it uses a

carry save adder. The final 2 rows are added issuing fast carry propagate adder. This forms a product. This becomes an advantage that is more used for multipliers that is bigger than 16 bits. In Wallace tree architecture, addition of all the partial products of all the bits in each column is summed together by a set of parallel counters without disseminating any carry. Later another set of parallel counters limits the initiating new matrix and so on. It is done until 2-row matrix is generated. To perform multiplication operation in Wallace method it uses three steps.

- The three steps in Wallace multiplication operation are:
- Arranging of bit products.
- By using carry save adder the bit product matrix is minimized to a 2-row matrix.
- Fast carry-propagate adders used to add the remaining rows in order to produce the product.

Multiplication Operation in Wallace tree multiplier

To multiply two integers Wallace tree is excellent hardware architecture in digital circuits. There are three steps in Wallace tree, they are:

- Each bit of the operands is multiplied i.e. AND by each bit of the other operand, which capitulate n^2 results. The wires carry different weights which is dependent in the position of the multiplied bit.
- The numbers of partial products by two layers of half and full adders are minimized.
- Clustering the wires of two integer and summing them with a conventional adder
- The phase two mechanism as follows, if there are three or more wires with similar weight are added to form a new layer.



- Input of a full adder is summing of any three wires with same weight. Therefore, the weight of the output wire will be of same weight as input wire and a higher weight of an output wire for each of input wire.
- Half adder is used if the weight of any two wires is same.
- If only one wire is left it is connected to the next layer.

Wallace proposed an unusual way of parallel addition. Partial product addition of bits is performed using tree of carry save adder. This implies multiplication of two integers using Wallace method. Partial product matrix is reduced to a two-row matrix by using a carry save adder. The left two layers are added using a fast carry propagate adder to form the product. 3:2 compressors are used to propagate the conventional Wallace tree algorithm. The propagation of higher order compressor is minimized by using Wallace tree algorithm. Wallace tree multiplier is described in Figure 3.9.

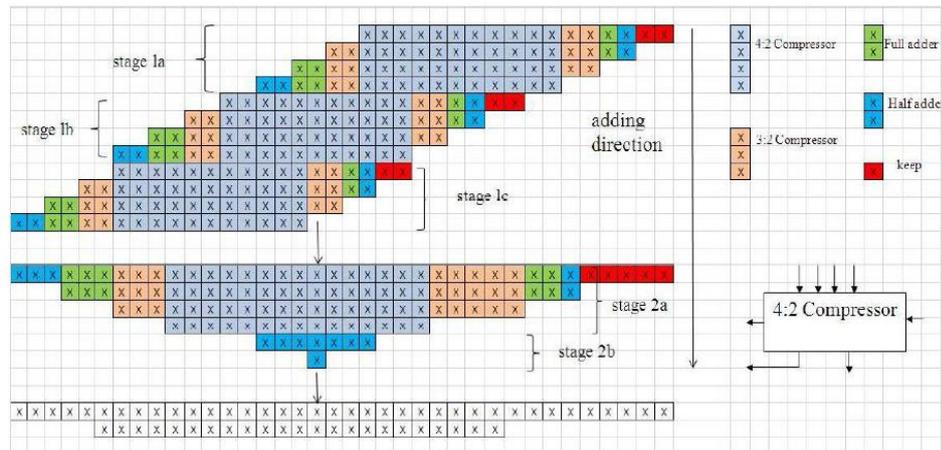


Figure 3.9 Wallace tree examples

In first stage the partial products are minimized by using compressors. The partial terms are indicated in various colors like red, light orange, blue and green. The partial term that marked in red are kept as such, and the partial term marked in green implies that the partial term is compressed with full adder. The light orange marked partial term is

compressed in 3:2 compressors and finally with blue marked partial term is for 4:2 compressors. Advantage of Wallace multiplier is the propagation delay is minimized compared to array multiplier. The limitation of Wallace multiplier is, Efficient layout is not feasible in Wallace multiplier because of its irregularity. Routing is complicated among the levels because of wires greater capacitance.

3.2.4 Dadda Multiplier Architecture

Most of the stages are similar in both Wallace tree multiplier and Dadda tree multiplier. But in Dadda tree multiplier, it does not attempt to reduce the partial product in each layer, where as it performs same minimization as feasible. This is comparatively economical due to its minimization of stages and longevity of numbers n each stage. Dadda tree multiplier requires carry save adders (CSA).

For Dadda tree multiplier architecture, the partial products formation is performed using the similar method as in Wallace tree multiplier. As on minimization stage, a series of widespread recursive stages were used to identify the heights of every stage and the number of additions required to achieve height d each stage. This is clearly explained as follows:

$$1. d_1 = 2 \text{ and } d_{i+1} = \lceil 1.5 \cdot d_i \rceil$$

Here, D indicated the height of the matrix in j th stage. Repetition of this stage till j^{th} stage is achieved, where the original N – matrix height (at least one column has more than d dots).

2. In the j^{th} stage from the last, (3, 2) and (2, 2) place counters are required to minimize matrix. Only columns with d_j dots, they receive carry from less significant (3, 2) and (2, 2) counters are reduced.



3. In this stage half adders are used once to ensure the total number of stages. These stages should not exceed that of a conventional Wallace multiplier. In few scenarios, half adder is added at the final stage of reduction only.

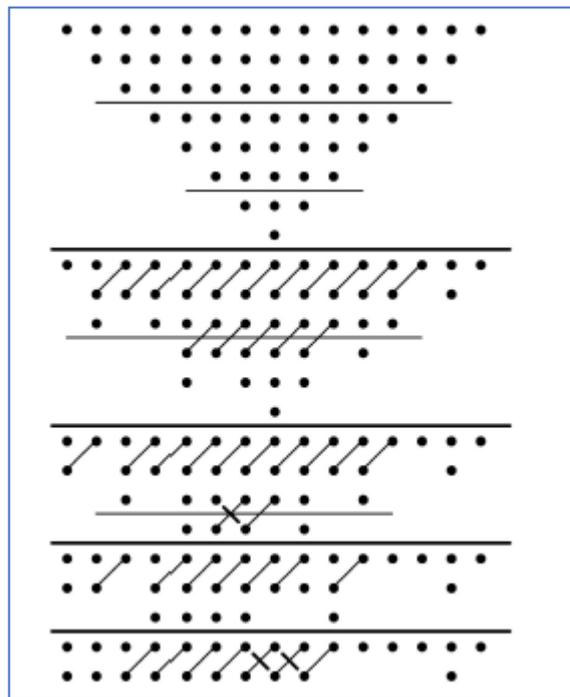


Figure 3.11 Reduced Complexity Wallace multiplier

3.3 IMPLEMENTATION OF KOGGE-STONE ADDER WITH REDUCED COMPLEXITY WALLACE MULTIPLIER

3.3.1 Kogge-stone Adder

Kogge-stone adder or KSA is parallel fix form that uses carry ahead adder. Carry is generated in $O(\log n)$ time and it is drastically considered as a quickest adder. It is used in industries for high performance arithmetic circuit industries. Carry is computer fast in KSA. As computing is done parallel in increased area. In high speed applications used these methods. It comes in cost of power and area. The delay in architecture is stated as $2 \log n$. This implementation posses $2[(n) (\log_2 n) - n + 1]$ nodes, that is computed

introduces recursive doubling algorithm in kogge-stone scheme to address fan-out problems. To limit the lateral fan-out issues, it uses idempotency property. But due to sudden increase in number of lateral wires at every stage, the cost is increased. It is because of excessive overlap of prefix sub-terms and pre-computation.

The comprehensive functionalities of KSA can be analyzed in three distinct parts. They are:

- Preprocessing
- Carry look ahead network
- Post processing

3.3.1.1 Preprocessing

In this stage one, the computation involves generating and broadcast signal communicate with all the pair of bits n register A and register B. These signals broadcasted to produce logical equations. These logical equations stated below are:

$$P_i = A_i \text{ xor } B_i$$

$$G_i = A_i \text{ and } B_i$$

3.3.1.2 Carry look ahead network

This is different from other adders. It works behind as a required force in high performance circuit. In this stage the computation involves to carry each bit correspondingly. It uses cluster propagation method and produce the signals as intermediate. These signals are represented as logical Equation such as:



$P_i: j = P_i: k+1$ and $P k: j$

$G_i: j = G_{i:k+1}$ or $(P_i: k+1$ and $G k: j)$

3.3.1.3 Post processing

This is end step and is general to all adders of this cluster (carry looks ahead). It implements computation by adding bits. The sums of bits which are computed b logical Equation given below are:

$S_i = p_i \text{ xor } c_{i-1}$

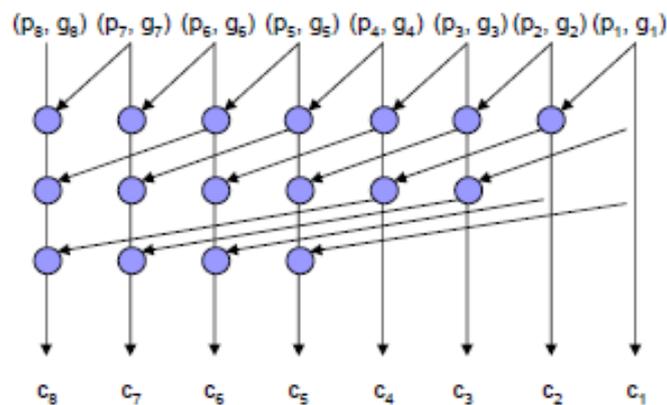


Figure 3.12 5. 8-bit Kogge-Stone adder's carry generation stage

3.4 RESULTS AND DISCUSSIONS

The conventional carry save adder is compared with kogge stone adder or KSA is implemented. Design I stipulate the multipliers with HDL – Hardware Description Language. This uses 8-bit unsigned data. It generates Power and speed output. To increase power and speed it uses XILINX ISE 10.1 as amalgamation tool and Model Sim XE III 6.3c for stimulation. It also implements FPGA-Spartan III. In 8x8 multiplier design structure, fixed block size was implemented in Kogge stone adder. Table-3.2 describes the output of

the conventional adder speed and power consumption. The Simulation Result of the proposed multiplier is shown in Figure.3.13.

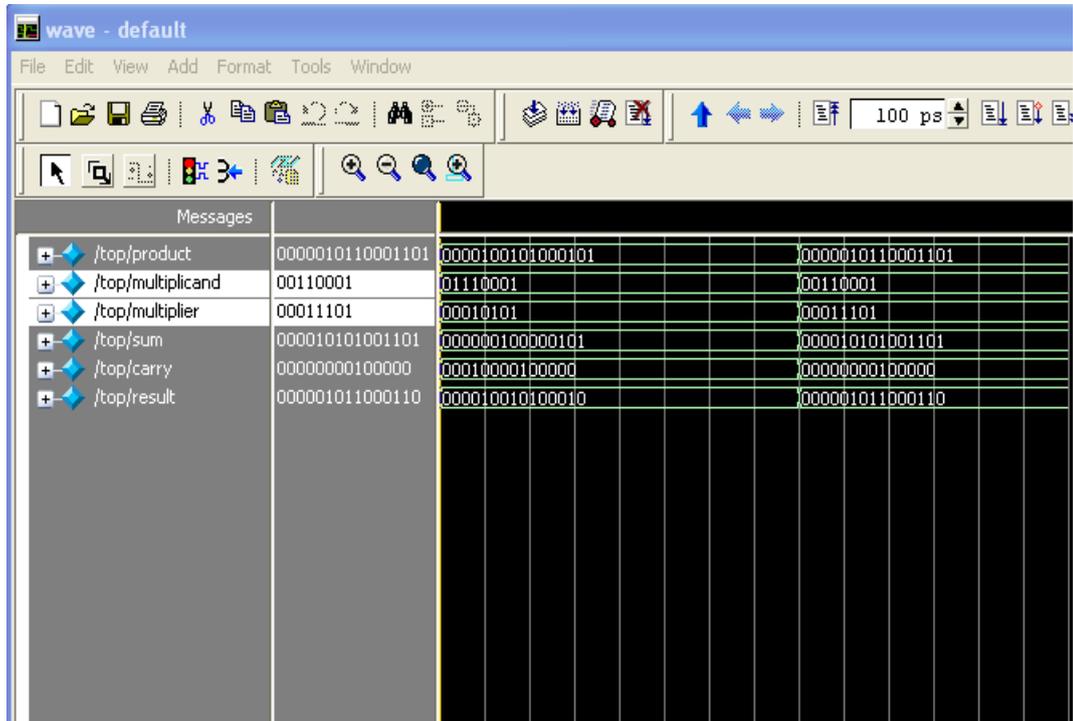


Figure 3.13 Reduced Wallace multiplier with Kogge Stone Adder (KSA)

Table 3.5 Comparison of Delay and Power of Proposed multiplier.

Various Multipliers	Delay(ns)	Power(w)
Proposed multiplier	22.809	0.157
Dadda multiplier	25.859	0.624
Multiplier using compressor	25.491	0.173
Booth multiplier	29.981	0.376

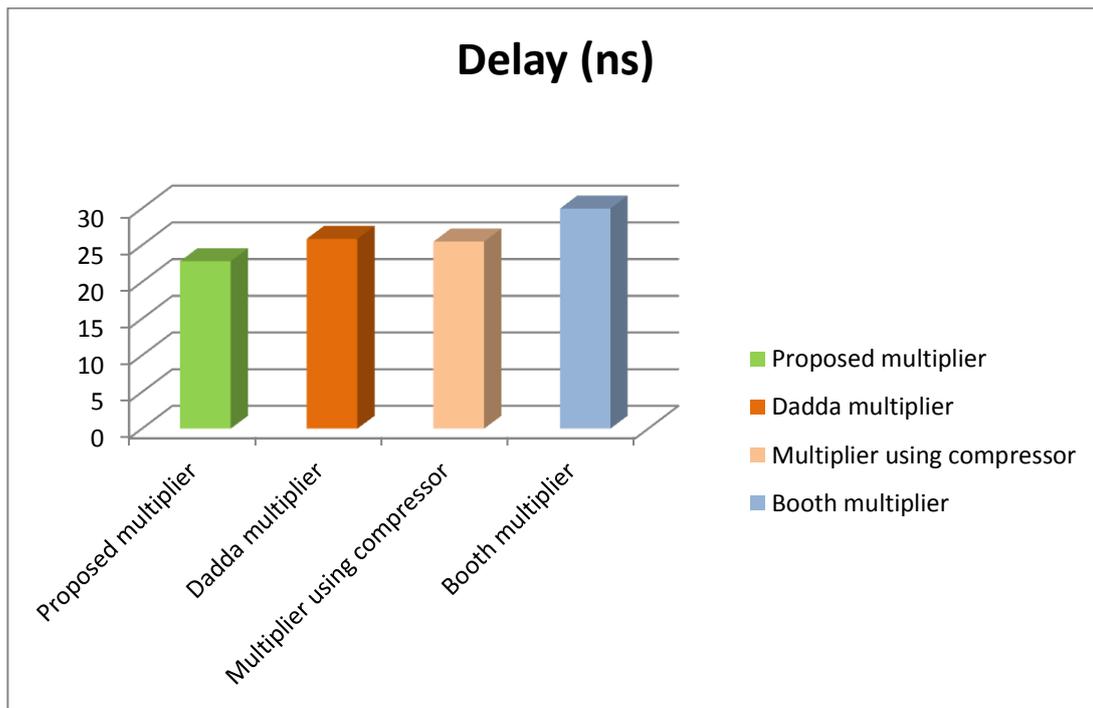


Figure 3.14 Comparison chart for Delay

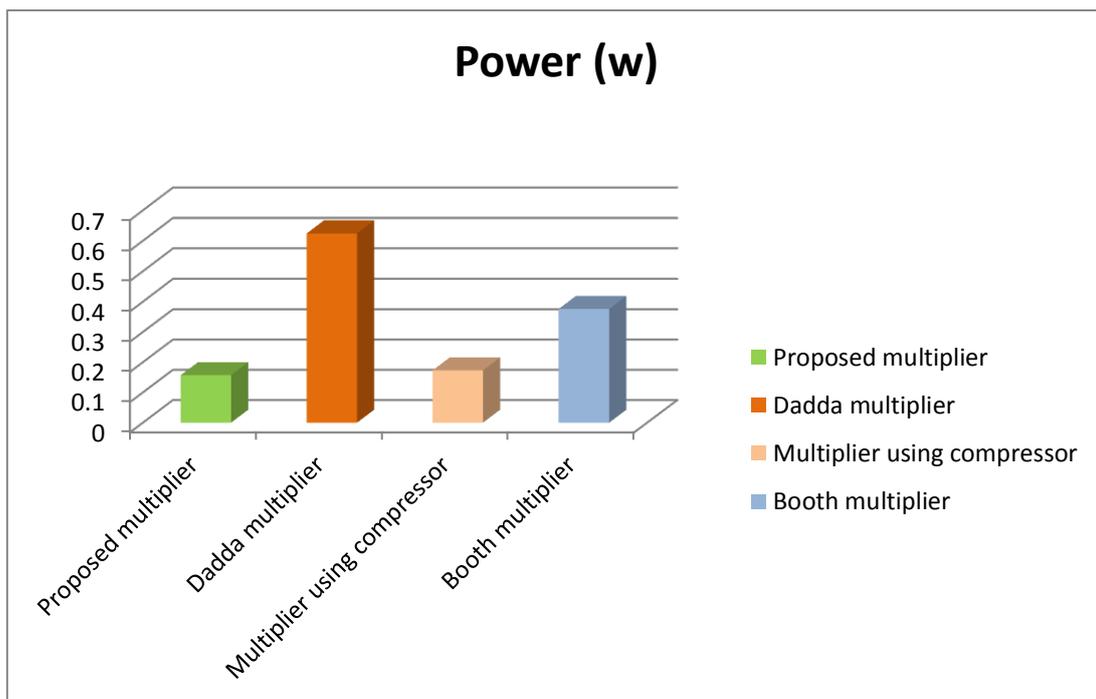


Figure 3.15 Comparison Chart for Power

3.5 SUMMARY

This chapter presents the detailed information about multipliers. It describes the complete analysis, comparison and discussion with various types of multipliers. From this chapter it is understand that how to design a VLSI circuit for effective multipliers. The next chapter discussed about FIR filters.



CHAPTER 4

HIGH SPEED MULTIPLICATION AND ACCUMULATION (MAC) DESIGN FOR DIGITAL FIR FILTER

4.1 OBJECTIVES

In the first stage of research work, it is motivated to design the efficient MAC (Multiplication and Accumulation) unit of digital FIR (Finite Impulse Response) filter to enhance the speed and throughput of the digital FIR filter. Recent days, there is an essential need of low power digital signal processing (DSP) architectures due to the enhancement of wireless applications. FIR filter plays the main role in the signal processing applications. To enhance the efficiency of the digital FIR filter, SQRT CSLA (Square Root Carry Select Adder) accumulation unit integrated with Wallace multiplier for addition procedure.

4.2 PROBLEM STATEMENT

MAC unit is the heart of the direct form of FIR filter. There is a need of proficient configuration of adder and multiplier in the design of VLSI circuit for the low power consumption, reduced size with increased speed. To avoid the computational delay of the direct form of FIR filter, need of efficient structuring of the adder and multiplier. In the previous research, in the method of MCM used in direct form of FIR filter, filter coefficients changed dynamically. In this work, reduced number of chip size and delay have



considered in the design of MAC unit for FIR filter. The effective Wallace multiplier proposed to enhance the process of multiplication in the Modified Booth Algorithm by using compressors.

SQRT CSLA (Square Root Carry Select Adder) adder used to offer the reduced size and power for the addition process. SQRT CSLA accumulation unit is personalized by modifying the carry selection block based on the BEC (Binary to Excess1 conversion) and integrated with Wallace multiplier for addition procedure with decreased complication. Thus, to decrease the size and delay with increase speed and also to decrease the complexity, Wallace multiplier is incorporated into digital FIR filter.

4.3 EXISTING REDUCED COMPLEXITY WALLACE MULTIPLIER

A Wallace multiplier is a parallel multiplier which performs multiplication operation effectively. The architecture of reduced complexity Wallace multiplier consists of less number of half adder and full adder to perform partial products. In reduced complex Wallace multiplier, N^2 AND gates are used for generating partial products and arranged in such a manner (triangle order).

The procedure for producing partial products using reduced complexity Wallace multiplier is as follows:

- The matrix is divided into three row groups in the reduced complexity Wallace multiplier.
- All three-bit combinations are added using full adder.
- Single bit and a group of two bits are moved to the next stage directly.



Finally, it requires effective digital adder structure for doing binary addition process. In existing system, modified Carry Save Adder is used for addition process. But this requires a greater number of chip size and delay for implementation. Hence to improve the performance of reduced complexity Wallace multiplier, still we require efficient adder structure. To fulfill this requirement, SQR T CSLA adder structure is re-designed in this paper. The modified SQR T CSLA adder effectively reduces the chip size and delay for addition process.

4.4 MODIFIED SQR T CSLA

General architecture of SQR TCSLA consists of Ripple Carry Adder unit when input carry 0 (RCA0), Ripple Carry Adder unit when input carry 1 (RCA1) and full sum generation unit (HSG). This method uses the two types RCA units for input carry 0 and input carry 1 respectively and therefore this causes more chip size and delay for select carry outputs. Further RCA1 unit has replaced by BEC unit for reduce the delay. Hence the existing system called as BEC based SQR T CSLA. However, this architecture also requires more chip size and low speed for addition process. Hence to overcome this problem, the circuit for SQR T CSLA is re-designed. The modified SQR T CSLA consists of Half Sum Generation (HSG) unit, FSG, Carry Generation (CG) unit for both input carry 0 and input carry 1 and Carry Selection (CS) unit. The design of modified SQR T CSLA consists only a smaller number of logic gates when compared to BEC-SQR T-CSLA. Figure-4.1 illustrates the architecture of the modified SQR T-CSLA model for 4-bit addition. Similarly, it can be extended this for 16-bit addition. This architecture consists of HSG, CG unit, CS unit and FSM unit. Also, the number of gates is reduced in the proposed design through common Boolean logic expressions. This indicated in CG unit of proposed SQR T CSLA. The common expressions $ab+c$ are used



for both CG units and CS unit. Hence, the proposed SQR CSLA offers less area and delay when compared to conventional BEC based SQR CSLA.

Similar modifications are made on 2-bit, 3-bit, 5-bit SQR CSLA and combining all these we get 16-bit SQR CSLA structure. Four sets of binary addition process are done concurrently, with help above mentioned modifications. Due to this, the proposed architecture is called as modified SQR-CSLA, and the entire functionality is shown in Figure-4.2.

4.5 REDUCED COMPLEXITY WALLACE MULTIPLIER USING MODIFIED SQR CSLA

Reduced complexity Wallace multiplier is a parallel multiplier in which complexity of multiplication process is reduced. To generate the partial product of reduced complexity Wallace multiplier, N^2 AND gates are used and they are arranged in a triangular position. The procedure for generating the partial products is same as existing reduced complexity Wallace multiplier. The method of partial product generation of reduced complexity multiplier is shown in Figure.4.3.

Further to add the partial generation output, efficient adder structure is essential in final stage of reduced complexity Wallace multiplier. Hence in this proposed work, the designed modified SQR CSLA is used for addition process of reduced complex multiplier structure. From the simulation-based experiment, it is identified that the efficiency of the proposed Wallace multiplier is better regarding power, delay and area utilization. It is better than the existing systems. Hence, it is concluded that the proposed MAC unit design is absolutely suitable for digital FIR filter.



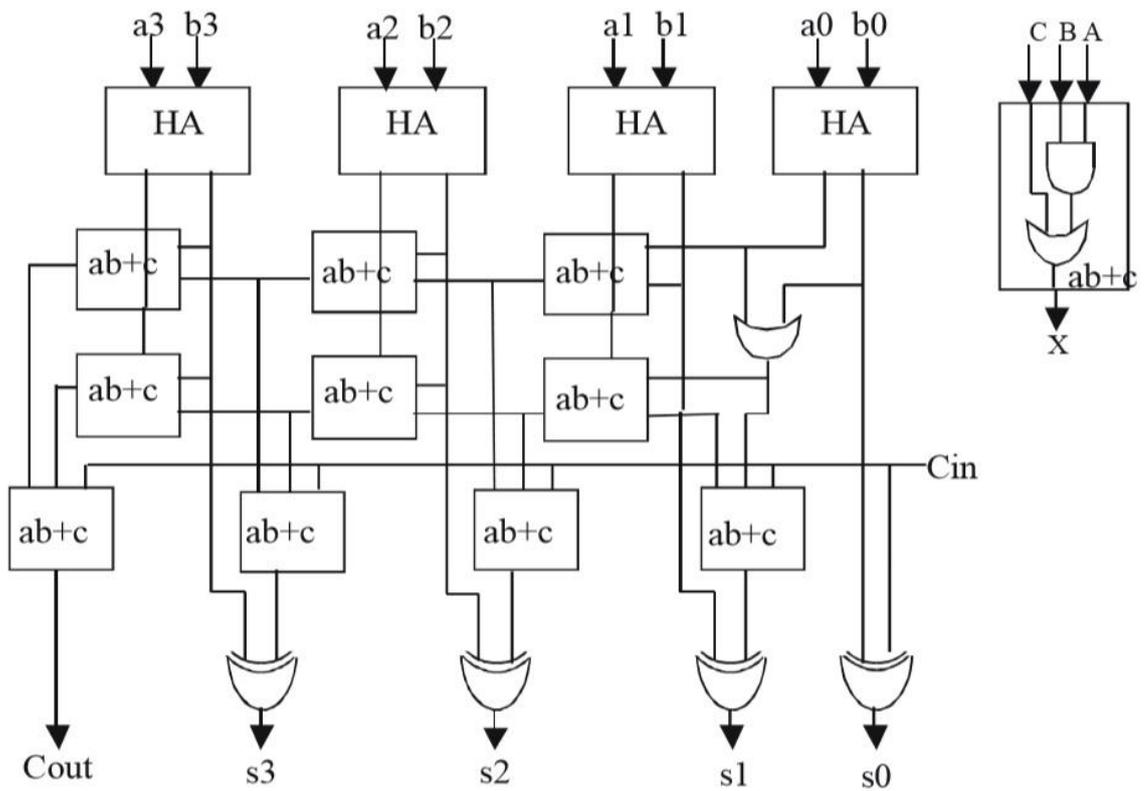


Figure 4.1 Architecture of modified Sqrt CSLA

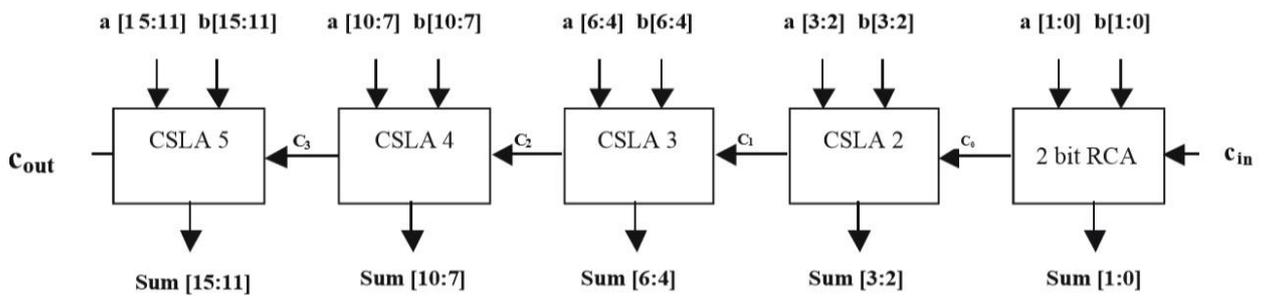


Figure 4.2 Block diagram of 16-bit modified Sqrt CSLA

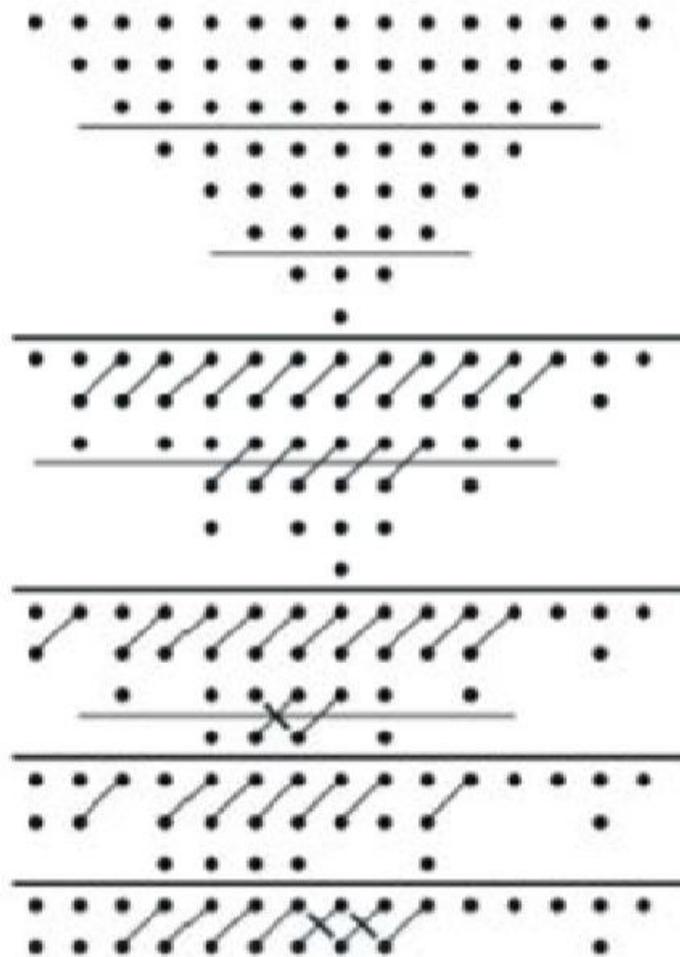


Figure 4.3 Partial products generation of reduced complexity Wallace multiplier

4.6 PROPOSED DIRECT FORM FIR FILTER

Figure-4.4 illustrates the structure of the direct form of FIR filter. It comprises of adders, multipliers and delay units in performing digital filter operations. From Figure 4.4, it is clear that the performance of direct form FIR filter is mostly depends on MAC unit. The low power or area schemes are developed for FIR filter in previous endeavours. To further improve the performance of digital FIR filter, proposed MAC (reduced complexity Wallace multiplier with help of modified SQRT CSLA) unit is incorporated into direct form FIR filter. When comparing direct form FIR filter using existing reduced complexity Wallace multiplier, the proposed direct form FIR

filter using modified SQR T CSLA based reduced complexity Wallace multiplier provides better results. In order to attain high spectral suppression and/or noise reduction, digital FIR filters with moderately large number of taps are essential as shown in Figure.4.4.

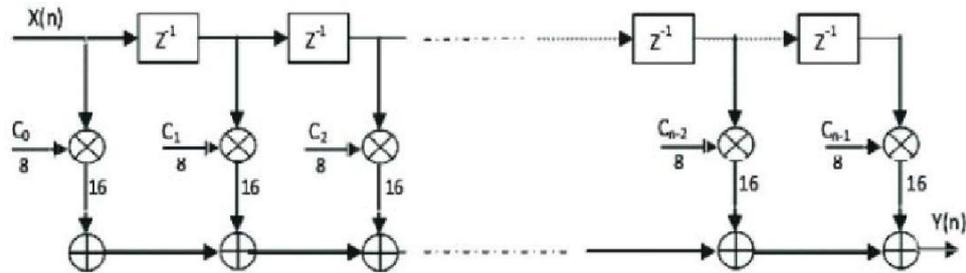


Figure 4.4 Direct Form of FIR Filter

4.7 RESULTS AND DISCUSSIONS

The main objective of this stage work is to increase the performance of MAC for digital FIR filter. In this paper, efficient MAC unit is designed with help of reduced complexity Wallace multiplier and modified SQR T CSLA. The design of both reduced complexity Wallace multiplier and modified SQR T CSLA is done by using Verilog Hardware Description Language (Verilog HDL). From simulation and synthesis tools, the results for MAC unit is analyzed and compared. Synthesis results for MAC unit are analyzed as follows:

Synthesis results for both BEC based SQR T CSLA and modified SQR T CSLA is analyzed and compared as shown in Table-4.1. From obtained results, it shows that the modified SQR T CSLA consumes less area and delay when compared to BEC based SQR T CSLA. These performances are graphically represented in Figure.4.5. From graphical representation, it is clear

that the modified SQR T CSLA offers 29.26% reduction in area and 4.23% reduction in delay when compared to BEC based SQR T CSLA.

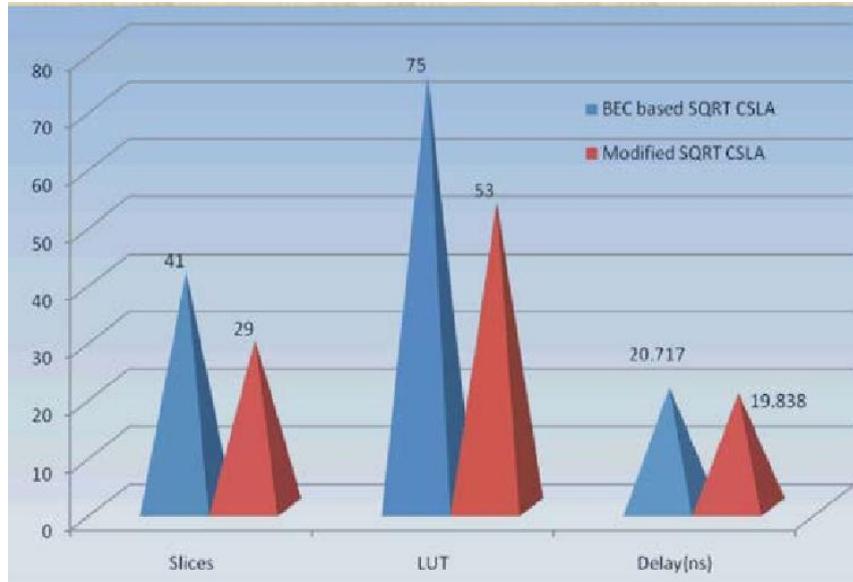


Figure 4.5 Performance of both BEC based SQR T CSLA and modified SQR T CSLA

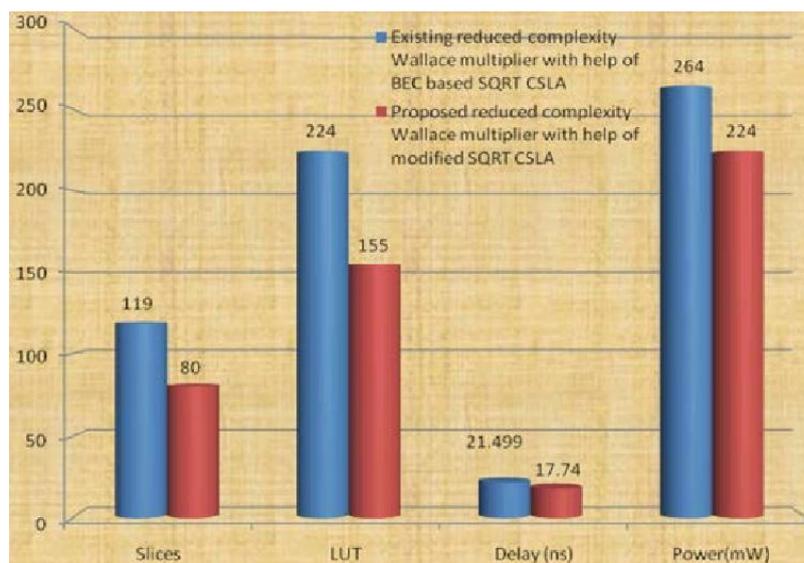


Figure 4.6 Performance of existing and proposed reduced complexity Wallace multiplier

Table 4.1 Comparison of BEC based Sqrt CSLA and modified Sqrt CSLA

Type	Slices	LUT	Delay(ns)
BEC based Sqrt CSLA	41	75	20.717
Modified Sqrt CSLA	29	53	19.839

Table 4.2 Reduced complexity Wallace multiplier and Modified Sqrt-CSLA

Type	Slices	LUT	Delay(ns)	Power (mW)
Existing reduced complexity Wallace multiplier with help of BEC based Sqrt CSLA	119	224	21.499	264
Proposed reduced complexity Wallace multiplier with help of modified Sqrt CSLA	80	155	17.74	224

The modified Sqrt CSLA is applied to reduced complexity Wallace multiplier for multiplication process. Then the results for both reduced complexity Wallace multiplier with help of BEC based Sqrt CSLA and modified Sqrt CSLA is analyzed and compared in Table-4.2. It shows that, proposed Wallace multiplier using BEC based Sqrt-CSLA bargains 32.73% reduction in area, 17.48% reduction in delay and 15.15% reduction in power when compared to existing reduced Wallace multiplier with help of BEC based Sqrt-CSLA. These performances are graphically represented in Figure-4.6.

In addition to that, the proposed WM is integrated with the DFFIR filter for increasing the performance of the FIR filter functions. Therefore, the proposed MAC unit is absolutely suitable for digital signal processing applications and wireless communication applications.



4.8 SUMMARY

FIR filter is the most significant part in communication systems and digital signal processing in mobile applications. To improve the efficiency of the FIR filter focused on the configuration of multipliers and adders in FIR filter architectures. Using efficient multiplier and adder circuits for an optimized area, power, delay and increase in speed in digital signal processing (DSP), this problem is considered and this research work focused on designing a direct-form Finite Impulse Response (FIR) digital filter. High speed and area efficient MAC unit is designed with help of reduced complexity Wallace multiplier and modified SQR T CSLA for digital FIR filter. This modified SQR T CSLA is incorporated into reduced complexity Wallace multiplier to improve the performance of digital multiplication process. The proposed reduced complexity Wallace multiplier offers 32.73% reduction in area, 17.48% reduction in delay and 15.15% reduction in power when compared to existing reduced complexity Wallace multiplier. Further the proposed reduced complexity Wallace multiplier is incorporated into digital FIR filter to improve the digital filtering performance.



CHAPTER 5

INCORPORATION OF REDUCED FULL ADDER AND HALF ADDER INTO WALLACE MULTIPLIER AND IMPROVED CARRY- SAVE ADDER FOR DIGITAL FIR FILTER

5.1 OBJECTIVES

In the second stage of research work, the design of direct form FIR filter with efficient MAC unit has been presented to reduce the area, delay and power utilization. Finite impulse response digital filter is the most important component in communication systems and applications of digital signal processing. Multiplication and Accumulation (MAC) unit of Finite Impulse Response (FIR) filter has been designed using efficient multiplier and adder circuits for optimized APT (Area, Power and Timing) product. Initially, full adder and half adder structures are shrunk down by reducing number of gates. These compact full adders and half adder structures are incorporated into Wallace Multiplier and Improved Carry-Save Adder. The proposed 16-bit Carry-Save Adder has been improved by splitting into four parallel phases. Consequently, the delay of enhanced Carry-Save Adder is reduced. Generation of carry output is performed using number of OR gates in a sequential manner. All these enhanced architectures are incorporated into the Digital FIR Filter to reduce the area, delay and power utilization.



5.2 PROBLEM STATEMENT

If the digital FIR filters provide limited power and area, then it is extensively used in several portable applications. The two fundamental FIR structures used for a linear phase FIR filter are transposed form and direct form. Direct form digital FIR filter is used for DSP applications in this research work. Multiplier-Accumulator (MAC) unit of FIR filter is the most important element. The efficiency of the MAC unit is affected by full adder. Full adder circuit power reduction is necessary for low power application. The heart of the processor is Arithmetic & Logic Unit (ALU). It contains elements for reckoning operations. It plays a very important role in computation time of the processor. Multiplication operation is more recurrent in Digital Signal Processing (DSP) application. Sinking delay in the multiplier shrinks the overall computation time. One of the fast multipliers is available such as Wallace multiplier. It works due to speeding up the addition process. Carry Propagating Adder has been used to sum the final two rows. A direct implementation needs a $(2N - 2)$ bit Carry Propagating Adder (CPA), where N is the number of bits of operands. Carry Propagating Adder obtains long time when the carry is required to get promulgated until the last adder. In this work, a fast carry-save adder is implemented at the last stage to obtain superior performance.

Modified Carry-Save Adder consumes more delay and area due to propagation delay and sequential process. Hence Improved Carry-Save Adder (ICSA) is designed in this work with parallel processing and without carry propagation delay. Our ICSA adder offers less area and higher speed than all other schemes. Regular Wallace and reduced Wallace Multipliers are designed using different high-speed adders. But it consumes more area, power and less delay. So compact full adder, half adder and ICSA adder are incorporated into Wallace to improve the efficiency of our multiplier. Several



previous endeavours for reducing area, delay and power consumption of digital FIR filter usually focus on the optimization of the filter coefficient while the filter order is fixed. FIR filter structures are simplified to, minimizing the number of additions/subtractions & Add and Shift operations which is the main focus of those approaches. However, one of the drawbacks encountered in those approaches is that once the filter architecture is determined, the coefficients cannot be altered. Consequently, those schemes are not appropriate to the FIR filter with programmable coefficients. Reconfigurable FIR filter with modified Amplitude Detector (AD) and control logic is introduced to reduce the area and power utilization. But it makes performance degradation. Previously described works have been focused on reducing the power consumption and improving the configuration of filter coefficients. However, all those architectures have more complexity, because of using traditional hardware structures to perform multiplication and accumulation functions. In order to reduce the hardware complexity of MAC unit, redundant logical functions are identified with the help of Boolean expressions. It is identified that half adder and full adder are used in every digital signal processing operation like MAC and ALU. Hence, the redundant Boolean logical expressions of half adder and full adder are identified to optimize the digital signal processing operations. So, our proposed Direct FIR filter offers optimum area, delay and power compared with the all other filter techniques also without any degradation. Because Enhanced Wallace Multiplier with Improved Carry-Save adder is incorporated into proposed FIR filter.

5.3 REDUCED FULL ADDER AND HALF ADDER STRUCTURE

Half adder and Full adder are the main building block of every adder and multipliers unit. Hence the design of efficient half adder and full



adder is performed to reduce the number of gates in order to achieve less area, delay and power utilization. Structure of reduced half adder is given in Figure-5.1(A) which reduces one AND gate and one INVERTER compared to existing full adder structure. Structure of reduced full adder is shown in Figure-5.1(B) which reduces one AND gate and one OR gate compared to conventional full adder. This compact full adder and half adder can be used in various adder and multiplier to achieve less area, delay and power consumption.

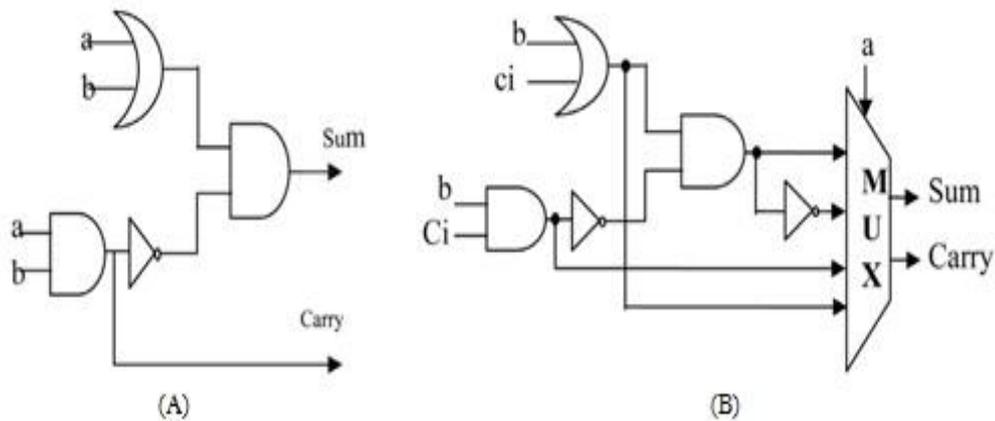


Figure 5.1 Structures of reduced Half Adder and Full Adder

5.1(A) Reduced Half Adder, 5.1(B) Reduced Full Adder

Reduced Half Adder structure is simplified by use of Demorgan's theorem and some Boolean logic. General expression to find the sum of half adder is given in Equation (5.1)

$$\begin{aligned}
 \text{Sum} &= A\bar{B} + \bar{B}A \\
 &= (A+B) \cdot (\bar{A} + \bar{B}) \cdot (A + \bar{A}) \cdot (B + \bar{B}) \\
 &= (A+B)(\bar{A} + \bar{B})
 \end{aligned}
 \tag{5.1}$$

Equations (5.2) and (5.3) are simplified Sum and Carry Expression for reduced half adder. Similarly, Full adder is shrinking down by introducing Boolean logic and Demorgan's Law.

$$Sum = (A + B) \cdot \overline{AB} \quad (5.2)$$

$$Carry = A \cdot B \quad (5.3)$$

Simplified expression of Sum and Carry of compact Full Adder are given in Equation (5.4) and Equation (5.5) which is derived as below.

$$Sum = \sum_{i=0}^1 X_i \cdot \overline{A_i} + \overline{A_i} \cdot A_i \quad (5.4)$$

where

$$\begin{aligned} X &= (B + C) \cdot \overline{BC} = B\overline{C} + C\overline{B} = B \oplus C \\ \overline{X} &= \overline{(B + C) \cdot \overline{BC}} = \overline{(B\overline{C} + C\overline{B})} = \overline{B\overline{C}} \cdot \overline{C\overline{B}} = \overline{B}C + B\overline{C} = B \oplus C \\ Carry &= \sum_{i=0}^1 BC_i + (B + C)A_i \end{aligned} \quad (5.5)$$

5.4 IMPROVED 16-BIT CARRY-SAVE ADDER

Conventional 16-bit Carry-Save Adder has been designed in the sequence manner. Hence the propagation delay of this adder is high. It has 15-full adders and 17-half adders. As the ripple carry adder is used in the last phases, this architecture yields maximum carry propagation delay. To minimize this delay, the last stage of CSA is separated into five sets. After splitting into 5 stages, chip size (area) and power utilization are maximum in the existing CSA. Consequently, this structure is split into four stages and parallel processing is performed in order to achieve less delay, area and power than the existing CSA. Construction of Improved Carry-Save Adder (ICSA) is shown in Figure-5.2. Enhanced 16-bit Carry-Save Adder consists of number of half adder, OR gate, 5-bit BEC & 2:1 MUX. The divided four groups of ICSA are listed below.



- 1). {c0, s[3:0]}
- 2). {c1, x[7:4]}
- 3). {c2, x[11:8]}
- 4). {c3, x[12:15]}

The 1st group of output s[3:0] are straightforwardly assigned as the final output; the 2nd group {c1,x[7:4]} controls the fractional result by allowing for c1 is 0; the 3rd group {c2,x[11:8]} influences the partial result through thinking c2 is 0; the 4th group {c3,x[12:15]} maneuvers the FPGA Implementation of partial result by considering c3 is 0.

Improved Carry-Save Adder is designed by using the below Equations (6) to (10) which are obtained from the Figure-5.2.

$$s_0 = a_0 \oplus b_0 \quad (5.6)$$

$$s_1 = x_0 \oplus c_0 \quad (5.7)$$

$$s_2 = x_1 \oplus c_1 \oplus (x_0 \cdot c_0) \quad (5.8)$$

$$s = x \oplus c \oplus x \cdot c \oplus x \oplus c \cdot c \quad (5.9)$$

$$s = c + (x \cdot c) + (x \oplus c) \cdot xc + (x \oplus c \oplus (x \cdot c)) \cdot (x \oplus c) \cdot (x \cdot c) \quad (5.10)$$

where



$$\begin{aligned}
 x_0 &= a_1 \oplus b_1 & c_1 &= a_1 \cdot b_1 \\
 x_1 &= a_2 \oplus b_2 & c_2 &= x_0 \cdot c_0 \\
 x_2 &= a_3 \oplus b_3 & c_3 &= a_2 \cdot b_2 \\
 c_0 &= a_0 \cdot b_0 & c_4 &= x_1 \cdot c_1 \\
 c_5 &= (x_1 \oplus c_1) \cdot c_2 & c_7 &= x_2 \cdot c_3 \\
 c_6 &= a_3 \cdot b_3 & c_8 &= (x_2 \oplus c_3) \cdot c_4 \\
 c_9 &= (x_2 \oplus c_3 \oplus c_4) \cdot c_5
 \end{aligned}$$

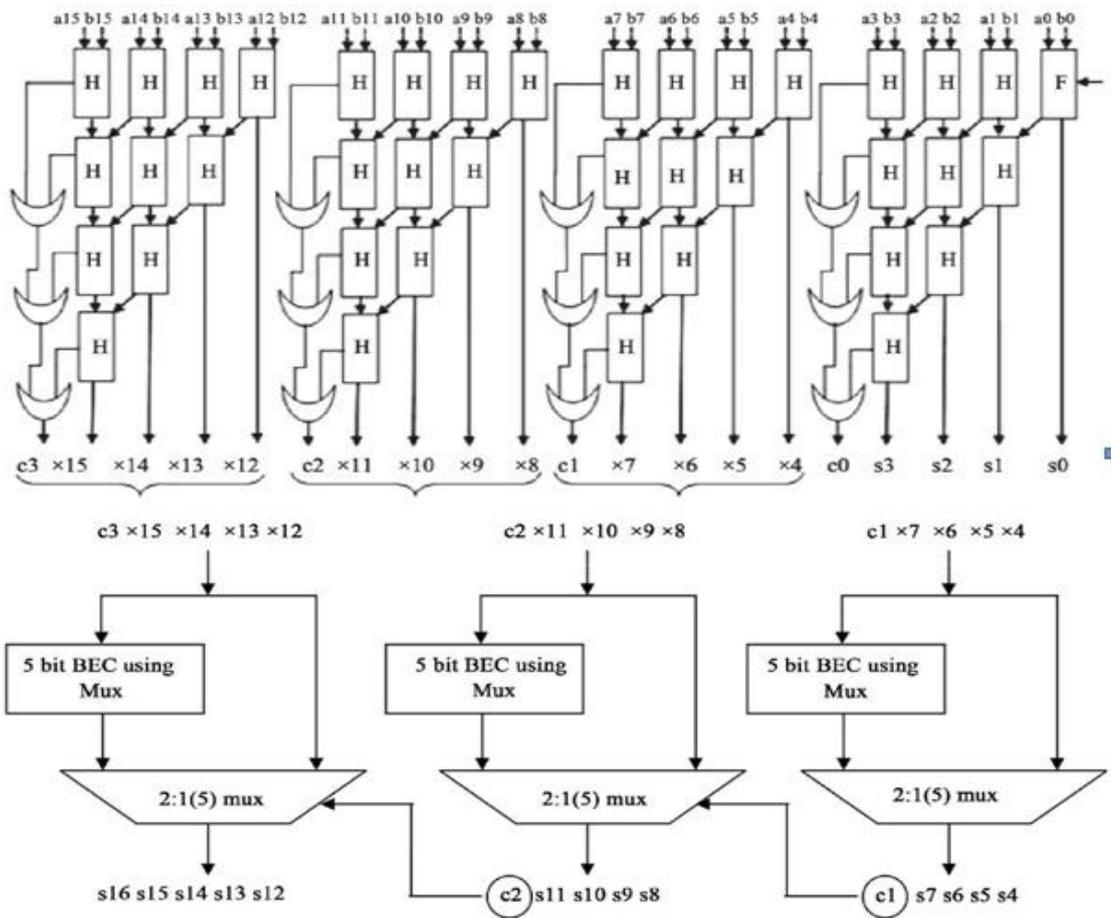


Figure 5.2 Architecture of enhanced 16-bit carry-save adder using modified 5-bit BEC structure and parallel processing

Depending on c₀ of the 1st group, the 2nd group mux provides the last result without the carry propagation de-ly from c₁ to c₂; depending on c₂ of the second group final result, the 3rd group MUX offers the final result without the carry propagation delay from c₂ to s₁₆. The major advantage of this logic is that every group calculates the limited results in

parallel and the MUXES are prepared to provide the last result without any delay of the mux. Once the C_{in} of every group enters, the last result will be finding instantaneously. Modified 5-bit BEC structure is shown in Figure-5.3 which consists of four modified XOR gate structures are connected in sequential order.

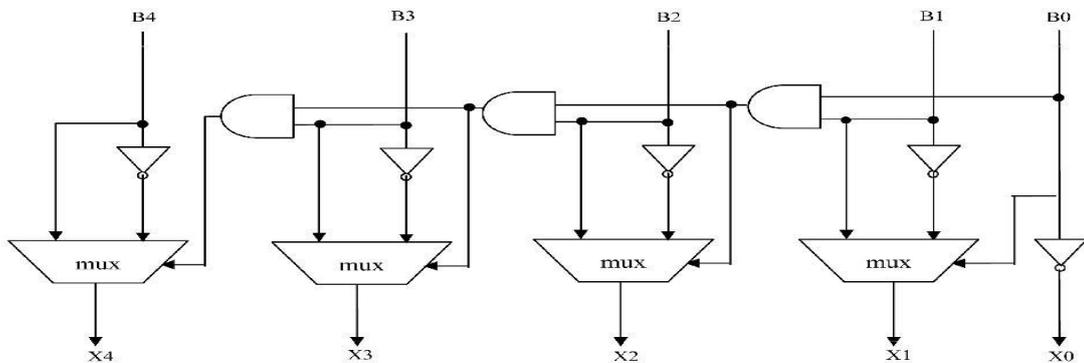


Figure 5.3 Structure of modified 5-bit binary to excess one code (BEC) converter

5.5 ENHANCED WALLACE MULTIPLIER

In this work, the design of Enhanced Wallace Multiplier with improved Carry-Save Adder is performed to evaluate best APT (Area, delay and timing) reduction. Proposed Wallace multiplier is designed by introducing the compact full adder, half adder and improved carry-save adder structures. Hence the proposed Wallace multiplier provides less area, delay and power than the existing Wallace multiplier techniques.

The adapted version of Wallace multiplier is called as Enhanced Wallace multiplier. It contains a less amount of half adders when compared to the regular Wallace multiplier is shown in Figure-5.4. Partial products are created through N^2 AND gates and they are located in an inverted triangle manner, which is separated into three row clusters in the modified Wallace reduction method.

- 1) Group of three bits are summed by applying a full adder.
- 2) Single bit and a group of 2 bits are stimulated to the next stage straightforwardly.

The Improved Carry-Save Adder (ICSA) with modified 5-bit BEC is incorporated in the final stage with the aim of low area, power and delay utilization. Enhanced Wallace Multiplier with Improved Carry-Save Adder (ICSLA) provides less area, delay and power than all other schemes which is confirmed by the results that follow. The enhanced Wallace Multiplier is applied in Digital FIR filter to analyze the efficiency of proposed methods. MAC unit of Digital FIR filter is vital for coefficient multiplication and addition. These efficient adders and Multipliers are integrated into MAC unit of the proposed Direct FORM FIR filter. The proposed FIR filter with Wallace Multiplier and Improved Carry-Save Adder (ICSA) is better for optimized APT product.

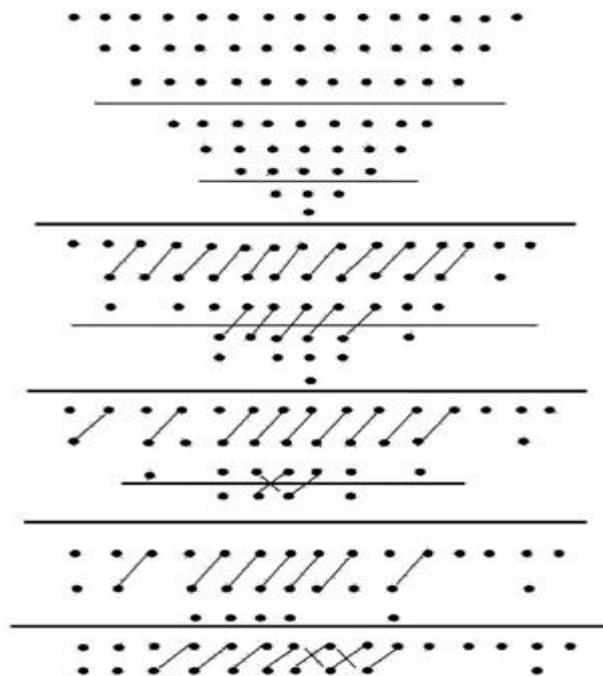


Figure 5.4 Reduced complexity Wallace multiplier

5.6 PROPOSED DIRECT FORM DIGITAL FIR FILTERS

FIR filter circuit must be able to drive at high sample rates, whereas in extra applications, the FIR filter architecture must be a low-power circuit operating at moderate sample rates. The low-power or low-area schemes developed particularly for digital filters. In order to further increase the effective throughput, decrease the power utilization and area of the original filter. Parallel processing can be applied to digital FIR filters. Direct Form Digital FIR filter is shown in Figure-5.5 which consists of delay unit, adder and multiplier units in the sequential manner.

In this paper, the design of Enhanced Wallace Multiplier with Improved Carry-Save Adder is presented. This effective multiplier is applied in Direct Form FIR Filter structure to analyze the Area, Power and Timing product. Proposed Direct Form FIR filter with enhanced Wallace Multiplier provides less area, power and delay than regular Direct Form FIR Filter.

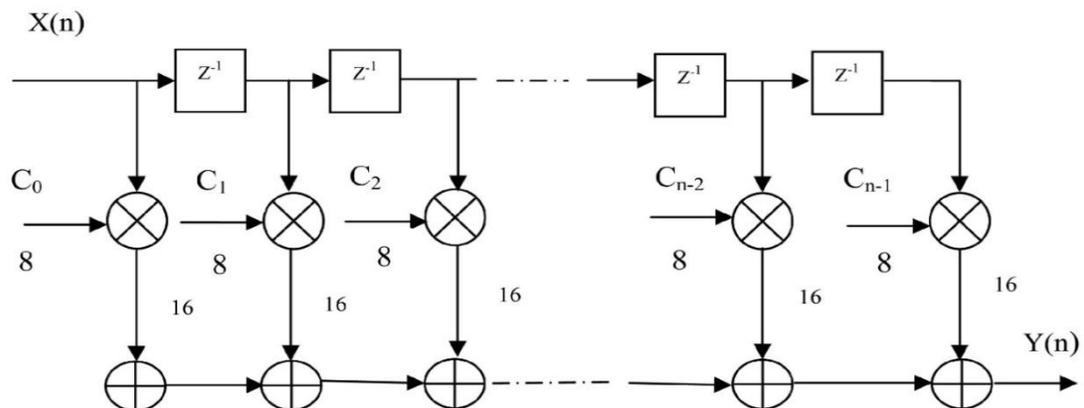


Figure 5.5 General structure of direct form digital FIR filter

5.7 RESULTS AND DISCUSSION

The aim of enhanced Wallace tree multiplier with Improved Carry-Save Adder (ICSA) is analyzed using Verilog and implemented in FPGA Spartan 3 XC3S50 using the Xilinx ISE 10.1i EDA (Electronic Design Automation) tool. Comparison between Conventional Carry-Save Adder and Improved Carry-Save Adder is performed to analyze the APT product as shown in Table-5.1. From the results, Improved Carry-Save Adder offers 25% area reduction and 15% delay reduction compared to conventional Carry-Save Adder.

Total equivalent LUT in case of enhanced Wallace multiplier with CSA is 162, which is improved to 152 using Improved Carry-Save Adder based Wallace Multiplier. The power consumption in case of enhanced Wallace multiplier with CSA is 264 mW, which is improved to 252 mW using ICSA based Wallace multiplier. The number of occupied slices used in enhanced Wallace multiplier with ICSA is also reduced. In case of reduced Wallace multiplier with Carry-Save Adder it is 87 and in enhanced Wallace multiplier with ICSA it is 79. Enhanced Wallace multiplier results are tabulated as shown in Table-5.2.

From the outcomes, Proposed Direct Form FIR Filter with Enhanced Wallace Multiplier provides 50% area reduction and 12% power reduction compared to conventional Direct Form FIR Filter and frequency utilization of proposed FIR Filter is improved up to 38%. Simulation result of proposed digital FIR filter is validated by using ModelSim 6.3C design tool. Simulation result of proposed digital FIR filter is shown in Figure-5.6. Table-5.3 shows the Comparison between proposed direct form FIR filter and conventional direct form FIR filter.



As shown in Figure-5.6, data input (x_{in}) is given as 8'd50 (00110010). The finite impulses are determined with the help of proposed MAC unit. Three constant filter coefficients are considered in the current research work such as 8'd5, 8'd2 and 8'd3 respectively. Hence, output is generated as 8'd250 ($8'd50 * 8'd5$), 8'd350 ($(8'd50 * 8'd5) + (8'd50 * 8'd2)$) and 8'd500 ($(8'd50 * 8'd5) + (8'd50 * 8'd2) + (8'd50 * 8'd3)$) respectively, which are shown in Figure-5.6. Similarly, for other combination of inputs, output parameters are validated.

Table 5.1 Comparison between conventional CSA and improved CSA

Parameters	Conventional Carry-Save Adder (CSA)	Improved Carry-Save Adder (ICSA)
Delay (ns)	23.854	20.655
Slices	40	32
LUT	71	57

Table 5.2 Comparison of conventional Wallace multiplier and modified Wallace multiplier

Parameters	Reduced Wallace Multiplier	Modified Wallace Multiplier
Slices	87	79
LUT	162	152
Delay (ns)	21.867	18.718
Power (mW)	264	252



with conventional full adder and half adder. These reduced adders are applied in the Wallace Multiplier to analyze the performance. After generating the partial product, Improved Carry-Save Adder (ICSA) with modified 5-bit BEC is applied to further reduce the area and delay. Enhanced Wallace Multiplier with Improved Carry-Save Adder is incorporated into Direct Form Digital FIR filter to examine the performance. Proposed Direct Form FIR filter offers less area, power and higher speed compared with conventional Direct Form FIR filter. This filter can be used in wireless communication techniques, signal processing and image processing mechanisms.



CHAPTER 6

PERFORMANCE EVALUATION

6.1 OBJECTIVES

This chapter presents the performance evaluation of the thesis by comparing the experimental results of the proposed approaches discussed in different stages. It is well known that one of the most important key factors in signal processing is FIR filter. The main objective of the research work is obtained by describing a novel design of FIR filter, which is efficient in terms of area, power, complexity and throughput.

6.2 SQRT CSLA BASED FIR FILTER

In the first stage of the research work, a modified SQRT CSLA is designed by incorporating with reduced complexity Wallace Multiplier. A Multiplication and accumulation unit-based FIR filter is used for increasing the speed and throughput whereas it is used for signal processing applications. To do that, a high speed and area efficient MAC unit is designed with help of reduced complexity Wallace multiplier. Then it is modified SQRT CSLA for digital FIR filter. Conventional BEC based SQRT CSLA is re-designed to reduce the chip size and delay for addition process. This modified SQRT CSLA is incorporated into reduced complexity Wallace multiplier to improve the performance of digital multiplication process.



The design of both reduced complexity Wallace multiplier and modified SQR T CSLA is done by using Verilog Hardware Description Language (Verilog HDL). From the simulation, the results for MAC unit is analysed and compared. Synthesis results for both BEC based SQR T CSLA and modified SQR T CSLA is analysed and compared is shown in Figure-6.1.

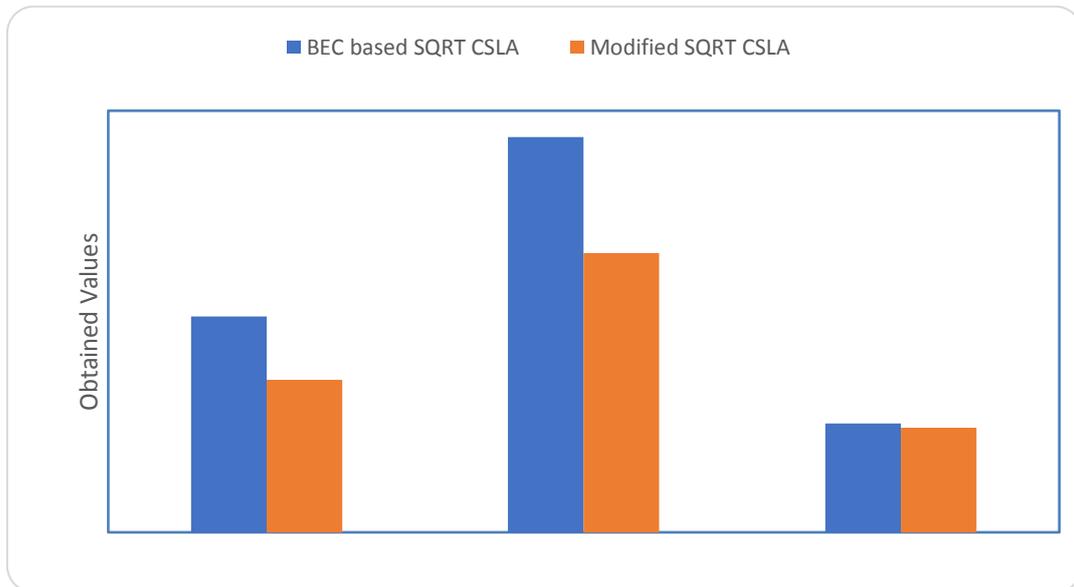


Figure 6.1 Comparison of BEC based SQR T CSLA and modified SQR T CSLA

From obtained results, it noticed that the modified SQR T CSLA consumes less area and delay when compared to BEC based SQR T CSLA. These performances are graphically represented in Figure-6.2. From graphical representation, it is clear that the modified SQR T CSLA offers 29.26% reduction in area and 4.23% reduction in delay when compared to BEC based SQR T CSLA.

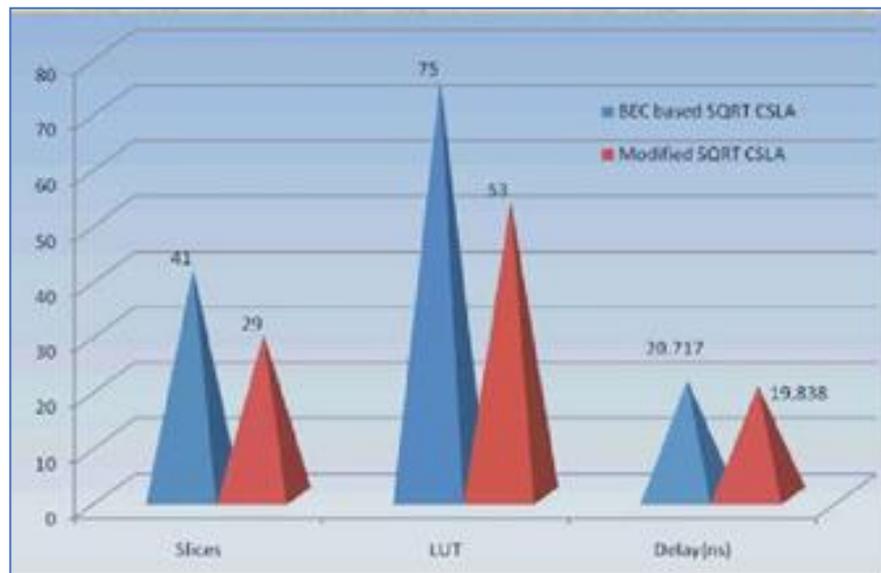


Figure 6.2 Performance of both BEC based Sqrt CSLA and modified Sqrt CSLA

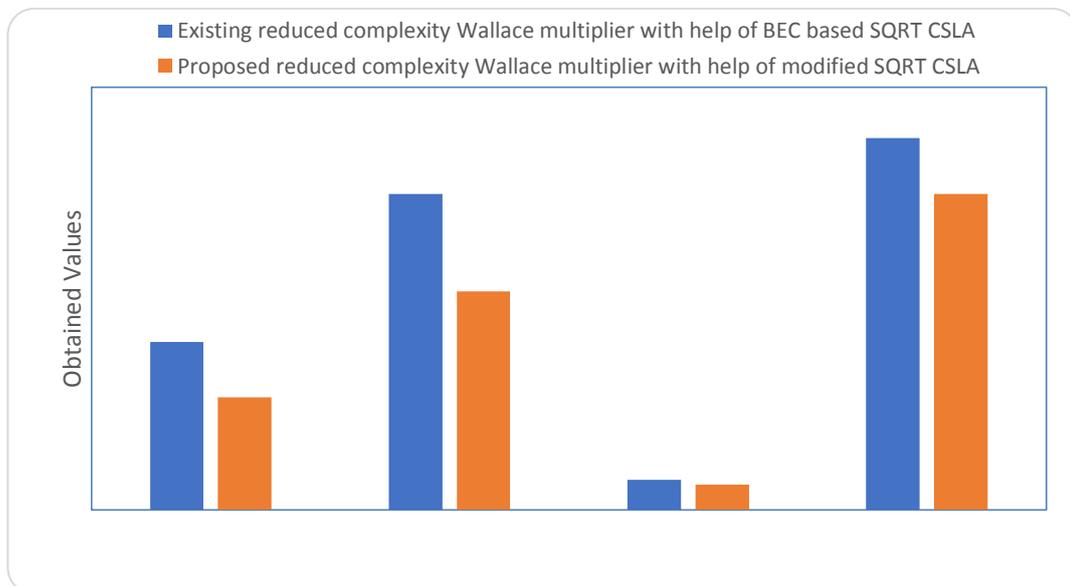


Figure 6.3 Comparison of both reduced complexity Wallace multiplier with help of BEC based Sqrt CSLA and modified Sqrt CSLA

The modified Sqrt CSLA is applied to reduced complexity Wallace multiplier for multiplication process. Then the results for both reduced complexity Wallace multiplier with help of BEC based Sqrt CSLA and modified Sqrt CSLA is analyzed and compared in Figure-6.3.

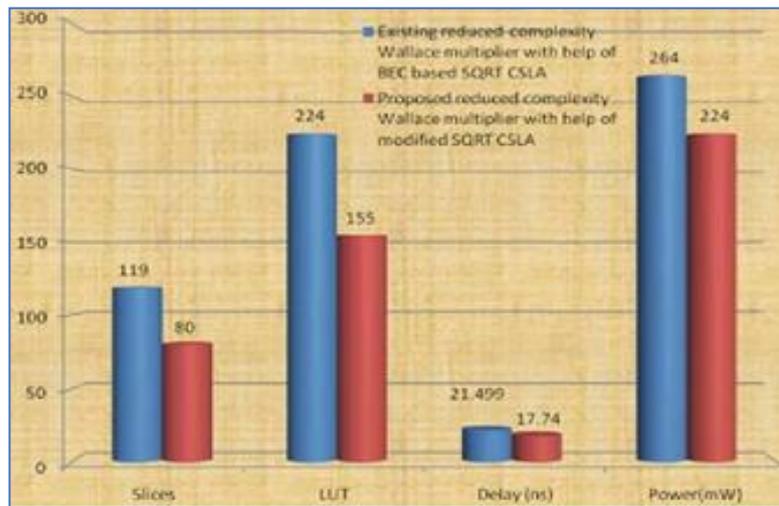


Figure 6.4 Performance of existing and proposed reduced complexity Wallace multiplier

Figure-6.4 shows that, proposed reduced complexity Wallace multiplier with help of BEC based SQR T CSLA offers 32.73% reduction in area, 17.48% reduction in delay and 15.15% reduction in power when compared to existing reduced Wallace multiplier with help of BEC based SQR T CSLA. From the results it is clear that the proposed FIR design is more efficient than the existing designs in terms of area, delay and power.

6.3 PROPOSED DIRECT FORM OF FIR FILTER

Similarly, the second stage of the research work is design and implemented. In the second stage of the research work, the design of direct form FIR filter with efficient MAC unit has been presented. A full adder and half adder structures are shrunk down by reducing number of gates and the structures are incorporated into Wallace Multiplier and Improved Carry-Save Adder. The proposed 16-bit Carry-Save Adder has been improved by splitting into four parallel phases. Consequently, the delay of enhanced Carry-Save Adder is reduced. Generation of carry output is performed using number of OR gates in a sequential manner. All these enhanced architectures are

incorporated into the Digital FIR Filter to reduce the area, delay and power utilization.

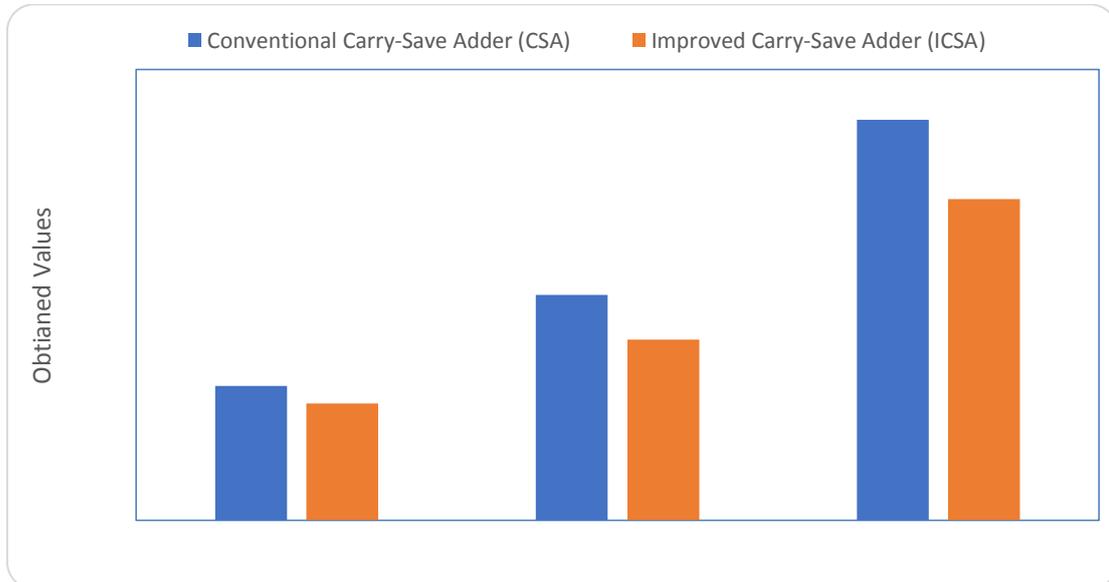


Figure 6.5 Comparison between conventional CSA and improved CSA

In order to further increase the effective throughput, decrease the power utilization and area of the original filter. Parallel processing can be applied to digital FIR filters initially, the design of Enhanced Wallace Multiplier with Improved Carry-Save Adder is presented. This effective multiplier is applied in Direct Form FIR Filter structure to analyse the Area, Power and Timing product. Proposed Direct Form FIR filter with enhanced Wallace Multiplier provides less area, power and delay than regular Direct Form FIR Filter.

The second stage of this research work is analyzed using Verilog and implemented in FPGA Spartan 3 XC3S50 using the Xilinx ISE 10.1i EDA (Electronic Design Automation) tool. Comparison between Conventional Carry-Save Adder and Improved Carry-Save Adder is performed to analyse the APT product as shown in **Figure-6.5**. From the results, Improved Carry-

Save Adder offers 25% area reduction and 15% delay reduction compared to conventional Carry-Save Adder.

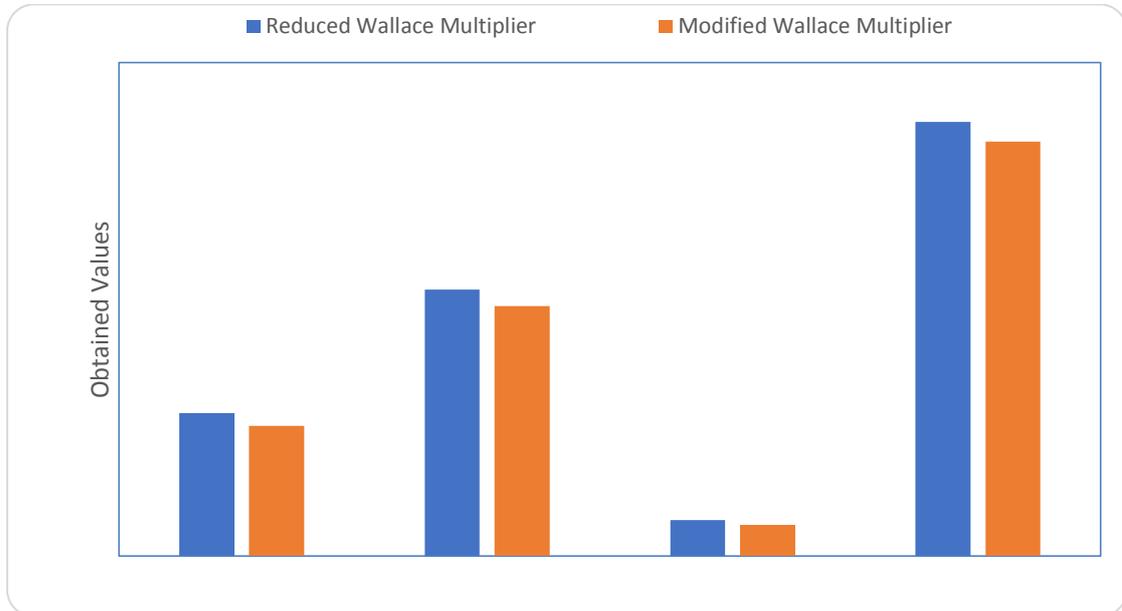


Figure 6.6 Comparison of conventional Wallace multiplier and modified Wallace multiplier

Total equivalent LUT in case of enhanced Wallace multiplier with CSA is 162, which is improved to 152 using Improved Carry-Save Adder based Wallace Multiplier. The power consumption in case of enhanced Wallace multiplier with CSA is 264 mW, which is improved to 252mW using ICOSA based Wallace multiplier. The number of occupied slices used in enhanced Wallace multiplier with ICOSA is also reduced. In case of reduced Wallace multiplier with Carry-Save Adder it is 87 and in enhanced Wallace multiplier with ICOSA it is 79. Enhanced Wallace multiplier results are tabulated as shown in **Figure-6.6**.

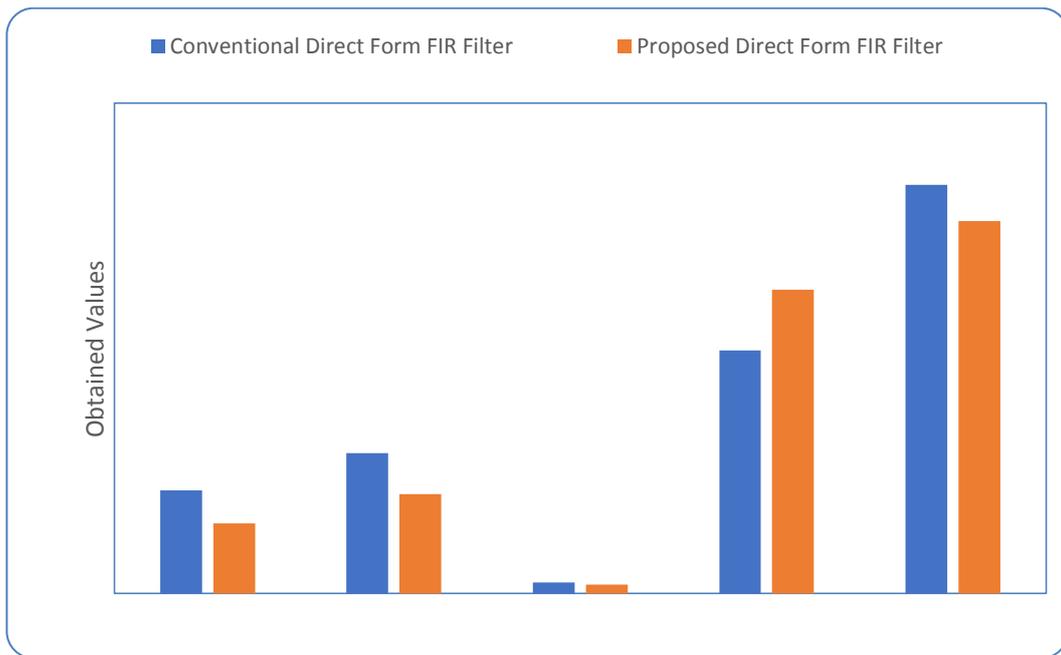


Figure 6.7 Comparison between proposed direct form FIR filter and conventional direct form FIR filter

From the outcomes, proposed Direct Form FIR Filter with Enhanced Wallace Multiplier provides 50% area reduction and 12% power reduction compared to conventional Direct Form FIR Filter and frequency utilization of proposed FIR Filter is improved up to 38%. Simulation result of proposed digital FIR filter is validated by using ModelSim-6.3C design tool. Simulation result of proposed digital FIR filter is shown in **Figure 6.7**. **Figure-6.8** shows the Comparison between proposed direct form FIR filter and conventional direct form FIR filter.

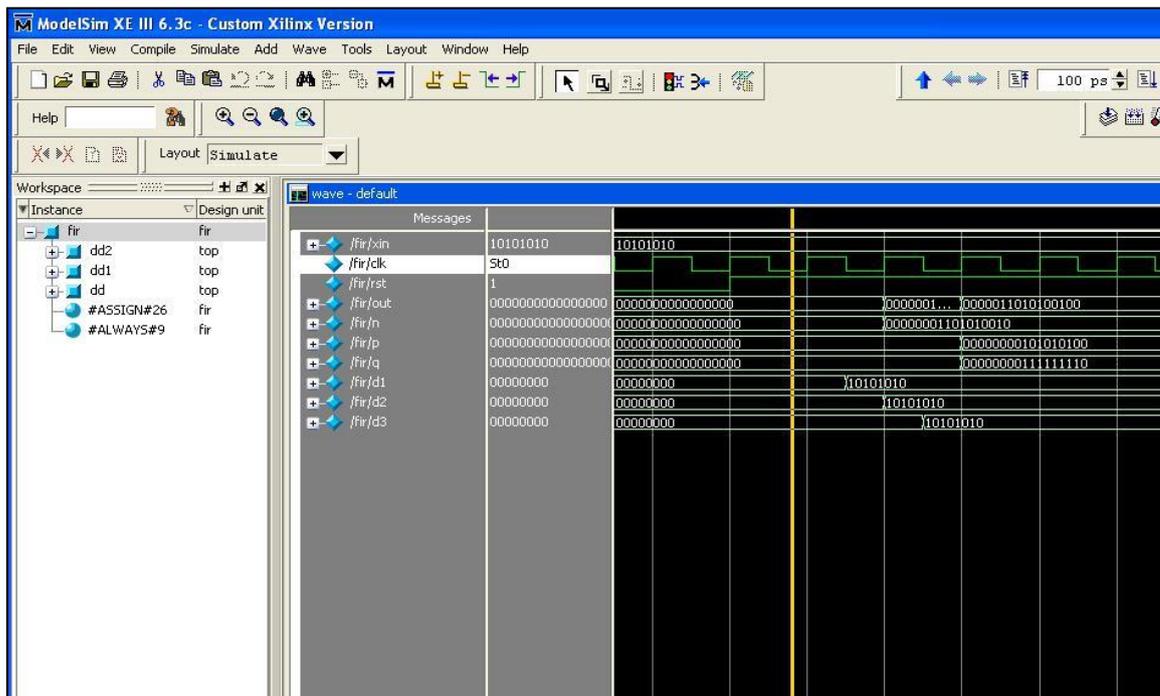


Figure 6.8 Simulation result of proposed digital FIR filter

As shown in **Figure 6.8**, data input (x_{in}) is given as 8'd50 (00110010). The finite impulses are determined with the help of proposed MAC unit. Three constant filter coefficients are considered in the current research work such as 8'd5, 8'd2 and 8'd3 respectively. Hence, output is generated as 8'd250 ($8'd50 * 8'd5$), 8'd350 ($(8'd50 * 8'd5) + (8'd50 * 8'd2)$) and 8'd500 ($(8'd50 * 8'd5) + (8'd50 * 8'd2) + (8'd50 * 8'd3)$) respectively, which are shown in **Figure 6.8**. Similarly, for other combination of inputs, output parameters are validated.

6.4 SUMMARY

This chapter describes the efficiency of the proposed FIR filter design in two different stages. Both the designs are simulated and experimented in MATLAB software and the results are verified. From the obtained results, the important factors such as delay, power and frequency are verified. From the comparison, it is noticed and obtained that the proposed

Direct Form Digital FIR Filters outperforms than the other FIR filters in terms of delay, LUT, power and frequency.



CHAPTER 7

CONCLUSION AND FUTURE WORK

7.1 CONCLUSION

The main objective of this research work is to design FIR filter architecture for emerging application under DSP domain. To do that the entire research work is carried out into two different stages such as: (i). Efficient MAC unit of digital FIR filter is designed to increase the speed and throughput of digital FIR filter. (ii). Design and implement the direct form FIR filter by incorporating reduced full adder and half adder into Wallace Multiplier and improved Carry- Save adder for digital FIR filter. Both the stages are experimented and the results are verified.

In the first stage, high speed and area efficient MAC unit is designed with help of reduced complexity Wallace multiplier and modified SQRD CSLA for digital FIR filter. Conventional BEC based SQRD CSLA is re-designed in this paper to reduce the chip size and delay for addition process. This modified SQRD CSLA is incorporated into reduced complexity Wallace multiplier to improve the performance of digital multiplication process. The proposed reduced complexity Wallace multiplier offers 32.73% reduction in area, 17.48% reduction in delay and 15.15% reduction in power when compared to existing reduced complexity Wallace multiplier. Further the proposed reduced complexity Wallace multiplier is incorporated into digital



FIR filter to improve the digital filtering performance. In future, the proposed MAC based digital filter will be useful to implementation of parallel FIR filter for wireless standard communication, signal and image processing applications.

In the second stage, high-speed and area-efficient Reduced Full Adder, Half Adder, Improved Carry-Save Adder (ICSA) and modified 5-bit BEC (Binary to Excess one code Converter) using mux are presented. Reduced full adder and half adder are designed using a smaller number of gates compared with conventional full adder and half adder. These reduced adders are applied in the Wallace Multiplier to analyze the performance. After generating the partial product, Improved Carry-Save Adder (ICSA) with modified 5-bit BEC is applied to further reduce the area and delay. Enhanced Wallace Multiplier with Improved Carry-Save Adder is incorporated into Direct Form Digital FIR filter to examine the performance. Proposed Direct Form FIR filter offers less area, power and higher speed compared with conventional Direct Form FIR filter. This filter can be used in wireless communication techniques, signal processing and image processing mechanisms.

Both the stages of the research work are the proposed approaches are experimented and the results are verified. From the experimental results, it is noticed that the enhanced Wallace Multiplier with Improved Carry-Save Adder is incorporated into Direct Form Digital FIR filter to examine the performance. Proposed Direct Form FIR filter offers less area, power and higher speed compared with conventional Direct Form FIR filter. This filter can be used in wireless communication techniques, signal processing and image processing mechanisms. From the experimental results, it is noticed that the proposed design in the initial stage reduced complexity Wallace multiplier offers 32.73% reduction in area, 17.48% reduction in delay and 15.15%



reduction in power when compared to existing reduced complexity Wallace multiplier.

7.2 FUTURE WORK

Further the proposed reduced complexity Wallace multiplier is incorporated into digital FIR filter to improve the digital filtering performance. In future, the proposed MAC based digital filter will be useful to implementation of parallel FIR filter for wireless standard communication, signal and image processing applications.



REFERENCES

1. Aksoy, L., Lazzari, C., Costa, E., Flores, P., & Monteiro, J. (2013). Design of digit-serial FIR filters: Algorithms, architectures, and a CAD tool. *IEEE transactions on very large-Scale integration (VLSI) systems*, 21(3), 498-511.
2. Ali, Md Raju, and Jobbin Abraham Ben, (2014), "Design of Parallel Linear Phase FIR Digital Filter of Odd Length based on Fast FIR Algorithm", *International Journal* (2014).
3. Ambika, R., & Ranjani, S. S. (2014). Design of Fir Filter Using Area and Power Efficient Truncated Multiplier. *International Journal of Engineering Sciences & Research Technology (IJESRT)*.
4. Andamuthu, A., & Rithanyaa, S. (2012). Design of 128-bit low power and area efficient carry select adder. *International Journal of Advanced Research in Engineering (IJARE) Vol, 1*, 31-34.
5. Anju, S., & Saravanan, M. (2013). High Performance Dadda Multiplier Implementation Using High Speed Carry Select Adder. *International Journal of Advanced Research in Computer and Communication Engineering*, 2(3).
6. Balasubramaniam, S., and R. Bharathi, (2012), "Performance Analysis of Parallel FIR Digital Filter using VHDL", *International Journal of Computer Applications* 39, no.9, February 2012.
7. Bedrij, O. J. (1962). Carry-select adder. *IRE Transactions on Electronic Computers*, (3), 340-346.
8. Bharti, D., & Anusudha, K. (2013). High Speed FIR Filter Based on Truncated Multiplier and Parallel Adder. *International Journal of Engineering Trends and Technology (IJETT)*–Volume, 5.
9. C. Cheng and K. K. Parhi, (2004), "Hardware efficient fast parallel FIR filter structures based on iterated short convolution", *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 51, no. 8, pp. 1492-1500, Aug. 2004.
10. C. Cheng and K. K. Parhi, (2005), "Further complexity reduction of parallel FIR filters", in *Proc. IEEE In. Symp. Circuits Syst. (ISCAS 2005)*, Kobe, Japan, May 2005.
11. Chu, T. A. (1987). *Synthesis of self-timed VLSI circuits from graph-theoretic specifications* (Doctoral dissertation, Massachusetts Institute of Technology).



12. D. A. Parker and K. K. Parhi, (1996), "Area-efficient parallel FIR digital filter implementations," In Proceedings International Conference-Application-Specific Systems, Architectures and Processors, pp. 93-111.
13. Dash, S. P., Rath, A., Pattnaik, G., Das, S., & Dash, A. (2014). Analysis and design of a low phase noise, low power, wideband CMOS voltage-controlled ring oscillator in 90 nm process. *IJSETR*, 3(5), 1264-1268.
14. Dhillon, H. S., & Mitra, A. (2008). A reduced-bit multiplication algorithm for digital arithmetic. *International Journal of Computational and Mathematical Sciences*, 2(2).
15. G.Thanuja, Mr. P.Ashok, Dr.V.S.R Kumari, (2016), "Implementation of Novel Distribute Arithmetic Based Reconfigurable FIR Digital Filter", *IJSRD*, Vol. 4, No. 7, PP. 813-818.
16. Gahlan, N. K., Shukla, P., & Kaur, J. (2012). Implementation of Wallace tree multiplier using compressor. *International journal on Computer Technology & Applications*, 3, 1194-1199.
17. Gnanasekaran, M. and Manikandan, M. (2014) Performance of FIR Filter with Wallace Multiplier over FIR Filters with Truncated Multiplier. *International Journal of Computer Science and Engineering Communications*, 2, 450-455.
18. Gnanasekaran, M., Manikandan, M., & St Peter's University, A. U. (2014). Performance of FIR Filter with Wallace Multiplier over FIR filter with truncated multiplier. *IJCSEC-International Journal of Computer Science and Engineering Communications*, 2(3).
19. He, Y., Chang, C. H., & Gu, J. (2005, May). An area efficient 64-bit square root carry-select adder for low power applications. In *Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on* (pp. 4082-4085). IEEE.
20. Hemalatha, A., & Shanmugam, A. (2011). Computer Aided Design for Low Power Fir Processor on System On-Chip Platform Architecture for High Performance DSP Applications [J]. *International journal of computer science and network security: IJCSNS*, 11(7), 38-42.
21. Hsiao, S. F., Jian, J. H. Z., & Chen, M. C. (2013). Low-cost FIR filter designs based on faithfully rounded truncated multiple constant multiplication/accumulation. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 60(5), 287-291.



22. J.I. Acha, (1989), "Computational structures for fast implementation of L-path and L-block digital filters", *IEEE Trans. Circuit Syst.*, vol. 36, no. 6, pp.805-812, Jun. 1989.
23. K.K.Parhi, (1999), "VLSI Digital Signal Processing systems", Design and implementation New York: Wiley.
24. Kannan, N., Seshadri, R. and Ramakrishnan, S. (2014) Improved Wallace Tree Multiplier Based Direct Fir Structure Using MCM Technique. Proceedings of ICSEM'14-2nd International Conference on Science, Engineering and Man-agement, March 2014.
25. Kashyap, S. and Maheshwari, M. (2014) Implementation of High Performance Fir Filter Using Low Power Multiplier and Adder. *International Journal of Engineering Research and Applications*, 4, 177-181.
26. Kashyap, S., & Maheshwari, M. (2014). Implementation of High-Performance Fir Filter Using Low Power Multiplier and Adder. *Research Scholar, Department of Electronics and Communication Jaipur National University, Jaipur, Rajasthan, India.*
27. Kharate, A.B. and Gumble, P.R. (2013) VLSI Design and Implementation of Low Power MAC for Digital FIR Filter. *International Journal of Electronics Communication and Computer Engineering*, 4, REACT-2013.
28. Kim, S., & Cho, K. (2010). Design of high-speed modified booth multipliers operating at GHz ranges. *World academy of science, Engineering and Technology*, 61, 1-4.
29. Kim, Y., & Kim, L. S. (2001). 64-bit carry-select adder with reduced area. *Electronics Letters*, 37(10), 614-615.
30. Kumar, A., & Raman, A. (2010, February). Low power ALU design by ancient mathematics. In *Computer and Automation Engineering (ICCAE), 2010 The 2nd International Conference on* (Vol. 5, pp. 862-865). IEEE.
31. M.Gnanasekaran, Dr. M. Manikandan, (2014), "Performance of FIR Filter with Wallace Multiplier over FIR filter with Truncated Multiplier", *IJCSEC International Journal of Computer Science and Engineering Communications*, Vol.2 Issue.3, May 2014. ISSN: 2347-8586.
32. Macpherson, K. N., & Stewart, R. W. (2006). Area efficient FIR filters for high speed FPGA implementation. *IEE Proceedings-Vision, Image and Signal Processing*, 153(6), 711-720.



33. Manikandan, M.S. , Dandapat, S. (2006), “Wavelet threshold-based ECG compression using USZZQ and Huffman coding of DSM”, *Biomedical Signal Processing and Control*. 1, 4, 261-270.
34. Mirzaei, S., Hosangadi, A., & Kastner, R. (2006, October). FPGA Implementation of High-Speed FIR Filters Using Add and Shift Method. In *ICCD* (pp. 308-313).
35. P Kiran Mojesh, N Rajesh Babu, (2017), “FPGA Implementation of Serial and Parallel FIR Filters by using Vedic and Wallace tree Multiplier”, *IJRSET*, Vol. 6, No. 3, PP. 4337-4342.
36. Pham, P. H., Song, J., Park, J., & Kim, C. (2013). Design and implementation of an on-chip permutation network for multiprocessor system-on-chip. *IEEE transactions on very large scale integration (VLSI) systems*, 21(1), 173-177.
37. Ramkumar, B., & Kittur, H. M. (2013). Faster and energy-efficient signed multipliers. *VLSI Design*, 2013, 13.
38. Rao, M. J., & Dubey, S. (2012, December). A high speed and area efficient Booth recoded Wallace tree multiplier for Fast Arithmetic Circuits. In *Microelectronics and Electronics (PrimeAsia), 2012 Asia Pacific Conference on Postgraduate Research in* (pp. 220-223). IEEE.
39. Sandhya Pridhini, Jeena Maria Cherian, Diana Alosius, (2014), “Efficient FIR filter design using Wallace tree compression”, *International Journal of Science, Engineering and Technology Research (IJSETR)*, Volume 3, Issue 4, April 2014.
40. Sankar, D. R., & Ali, S. A. (2013). Design of Wallace tree multiplier by Sklansky adder. *Int. J. Eng. Res. Appl*, 3(1), 1036-1040.
41. Sejal D. Patel and M.C. Patel, (2017), “Research Trends in Area Optimized FIR filter Implementation on FPGA”, *IJSETR*, Vol. 6, No. 3, PP. 293-296.
42. Shrividhya M. Pothuri, Prachi Palsodkar, (2015), “Area-reduced Parallel FIR Digital Filter Structures Based on Modified Winograd Algorithm”, *IEEE ICCSP- 2015*, PP. 588 -591.
43. Sreenivasulu, P. (2012). Krishna veni, Dr. K. Srinivasa Rao and Dr. A. VinayaBabu, “Low Power Design Techniques Of Cmos Digital Circuits” *International journal of Electronics and Communication Engineering & Technology (IJECET)*, 199-208.



44. Srinivasan, S., Bhudiya, K., Ramanarayanan, R., Babu, P. S., Jacob, T., Mathew, S. K., ... & Errgauntla, V. (2013, April). Split-path fused floating point multiply accumulate (FPMAC). In Computer Arithmetic (ARITH), 2013 21st IEEE Symposium on (pp. 17-24). IEEE.
45. Srinivasan, S., *et al.* (2013) Split-Path Fused Floating Point Multiply Accumulate (FPMAC). IEEE 21st Symposium on Computer Arithmetic, 7-10 April 2013, 17-24.
46. T. N. Priyatharshne, L. Raja,, and A. Vinodhini, "An Optimized Wallace Tree Multiplier using Parallel Prefix Han-Carlson Adder for DSP Processors" International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE), Vol. 3, Issue. 11, pp. 1700-1704, 2014.
47. TARUMI, K., HYODO, A., MUROYAMA, M., & YASUURA, H. (2004), "A design method for a low power digital FIR filter in digital wireless communication systems", Graduate School of Information Science & Electrical Engineering, Kyushu University.
48. Thakur, R., & Khare, K. (2013). High speed FPGA implementation of FIR filter for DSP applications. International Journal of Modeling and Optimization, 3(1), 92-94.
49. Tian, Jingjing, Guangjun Li, and Qiang Li, (2013), "Hardware-efficient parallel structures for linear-phase FIR digital filter", In Circuits and Systems (MWSCAS), 2013 IEEE 56th International Midwest Symposium on, pp. 995-998. IEEE, 2013.
50. Tiwari, H. D., Gankhuyag, G., Kim, C. M., & Cho, Y. B. (2008, November). Multiplier design based on ancient Indian Vedic Mathematics. In SoC Design Conference, 2008. ISOC'08. International (Vol. 2, pp. II-65). IEEE.
51. V. S. kanchana Bhaaskaran, (2013), "Modified Carry Select Adder using Binary Adder as a BEC-1", European Journal of Scientific Research, Vol.103, No.1, pp.156-164.
52. Waters, R. S., & Swartzlander, E. E. (2010), "A reduced complexity Wallace multiplier reduction", IEEE transactions on Computers, 59(8), 1134-1137.
53. Yu-Chi Tsao and Ken Choi, (2012), "Area-Efficient VLSI Implementation for parallel Linear-Phase FIR digital filters of odd length based on Fast FIR algorithm", Circuits and Systems II: Express Briefs, IEEE Trans. on page(s): 371-375 vol.59, Issue: 6, June 2012.



LIST OF PUBLICATIONS

International Journals

1. **Chinnapparaj, S** & Dr.D.Somasundareswari, 2016, 'Incorporation of Reduced Full Adder and Half Adder into Wallace Multiplier and Improved Carry Save Adder for Digital FIR Filter', Circuits and Systems, vol. 7, no. 9, pp. 2467-2475. **Annexure 1 IF : 0.33.**

