

**EFFICIENT TRANSMITTER/RECEIVER ARCHITECTURES
FOR HIGH PERFORMANCE WIRELESS APPLICATIONS**

A thesis submitted by

SHAHANA T. K.

for the award of the degree of

DOCTOR OF PHILOSOPHY

Under the guidance of

Dr. K. POULOSE JACOB and Dr. SREELA SASI

**DEPARTMENT OF COMPUTER SCIENCE
FACULTY OF TECHNOLOGY
COCHIN UNIVERSITY OF SCIENCE AND TECHNOLOGY
KOCHI 682 022, INDIA**

OCTOBER 2008

EFFICIENT TRANSMITTER/RECEIVER ARCHITECTURES FOR HIGH PERFORMANCE WIRELESS APPLICATIONS

Ph. D. Thesis in the field of Wireless Communication

Author

Shahana T.K.

Research Scholar

Department of Computer Science

Cochin University of Science and Technology

Kochi – 682 022, Kerala, India

E-mail: shahanatk@cusat.ac.in

Research Advisors

Dr. K. Poullose Jacob

Professor and Head

Department of Computer Science

Cochin University of Science and Technology

Kochi – 682 022, Kerala, India

E-mail: kpj@cusat.ac.in

Dr. Sreela Sasi

Associate Professor

Department of Computer & Information Science

Gannon University

PA, USA

E-mail: sasi001@gannon.edu

October 2008

CERTIFICATE

This is to certify that this thesis entitled “EFFICIENT TRANSMITTER/RECEIVER ARCHITECTURES FOR HIGH PERFORMANCE WIRELESS APPLICATIONS” is a bonafide record of the research work carried out by Ms. Shahana T. K. under my supervision and guidance in the Department of Computer Science, Cochin University of Science and Technology with Dr. Sreela Sasi, Associate Professor, Gannon University, PA, USA as co-guide. The results presented in this thesis or parts of it have not been presented for the award of any other degree.

30th October 2008



Dr. K. POULOSE JACOB

(Supervising Guide)

Professor and Head

Department of Computer Science

Cochin University of Science and Technology

Kochi-682 022, Kerala.

CERTIFICATE

This is to certify that this thesis entitled “EFFICIENT TRANSMITTER/RECEIVER ARCHITECTURES FOR HIGH PERFORMANCE WIRELESS APPLICATIONS” is a bonafide record of the research work carried out by Ms. Shahana T. K. under the supervision and guidance of Dr. K. Poulose Jacob, Professor and Head, Department of Computer Science, Cochin University of Science and Technology with myself as co-guide. The results presented in this thesis or parts of it have not been presented for the award of any other degree.

30th October 2008

A handwritten signature in black ink that reads "Sreela Sasi". The signature is written in a cursive style and is underlined with a single horizontal line.

Dr. SREELA SASI

Associate Professor

Department of Computer & Information Science

Gannon University, PA, USA.

DECLARATION

I hereby declare that the work presented in this thesis entitled “EFFICIENT TRANSMITTER/RECEIVER ARCHITECTURES FOR HIGH PERFORMANCE WIRELESS APPLICATIONS” is based on the original research work carried out by me under the supervision and guidance of Dr. K. Poullose Jacob, Professor and Head, Department of Computer Science, Cochin University of Science and Technology with Dr. Sreela Sasi, Associate Professor, Department of Computer and Information Science, Gannon University, PA, USA as co-guide. The results presented in this thesis or parts of it have not been presented for the award of any other degree.

Kochi - 682 022

30th October 2008



SHAHANA T. K.

ACKNOWLEDGEMENT

I am deeply indebted and grateful to many people who supported me during the research work and preparation of the thesis.

First and foremost, I give special thanks and glory to the God Almighty for giving me the grace, wisdom and health to complete this endeavour.

I would like to express sincere gratitude and appreciation to my supervising guide Dr. K. Poulouse Jacob, Professor and Head, Department of Computer Science, Cochin University of Science and Technology for his constant encouragement, support and guidance. His sincerity, positive and supportive attitude, calmness and scholarly advice have been a steady source of inspiration to me.

My deepest gratitude and respect also goes to Dr. Sreela Sasi for her guidance and assistance as co-supervisor. Her creative comments and suggestions from the initial conception to the end of this work are highly appreciated. I am greatly indebted to her for financial assistance which enabled me to register for several international conferences abroad, and also for endless hours of help for reviewing the research papers and the thesis.

I am highly indebted and grateful to Dr. Jimson Mathew, Department of Computer Science, University of Bristol, UK, for his tremendous support and encouragement. Special thanks to him for his patient listening to my questions, giving necessary reference materials and constructive comments which helped a lot to keep me on track.

I am thankful to Dr. R. Gopikakumari, Division Head, and all my colleagues at the Division of Electronics and Communication Engineering, School of Engineering, Cochin University of Science and Technology for their encouragement and support.

I am very grateful to Anju Pradeep, Babita R Jose, Mridula S., Rekha K. James and Sheena Mathew for their friendship, cooperation, support and care which helped me to endure this program.

I appreciate the technical and non-technical staff of the Department of Computer Science, Cochin University of Science and Technology for their timely help.

I owe heartfelt thanks to my parents and my mother-in-law for their motivation, encouragement and understanding when it was mostly required. Also, I remember my sisters with gratitude for their affectionate support.

A special mention to my husband Zakir, and my children Tasneem and Hasna for their love, understanding and support.

SHAHANA T. K.

ABSTRACT

Wireless communication has become an integral part of modern society with an ultimate aim of global roaming. The developments in satellite transmission, radio and television broadcasting, and the new generation mobile systems have revolutionized global communication. The recent trend in wireless communication is to implement a single transceiver hardware platform that can support multiple communication standards. The progress in CMOS technology opens opportunities to implement flexible platforms at low cost and low power. In this research, analysis has been done on architectures and design techniques that enhances integration and programmability of RF transceivers for wireless communication; and compact, inexpensive, low power communication devices that are robust, testable and capable of handling multiple standards have been developed.

The thesis focuses on the development of reconfigurable architectures and high performance building blocks suitable for wireless communication. For efficient multi-mode operation, advanced analog-to-digital converter (ADC) solutions are required with flexibility to adapt to the bandwidth and speed of the high data rate standards. Sigma-delta ADC is the most promising solution to achieve high resolution over a wide variety of bandwidth requirements. The most complex part of a sigma-delta ADC is the digital decimation filter. So, efficient, high speed and reconfigurable implementation of decimation filter is a key to achieve high performance. The main objectives of the research are to:

- Reduce the complexity of decimation filter design by developing a toolbox

- Design Residue Number System (RNS) based dual-mode decimation filters for high speed, small die area and low power operation
- Reduce the complexity of RNS conversion circuitry
- Improve the performance of the communication system through an efficient concatenated coding scheme
- Realize easily testable circuits using Reed-Muller (RM) logic

The major achievements in this research are the following:

- A decimation filter design toolbox is developed in MALAB[®] Graphical User Interface Development Environment (GUIDE) which enables the user to perform quick filter design and analysis for six popular wireless standards.
- A reconfigurable RNS based decimation filter is designed and implemented for WCDMA/WiMAX and WCDMA/WLAN dual-mode operation. The performance analysis shows that it has high speed, less area and low power dissipation compared to the implementation in traditional binary number system.
- A direct analog-to-residue converter based on sigma-delta ADC is designed to reduce the complexity of RNS conversion circuitry. It has high resolution, high conversion speed, medium hardware complexity and low cost of implementation compared to the existing Nyquist rate converters.
- An Orthogonal Frequency Division Multiplexing (OFDM) based communication system is simulated using a novel Redundant Residue Number System (RRNS) – Convolutional concatenated coding

scheme. This coding scheme offers significant improvements in bit error rate (BER) under different operating conditions.

- Easily testable circuit realization for multiply and accumulate units of the filter is achieved by implementing in RM form. New algorithms for combinational logic synthesis in RM form are also developed using exhaustive branching and genetic algorithm based approaches.

The performance evaluations show that these efficient design methods and reconfigurable implementations in RNS domain yield high speed operation in addition to reduction in area and power consumption. Also, the direct analog-to-residue conversion technique reduced the overall hardware complexity. The RRNS-convolutional concatenated coding designed for forward error correction in an OFDM communication system achieved significant BER improvement. Hence these design techniques and circuits are dependable alternatives that could be used for high performance wireless applications.

CONTENTS

	Page No.
LIST OF FIGURES	v
LIST OF TABLES	ix
LIST OF ABBREVIATIONS	xi
Chapter 1	
INTRODUCTION	1
1.1 History of Wireless Communication	3
1.2 Wireless System Developments	5
1.3 Conventional RF Receiver Architectures	5
1.3.1 Superheterodyne Receiver	6
1.3.2 Direct Conversion Homodyne Receiver	7
1.3.3 Low IF Receiver	8
1.3.4 Wideband IF Double Conversion Receiver	8
1.4 Analog-to-Digital Converters	9
1.4.1 Nyquist Rate Analog-to-Digital Converters	10
1.4.2 Oversampling Analog-to-Digital Converters	10
1.4.2.1 Sigma-Delta Analog-to-Digital Converters	12
1.5 Layout of the Thesis	14
Chapter 2	
DECIMATION FILTER DESIGN: A TOOLBOX APPROACH	17
2.1 Decimation Filter Design Considerations	19
2.2 Receiver Architecture for Multi-standard operation	21
2.2.1 Reconfigurable Sigma-Delta ADC	22
2.3 Multistage Decimation Filter	24
2.3.1 Cascaded Integrator Comb Filter	26
2.3.2 Halfband Filter	28
2.3.3 FIR Filter	29
2.4 Decimation Filter Design Specification	30
2.5 Multi-standard Decimation Filter Design Toolbox	32

2.6	Polyphase Implementation of Non-recursive Comb Decimators	40
2.6.1	Classical Recursive CIC Filter	42
2.6.2	Non-recursive CIC Filter	43
2.6.3	Polyphase Non-recursive CIC Architecture	44
2.7	Summary	47

Chapter 3

	RNS BASED PROGRAMMABLE MULTI-MODE DECIMATION FILTERS	49
3.1	Residue Number System	51
3.1.1	RNS Basics	51
3.1.2	RNS Arithmetic	52
3.1.3	Forward and Reverse Conversions	53
3.1.4	Choice of RNS Moduli	54
3.2	FIR Digital Filter Design: RNS Versus Traditional	55
3.2.1	FIR Filter Architecture	57
	3.2.1.1 Forward Converter	58
	3.2.1.2 Modulo Addition	59
	3.2.1.3 Modulo Multiplication	60
	3.2.1.4 Reverse Converter	61
3.3	RNS based Dual-mode Decimation Filters	63
3.3.1	Receiver Architecture	65
3.3.2	Dual-mode Decimation Filter for WCDMA/WiMAX	65
	3.3.2.1 Design Considerations	66
3.3.3	Dual-mode Decimation Filter for WCDMA/WLANa	69
	3.3.3.1 Design Considerations	70
3.4	RNS Multiplier using Index Calculus	72
3.5	Programmable Decimation Filter using Index Calculus Multipliers	74
	3.5.1 Design Considerations	75
3.6	Summary	78

Chapter 4

RRNS-CONVOLUTIONAL CONCATENATED CODED OFDM WIRELESS	81
COMMUNICATION SYSTEM WITH A DIRECT ANALOG-TO-RESIDUE CONVERTER	
4.1 Direct Analog-to-Residue Converters	83
4.1.1 Nyquist Rate Analog-to-Residue Converters	83
4.1.1.1 Multiple-Residue Flash A/R Converter	84
4.1.1.2 Successive Approximation based A/R Converter	86
4.1.1.3 Iterative Subranging Flash A/R Converter	87
4.1.2 A Novel Sigma-Delta based parallel Analog-to-Residue Converter	90
4.1.2.1 Sigma-Delta Modulator	90
4.1.2.2 RNS based Decimation Filter	92
4.2 RRNS-Convolutional encoded Concatenated Code for OFDM based wireless Communication	94
4.2.1 OFDM Communication System	95
4.2.2 Error Detection and Correction with RRNS	97
4.2.3 System Description	100
4.2.3.1 Transmitter Model	101
4.2.3.2 Receiver Model	102
4.3 Summary	105

Chapter 5

EASILY TESTABLE CIRCUITS FOR MAC UNITS	107
5.1 Reed-Muller Expressions	109
5.2 Easily Testable Circuits	112
5.3 Combinational Logic Synthesis using Reed-Muller Universal Logic Modules	113
5.3.1 N-ary Exhaustive Branching Technique	115
5.3.2 Exhaustive Branching Algorithm	117
5.4 Genetic Algorithm based Approach for Combinational Logic Synthesis	119
5.4.1 Universal Logic Modules for Logic Synthesis	120
5.4.2 GA based Approach for Logic Synthesis	121
5.5 Summary	124

Chapter 6

SIMULATION RESULTS AND ANALYSIS	127
6.1 Decimation Filter Implementation and Analysis	129
6.2 Simulation Results and Analysis of Polyphase Non-recursive Comb Decimation Filter	131
6.3 Performance Analysis of FIR Filter Implementation: RNS Versus Traditional	135
6.4 Simulation Results and Analysis of Dual-mode Decimation Filter for WCDMA/WIMAX	141
6.5 Simulation Results and Analysis of Dual-mode Decimation Filter for WCDMA/WLANa	144
6.6 Implementation of Programmable Decimation Filter using Index Calculus Multipliers	146
6.7 Simulation Results for Sigma-Delta based Parallel Analog-to-Residue Converter	150
6.8 Simulation Results and Performance Analysis of RRNS-Convolutional Concatenated Coding for OFDM System	154
6.8.1 Additive White Gaussian Noise Tolerance	154
6.8.2 Multipath Delay Spread Immunity	155
6.8.3 Effects of Frame Synchronization Errors	157
6.8.4 Peak Power Clipping for PAPR Reduction	158
6.9 Simulation Results for Easily Testable MAC Units	160
6.10 Combinational Logic Synthesis Results using Exhaustive Branching Algorithm	163
6.11 GA based Combinational Logic Synthesis Results using ULMs	167
6.12 Summary	172

Chapter 7

CONCLUSIONS AND SUGGESTIONS FOR FURTHER WORK	175
7.1 Conclusions	177
7.2 Suggestions for Further Work	179
REFERENCES	181
LIST OF PUBLICATIONS OF THE AUTHOR	195
INDEX	199

LIST OF FIGURES

Figure 1.1	Block diagram of a RF Transceiver	6
Figure 1.2	Superheterodyne receiver	7
Figure 1.3	Direct conversion homodyne receiver	8
Figure 1.4	Low IF receiver	8
Figure 1.5	Wideband IF double conversion receiver	9
Figure 1.6	Block diagram of analog-to-digital converter	9
Figure 1.7	Baseband quantization noise power	12
Figure 1.8	Block diagram of first order sigma-delta ADC	12
Figure 1.9	Noise shaping in sigma-delta modulators	14
Figure 2.1	Direct conversion homodyne receiver architecture	22
Figure 2.2	Multistage decimation filter	25
Figure 2.3	CIC decimation filter	27
Figure 2.4	CIC magnitude response for GSM with $F_s = 34.667\text{MHz}$, $R = 32$, $M = 1$ and $k = 3$	28
Figure 2.5	Magnitude response of halfband filter	29
Figure 2.6	Magnitude response of droop compensating FIR filter with $M = 1$ and $k = 4$	30
Figure 2.7	GUI for Multi-standard Decimation Filter Design Toolbox	33
Figure 2.8	Pop-up menu for standard selection	33
Figure 2.9	Decimation filter details for GSM	33
Figure 2.10	Cost of decimation filter implementation for GSM	34
Figure 2.11	Message box displaying filter coefficients for GSM	34
Figure 2.12	Pop-up menu for magnitude response selection	35
Figure 2.13	Individual Filter response for GSM displayed on front panel of GUI (a) CIC filter (b) Halfband filter (c) FIR filter	36
Figure 2.14	Display of the cascaded filter response	37
Figure 2.15	Pop-up menu for pole-zero plots	38
Figure 2.16	Pole-Zero plot of individual filters for GSM (a) CIC filter (b) Halfband filter (c) FIR filter	39
Figure 2.17	Non-recursive comb decimator	44
Figure 2.18	Polyphase realization of non-recursive comb decimator	45
Figure 2.19	Polyphase realization of non-recursive comb decimator for $R = 64$, $k = 4$	47
Figure 3.1	Traditional FIR filter architecture	57

Figure 3.2	RNS implementation of FIR filter	58
Figure 3.3	i^{th} modulo filter channel	58
Figure 3.4	Forward conversion logic for $\lfloor X \rfloor_{m_i}$	59
Figure 3.5	Modulo adder	60
Figure 3.6	Hardware efficient reverse converter	62
Figure 3.7	Dual-mode programmable decimation filter for WCDMA/WiMAX	67
Figure 3.8	RNS based FIR filter	68
Figure 3.9	i^{th} filter channel of stage 1 for WCDMA/WiMAX decimator	69
Figure 3.10	Dual-mode programmable decimation filter for WCDMA/WLANa	71
Figure 3.11	i^{th} filter channel of stage 1 for WCDMA/WLANa decimator	71
Figure 3.12	Index calculus multiplier	73
Figure 3.13	i^{th} filter channel of stage 1 programmable for WCDMA/WLANa	76
Figure 4.1	Multiple-residue flash converter	85
Figure 4.2	Successive approximation A/R converter	87
Figure 4.3	Iterative flash A/R converter	89
Figure 4.4	Sigma-delta based parallel A/R converter	90
Figure 4.5	A 2-2 cascaded MASH architecture	91
Figure 4.6	RNS based decimation filter for A/R converter	93
Figure 4.7	i^{th} modulo filter channel for A/R converter	93
Figure 4.8	Principle of error correction with RRNS	100
Figure 4.9	Block diagram of transmitter section	101
Figure 4.10	Convolutional encoder	102
Figure 4.11	Block diagram of receiver section	104
Figure 5.1	Logic symbol of RM-ULM(c)	114
Figure 6.1	Power consumption for CIC architectures with $k = 4$ and $B_{in} = 4$	132
Figure 6.2	Area requirement for CIC architectures with $k = 4$ and $B_{in} = 4$	134
Figure 6.3	Frequency response of FIR filter	135
Figure 6.4	PSD plot of original and filtered output (a) Traditional (b) RNS	136
Figure 6.5	Critical Path delay: Traditional Vs RNS implementation	137
Figure 6.6	Area: Traditional Vs RNS implementation	137
Figure 6.7	Speed up factor for RNS filter Vs traditional filter	137

Figure 6.8	Percentage area improvement of RNS filter vs. traditional	137
Figure 6.9	Filter responses for WCDMA mode in WCDMA/WiMAX decimator	141
Figure 6.10	Filter responses for WiMAX mode in WCDMA/WiMAX decimator	141
Figure 6.11	Filter responses for WCDMA mode in WCDMA/WLANa decimator	144
Figure 6.12	Filter responses for WLANa mode in WCDMA/WLANa decimator	144
Figure 6.13	Area requirements for RNS decimation filter	147
Figure 6.14	Placed cell structure for RNS based decimation filter	149
Figure 6.15	Routed view of RNS decimation filter	150
Figure 6.16	Power spectral density (PSD) plot for filter input and output	151
Figure 6.17	PSD plot for decimation filter output at Nyquist rate	151
Figure 6.18	BER versus SNR for uncoded, convolutional coded and RCCC OFDM system	155
Figure 6.19	BER versus multipath delay spread for uncoded, convolutional coded and RCCC OFDM system	156
Figure 6.20	BER versus frame start time error for uncoded, convolutional coded and RCCC OFDM system	157
Figure 6.21	BER versus peak power clipping for uncoded, convolutional coded and RCCC OFDM system	158
Figure 6.22	BER versus channel noise of the RCCC OFDM system for different peak power clipping levels	159
Figure 6.23	Exhaustive branched implementation for $F = \oplus \sum (13, 14)$	163
Figure 6.24	Tree implementation for $F = \oplus \sum (13, 14)$	164
Figure 6.25	Exhaustive branched implementation for $F = \oplus \sum (5, 6, 9, 10)$	164
Figure 6.26	Tree implementation for $F = \oplus \sum (5, 6, 9, 10)$	165
Figure 6.27	Exhaustive branched implementation for $F = \oplus \sum (0, 1, 2, 4, 6)$	165
Figure 6.28	Tree implementation for $F = \oplus \sum (0, 1, 2, 4, 6)$	165
Figure 6.29	Exhaustive branched and tree implementation for $F = \oplus \sum (0, 2, 3, 4, 5)$	166
Figure 6.30	GA implementation for $F = \sum_m (6, 7, 8, 12, 13, 14, 15)$	167
Figure 6.31	GA implementation for $F = \sum_m (1, 2, 4)$	168
Figure 6.32	GA implementation for $F = \sum_m (1, 2, 3, 4, 5, 6, 7)$ using NOR gates	169
Figure 6.33	GA implementation for $F = \sum_m (1, 2, 3, 4, 5, 6, 7)$ using NAND gates	169
Figure 6.34	GA implementation for $F = \sum_m (1, 2, 3, 4, 5, 6, 7)$ using multiplexers	169

Figure 6.35	GA implementation for $F = \sum m (1, 2, 3, 4, 5, 6, 7)$ using RM-ULMs	170
Figure 6.36	Comparison in terms of number of modules and levels required for implementing $F = \sum m (1, 2, 3, 4, 5, 6, 7)$ with various ULMs	170
Figure 6.37	GA implementation for $F = \sum m (7)$ using (a) NOR (b) NAND (c) MUX (d) RM-ULM	171
Figure 6.38	Comparison in terms of number of modules and levels required for implementing $F = \sum m (7)$ using NOR, NAND, MUX, and RM-ULM	171

LIST OF TABLES

Table 1.1	Milestones in wireless communication	4
Table 2.1	Multi-standard specifications and decimation filter design parameters	31
Table 2.2	Interference profile and C/N ratio	32
Table 3.1	Standard specification and decimation filter design parameters for WCDMA/WiMAX transceiver	66
Table 3.2	Standard specification and decimation filter design parameters for WCDMA/WLANa transceiver	70
Table 3.3	Primitive roots for the selected moduli set	77
Table 6.1	Decimation filter implementation results for multiple standards	130
Table 6.2	Total dynamic power consumption for CIC architectures	132
Table 6.3	Highest operating frequency for CIC architectures with $k = 4$, $R = 64$ and $B_{in} = 4$	133
Table 6.4	Area requirement for CIC architectures	133
Table 6.5	Critical path delay and area for 64 taps RNS FIR filter	138
Table 6.6	Critical path delay and area of reverse converter	139
Table 6.7	Critical path delay and area for 64 taps traditional FIR filter	140
Table 6.8	Area and critical path delay of RNS decimation filter for WCDMA/WiMAX transceiver	142
Table 6.9	Area requirement for programmability	143
Table 6.10	Area requirement for WCDMA/WiMAX decimation filter: Traditional Vs RNS	143
Table 6.11	Area and critical path delay of RNS decimation filter for WCDMA/WLANa transceiver	145
Table 6.12	Area requirement for programmability	146
Table 6.13	Area requirement for WCDMA/WLANa decimation filter: Traditional Vs RNS	146
Table 6.14	Area, critical path delay and dynamic power dissipation for RNS decimation filter	147
Table 6.15	Area requirement for programmability	148
Table 6.16	Area, critical path delay and dynamic power dissipation for traditional decimation filter	149
Table 6.17	Sigma-delta modulator complexity for A/R converters of various resolutions	151
Table 6.18	Decimation filter complexity for A/R converters with various resolutions	152
Table 6.19	Performance comparison of A/R converters	153
Table 6.20	PAPR for different peak power clipping	159

Table 6.21	Number of test patterns for various adders in AND-OR and AND-XOR logic	162
Table 6.22	Comparison in terms of delay and hardware for standard, tree and exhaustive branched implementations	166

LIST OF ABBREVIATIONS

A/R	Analog-to-residue
ADC	Analog-to-digital converter
ATPG	Automatic test pattern generator
AWGN	Additive white Gaussian noise
BER	Bit error rate
C/N ratio	Carrier to noise ratio
CF	Correction factor
CIC	Cascaded integrator comb
COFDM	Coded orthogonal frequency division multiplexing
CPA	Carry propagate adder
CR	Clip compression ratio
CRT	Chinese remainder theorem
CSA	Carry save adder
DAB/DVB	Digital audio/video broadcasting
DAC	Digital-to-analog converter
DECT	Digital enhanced cordless telecommunication
DFT	Discrete Fourier transform
DR	Dynamic range
DS-CDMA	Direct sequence – code division multiple access
DSP	Digital signal processing
EDGE	Enhanced data rate for GSM environment
ESOP	Exclusive OR sum-of-products
FA	Full adder
FAN	Fan-out oriented test generator
FEC	Forward error correction
FFT	Fast Fourier transform
FIR	Finite impulse response
FPGA	Field programmable gate array
FPRM	Fixed polarity Reed-Muller
GA	Genetic algorithm

GF	Galois field
GRM	Generalized Reed-Muller
GSM	Global system for mobile communication
GUI	Graphical user interface
GUIDE	Graphical user interface development environment
ICI	Inter carrier interferences
IF	Intermediate Frequency
IFFT	Inverse fast Fourier transform
IIR	Infinite impulse response
IR	Image reject
ISI	Inter symbol interferences
ITU	International telecommunication union
LO	Local oscillator
LUT	Look up table
LNA	Low noise amplifier
MAC	Multiply and accumulate unit
MASH	Multistage noise shaping
MIMO	Multiple input / multiple output
MRC	Mixed radix conversion
MVL	Multiple valued logic
NTF	Noise transfer function
OFDM	Orthogonal frequency division multiplexing
OSR	Oversampling ratio
PA	Power amplifier
PAPR	Peak to average power ratio
PLA	Programmable logic array
PPRM	Positive polarity Reed-Muller
PSD	Power spectral density
QAM	Quadrature amplitude modulation
QPSK	Quadrature phase shift keying
RCCC	RRNS – Convolutional concatenated coding
RF	Radio frequency

RM	Reed-Muller
RM-ULM	Reed-Muller universal logic module
RNS	Residue number system
ROM	Read only memory
RRNS	Redundant residue number system
SD-ADC	Sigma-delta analog-to-digital converter
SDR	Software defined radio
SNR	Signal to noise ratio
SRC	Sampling rate conversion
STF	Signal transfer function
ULM	Universal logic module
VGA	Variable gain amplifier
VLSI	Very large scale integration
WCDMA	Wideband code division multiple access
WIF	Wideband intermediate frequency
WiMAX	Worldwide interoperability for microwave access
WLAN	Wireless local area network

Chapter 1

Introduction

This chapter serves to explore the history of wireless communication and the important milestones in its evolution. Radio Frequency (RF) receiver architectures suitable for high integration and multi-standard capability are presented. The need for Sigma-Delta analog-to-digital converter for multi-standard architectures addressing the various dynamic range requirements and sampling rates is discussed. The basic concepts of a sigma-delta modulator that achieves high resolution and signal to noise ratio in the baseband are also presented.

1.1 History of Wireless Communication

Telecommunication is defined by the International Telecommunication Union (ITU) as the transmission, emission or reception of any signs, signals or messages by electromagnetic systems. The demonstration of electrical telegraphy by Joseph Henry and Samuel F.B. Morse in 1832 followed shortly after the discovery of electromagnetism by Hans Christian Oersted and Andre-Marie Ampere early in the 1820's. In 1864 James Clerk Maxwell proved the existence of electromagnetic waves and postulated wireless propagation. This was verified and demonstrated by Heinrich Hertz in 1887. Guglielmo Marconi started experiments with the radio-telegraph shortly thereafter, and was awarded patent for wireless telegraph system in 1897. In mid-December 1901, he startled the world with transatlantic transmission [Palma, 2001]. In 1876, Alexander Graham Bell patented the telephone. The invention of the diode by Fleming in 1904 and the triode by Lee de Forest in 1906 led to the rapid development of long-distance radio telephony. The invention of the transistor by Bardeen, Braitain and Shockley that later led to the development of integrated circuits, paved the way for miniaturisation of electronic systems.

Wireless communication has developed into a key element of modern society. The developments in satellite transmission, radio and television broadcasting, and the new generation mobile systems have revolutionized global communication. The milestones in the evolution of wireless communication are listed in Table 1.1. The advances in micro-electronic circuits have recently undergone rapid development that made mobile and personal communication systems feasible. The critical attributes of such systems include high speed to transmit information in real time, world wide coverage, reliability, cost and security.

Table 1.1 Milestones in wireless communication

1921	Police car dispatch radios installed in Detroit
1930s	Mobile transmitters developed; radio equipment occupied most of police car trunk
1935	Frequency modulation (FM) demonstrated by Armstrong
1946	First interconnection of mobile users to Public Switched Telephone Network (PSTN)
1947	The advent of the cellular concept by D. H. Ring of AT&T Bell Laboratories
1949	FCC (Federal Communications Commission) recognizes mobile radio as new class of service
1960s	Improved Mobile Telephone Service (IMTS) introduced; supports full-duplex, auto dial, auto trunking
1974	FCC allocates 40 MHz for cellular telephony
Analog Cellular Age: first-generation (1G) for voice-only service	
1979	NTT (Nippon Telegraph and Telephone) Japan deploys first cellular communication system
1981	Commercial operation of NMT450 (Nordic Mobile Telephone – 450 MHz band)
1983	AMPS (Advanced Mobile Phone System) deployed in US in 900 MHz band: supports 666 duplex channels
1986	NMT900 system introduced in Scandinavia
1988	TACS (Total Access Communication System) introduced in UK and Japan
Digital Cellular Age: Second-generation (2G) for digital voice and data communications	
1991	First GSM (Global System for Mobile Communications) network, Radiolinja in Finland was officially opened
1992	All major European operators start commercial operation of GSM networks
1993	GSM1800 system in commercial operation in UK
1993	IS-95 CDMA(Interim Standard-95 Code Division Multiple Access) standard finalised
High Speed Cellular Age: Third-generation (3G) and higher for more advanced mobile broadband services such as Internet and wireless access	
1999	Finland gave out the world's first 3G mobile technology licenses
2000s	First commercial GPRS and cdma2000 networks launched
2001	Ericsson and Vodafone UK claim to have made the world's first WCDMA (Wideband-CDMA) voice call over commercial network. Nokia and AT&T Wireless complete first live 3G EDGE (Enhanced Data rate for GSM Evolution) call.
2002	Qualcomm announces world's first Bluetooth WCDMA (UMTS) and GSM Voice Calls
2003	LG introduced the world's first dual band, dual mode phone for both CDMA and WCDMA
2004	Nortel Networks has completed the industry's first 3G partnership project (3GPP) compliant UMTS (Universal Mobile Telecommunication Systems) Assisted Global Positioning System (A-GPS) calls at 2100 and 1900 MHz
2005	Ericsson demonstrates 9 Mbps with WCDMA, HSDPA (High-Speed Downlink Packet Access) phase 2
2006	Ericsson is the first to complete WCDMA calls on all 3GPP-defined frequency bands
2007	4G Technology beginning to shape up based on OFDM (Orthogonal Frequency Division Multiplexing) and aiming at 100 Mbps for wide area mobile applications.
2008	Aims to launch WiMAX network offering 185 million subscribers 4G wireless data speeds up to 5 times faster than current wireless networks.

1.2 Wireless System Developments

The Fourth Generation (4G) wireless networks aim to support global roaming across multiple wireless and mobile networks. Technologies employed by 4G include software defined radio (SDR) receivers, orthogonal frequency division multiplexing (OFDM) and multiple input/multiple output (MIMO) to attain reconfigurability and high data rate transmission. The recent advances in device scaling and integration together with the increasing demand for more services in portable communication favors the development of multi-mode transceivers [Soudris et al., 2000]. Reconfigurable wireless system is a new wireless technology that has great research potential and market interest. The trend for multi-mode support and flexible design increases the digital signal processing (DSP) portion of the transceiver. This implies moving the DSP more close to the front-end by placing analog-to-digital converter close to the antenna.

Sigma-delta analog-to-digital converters (SD-ADCs) are widely used in multi-standard transceivers to adapt to the requirements of different standards. The SD-ADCs are promising candidates because of its readiness to reconfigure by changing the oversampling ratio, loop filter coefficients, loop filter order and number of quantizer bits. The most common architectures of Radio Frequency (RF) receivers with an emphasis on multi-standard capability and the principles of sigma-delta converter are explained in the following Sections 1.3 and 1.4.

1.3 Conventional RF Receiver Architectures

The general block diagram for an RF transceiver is shown in Figure 1.1. The incoming RF signal from the antenna is fed to the signal conditioning

block. The analog signal conditioning block performs frequency translation, amplification and filtering. Then it is digitized by an analog-to-digital converter (ADC). Sigma-delta ADCs are widely used in wireless communication transceivers as it provides high resolution and wide dynamic range. The DSP block performs further processing of the signal in digital domain. The sub-sections 1.3.1 – 1.3.4 describe the various RF receiver architectures with emphasis on its capability for high level of integration and multi-standard operation [Barrett, 1997].

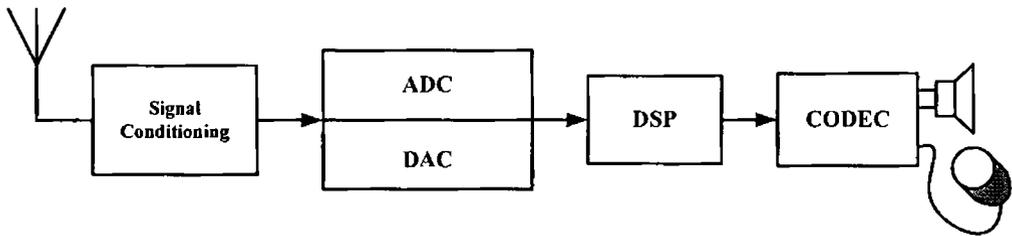


Figure 1.1 Block diagram of a RF transceiver

1.3.1 Superheterodyne Receiver

The traditional superheterodyne receiver architecture is shown in Figure 1.2. The received signal is passed through an RF filter and a low noise amplifier (LNA) that determines the receiver sensitivity. The signal is then passed through an image rejection (IR) filter that removes any image frequency signal present. The incoming signal frequency is translated to an intermediate frequency (IF) using a mixer and local oscillator. The IF is assigned a suitable high value to get adequate image rejection. The first local oscillator (LO_1) is tuned to select the desired channel so that the signal is centred on the IF after mixing. The IF bandpass filter passes the desired signal rejecting the interference signals. The amplified IF signal is given to the second local oscillator (LO_2) to split it into in-phase (I) and quadrature-phase

(Q) baseband components. This is followed by an antialiasing filter and analog-to-digital converter. Since the RF filter, IR filter and IF filter are off-chip filters that are specific to a particular standard, this architecture is not suitable for high integration and multi-standard operation.

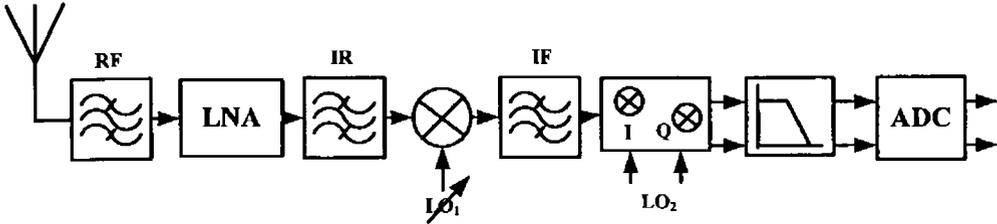


Figure 1.2 Superheterodyne receiver

1.3.2 Direct Conversion Homodyne Receiver

The direct conversion homodyne receiver architecture is shown in Figure 1.3. The incoming signal after pre-filtering and amplification is directly translated to the baseband frequency without any IF. Hence it is also called a zero-IF receiver. The local oscillator is tuned to the centre frequency of the desired signal. The desired signal in-phase and quadrature-phase components are filtered via an on-chip lowpass filter and are digitized by ADC for further processing. As the received RF signal is directly translated to baseband, it does not suffer from image signal interferences. This eliminates the need for off-chip image reject filter. Also, it can be programmed for a multi-standard operation by tuning the local oscillator to select different standards. So the direct conversion receiver is suitable for higher integration and multi-standard operation. On the other hand, careful design considerations are required to control the DC offset and flicker noise created at the output of the mixer. The direct conversion homodyne receiver is the receiver architecture considered for analysis through out this research work.

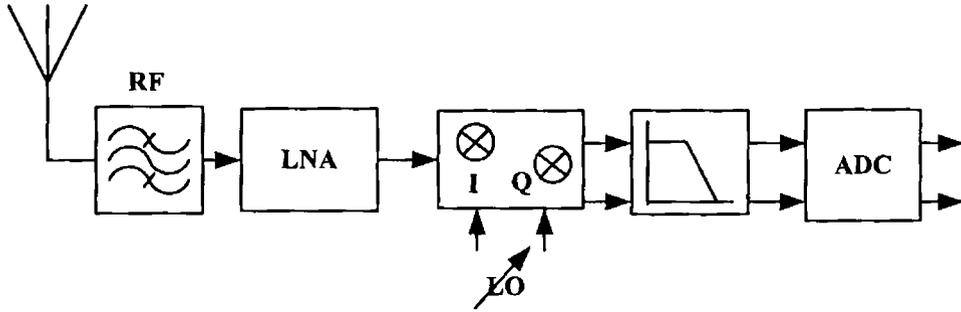


Figure 1.3 Direct conversion homodyne receiver

1.3.3 Low IF Receiver

The low IF receiver architecture is shown in Figure 1.4. In this topology, the incoming signal is translated to a suitably low IF, avoiding the DC offset and flicker noise associated with direct conversion. A low IF on-chip bandpass filter is used to perform baseband channel selection. Hence this architecture offers high level of integration. But it is not attractive for wide bandwidth standards, as the power dissipation in the bandpass filter and ADC becomes intensive. So, the low IF receiver has only limited multi-standard capability.

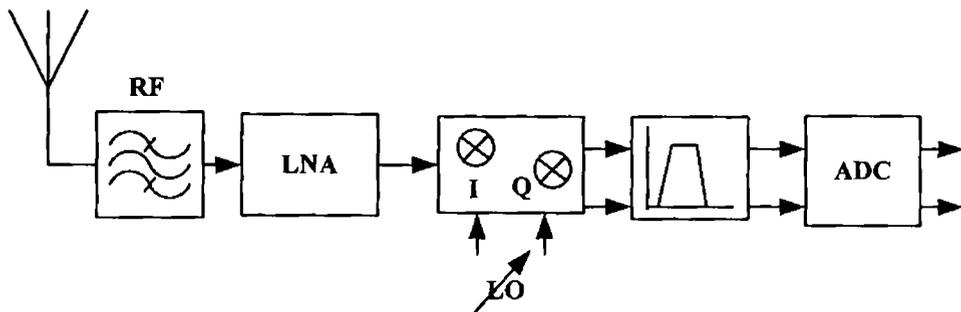


Figure 1.4 Low IF receiver

1.3.4 Wideband IF Double Conversion Receiver

The wideband IF (WIF) double conversion receiver architecture is shown in Figure 1.5. This architecture combines the strength of

superheterodyne and direct conversion receivers through a two-stage frequency translation process. The first mixer mixes the RF signal with a fixed LO signal to produce IF which is then filtered by a lowpass filter to remove the harmonics and high frequency noise. The second mixer mixes this signal with variable LO signal to baseband, centring the desired channel at DC. The channel select filtering is performed at baseband with a programmable lowpass filter, making it suitable for multi-standard operation. This architecture also offers a high level of integration, as the off-chip IF and IR filters are eliminated.

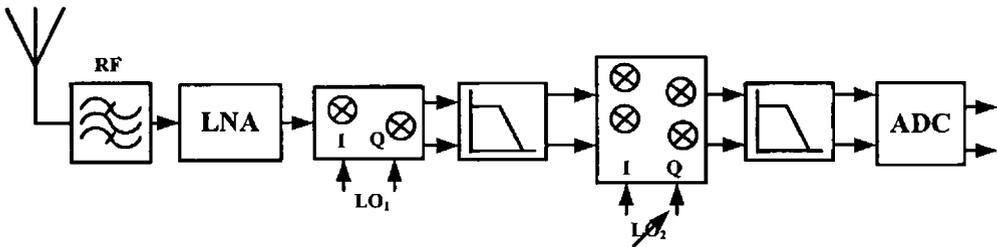


Figure 1.5 Wideband IF double conversion receiver

1.4 Analog-to-Digital Converters

The process of converting an analog signal to digital representation encompasses sampling of analog waveform in time and quantizing it in amplitude [Rabii and Wooley, 1998]. The minimum rate at which the signal can be sampled is governed by its bandwidth. The block diagram of an ADC is shown in Figure 1.6.

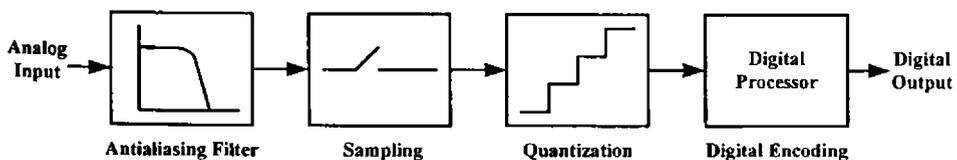


Figure 1.6 Block diagram of analog-to-digital converter

The antialiasing filter is used to limit the bandwidth of the analog input to less than half of the sampling frequency. This ensures that the sampling process will not alias the out-of-band signals back into the baseband of the ADC. The width of the transition band of the antialiasing filter increases with the sampling rate, thereby decreasing the complexity of the filter. The quantized signal is digitally encoded based on the resolution which in turn dictates the quantization error. The ADCs are classified as Nyquist rate ADCs and oversampling ADCs based on the rate at which the signal is sampled relative to the signal bandwidth.

1.4.1 Nyquist Rate Analog-to-Digital Converters

Nyquist rate ADCs sample the signal at approximately twice the signal bandwidth, which is the minimum rate required for reconstruction of the signal according to Nyquist theorem. Such converters are used in data conversion systems where the conversion process is constrained by the bandwidth limitations of the implementation technology. Nyquist rate converters include parallel flash converters, successive approximation converters, counting or ramp converters etc. It generally requires operations such as comparison, amplification or subtraction to be performed at the overall precision of the converter. This translates into the need for precise component matching which limits the Nyquist rate converters to about 10 to 12-bits of resolution. So, alternate techniques are to be used for high resolution applications.

1.4.2 Oversampling Analog-to-Digital Converters

Oversampling ADCs sample the analog input at a rate much higher than the Nyquist rate. The ratio of sampling rate to the Nyquist rate is called the oversampling ratio. Oversampling ADCs exchange resolution in time for

resolution in amplitude [Allen and Holberg, 2002]. Here each output is produced from a sequence of coarsely quantized input samples, whereas in Nyquist rate converters each digital word is obtained from an accurately sampled single sample of input. In oversampling ADCs the analog part is simple and do not require stringent component matching by performing most of the conversion process in digital domain. Hence these converters are suitable for high resolution implementations and today's VLSI technology tailored towards high speed and high density digital circuits. Oversampling converter relaxes the requirements of antialiasing filter. Another advantage is that they do not require a sample and hold circuit at the input.

Oversampling ADCs are classified into three main groups as straight-oversampling ADCs, predictive ADCs and noise shaping ADCs. Straight-oversampling ADCs exploit the fact that quantization noise is equally distributed over the entire frequency range, from DC to half the sampling frequency. So quantization noise power per frequency is reduced as shown in Figure 1.7. The shaded portion shows the baseband noise power (N_B) for oversampling ADC. Overall resolution is improved by removing the out-of-band noise by a digital filter. Predictive and noise shaping ADCs utilize oversampling and noise shaping concepts to attain a more efficient accuracy-speed trade off. Noise shaping is performed by placing the quantizer in a feedback loop in conjunction with a loop filter. In predictive ADCs loop filter is placed in the feedback path, and both signal and quantization noise spectra are shaped. In noise shaping ADCs, also called sigma-delta ADCs (SD-ADCs), the loop filter is placed in the feedforward path. Here, only the noise spectrum is shaped preserving the signal spectrum.

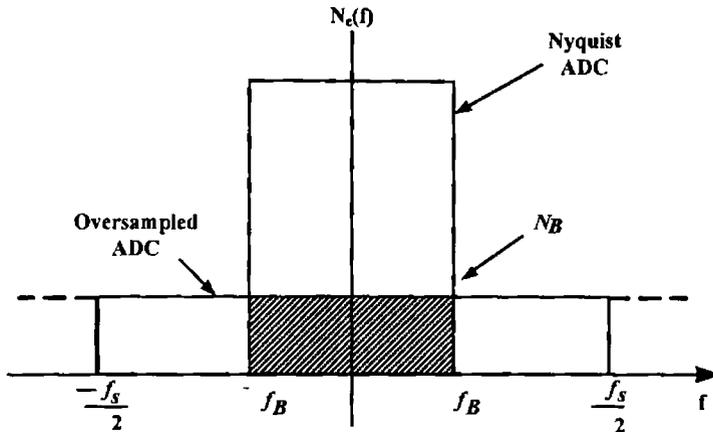


Figure 1.7 Baseband quantization noise power

1.4.2.1 Sigma-Delta Analog-to-Digital Converter

The SD-ADC is the most widely used oversampling ADC as it is more robust against circuit imperfections. It consists of an analog sigma-delta modulator part and a digital decimator part. A simple first order sigma-delta ADC is depicted in Figure 1.8. The first order sigma-delta ($\Sigma\Delta$) modulator consists of an integrator and a coarse quantizer placed in a feedback loop. The quantizer is realized as either a one-bit comparator or a multi-bit quantizer.

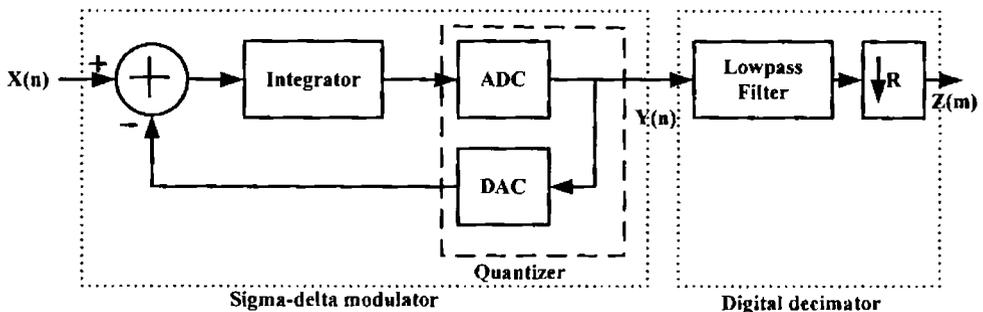


Figure 1.8 Block diagram of first order sigma-delta ADC

When the integrator output is positive, quantizer feedbacks a positive signal which is subtracted from the input signal. This moves the integrator

output in the negative direction. Similarly, when the integrator output is negative, the quantizer feedbacks a negative signal that is added at the input to move integrator output in the positive direction. Hence, the integrator accumulates the difference between input and quantizer output, and maintains the integrator output around zero. A zero integrator output implies that the difference between input and quantizer output is zero. In the sampled data model of modulator, quantization is represented as an additive error q , defined as the difference between the modulator output y and the quantizer input v . The input-output relationship of modulator as a difference equation is given in (1.1). The corresponding z -transform is given (1.2).

$$y[nT_s] = x[(n-1)T_s] + q[nT_s] - q[(n-1)T_s] \quad (1.1)$$

$$Y(z) = z^{-1}X(z) + (1 - z^{-1})Q(z) \quad (1.2)$$

From (1.2), the signal transfer function (STF) is z^{-1} which represents unit delay and the noise transfer function (NTF) is $(1 - z^{-1})$ which represents a highpass characteristics. Hence the noise is shaped in such a way that the highpass nature allows noise suppression at low frequencies. This results in high signal to noise ratio (SNR) in the baseband. In general, L^{th} order noise shaping is achieved by placing L integrators in the forward path of the modulator. The noise transfer function becomes $(1 - z^{-1})^L$. As the order of the modulator L increases, the baseband noise power decreases as shown in Figure 1.9. The modulator is followed by a digital decimation filter which removes the out-of-band noise and reconstructs the signal back at Nyquist rate. While the simple analog part of the ADC generally determines the resolution, the digital part occupies most of the die area and consumes much power. Hence, the design techniques that can reduce the die area and dynamic power consumption draw a great deal of research interest.

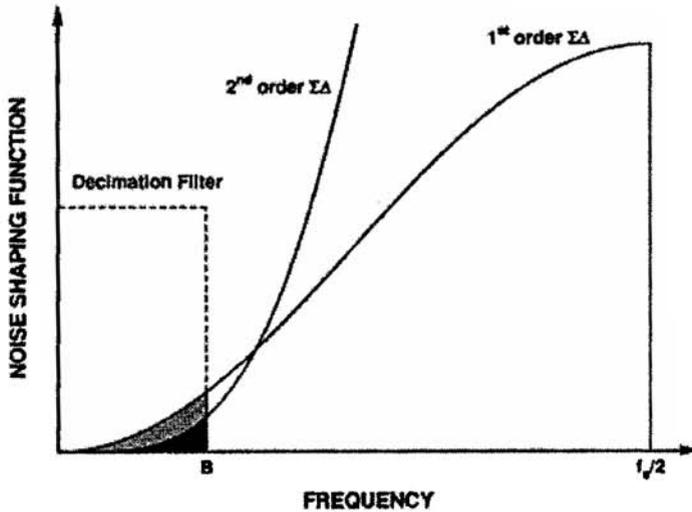


Figure 1.9 Noise shaping in sigma-delta modulators

1.5 Layout of the Thesis

The layout of the thesis is as follows: Chapter 2 presents a decimation filter design toolbox developed in MATLAB for six popular wireless standards, namely GSM, WCDMA, WLANa, WLANb, WLANg and WiMAX. The toolbox enables the communication system designer to perform a quick visual analysis of the filter performance without doing the complicated design calculations. A polyphase implementation of the non-recursive comb decimator for high speed operation that can be used as the first stage of a multistage decimator is also presented.

Chapter 3 illustrates the use of Residue Number System (RNS) for the high speed and area efficient implementation of digital filters. Dual-mode RNS based decimation filters reconfigurable for WCDMA/WiMAX and WCDMA/WLAN standards are designed and implemented. Modulo multipliers based on index calculus approach are used for increased programmability needed in multi-standard operation. A dual-mode decimator

programmable for WCDMA/WLAN standards is designed using index calculus approach.

Chapter 4 introduces a novel parallel analog-to-residue converter based on sigma-delta converter for high resolution. An OFDM based communication system that explores the error detection and correction properties of Redundant Residue Number System (RRNS) is presented. The improvements in system performance obtained through the use of RRNS-Convolutional Concatenated Coding (RCCC) scheme for forward error correction under different operating conditions are also shown.

Chapter 5 describes easily testable circuit design using Reed-Muller (RM) logic. An exhaustive branching algorithm for implementing logic functions using Reed-Muller Universal Logic Modules (RM-ULMs) is presented. Genetic programming approach to realize logic functions using a particular ULM that implements the function using minimum number of modules and levels is also presented.

Chapter 6 presents the simulation results of various reconfigurable architectures and circuits described in Chapters 2, 3, 4 and 5. The performances of the new systems are compared with that of existing systems and the results are tabulated.

Chapter 7 concludes the thesis by drawing the conclusions from the results, and suggests possible extensions of the research work for further investigation. References are provided following this chapter along with the 'List of Publications' of the author.

Chapter 2

Decimation Filter Design: A Toolbox Approach

A multi-standard decimation filter design often involves extensive system level analysis and architectural partitioning, typically requiring extensive calculations. This chapter describes a multistage decimation filter design tool developed in MATLAB[®] using Graphical User Interface Development Environment (GUIDE) for visual analysis. The toolbox is designed for six popular wireless communication standards consisting of GSM, WCDMA, WLANa, WLANb, WLANg and WiMAX.

This chapter also presents a computationally efficient polyphase implementation of non-recursive cascaded integrator comb (CIC) decimator for Sigma-Delta Converters. This polyphase implementation offers high speed operation and low power consumption for the first stage of a multistage decimator.

2.1 Decimation Filter Design Considerations

Software defined radio (SDR) is a wireless interface technology in which software-programmable hardware is used to provide flexible radio solutions in a single transceiver system. New telecommunication services requiring higher capacities, data rates and different operating modes have motivated the development of new generation multi-standard wireless transceivers. Current RF transceivers demand higher integration for low cost and low power operations, and adaptability to multiple communication standards. Multi-standard operation is achieved by using a receiver architecture that performs channel selection on chip at baseband [Gray and Meyer, 1995]. This baseband channel filtering is performed in digital domain to adapt to the channel bandwidths, sampling rates, carrier to noise (C/N) ratio, and blocking and interference profiles of multiple communication standards [Barrett, 1997].

Sigma-delta ADCs are used in multi-standard transceivers to adapt to the requirements of different standards. The dynamic range of a SD-ADC can be easily adjusted by selecting different oversampling ratios. Sigma-delta modulator based on oversampling technique provides high resolution over wide bandwidth that is required in multi-mode receivers. High signal to noise ratio (SNR) is achieved in the signal band through noise shaping. The digital decimation filter selects a desired channel and removes the out-of-band quantization noise produced by the modulator. Further, it reduces the sampling rate from oversampled frequency of the modulator to the Nyquist rate of the channel [Norsworthy et al., 1997]. Therefore in a multi-mode transceiver, SD-ADC requires a decimation filter with programmable decimation ratios.

The design issues of decimation filters for wireless communication transceivers are well studied in literature. A low power fifth order comb decimation filter with programmable decimation ratios and sampling rates for GSM (Global System for Mobile communications) and DECT (Digital Enhanced Cordless Telecommunication) standards is presented by Gao [Gao et al., 2000]. They have developed non-recursive architecture for comb filter to achieve a low power VLSI implementation. Ghazel has presented the design and implementation of digital filter processors that can be used as downsamplers in wireless transceivers. The method is detailed for DECT standard [Ghazel et al., 2003]. Low complexity decimation filter architecture is presented [Xihuitl, 2005] by using infinite impulse response (IIR) filters implemented by all-pass sum that avoids multiplications. A low-power high linearity variable gain amplifier (VGA) embedded in a multi-standard receiver that meets the standard requirements has been reported [Amico et al. 2006]. Tao et al. have given an overview on the design considerations of the decimation filters for GSM, WCDMA (Wideband Code Division Multiple Access), 802.11a, 802.11b, 802.11g and WiMAX (Worldwide Interoperability for Microwave Access) standards [Tao et al., 2006].

As a part of the research, a decimation filter design toolbox is developed in MATLAB[®] Graphical User Interface Development Environment (GUIDE) addressing the design issues presented in the above papers. The toolbox includes six wireless standards consisting of GSM, WCDMA, WLANa, WLANb, WLANg and WiMAX, and provides an appropriate multistage decimation filter for each standard. The toolbox will help the user or design engineer to perform a quick design and analysis of decimation filter for multiple standards without doing extensive calculation of the underlying methods. Decimation is done in two or three stages to reduce the hardware

complexity and power dissipation. Each stage is implemented with optimized filters so that the overall cascaded filter response meets the specification for a particular standard. The implementation complexity in terms of filter length that meets the specification for any of these standards is computed using this tool, and is tabulated.

2.2 Receiver Architecture for Multi-standard Operation

The receiver architecture that emphasizes high integration and multi-standard capability is required for new generation wireless applications. High integration can be achieved by utilizing a receiver architecture that performs baseband channel select filtering on chip. This enhances the programmability to different dynamic range, linearity and signal bandwidth to meet the requirements of multiple RF standards. A wideband high dynamic range sigma-delta modulator can be used to digitize both the desired signal and potentially stronger adjacent channel interferers.

A direct conversion homodyne receiver architecture which is an example of a receiver suitable for high integration and adaptability [Barrett, 1997] is shown in Figure 2.1. This architecture translates the incoming frequency to baseband directly to eliminate external components within the receive path. It can be programmed for multi-standard operation since the local oscillator (LO) is tuned to the same frequency as the incoming RF frequency to select different standards. The incoming RF signal is multiplied by one sided LO signal of a frequency equal to the centre frequency of the desired signal band, and hence does not suffer from image signal interference. The down-conversion with a one sided LO signal is achieved by a quadrature mixer in which the incoming signal is multiplied by two LO signals with 90

degrees out of phase. These in-phase and quadrature-phase components are then lowpass filtered and sent to ADCs. The digital signal from ADC is given to digital signal processing section for demodulation. Homodyne receivers are multi-standard capable because the channel filtering is done at baseband. However, the noise and DC offset are to be reduced to achieve adequate dynamic range.

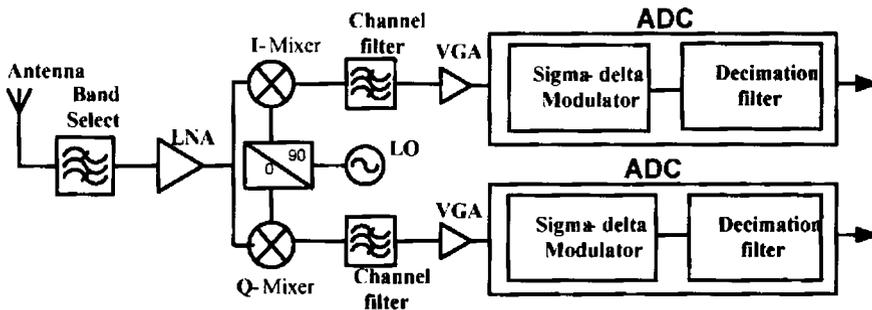


Figure 2.1 Direct conversion homodyne receiver architecture

2.2.1 Reconfigurable Sigma-Delta ADC

The sigma-delta ($\Sigma\Delta$) analog-to-digital converters are widely used in wireless systems because of their superior linearity, robustness to circuit imperfections, inherent resolution-bandwidth trade off and increased programmability in digital domain. A highly linear sigma-delta modulator for multi-standard operation that can achieve high resolution over a wide variety of bandwidth requirements remains challenging. A reconfigurable ADC is a promising solution to keep the power dissipation as low as possible [Xotta et al., 2005], [Zhang et al., 2004].

Single loop and multistage noise shaping (MASH) topologies are two different approaches for implementing $\Sigma\Delta$ modulators. Single loop structures with a higher-order noise transfer function combined with multi-bit feedback can achieve higher dynamic range (DR) with low oversampling ratio (OSR).

But the linearity and resolution of the overall $\Sigma\Delta$ modulator are limited by the precision of the multi-bit digital-to-analog converter (DAC). MASH topology is preferred over single loop structures since the coefficients are optimized for a specific OSR. It has flexibility to handle different OSRs with little modification. MASH structures are adopted for multi-mode receivers considering the stability and reconfigurability.

The theoretical dynamic range has been used in conjunction with the implementation attributes to choose the optimal topology for different RF standards. The dynamic range DR of a $\Sigma\Delta$ modulator is given by

$$DR = \frac{3}{2} \frac{2L+1}{\pi^{2L}} M^{2L+1} (2^B - 1)^2 \quad (2.1)$$

where L is the order of modulator, M is the OSR and B is the number of bits of the quantizer. The six popular standards considered for the toolbox are GSM, WCDMA, 802.11a, 802.11b, 802.11g and WiMAX. These standards have different bandwidth requirements. Since the bandwidth requirements of WLAN-a, b, g and WiMAX are more or less the same, the same topology can be adopted with different OSRs. This will reduce the DR calculation for the main three standards GSM, WCDMA and WLAN (Wireless Local Area Network) whose DR requirements are chosen as 94dB, 79dB and 69dB respectively.

OSR can be selected as 128 for low data rate application, such as GSM receiver, due to a much smaller signal bandwidth. A basic second order modulator with 1-bit quantization is sufficient for this kind of application. In order to meet the DR requirements demanded by WCDMA, a fourth order cascaded MASH topology will be enough with an OSR of 16. If WLANa becomes the target standard, a fifth order topology is a good compromise to achieve the required DR with a 4-bit quantizer and an OSR of 8. The sigma-

delta modulator can be made programmable, and all the blocks are switched to operation only in the WLAN mode. This results in power saving when the receiver is operating in other modes. Similar considerations apply for other standards also. The OSR is chosen as 12, 12 and 8 for WLANb, WLANg and WiMAX respectively. Sigma-delta modulator is followed by a programmable decimation filter operating in the digital domain. The toolbox focuses on the design of multistage decimation filter for multiple standards, which is highlighted in Figure 2.1.

2.3 Multistage Decimation Filter

The sampling rate is downconverted from the oversampled rate of sigma-delta modulator to a data rate that can be conveniently processed by existing DSP processors. This minimizes the power consumption of DSP processors for demodulation and equalization. The purpose of decimation filter is to remove all the out-of-band signals and noise, and to reduce the sampling rate from oversampled frequency of the sigma-delta modulator to Nyquist rate of the channel. The decimation filter consists of a lowpass filter and a downsampler. It is possible to perform noise removal and downconversion with a single stage finite impulse response (FIR) filter. The filter order N of FIR lowpass filter is given in (2.2), where D_∞ is a function of the required ripples δ_p and δ_s in the passband and stopband respectively, F_s is the sampling frequency and Δf is the width of transition band.

$$N \approx D_\infty(\delta_p, \delta_s) \left(\frac{F_s}{\Delta f} \right) \quad (2.2)$$

As the sigma-delta modulators are oversampled, the transition band is small relative to sampling frequency leading to excessively large filter orders.

The power consumption of the filter depends on the number of taps as well as the rate at which it operates. So computational complexity is high for single stage implementation of decimation filter and consumes much power. This can be overcome by multistage approach.

Implementing decimation filter in several stages reduces the total number of filter coefficients. The filters operating at higher sampling rates have larger transition bands, and the filters with lower transition bands operate at reduced sampling frequencies. Subsequently, the hardware complexity and computational effort are reduced in multistage approach. This will lead to low power consumption. A multistage sampling rate conversion (SRC) system consists of a cascade of single stage SRC systems as shown in Figure 2.2. The ' i^{th} ' stage performs decimation by a factor of ' R_i ' such that the overall decimation factor ' R ' is given by $R = \prod_{i=1}^P R_i$, where ' P ' is the total number of stages. The individual filter of each stage is designed within the frequency band of interest in order to prevent aliasing in the overall decimation process.

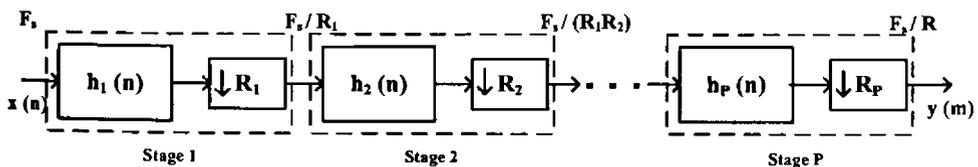


Figure 2.2 Multistage decimation filter

The performance of a decimation filter depends on the filter architecture and the order of each stage of a multistage decimator. FIR filters are widely used in decimators as most of the modulation schemes require linear phase characteristics. The different filter architectures used in this work are given below.

2.3.1 Cascaded Integrator Comb Filter

Hogenauer devised a flexible, multiplier free Cascaded Integrator Comb (CIC) filter that can handle large sampling rate changes suitable for hardware implementation [Hogenauer, 1981]. The basic structure of the Hogenauer CIC filter is shown in Figure 2.3. This consists of an integrator and a comb filter as two basic building blocks. So, it is an infinite impulse response (IIR) filter followed by a finite impulse response (FIR) filter. In a CIC filter of order k , the integrator section consists of a cascade of ' k ' digital integrators operating at the high sampling rate F_s . Each integrator is a one-pole filter with unity feedback coefficient, and the transfer function is

$$H_I(z) = \frac{1}{1 - z^{-1}} \quad (2.3)$$

The comb section consists of ' k ' comb stages with a differential delay of ' M ' and operates at the low sampling rate F_s / R , where ' R ' is the rate change or decimation factor. The transfer function of a comb stage referenced to high sampling rate is

$$H_C(z) = 1 - z^{-RM} \quad (2.4)$$

The rate change switch between the two filter sections subsamples the output of the integrator stage reducing the sample rate from F_s to F_s / R . In practice, the differential delay, M is usually held equal to 1 or 2. Using (2.3) and (2.4), the system transfer function of the CIC filter with respect to the high sampling rate F_s is given by

$$H(z) = H_I^k(z) H_C^k(z) = \frac{(1 - z^{-RM})^k}{(1 - z^{-1})^k} = \left[\sum_{i=0}^{RM-1} z^{-i} \right]^k \quad (2.5)$$

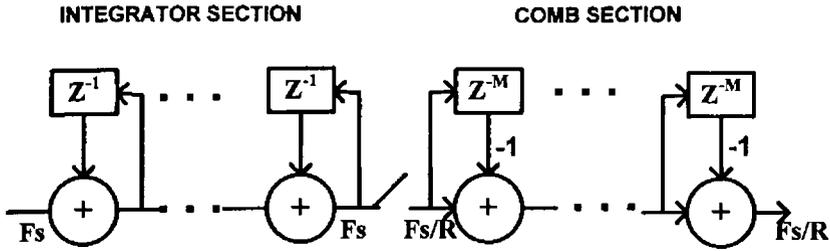


Figure 2.3 CIC decimation filter

The working of CIC filters is based on the fact that perfect pole/zero cancellation can be achieved. From the transfer function in (2.5), it is clear that RM zeros are generated by the numerator term with a multiplicity of k . The k poles at $z = 1$, generated by the denominator are cancelled by the k zeros of the CIC filter [Meyer-Baese, 2001]. On evaluating the frequency response given by (2.5) at $z = \exp(j2\pi f/R)$, where ' f ' is the frequency relative to low sampling rate (F_s/R), the magnitude response of CIC filter is obtained as

$$|H(f)| = \left| \frac{\sin \pi M f}{\sin \frac{\pi f}{R}} \right|^N \quad (2.6)$$

As for small values of ' x ', $\sin x \approx x$, the magnitude response given in (6) can be approximated for large ' R ' as

$$|H(f)| = \left| RM \frac{\sin \pi M f}{\pi M f} \right|^N \text{ for } 0 \leq f < \frac{1}{M} \quad (2.7)$$

The output spectrum has nulls at multiples of $f = \frac{1}{M}$. The aliasing or imaging occurs in the region around the nulls. An example of CIC response used for GSM case, with $F_s = 34.667\text{MHz}$, $R = 32$, $M = 1$ and $k = 3$ is shown in Figure 2.4.

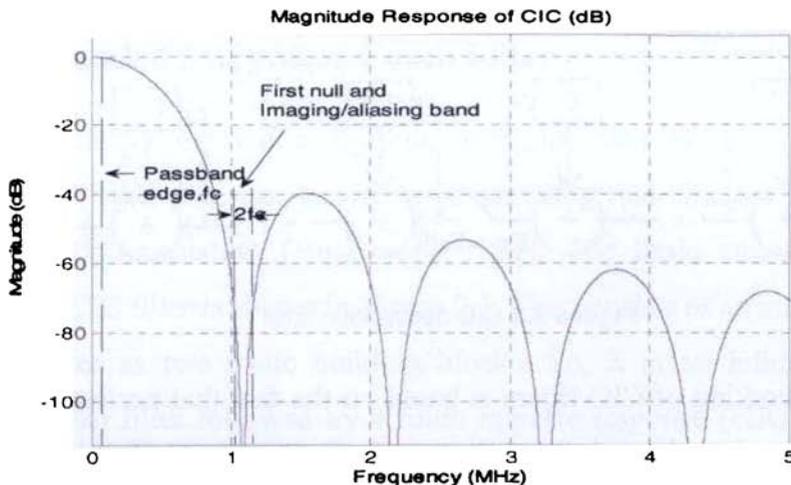


Figure 2.4 CIC magnitude response for GSM with $F_s = 34.667\text{MHz}$, $R = 32$, $M = 1$ and $k = 3$

The amount of passband aliasing or imaging error can be brought within prescribed bounds by increasing the number of stages in the CIC filter. It will also increase the passband droop. The width of the passband and the frequency characteristics outside the passband are severely limited. So, CIC filters are used only to facilitate transition between high and low sampling rates. The CIC filter is followed by one or two stages of finite impulse response (FIR) filters operating at low sampling rates. These are designed to attain the required transition bandwidth and stopband attenuation.

2.3.2 Halfband filter

Halfband filters are a special class of symmetric FIR filters used in second stage of multistage decimators. Halfband filters are characterized by equal passband and stopband ripples ($\delta_p = \delta_s$), and the transition band is symmetrical about $\pi/2$ such that $\omega_p + \omega_s = \pi$, where ω_p and ω_s correspond to the passband and stopband edges. The impulse response $h(n)$ exhibits symmetry with almost 50% of coefficients 'zero' and with a magnitude of 0.5

at $F_s/4$. This implies reduced number of filter taps, lesser hardware and low power consumption. Halfband filters are used to perform decimation by a factor of 2 [Norsworthy et al., 1997]. The ideal halfband filter characteristic is as shown in Figure 2.5, where Δf is the width of the transition band and F_s is the sampling frequency.

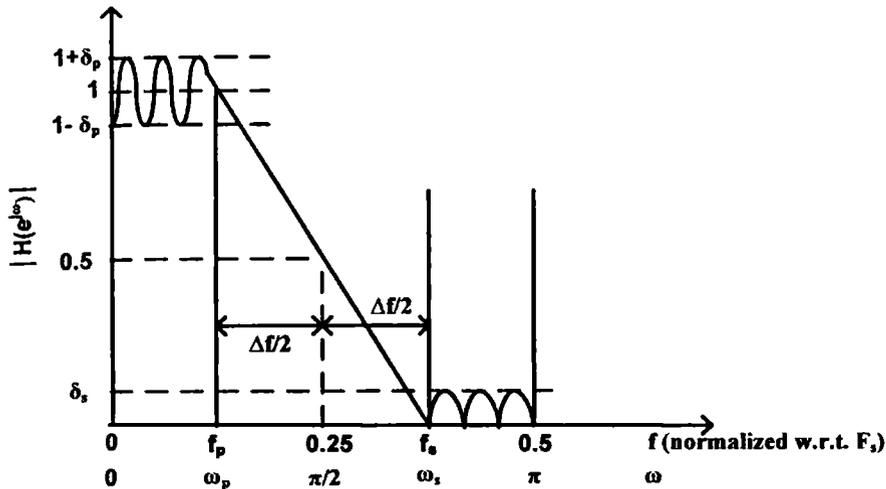


Figure 2.5 Magnitude response of halfband filter

2.3.3 FIR filter

The third type of filter used in the multistage decimator is a FIR filter. The CIC filter response exhibits a droop in the passband which progressively attenuates the signals. The passband droop and stopband attenuation increases as the number of sections of CIC filters increases. The FIR filter used in the last stage performs decimation and CIC droop compensation. This FIR filter is designed according to the differential delay and number of sections of CIC filter along with the passband ripple and stopband attenuation to meet the overall specification of a particular standard. So, a low computational complexity multistage decimator is obtained with a CIC filter followed by

halfband and droop correct FIR filter. The magnitude response of a droop compensating FIR filter designed to compensate the passband droop produced by a CIC filter with a differential delay of $M = 1$, and number of sections $k = 4$, is shown in Figure 2.6.

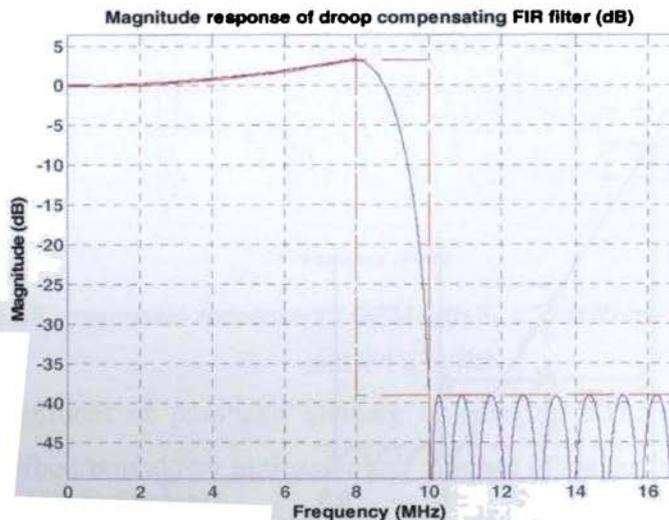


Figure 2.6 Magnitude response of droop compensating FIR filter with $M = 1$ and $k = 4$

2.4 Decimation Filter Design Specification

The specifications for all six standards considered in this toolbox and their corresponding decimation filter design parameters are given in Table 2.1. The oversampling ratio (OSR) for each standard is selected so as to get the required dynamic range for the sigma-delta modulator of a particular order and number of quantizer bits. The receiver specifications and the blocking and interference profiles are defined first in order to set the parameters for the decimation filter. There are large undesired signals called ‘blockers’ within the same cell, and large undesired signals known as ‘adjacent channel interferers’ from the neighbouring cells. These interference signals are to be limited within a certain range for each standard for proper reception of the desired signals.

The decimation filter is generally designed to minimize the undesired signals in the desired band of operation. The output carrier to noise (C/N) ratio is calculated from the bit error rate (BER) of each standard and the modulation scheme used. Table 2.2 gives the interference profile and the C/N ratio for all the six standards [Tao et al., 2006]. The passband frequency edge is taken as 80% of the bandwidth for each standard. The passband ripples are chosen to minimize signal distortions in the signal band. The stopband attenuations shown in Table 2.1 are selected according to the interference profile and C/N ratio given in Table 2.2 for each standard.

Table 2.1 Multi-standard specifications and decimation filter design parameters

Standards	Frequency range (GHz)	Channel Spacing (MHz)	Symbol rate / Chip rate	OSR	Input sampling frequency, F_s (MHz)	Passband edge (MHz)	Stopband edge (MHz)	Passband ripple (dB)	Stopband attenuation (dB)
GSM	DL: 0.935-0.96 UL: 0.89-0.915	0.2	270.833 Ksymbols/s	128	34.667	0.08	0.1	0.1	65
WCDMA	DL: 2.11-2.17 UL: 1.92-1.98	5	3.84 Mchips/s	16	61.44	2	2.5	0.5	55
WLANa	5.15-5.35	20	12 Msymbols/s	8	96	8	10	0.5	44
WLANb	2.4-2.4835	25	11 Mchips/s	12	132	10	12.5	0.5	42
WLANg	2.4-2.4835	25	12 Msymbols/s	12	144	10	12.5	0.5	44
WiMAX	10-66	20	16.704 Msymbols/s	8	133.632	8	10	0.5	39

Table 2.2 Interference profile and C/N ratio

Standard	Offset from central frequency (MHz):				C/N ratio (dB)
	Interference magnitude (dBm)				
GSM	0.2 : -90	0.4 : -58	0.6 : -46	1 : -42	9.7
WCDMA	5 : -63	10 : -56	12.5:-44		7.2
WLANa	20 : -63	40 : -47			28
WLANb	25 : -35				7
WLANg	20 : -63	40 : -47			28
WiMAX	20 : -68	40 : -49			21

2.5 Multi-standard Decimation Filter Design Toolbox

The 'Multi-standard Decimation Filter Design Toolbox' is designed using the Signal Processing Toolbox and Filter Design Toolbox from MATLAB® using GUIDE environment. The user can select a required wireless communication standard and obtain the corresponding multistage decimation filter implementation using this toolbox. The toolbox will help the user or design engineer to perform a quick design and analysis of a decimation filter for multiple standards without doing extensive calculation of the underlying methods. The front panel of the graphical user interface (GUI) is shown in Figure 2.7 and the features of the toolbox are detailed below.

Multistage decimation filter design

The toolbox is designed for six popular wireless communication standards, namely GSM, WCDMA, WLANa, WLANb, WLANg and WiMAX. Initially, the desired standard is selected from the pop-up menu as in Figure 2.8 and the filter design is obtained by pressing the push button named *Multistandard Decimation Filter Design*. The filter details such as the required channel spacing for a selected standard, passband edge, stopband edge, input

sampling frequency, OSR, number of stages and type of filter used in each stage, decimation factors for each stage, and filter complexity are displayed on the GUI as in Figure 2.9.

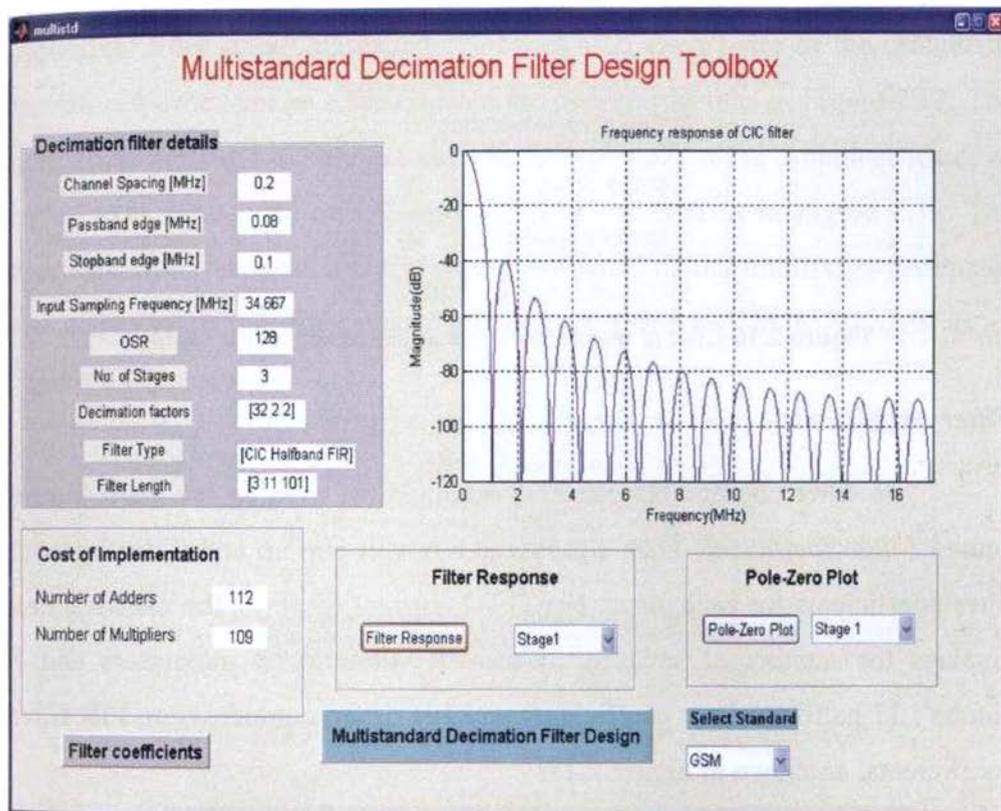


Figure 2.7 GUI for Multi-standard Decimation Filter Design Toolbox



Figure 2.8 Pop-up menu for standard selection

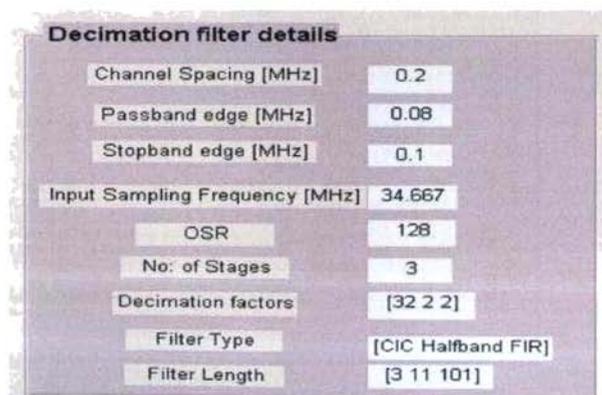
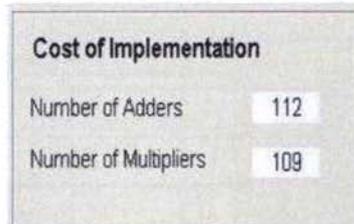


Figure 2.9 Decimation filter details for GSM

Cost of implementation

The cost of implementation of the multistage decimator is displayed as in Figure 2.10, in terms of total number of adders and multipliers required for all stages.



Cost of Implementation	
Number of Adders	112
Number of Multipliers	109

Figure 2.10 Cost of decimation filter implementation for GSM

Filter coefficients

The filter coefficients can be visualized by pressing the push button named *Filter coefficient*. Then a message box will pop up and it displays the filter coefficients for each stage. For GSM (current display), the message box displays the number of sections of the CIC filter as '3 integrators and 3 combs', 11 halfband filter coefficients and 101 droop compensation FIR filter coefficients, as shown in Figure 2.11.



Figure 2.11 Message box displaying filter coefficients for GSM

Filter response

The push button named *Filter response* is used to display the magnitude response. The desired response such as the magnitude response for individual filter stages, cascaded responses after each stage or the multistage overall response, can be selected from the pop-up menu as in Figure 2.12. The magnitude response of individual filter is displayed on the graphical window, called axes, embedded on the front panel of the GUI as in Figure 2.13. The cascaded filter response and the overall response of the multistage decimator are displayed using filter visualization tool (FVTool) in MATLAB as in Figure 2.14.

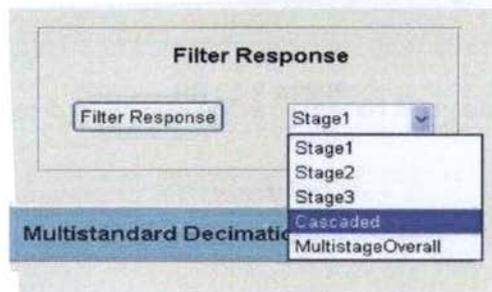


Figure 2.12 Pop-up menu for magnitude response selection

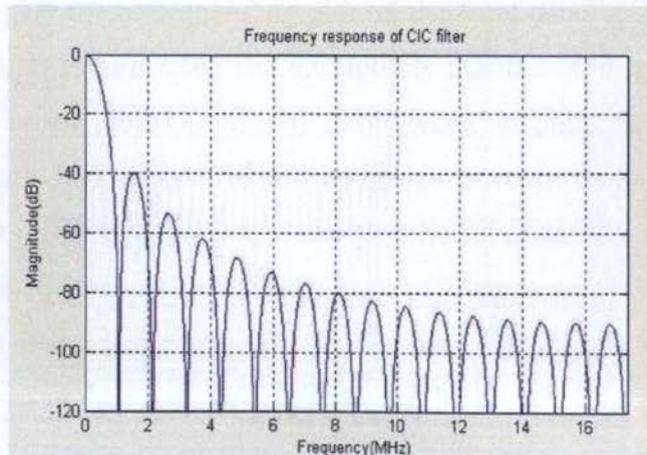


Figure 2.13 (a)

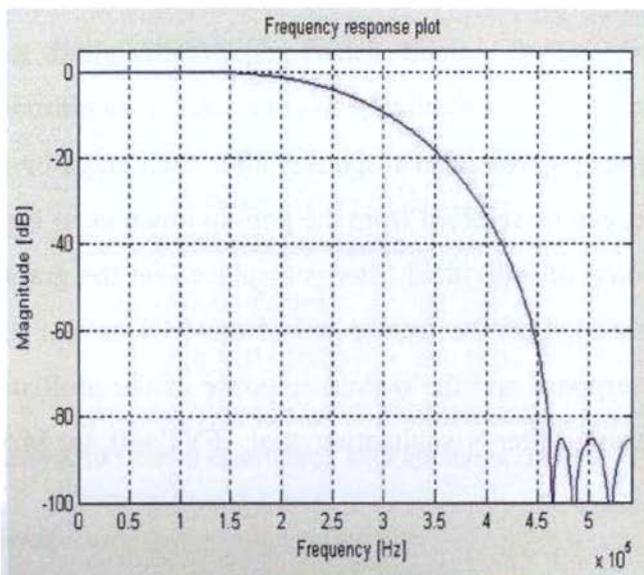


Figure 2.13 (b)

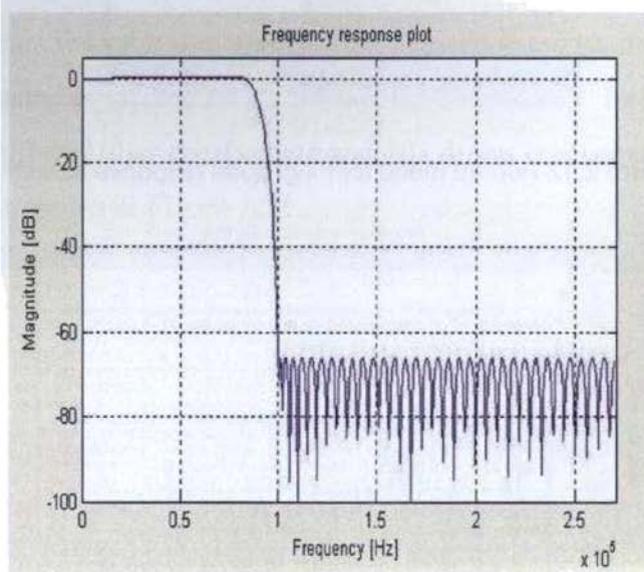


Figure 2.13 (c)

Figure 2.13 Individual Filter response for GSM displayed on front panel of GUI

(a) CIC filter (b) Halfband filter (c) FIR filter

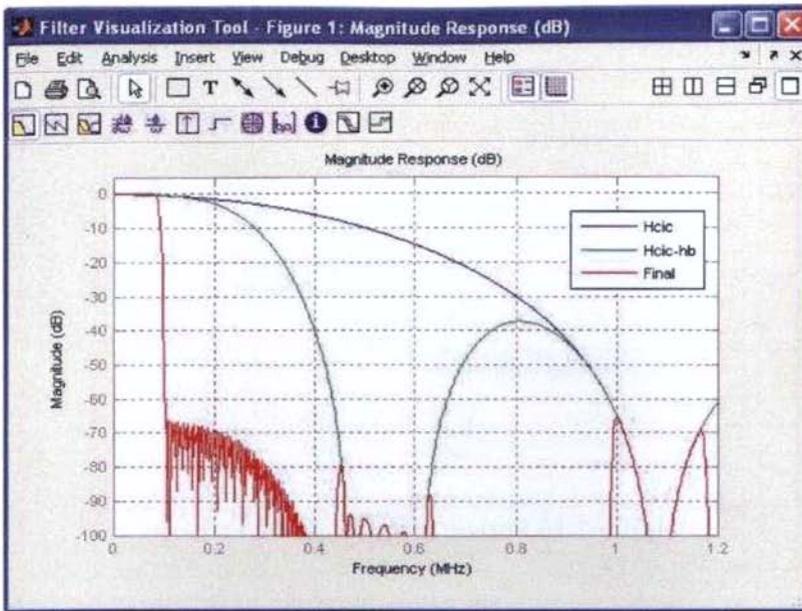


Figure 2.14 Display of the cascaded filter response

Pole-Zero plots

To get the pole-zero plot of individual filter, each stage can be selected from a pop-up menu as in Figure 2.15. The push button named *Pole-Zero Plot* is used to display the corresponding plot on the front panel graphical window of the GUI as in Figure 2.16. The multiplicity number of each pole and zero are indicated in the plot. The filter is stable when the poles lie inside the unit circle in z -plane. FIR filters are stable by design since the transfer functions do not have denominator polynomials, and thus no feedback to cause instability. CIC filters are stable even with the presence of integrators, as the poles on unit circle due to the denominator of transfer function are cancelled by equal number of zeros at the same position produced by the numerator.

The multistage decimation filter implementation results obtained for each of the six standards using the new toolbox are given in Section 6.1.

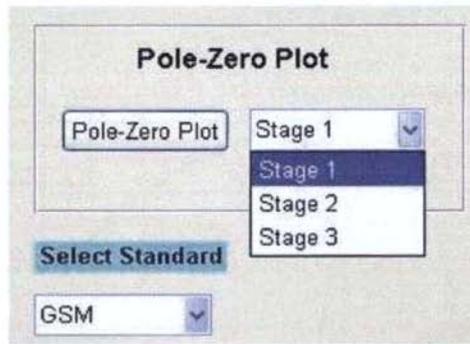


Figure 2.15 Pop-up menu for pole-zero plots

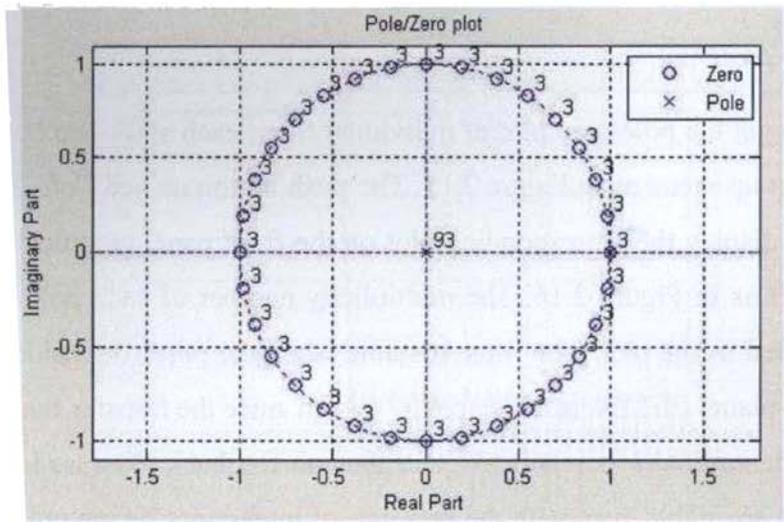


Figure 2.16 (a)

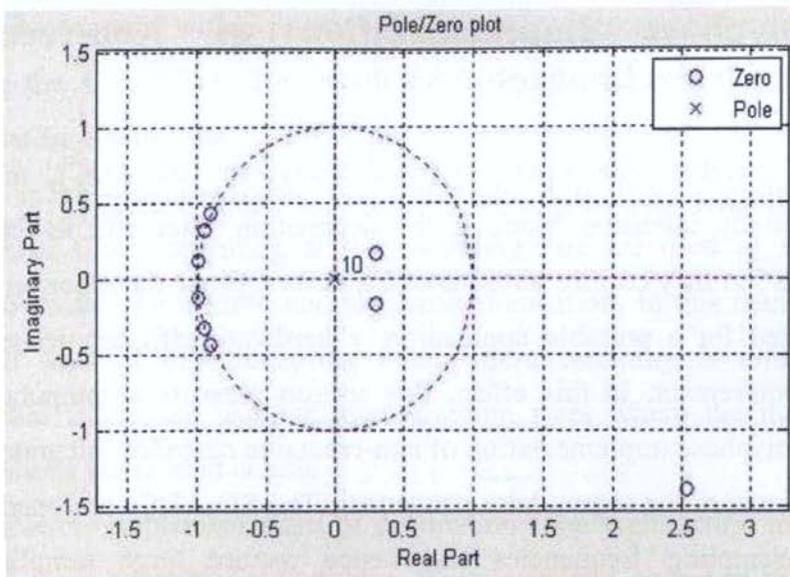


Figure 2.16 (b)

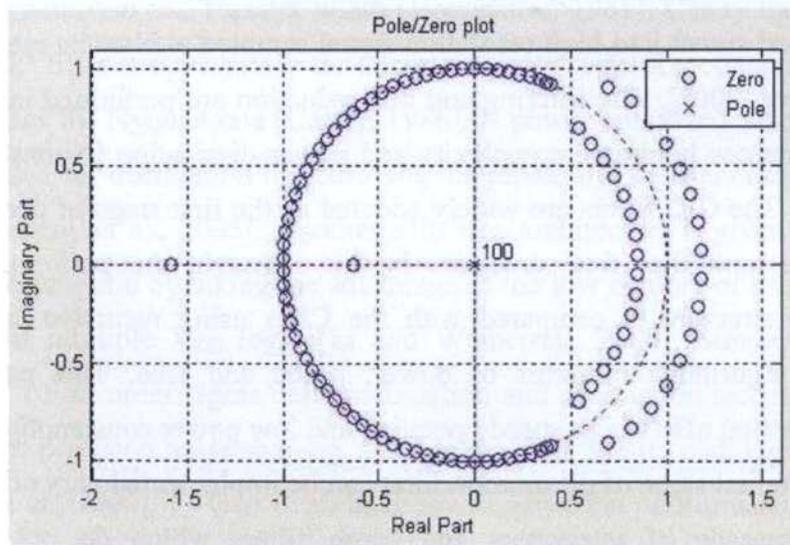


Figure 2.16 (c)

Figure 2.16 Pole-Zero plot of individual filters for GSM (a) CIC filter (b) Halfband filter (c) FIR filter

2.6 Polyphase Implementation of Non-recursive Comb Decimators

In a sigma-delta analog-to-digital converter (SD-ADC), the most computationally intensive block is the decimation filter and its hardware implementation may require millions of transistors. Since these converters are now targeted for a portable application, a hardware efficient design is an implicit requirement. In this effect, this section presents a computationally efficient polyphase implementation of non-recursive cascaded integrator comb (CIC) decimators for sigma-delta converters. The SD-ADCs are operating at high oversampling frequencies and hence require large sampling rate conversions. The digital decimator part consists of a lowpass filter and a downsampler that is responsible for transforming the low resolution oversampled signal into high resolution signal sampled at Nyquist rate [Allen and Holberg, 2002]. The filtering and rate reduction are performed in several stages to reduce hardware complexity and power dissipation [Norsworthy et al., 1997]. The CIC filters are widely adopted as the first stage of decimation due to its multiplier free structure. In this research, the performance of polyphase structure is compared with the CICs using recursive and non-recursive algorithms in terms of power, speed and area. This polyphase implementation offers high speed operation and low power consumption.

The first stage of decimation filter can be implemented very efficiently using a cascade of integrators and comb filters which do not require multiplication or coefficient storage. The remaining filtering is performed either in single stage or in two stages with more complex FIR or IIR filters according to the requirements. The amount of passband aliasing or imaging

error can be brought within prescribed bounds by increasing the number of stages in the CIC filter. The width of the passband and the frequency characteristics outside the passband are severely limited. So, CIC filters are used to make the transition between high and low sampling rates. Conventional filters operating at low sampling rate are used to attain the required transition bandwidth and stopband attenuation. In this manner, CIC filters are used at high sampling rates where economy is critical, and conventional filters are used at low sampling rates where the number of multiplications per second is less.

Different implementations of decimation filter architecture for sigma-delta ADCs are available in literature. Hogenauer has described the design procedures for decimation and interpolation CIC filters with emphasis on frequency response and register width [Hogenauer, 1981]. Candy has proved that a Sinc^k filter is appropriate for decimating sigma-delta modulation down to four times the Nyquist rate [Candy, 1986]. A power optimized Sinc^4 filter is implemented for decimation by removing the pipelining registers between the adders [Gursoy et al., 2005]. Another FIR-Sinc architecture is given for low-power consumption by taking the advantage of the low number of bits at input and use of multiple V_{DD} logic [Li and Wetherrell, 2000]. Simon Foo has presented a fifth order sigma delta modulation and decimation technique with a very high precision noise shaping, suitable for high fidelity audio application [S. Foo et al., 2004]. F Gao et al. have investigated the performance of non-recursive algorithm for comb decimators [Gao et al., 1999]. The comparison results with recursive CIC structure show that the non-recursive implementation provides reduced power consumption and increased circuit speed. Laddomada has performed performance comparison of various comb-

based decimation filter schemes for sigma-delta ADCs. The use of a combination of sharpened filter cells and modified-comb cells which diminishes the filter passband droop and increases the quantization noise rejection is presented [Laddomada, 2007].

To reduce power consumption in a circuit either the clock rate or the operating voltage has to be decreased. But sigma-delta ADCs utilize oversampling at high clock rates, and hence power consumption will increase. Lowering the operating voltage increases the circuit delay that will put a bound on operating frequency. One solution to this problem is to use parallel processing. Polyphase decomposition has been traditionally used to implement parallel structures in digital signal processing. Yang proposed a polyphase CIC implementation for high speed operation [Yang and Snelgrove, 1996], but the complete rate reduction is achieved by using another CIC which is again a recursive structure.

2.6.1 Classical Recursive CIC Filter

Hogenauer devised a flexible, multiplier free recursive filter suitable for hardware implementation that can handle large sampling rate changes as detailed in Section 2.3.1. The major problems encountered with the Hogenauer CIC filter include the following. The first problem is that the register widths can become large for large rate change factors. The register growth is considered in filter design process to ensure that no data are lost due to register overflow. The maximum register growth G_{max} from the first stage up to and including the last stage is approximated as in (2.8), where R is the rate change factor, M is the differential delay and k is the number of stages of the CIC filter.

$$G_{\max} = (RM)^k \quad (2.8)$$

If the number of bits in the input data stream is B_{in} , then the register growth can be used to calculate B_{max} , the most significant bit at the filter output. It is given by $B_{max} = \lceil k \log_2 RM + B_{in} - 1 \rceil$, where the least significant bit of the input register is considered to be bit number 'zero'. Since the first ' k ' stages of the filter are integrators with unity feedback, the integrator outputs grow without bound for uncorrelated input data. It can be concluded that B_{max} is the most significant bit not only for the integrators, but also for the combs that follow. B_{max} is large for many practical cases, and can result in large register widths. So, truncation or rounding has to be used at each filter stage to reduce the register widths.

Second problem with the recursive CIC filter is the higher power consumption since the integrator stage works at the highest oversampling rate with a large internal word length. As the decimation ratio and filter order are increasing, power consumption increases significantly. Third problem is that the circuit speed will be limited by the large word length and recursive loop of the integrator stage.

2.6.2 Non-Recursive CIC filter

The non-recursive CIC filter reduces power dissipation and increases speed of operation by avoiding the IIR part in the recursive structure [Gao et al., 1999]. The difference between the non-recursive and recursive algorithms is that they use different VLSI structures to implement the transfer function in (2.5). Taking differential delay $M = 1$ and rate change factor, $R = 2^N$, the transfer function can be rewritten as

$$\begin{aligned}
 H(z) &= \left(\frac{1 - z^{-R}}{1 - z^{-1}} \right)^k = \left(\sum_{i=0}^{R-1} z^{-i} \right)^k \\
 &= \left(\sum_{i=0}^{2^{N-1}-1} z^{-i} \right)^k = \prod_{i=0}^{N-1} (1 + z^{-2^i})^k \quad (2.9)
 \end{aligned}$$

The non-recursive CIC architecture is shown in Figure 2.17. Every stage is a FIR filter but operates at different sampling rate. After each stage, the sampling rate is reduced by a factor of 2. The output from a sigma-delta modulator of word length B_{in} is given as input to the filter. The word length increases through every stage by ‘ k ’ bits, but the sampling rate decreases through every stage by a factor of 2 starting from the oversampling rate f_s . Thus the word length is short when the sampling rate is high, and when the word length increases the sampling rate decreases. In the recursive algorithm, the IIR part has to operate with the oversampling rate and has a word length of $\lceil k \log_2 R + B_{in} \rceil$ bits. In the non-recursive algorithm, the first stage works at the oversampling rate but has only a word length of $(B_{in} + k)$ bits. This helps to reduce the power consumption and to increase the maximum speed of operation for non-recursive decimator.

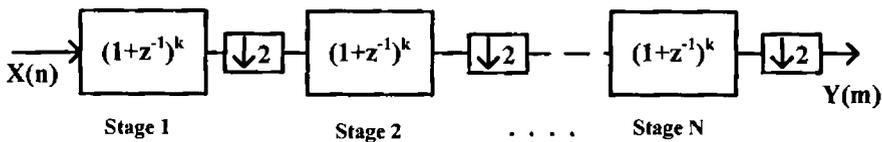


Figure 2.17 Non-recursive comb decimator

2.6.3 Polyphase Non-Recursive CIC Architecture

The average power consumption of a digital signal processing system is determined by the number of computations performed per sample, the word

length and the sampling frequency. Parallel processing through polyphase decomposition is an efficient way to achieve high speed and lower power consumption [Meyer-Baese, 2001]. In this research, polyphase decomposition is done for each FIR filter stage of the non-recursive decimator as shown in Figure 2.18. Here, decimation occurs at the input of each filter reducing the sampling frequency by a factor of 2. So the number of computations per sample is also reduced to half of that for non-recursive implementation leading to low power consumption. As in non-recursive structure, polyphase implementation is also not having any register overflow problems, and the word length of initial stages is limited to a few bits. Since the use of polyphase decomposition has reduced the operating frequency of the filters significantly at the last stages, the critical path is no longer a problem. So, the polyphase CIC filter can operate at a higher speed.

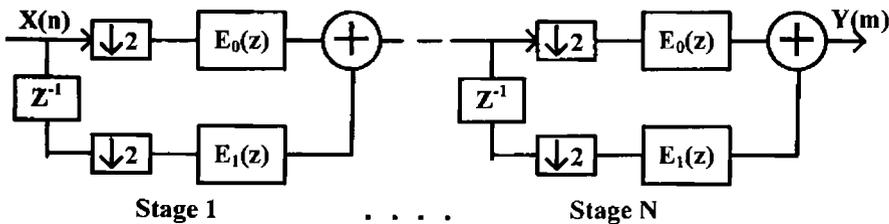


Figure 2.18 Polyphase realization of non-recursive comb decimator

In general, an L -branch polyphase decomposition of the transfer function of FIR filter of order N is of the form

$$\begin{aligned}
 H(z) &= \sum_{k=0}^N h(k)z^{-k} \\
 &= \sum_{m=0}^{L-1} z^{-m} E_m(z^L)
 \end{aligned} \tag{2.10}$$

where

$$E_m(z) = \sum_{n=0}^{\lfloor (N+1)/L \rfloor} h(Ln+m)z^{-n}, 0 \leq m \leq L-1, \text{ with } h(n) = 0, \text{ for } n > N \quad (2.11)$$

Performing two-branch polyphase decomposition of each FIR block of the non-recursive comb decimator, the transfer function in (2.9) can be rewritten as

$$H(z) = \sum_{m=0}^1 z^{-m} E_m(z^2) \quad (2.12)$$

Consider the comb decimator with decimation factor $R = 64$, and order $k = 4$, so that the polyphase filter equations are:

$$E_0(z) = h(0) + h(2)z^{-1} + h(4)z^{-2} = 1 + 6z^{-1} + z^{-2} \text{ and ,}$$

$$E_1(z) = h(1) + h(3)z^{-1} = 4 + 4z^{-1}.$$

The corresponding polyphase CIC filter architecture is shown in Figure 2.19. The multipliers in the polyphase filter are implemented using shift and add method, which require only adder circuit as shifting can be achieved by properly routing the input bits. For example, multiplication of input 'x' by 6 is carried out by adding '4x' and '2x'. So, the operation is only an addition as the numbers '4x' and '2x' are easily obtained by inserting zeros at least significant bit positions.

The simulation results obtained for a 4th order CIC filter using the three different architectures are presented in Section 6.2. Performance evaluation of three architectures for different decimation factors as $R = 64, 128$ and 256 are also given.

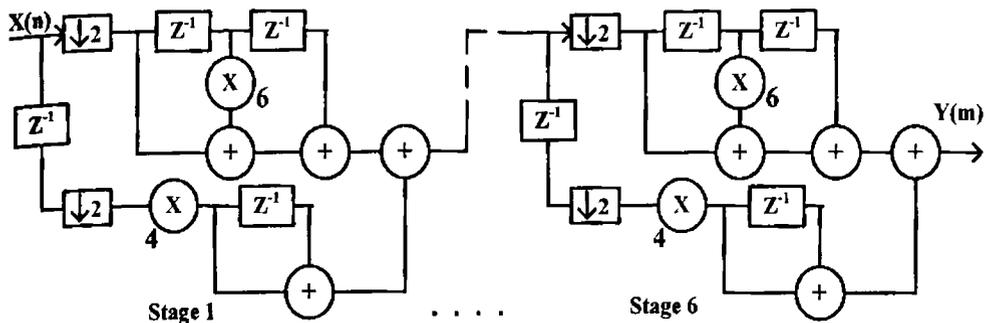


Figure 2.19 Polyphase realization of non-recursive comb decimator for $R = 64$, $k = 4$

2.7 Summary

A multistage decimation filter design toolbox is developed for six popular wireless communication standards, namely GSM, WCDMA, WLANa, WLANb, WLANg and WiMAX. The toolbox allows the user or design engineer to perform a quick design and analysis of decimation filters for different standards without doing extensive calculation of the underlying methods. The tool provides the user with all necessary details of decimation filter designed for the selected standard including filter coefficients, frequency response, pole-zero plot, cost of implementation etc. The implementation of multistage decimation filter reduces the hardware and computational effort while meeting the standard requirements. A computationally efficient polyphase implementation of non-recursive CIC filter is presented. The polyphase CIC filter has higher speed of operation, lower power consumption and more area requirement. So, the designer can trade and select the CIC architecture based on the overall system requirements. The implementation results obtained for the multistage decimators using the toolbox and the polyphase non-recursive comb decimator are presented in Section 6.1 and 6.2 respectively.

Chapter 3

RNS based Programmable Multi-mode Decimation Filters

The growth of wireless communication for global roaming has led to the coexistence of multiple standards in a single transceiver. A programmable decimator is required in such transceivers to adapt to the requirements of different standards. In Residue Number System (RNS) arithmetic operations are performed in parallel without any carry propagation between residue digits. This leads to significant speed up of multiply and accumulate (MAC) operations in RNS domain. This has motivated the development of RNS based multistage programmable decimation filter.

In this chapter, the performance of FIR filter operating in RNS domain is compared against the traditional implementation in terms of area requirement and delay. Dual-mode decimators programmable for WCDMA/WiMAX and for WCDMA/WLAN standards are implemented. Here, modulo multiplication is realized by look up table approach. The RNS multipliers are also implemented for increased programmability by index addition, utilizing the arithmetic benefits associated with Galois field. A reconfigurable three-stage decimator for WCDMA/WLAN mode of operation using index calculus multipliers is implemented.

3.1 Residue Number System

The digital signal processing (DSP) hardware architectures are mostly based on the two's complement fixed point number system. While two's complement is easy to use, it suffers from the drawback of speed limitations for arithmetic operations on long word lengths due to carry propagate delays. Also, there is a quadratic growth of die area with operand word length. Most of the real time DSP algorithms are based on intensive multiplication and addition operations. In real time systems digital convolution, finite impulse response (FIR) filtering, discrete Fourier transforms (DFT) and similar computations are performed at high sample rate on long word lengths. The carry propagating multipliers and adders in binary number system become the bottle-neck for high sampling rate computations.

In this research a nonconventional number system called Residue Number System (RNS) is chosen to eliminate the long carry propagate delays involved in arithmetic computations. It simplifies long arithmetic computations by splitting the operands into smaller residue numbers and performing parallel independent modulo operations on them [Soderstrand et al., 1986], [Szabo and Tanaka, 1967]. The use of Residue Number System (RNS) draws a great deal of interest in computationally intensive DSP applications as it significantly speeds up multiply and accumulate (MAC) operations [Parhami, 2000].

3.1.1 RNS Basics

RNS is defined by a set of ' r ' relatively prime integers (m_1, m_2, \dots, m_r) which are called the moduli. Any integer ' X ' is represented in RNS as a set of ' r ' residues (x_1, x_2, \dots, x_r), where each x_i is a nonnegative integer satisfying the

relationship $X = m_i \cdot q_i + x_i$ for $i = 1, \dots, r$ and q_i is the largest integer with $0 \leq x_i \leq (m_i - 1)$. Hence x_i is the residue of X modulo m_i , denoted by $X \bmod m_i$ or $|X|_{m_i}$. The product of the ‘ r ’ relatively prime moduli gives the number of different representations possible in RNS, called the dynamic range M .

$$\text{Dynamic range, } M = \prod_{i=1}^r m_i \quad (3.1)$$

There exists a unique representation for integers in the range $[0, M)$ where ‘[’ indicates 0 is included in the range and ‘)’ indicates M is excluded from the range. Negative numbers can be represented by partitioning the dynamic range into two sets. For a signed number system, the dynamic range is $[-(M-1)/2, (M-1)/2]$ for an odd M and $[-M/2, (M/2)-1]$ for an even M . In other words, a common assignment is that all numbers X , $0 \leq X \leq (M-1)/2$ for odd M and $0 \leq X \leq (M/2)-1$ for even M , are considered positive and the rest are considered as negative.

3.1.2 RNS Arithmetic

In RNS, arithmetic operations are computed by the formula:

$$(x_1, x_2, \dots, x_r) \ominus (y_1, y_2, \dots, y_r) = (z_1, z_2, \dots, z_r) \quad (3.2)$$

where $z_i = |x_i \ominus y_i|_{m_i}$ and \ominus denotes one of the modulo operations of addition, subtraction or multiplication. The i^{th} residue of the result z_i depends only on the i^{th} residues of the operands x_i and y_i , corresponding to the modulus m_i , and is independent of the remaining residues. Thus arithmetic operations are performed on smaller residues instead of the large number. Since the operations in each modulo channel are independent of the others, these can be performed in parallel without any carry propagation among the residue digits.

This leads to significant speed-up of the whole operation. Also, an error occurring in one modulo channel is not propagated to any other channel, thereby providing fault isolation.

3.1.3 Forward and Reverse Conversions

The RNS offers very high speed parallel arithmetic processing. However, since most of the digital systems use conventional binary number system, the conversions between binary and RNS representations are required. Initially the number is converted from binary to residue by performing modulo operations with respect to each modulus in the moduli set. The process of translating a binary integer X , in the range $[0, M)$ to the residue representation (x_1, x_2, \dots, x_r) with respect to a relatively prime moduli set (m_1, m_2, \dots, m_r) is called forward conversion. The integer X is represented in binary as:

$$X = \sum_{j=0}^{n-1} x_j 2^j \quad \text{where } x_j \in \{0,1\} \quad (3.3)$$

The corresponding RNS representation is obtained as:

$$|X|_{m_i} = \left| \sum_{j=0}^{n-1} x_j 2^j \right|_{m_i} \quad \text{for } i = 1, \dots, r \quad (3.4)$$

To avoid time consuming divisions the values $|2^j|_{m_i}$ for each i and j are precomputed. Then $|X|_{m_i}$ are calculated by modulo m_i addition of all the values corresponding to the individual bits x_j which are one's in the binary representation of X . After forward conversion, arithmetic operations are performed in parallel independent modulo channels.

Getting back to the weighted representation of ‘ X ’ from a given residue representation is referred to as reverse conversion. This involves finding the solutions for a set of simultaneous congruences:

$$X \equiv x_1 \pmod{m_1}, \quad X \equiv x_2 \pmod{m_2}, \quad \dots, \quad X \equiv x_r \pmod{m_r} \quad (3.5)$$

The reverse conversion can be done using Mixed Radix Conversion (MRC) and Chinese Remainder Theorem (CRT). MRC is usually used for sign determination, magnitude comparison and overflow detection while CRT is more adapted for generation of binary number directly from its residue. CRT is based on the general formula:

$$X = \left| \sum_{i=1}^r a_i x_i \hat{M}_i \right|_M, \quad \text{where } \hat{M}_i = \frac{M}{m_i} \quad \text{and} \quad a_i = \left| \frac{1}{\hat{M}_i} \right|_{m_i} \quad (3.6)$$

3.1.4 Choice of RNS Moduli

The set of moduli chosen for RNS affects both the representational efficiency and the complexity of arithmetic algorithms. The strategy for selecting RNS moduli is to choose the largest modulus with the smallest possible number of bits for maximizing the speed of RNS arithmetic. Since the magnitude of the largest modulus decides the speed of arithmetic operations, all the remaining moduli can be chosen so that they are comparable with the largest one. The moduli of the form 2^m , $(2^m - 1)$ and $(2^m + 1)$ have some interesting properties that lead to easy and efficient implementation of forward and reverse converters [Parhami, 2000], [Radhakrishnan et al., 1999], [Mohan and Premkumar, 2007]. These are called low cost moduli. The optimal choice is dependent on both the application and the target implementation technology.

3.2 FIR Digital Filter Design: RNS Versus Traditional

This research is motivated by the importance of an efficient filter implementation for digital signal processing. Finite Impulse Response (FIR) digital filters have attracted a great deal of interest because they are inherently stable structures that are much less sensitive to quantization errors than filters of the recursive type. The major disadvantage of an FIR design is that usually a large number of nonzero terms are required in the impulse response in order to adequately control the frequency response of the filter. This results in large number of multiplications and additions that must be executed during a short sample interval which in turn will seriously limit the speed of the filter. An FIR filter is described by (3.7), where $X(n)$ is the input to the filter, $H(k)$ represents the filter coefficients, N is the order of the filter and $Y(n)$ is the output from the filter.

$$Y(n) = \sum_{k=0}^N H(k)X(n-k) \quad (3.7)$$

For very large N , filters implemented in the traditional binary number system suffer from the disadvantages of the carry propagation delay in binary adders and multipliers. In RNS, a large integer is broken into smaller residues which are independent of each other, and each digit is processed in parallel channels without any carry propagation from one to another [Soderstrand et al., 1986]. This leads to significant speed-up of MAC operations which in turn results in high data rate for RNS based FIR filters.

Several researchers have proposed various applications of RNS for Digital Signal Processing. RNS based digital signal processors are reported in literatures. Claudio et al. presented combinatorial or look up table approaches for RNS tailored to small designs or special applications, while pseudo-RNS

approach remains competitive also for complex systems [Claudio et al., 1995]. Ramirez et al. explored RNS for implementation of fast digital signal processors with the design of an RNS-based single instruction multiple data (SIMD) RISC processor [Ramirez et al., 2002]. A number of FIR filter designs based on residue arithmetic are available in the literature. Soderstrand presented the implementation of high speed recursive digital filters using residue arithmetic properties and high speed multiplication by a fraction using a table look up [Soderstrand, 1977]. Another technique is presented [Jenkins and Leon, 1977] for implementing a FIR digital filter in residue number system with a modified hardware implementation of the Chinese Remainder Theorem for translation of residue coded outputs into natural numbers. Soderstrand explored the feasibility of combining multiple valued logic (MVL) with RNS arithmetic, and developed a detailed block diagram representation of an MVL-RNS digital filter independent of the choice of levels in MVL or moduli in RNS [Soderstrand and Escott, 1986]. Structurally passive digital filters are realized using only RNS based rotators and delays [Cardarilli et al., 1988]. Low power realization of RNS based FIR filter is presented in [Mahesh and Mehendale, 2000] with coefficient ordering and coefficient encoding techniques. Cardarilli et al. has implemented a polyphase filter bank in the Quadratic Residue Number System (QRNS) and compared in terms of area and power dissipation to the implementation of a polyphase filter bank in the traditional two's complement system [Cardarilli et al., 2004]. Also, they presented the implementation of RNS FIR filters with low static and dynamic power dissipation using standard cell libraries with dual threshold transistors [Cardarilli et al., 2005].

Most of the work in the current literature addresses only the implementation in RNS domain and none makes a fair comparison of RNS

based FIR filter with a traditional implementation of FIR filter. Hardware implementations and graphical analysis of speed and area for FIR filters implemented in RNS versus traditional binary number system are presented.

3.2.1 FIR Filter Architecture

The architecture of a FIR filter based on traditional binary number system is shown in Figure 3.1. As the input samples are multiplied by the filter coefficient, the advantage of multiplication by a constant is considered while designing the multipliers. Carry Save Adder (CSA) tree based multiplier is used for fast multiplications and Carry Propagate Adder (CPA) is used for addition.

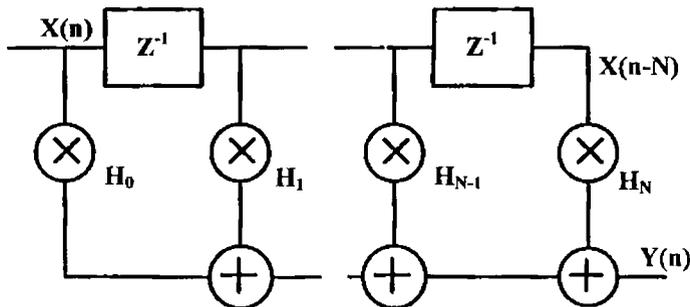


Figure 3.1 Traditional FIR filter architecture

For RNS based FIR filter, the first step is to choose appropriate moduli set that provides sufficient dynamic range for the filter. Let the moduli set be (m_1, m_2, \dots, m_r) . Then there will be ' r ' parallel filter channels as in Figure 3.2, which process the signals from forward converter. Finally, the reverse converter combines the signals from all the channels and puts the output signal back in binary form. A single channel corresponding to modulus ' m_i ' of such a filter is shown in Figure 3.3, where \otimes and \oplus represent modulo multiplication and modulo addition respectively. The filter coefficients are directly represented in residue form.

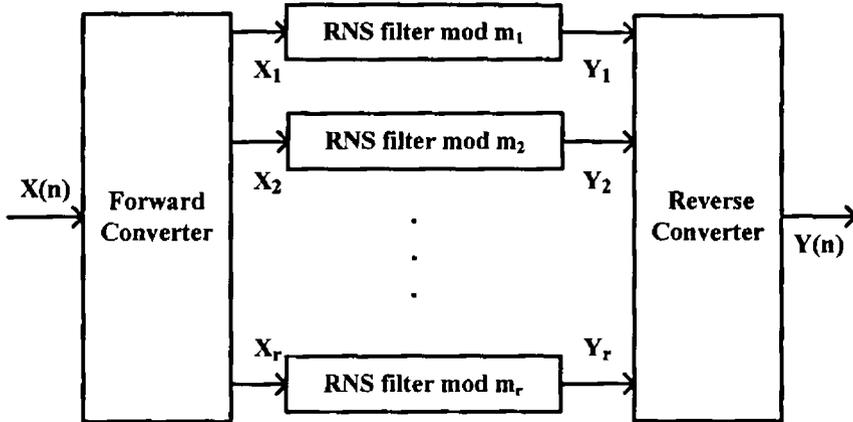


Figure 3.2 RNS implementation of FIR filter

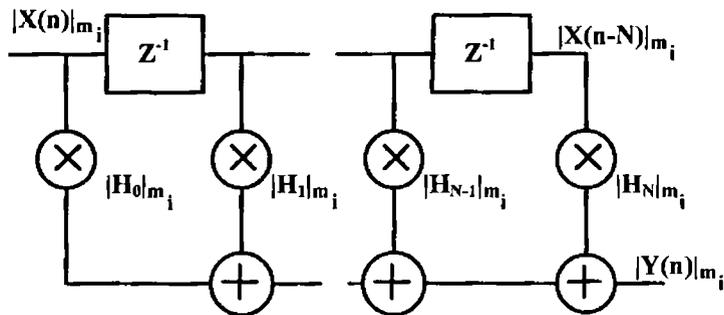


Figure 3.3 i^{th} modulo filter channel

3.2.1.1 Forward Converter

The forward conversion logic used to generate the residues [Preethy and Radhakrishnan, 1999] is shown in Figure 3.4. The input binary word is stored in a register of k -bits wide and partitioned into ' p ' partitions each of n -bits wide, where $n = \lceil \log_2 m_i \rceil$. Each partition of n -bits addresses a ROM of size $2^n \times n$ which is programmed to produce its residue with respect to m_i . The residues obtained simultaneously from the ROMs are added using a tree of modulo- m_i adders to get the final residue. Similar stages are used to generate all the residues with respect to each modulus. The modulus of the form 2^n does

not require extra logic as the least significant n -bits of the binary input directly represent the residue. Moduli of the form $(2^n - 1)$ are also desirable as it eliminates the requirement of ROMs for residue generation. The residue is obtained by performing modulo $(2^n - 1)$ addition of the n -bit partitions of binary input. The modulo $(2^n - 1)$ addition can be easily performed using a standard n -bit binary adder with end-around carry.

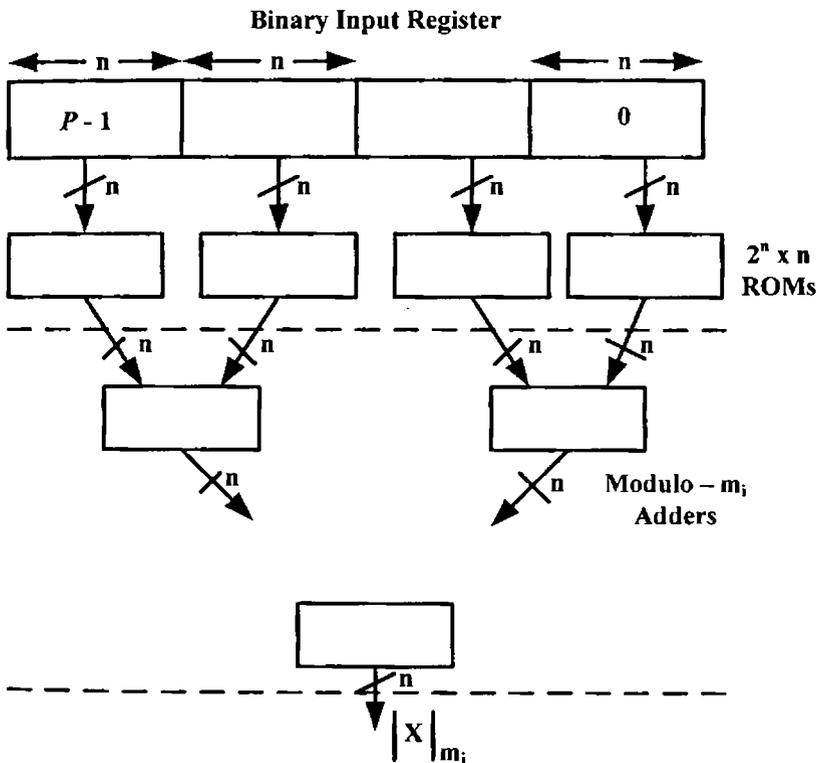


Figure 3.4 Forward conversion logic for $|X|_{m_i}$

3.2.1.2 Modulo Addition

The modulo addition with respect to modulus m_i of two numbers x and y belonging to $\{0, 1, \dots, m_i - 1\}$ is defined as:

$$(x + y) \bmod m_i = \begin{cases} x + y & \text{if } x + y < m_i \\ x + y - m_i & \text{if } x + y \geq m_i \end{cases} \quad (3.8)$$

The modulo adder circuit suitable for VLSI implementation is shown in Figure 3.5. This architecture requires only two carry propagate adders and a multiplexer [Beuchat, 2003]. The first adder performs normal binary addition of inputs x and y . The second adder helps to perform modulo correction by adding a correction factor of $(2^n - m_i)$ with the first adder output where $n = \lceil \log_2 m_i \rceil$.

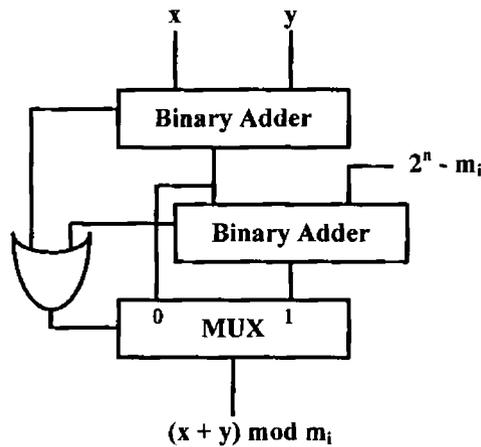


Figure 3.5 Modulo adder

3.2.1.3 Modulo Multiplication

Modulo multiplication is implemented with look up table (LUT) for faster multiplication. The filter coefficient is scaled by a suitable power of 2, 2^{17} in this case, to get RNS representation. One of the operands for multiplication in the MAC unit is the filter coefficient which is a constant value. Hence, the size of multiplier LUT is small as the operation is in the residue domain with one of the operands constant.

3.2.1.4 Reverse Converter

The reverse conversion is performed using Chinese Remainder Theorem (CRT). The CRT implementation is done by using ‘ r ’ ROMs to store the precomputed values $\left| a_i x_i \hat{M}_i \right|_M$ each indexed by x_i [Parhami and Huang, 1994]. This reduces CRT implementation to a summation of ‘ r ’ values, followed by modulo correction with ‘ M ’ using carry save adder stages and a final carry propagate adder. The overall hardware including the size of ROM stage used in the design is further reduced by selecting one of the moduli of the form 2^n , so that the least significant ‘ n ’ bits of the binary number are directly available [Radhakrishnan et al., 1999]. The scheme of reverse conversion is shown in Figure 3.6. If the RNS representation of integer X is (x_1, x_2, \dots, x_r) with respect to a relatively prime moduli set (m_1, m_2, \dots, m_r) with $m_r = 2^n$, then the expression for X using CRT is given by:

$$X = \left| \sum_{i=1}^r a_i x_i \hat{M}_i \right|_M \quad \text{where } \hat{M}_i = \frac{M}{m_i} \text{ and } a_i = \left| \frac{1}{\hat{M}_i} \right|_{m_i} \quad (3.9)$$

This is rewritten as: $X = \sum_{i=1}^r a_i x_i \hat{M}_i - KM$, where K is an integer

$$X + KM = \sum_{i=1}^r a_i x_i \hat{M}_i \quad (3.10)$$

On dividing both sides of (3.10) by m_r ,

$$\frac{X}{m_r} + K \frac{M}{m_r} = \frac{1}{m_r} \sum_{i=1}^r a_i x_i \hat{M}_i \quad (3.11)$$

Taking integer values on both sides of (3.11),

$$\left\lfloor \frac{X}{m_r} \right\rfloor + K \hat{M}_r = \frac{1}{m_r} \sum_{i=1}^{r-1} a_i x_i \hat{M}_i + \left\lfloor \frac{1}{m_r} a_r x_r \hat{M}_r \right\rfloor \quad (3.12)$$

Now on taking modulo operation with respect to \hat{M}_r , on both sides of (3.12), the equation becomes:

$$\left\lfloor \frac{X}{m_r} \right\rfloor = \left\lfloor \frac{1}{m_r} \sum_{i=1}^{r-1} a_i x_i \hat{M}_i + \left\lfloor \frac{1}{m_r} a_r x_r \hat{M}_r \right\rfloor \right\rfloor_{\hat{M}_r} \quad (3.13)$$

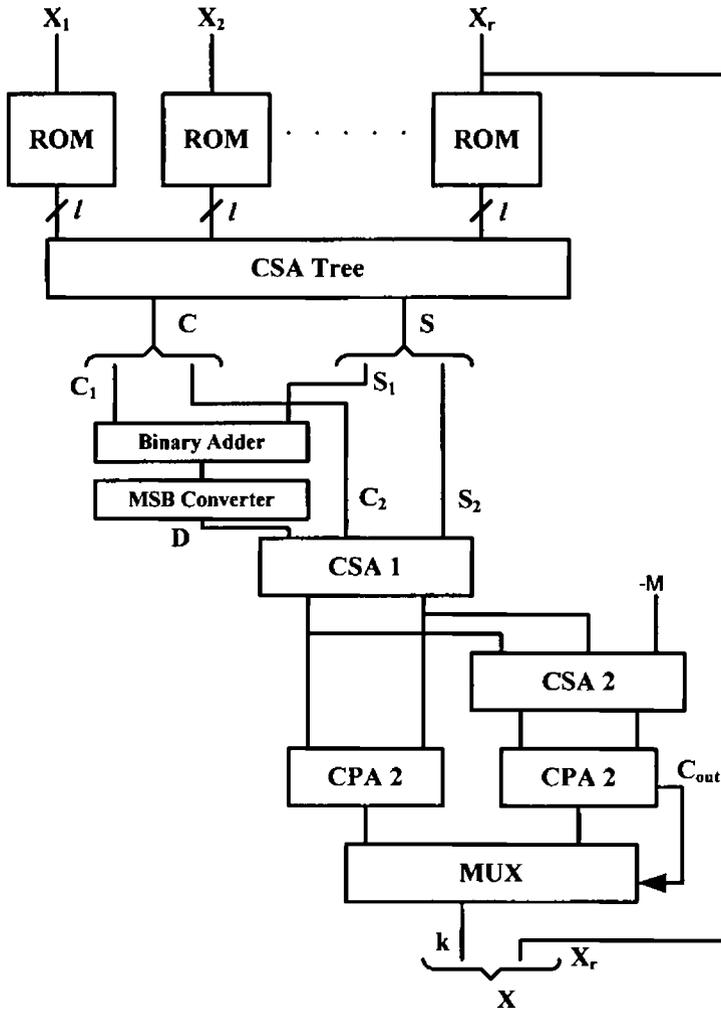


Figure 3.6 Hardware efficient reverse converter

The integer X can be represented as:

$$X = km_r + x_r \quad (3.14)$$

where k is an integer and $m_r = 2^n$. The value of

$\left\lfloor \frac{1}{m_r} \sum_{i=1}^{r-1} a_i x_i \hat{M}_i + \left\lfloor \frac{1}{m_r} a_r x_r \hat{M}_r \right\rfloor \right\rfloor_{\hat{M}_r}$ corresponds to the integer value k . The

values of $\left\lfloor \frac{1}{m_r} a_i x_i \hat{M}_i \right\rfloor_{\hat{M}_r}$ are precomputed and stored in ROMs of $l = \lceil \log_2(M-1) \rceil - n$ bits wide. The ROM outputs are added and modulo corrected with respect to \hat{M}_r using CSA tree as well as final CPA. Once k is calculated, X is obtained by concatenating k and x_r .

Performance analysis of FIR filters with different filter orders operating in RNS and traditional binary number system are given in Section 6.3 in terms of critical path delay and area requirement.

3.3 RNS based Dual-mode Decimation Filters

The recent trends envisage multi-standard architectures as a promising solution for the future wireless transceivers to attain higher system capacities and data rates. The computationally intensive decimation filter plays an important role in channel selection for multi-mode systems. An efficient reconfigurable implementation is a key to achieve low power consumption. To this end, the design considerations and implementation results for dual-mode Residue Number System (RNS) based decimation filters are presented. Two different decimation filters are designed: one programmable for WCDMA/802.16e standards, and the other programmable for WCDMA/802.11a standards. Decimation is done using multistage, multirate FIR filters. These FIR filters implemented in RNS domain offers reduction in chip area and high speed due to its carry free operation on smaller residues in parallel channels [Shahana et al., 2007]. Also, the FIR filters exhibit programmability to a selected standard by reconfiguring the hardware architecture. In each mode, the unused parts of the overall architecture is powered down and bypassed to attain power saving.

A programmable decimation filter is required in multi-mode transceiver as the channel bandwidth, sampling rates, and interference profile are different for each standard. Some of the previous works on programmable decimation filters include the following. Li et al. presented a digital IF down-converter with quadrature sampling based on polyphase filter, high rate CIC filter and interpolation filters, and compatible with WCDMA (Wideband Code Division Multiple Access) and EDGE (Enhanced Data rates for GSM Environment) [Li et al., 2004]. Sheikh and Masud introduced a decimation filter structure based on CIC filters and polynomial interpolation filters to perform fractional sample rate conversion [Sheikh and Masud, 2007]. Multi-rate digital filters and fractional frequency conversion techniques are adopted to implement the front end of a dual-mode receiver for WCDMA/cdma2000 [Kim and Lee, 2004]. Ramirez et al. presented a fast RNS field programmable logic (FPL) based communication receiver design and implementation [Ramirez et al., 2002].

RNS based programmable decimation filters for dual-mode WCDMA/WiMAX and WCDMA/WLANa receivers are designed and implemented as a part of this research. This technique fundamentally differs from the implementation presented by Ramirez in two critical issues [Ramirez et al., 2002]. Firstly, this technique addresses the problem of multi-standard decimation filtering. Secondly, and more importantly, since the implementation is multi-rate, the subsequent filters operate at lower sampling rates. Thus it reduces power consumption compared to the single stage implementation presented by Ramirez. Furthermore, as the front end of this architecture is a sigma-delta ADC, the forward converter that takes around 10% area of traditional RNS filter is eliminated by suitably selecting the moduli set.

3.3.1 Receiver Architecture

Direct conversion homodyne receiver architecture is considered for this research, and is detailed in Section 2.2. The theoretical dynamic range has been used in conjunction with the implementation attributes to choose the optimal topology for different RF standards. The dynamic range DR of a $\Sigma\Delta$ modulator is given by

$$DR = \frac{3}{2} \frac{2L+1}{\pi^{2L}} M^{2L+1} (2^B - 1)^2 \quad (3.15)$$

where L is the order of the modulator, M is the oversampling ratio (OSR), and B is the number of bits of the quantizer. The dynamic range requirements are chosen as more than 75dB for WCDMA, and more than 50dB for WiMAX and WLAN standards. In order to meet the DR requirements demanded by the WCDMA standard, a fourth order cascaded MASH topology is sufficient with a single bit quantizer and an OSR of 16. If WiMAX or WLAN becomes the target standard, a fifth order topology is a good compromise to achieve the required DR with a 4-bit quantizer and an OSR of 8. The sigma-delta modulator can be made programmable, and all the blocks are switched to operation only in the WiMAX or WLAN mode. This results in power saving when the receiver is operating in the other mode. The sigma-delta modulator is followed by a programmable decimation filter operating in the digital domain.

3.3.2 Dual-mode Decimation Filter for WCDMA/WiMAX

The specifications for WCDMA and WiMAX standards and the corresponding decimation filter design parameters are given in Table 3.1. The oversampling ratio (OSR) for each standard is selected so as to get the required dynamic range for the sigma-delta modulator of a particular order and

number of quantizer bits. In order to set the parameters for decimation filter, the receiver specifications and the blocking and interference profiles for each standard are considered. The decimation filter is designed to minimize undesired signals in the desired band of operation. The output carrier to noise (C/N) ratio is calculated from the bit error rate (BER) of each standard and the modulation scheme used. The passband frequency edge is taken as 80% of the bandwidth. The passband ripples are chosen to minimize signal distortions in the signal band. The stopband attenuations are selected according to the interference profile and C/N ratio for each standard.

Table 3.1 Standard specification and decimation filter design parameters for WCDMA/WiMAX transceiver

Specification	WCDMA	WiMAX
Frequency range(GHz)	DL:2.11-2.17 UL:1.92-1.98	10 - 66
Channel Spacing	5 MHz	20 MHz
Data rate	3.84 Mchips/s	16.704 Msymbols/s
OSR	16	8
Input sampling frequency, F_s	61.44 MHz	133.632 MHz
Passband edge	2 MHz	8 MHz
Stopband edge	2.5 MHz	10 MHz
Offset frequency (MHz) : Interference magnitude(dBm)	5 : -63 10 : -56 12.5:-44	20 : -68 40 : -49
C/N ratio	7.2 dB	21 dB
Passband ripple	0.5 dB	0.5 dB
Stopband attenuation	55 dB	39 dB

3.3.2.1 Design Considerations

The multistage, multirate, programmable decimator employs FIR filters in all the stages. Each individual filter stage is designed within the

frequency band of interest in order to prevent aliasing in the overall decimation process. The passband frequency remains the same for all stages. The cut off frequency for the first stage can be less constraining than the overall filter specification. The final stage filter is responsible for attaining the overall filter requirements, while operating at the lower sampling rate. For stage 'i', the passband is from $0 \leq F \leq F_{pc}$, where F_{pc} is the passband edge. If F_{i-1} and F_i are the input and output sampling frequencies for stage 'i', and F_{sc} is the stopband edge, the transition band for stage 'i' is from $F_{pc} \leq F \leq F_i - F_{sc}$ and the stopband is from $(F_i - F_{sc}) \leq F \leq (F_{i-1} / 2)$.

The decimation factor is 16 for WCDMA and 8 for WiMAX. Decimation is done in 3 stages with decimation factors of 4, 2 and 2 for WCDMA, and in 2 stages with decimation factors of 4 and 2 for WiMAX. Remez Parks-McClellan optimal equiripple FIR filter is chosen for implementation. The filter orders obtained for WCDMA are 14, 11 and 37 for first, second and third stages respectively. For WiMAX filter orders are 15 and 31 respectively. The block diagram for the programmable decimation filter is shown in Figure 3.7, where N1, N2 and N3 denote the filter orders of each stage in each mode. The third filter will be operating only in WCDMA mode and will be bypassed in WiMAX mode using switch 'S'. The switch can be a transmission gate. The first 14 MAC units of first stage filter and first 11 MAC units of the second stage are shared for both modes. The unused hardware in each mode are bypassed to get power saving.

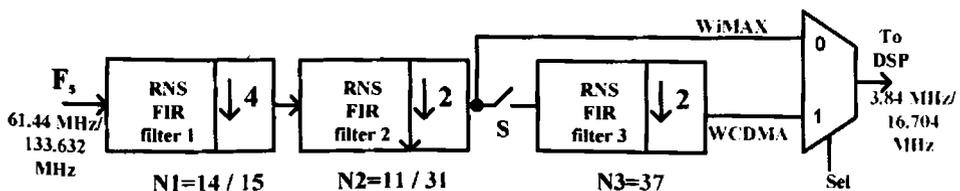


Figure 3.7 Dual-mode programmable decimation filter for WCDMA/WiMAX

All the FIR filters are implemented in RNS domain. The general block diagram for RNS based FIR filter is shown Figure 3.8. Let the moduli set be (m_1, m_2, \dots, m_r) . Then there will be ‘ r ’ parallel filter channels, which process the signals from the forward converter [Bernocchi et al., 2007]. The forward converter is shown in dotted lines as it is not used in the new design. Finally, the reverse converter combines the signals from all the channels and puts the output signal back in binary form.

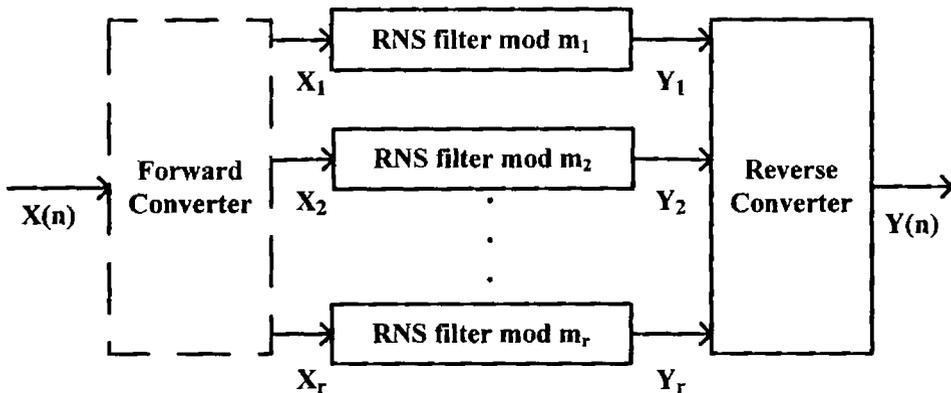


Figure 3.8 RNS based FIR filter

The moduli set selected for implementation of all the three filters is $(25, 29, 31, 37, 43, 47, 59, 64)$, which provides 43-bit dynamic range. The filter coefficients are taken with 14-bit accuracy. As input to the filter has maximum of 4-bits and the moduli set consists of 5-bit and 6-bit numbers, no forward converter is required in the new filter. The reverse converter at the last stage converts filtered outputs from parallel channels to binary form. A filter channel corresponding to modulus ‘ m_i ’ of the first stage is shown in Figure 3.9, where \otimes and \oplus represent modulo multiplication and addition respectively. Modulo multiplication is implemented with look up table (LUT). The LUT

contents can be easily reprogrammed as the mode changes, by implementing the LUTs in FPGA RAM blocks. The modulo adder receives the residue digits and performs usual binary addition, followed by modulo correction if a carry out is produced. The simulation results and analysis of dual-mode decimation filter for WCDMA/WiMAX standards are presented in Section 6.4.

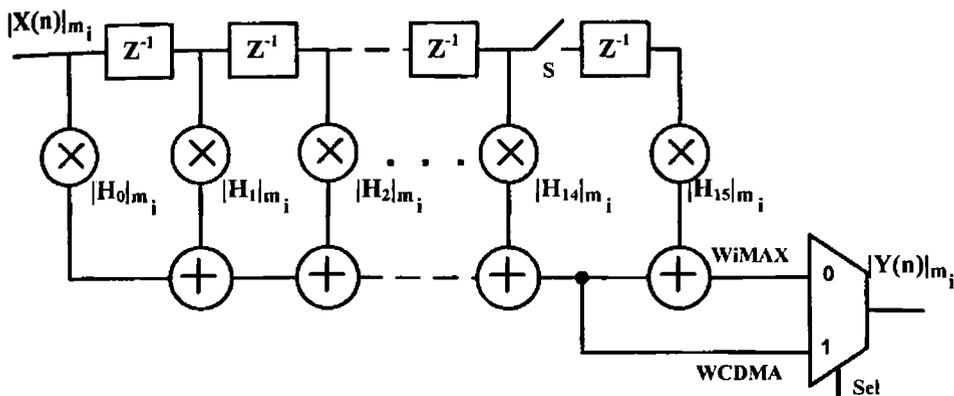


Figure 3.9 i^{th} filter channel of stage 1 for WCDMA/WiMAX decimator

3.3.3 Dual-mode Decimation Filter for WCDMA/WLANa

A dual-mode RNS based decimation filter that can be programmed for WCDMA and 802.11a standards is presented. The same design and implementation procedures are followed here as in the case of dual-mode WCDMA/WiMAX transceiver [Shahana et al., 2008a]. A reconfigurable sigma-delta modulator with 2-2-1 cascaded MASH topology is assumed as the front end. It reconfigures as a fourth order modulator with 1-bit quantizer and an OSR of 16 to offer a dynamic range of 79 dB in WCDMA mode of operation. In WLAN mode of operation, it reconfigures as a fifth order modulator with 4-bit quantizer and an OSR of 8 to achieve a dynamic range of

69 dB. The specifications for WCDMA and WLANa standards and the corresponding decimation filter design parameters are given in Table 3.2.

Table 3.2 Standard specification and decimation filter design parameters for WCDMA/WLANa transceiver

Specification	WCDMA	WLANa
Frequency range(GHz)	DL:2.11-2.17 UL:1.92-1.98	5.15-5.35
Channel Spacing	5 MHz	20 MHz
Data rate	3.84 Mchips/s	12 Msymbols/s
OSR	16	8
Input sampling frequency, F_s	61.44 MHz	96 MHz
Passband edge	2 MHz	8 MHz
Stopband edge	2.5 MHz	10 MHz
Offset frequency (MHz) : Interference magnitude(dBm)	5 : -63 10 : -56 12.5:-44	20 : -63 40 : -47
C/N ratio	7.2 dB	28 dB
Passband ripple	0.5 dB	0.5 dB
Stopband attenuation	55 dB	44 dB

3.3.3.1 Design Considerations

The OSR is chosen as 16 for WCDMA and 8 for WLANa. Multistage decimation is done in 3 stages with decimation factors of 4, 2 and 2 for WCDMA, and in 2 stages with decimation factors of 4 and 2 for WLANa. The FIR filter in each stage is designed using Remez Parks-McClellan optimal equiripple FIR filter implementation algorithm. The filter orders obtained for WCDMA are 14, 11 and 37 for first, second and third stages respectively. For WLANa filter orders are 33 and 25 respectively. The block diagram for the programmable decimation filter is shown in Figure 3.10, where N1, N2 and N3 denote the order of filters in each mode. The third filter will be operating only in WCDMA mode and will be bypassed in WLAN mode using switch S.

The switch can be a transmission gate. The first 14 MAC units of first stage filter and first 11 MAC units of the second stage are shared for both modes.

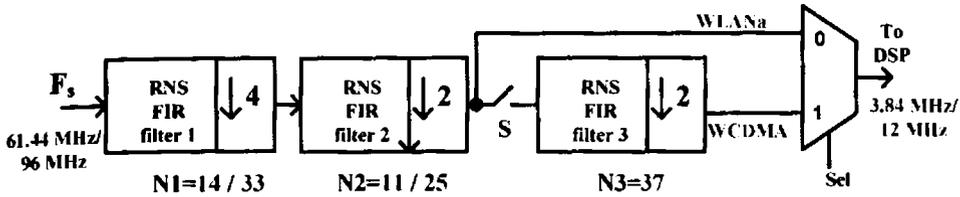


Figure 3.10 Dual-mode programmable decimation filter for WCDMA/WLANa

The same moduli set as in Section 3.3.2.1 is selected for implementation of all the three filters. The moduli set is (25, 29, 31, 37, 43, 47, 59, 64), which provides 43 bit dynamic range. The filter coefficients are taken with 14 bit accuracy. No forward converter is required as input to the filter has maximum of 4-bits and the moduli set consists of 5-bit and 6-bit numbers. A filter channel in the first stage corresponding to modulus ' m_i ' is shown in Figure 3.11. Modulo multiplication is implemented with LUT approach. The simulation results and analysis of dual-mode decimation filter for WCDMA/WLANa standards are presented in Section 6.5.

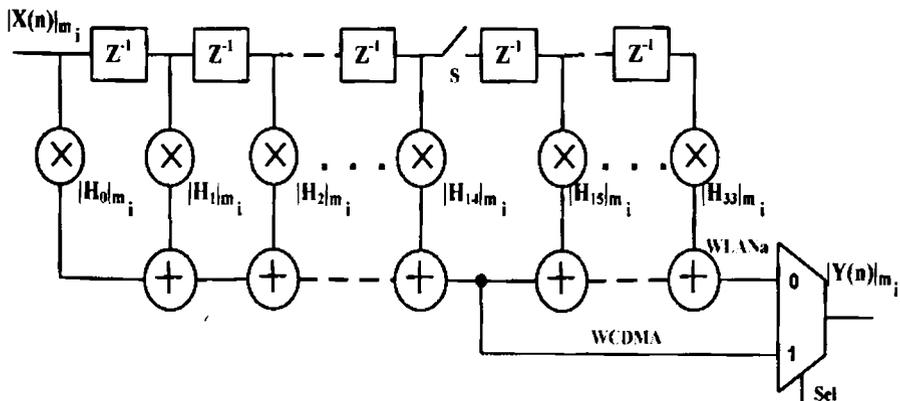


Figure 3.11 i^{th} filter channel of stage 1 for WCDMA/WLANa decimator

3.4 RNS Multiplier using Index Calculus

An algebraic field with finite number of elements is called a finite field or Galois field. There are two types of Galois fields: prime fields $GF(p)$ and polynomial fields $GF(p^m)$, where p is a prime number and m is any positive integer. It has the property that all non-zero elements of the field can be generated by non-negative integer powers of certain elements, say g , called primitive roots. This property is exploited to perform multiplication over $GF(p)$ using the isomorphism between a multiplicative group $\{q_n\} = \{1, 2, \dots, p - 1\}$, with multiplication modulo p , and the additive group $\{i_n\} = \{0, 1, \dots, p - 2\}$, with addition modulo $(p - 1)$ [Radhakrishnan and Yuan, 1990]. The relatively prime moduli in an arbitrary moduli set take any of the three forms p , 2^m and p^m , or a value with any of these as a factor. Number theoretic approach shows that the groups formed by p , 2^m and p^m integer elements fall into the category of Galois field $GF(p)$, and integer rings Z_2^m and Z_p^m [Hardy and Wright, 1979], [Koblitz, 1994].

For prime modulus the normal index mapping in $GF(p)$ is done as $q_n = |g^{i_n}|_p$. Multiplication of two numbers q_j and q_k is performed by adding their indices i_j and i_k modulo $(p - 1)$, and then by doing the inverse index operation. This approach is shown in Figure 3.12. Hence by index calculus approach, the product is represented as

$$|q_j q_k|_p = g^{|i_j + i_k|_{p-1}} \quad (3.16)$$

The elements of the integer ring Z_2^m are represented by a triplet index code $\langle \alpha, \beta, \gamma \rangle$ [Radhakrishnan, 1998], [Preethy et al., 2001a]. Any integer $X \in \{1, 2, \dots, 2^m - 1\}$ can be coded using the triplet index set as

$$X = 2^\alpha |5^\beta (-1)^\gamma|_{2^m} \quad (3.17)$$

where $\alpha \in \{0,1,\dots,m-1\}$, $\beta \in \{0,1,\dots,(2^{m-2}-1)\}$ and $\gamma \in \{0,1\}$. Multiplication of two integers X_1 and X_2 is carried out as follows: $X_1, X_2 \in Z_{2^m}$ where $X_1 \neq 0$, $X_2 \neq 0$, $X_1 = 2^{\alpha_1} |5^{\beta_1} (-1)^{\gamma_1}|_{2^m}$ and $X_2 = 2^{\alpha_2} |5^{\beta_2} (-1)^{\gamma_2}|_{2^m}$. Then the product is given by

$$|X_1 X_2|_{2^m} = 2^{\alpha_1 + \alpha_2} |5^{\beta_1 + \beta_2} (-1)^{\gamma_1 + \gamma_2}|_{2^m} \quad (3.18)$$

The index addition is performed with the following constraints: β_1 and β_2 are added modulo 2^{m-2} , γ_1 and γ_2 are added modulo 2, and α_1 and α_2 are added in normal binary mode. When the sum of 'α' indices is equal to $(m-1)$ the corresponding 'β' and 'γ' are made 'zero', and when the sum exceeds $(m-1)$ the final product is made 'zero'.

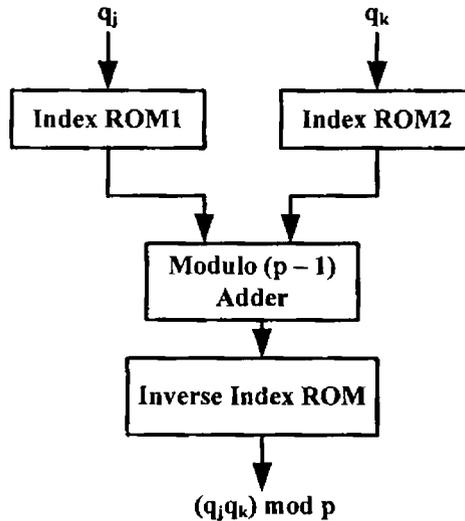


Figure 3.12 Index calculus multiplier

Similarly, the elements of the integer ring Z_p^m , where p is odd, are represented by an index pair $\langle \alpha, \beta \rangle$ [Radhakrishnan, 1998], [Preethy et al., 2001a] and any integer X is represented by

$$X = \left| g^\alpha p^\beta \right|_{p^m} \quad (3.19)$$

where g is primitive root of p , $\alpha \in \{0, 1, \dots, \phi(p^m) - 1\}$ where $\phi(p^m) = (p-1)p^{m-1}$, and $\beta \in \{0, 1, \dots, m-1\}$. The modulo multiplication of X_1 and X_2 in this finite field is carried out as follows: $X_1, X_2 \in Z_{p^m}$ where $X_1 \neq 0, X_2 \neq 0, X_1 = \left| g^{\alpha_1} p^{\beta_1} \right|_{p^m}$ and $X_2 = \left| g^{\alpha_2} p^{\beta_2} \right|_{p^m}$.

Then the product is given by

$$\left| X_1 X_2 \right|_{p^m} = \left| g^{\alpha_1 + \alpha_2} p^{\beta_1 + \beta_2} \right|_{p^m} \quad (3.20)$$

The index additions are performed subject to the following constraints: α indices are added modulo $\phi(p^m)$, and β indices are added in normal binary. When the sum of β indices exceeds $(m-1)$ the final product is made 'zero'.

3.5 Programmable Decimation Filter using Index Calculus Multipliers

The design and implementation of RNS based decimation filter programmable for WCDMA/WLANa standards where modulo multiplication is performed by index addition is presented. This offers increased programmability for multimode transceivers compared to the LUT approach for modulo multiplication. There are several techniques to perform modulo multiplication as reported in [Alia and Martinelli, 1991], [Paliouras and Stouraitis, 1997], [Bobin and Radhakrishnan, 1989], [Hiasat, 2000]. This research uses index calculus based multiplier as it has a simple structure and is fast. The specifications for WCDMA and WLANa standards and the corresponding decimation filter design parameters are given in Table 3.2.

3.5.1 Design Considerations

The design considerations for the programmable decimator are as detailed in Section 3.3.3.1. The multistage decimator is implemented in three reconfigurable stages as shown in Figure 3.10 [Shahana et al., 2008b]. The FIR filters used in all the three stages are implemented in residue number system defined by the moduli set (25, 29, 31, 37, 43, 47, 59, 64), which provides 43-bit dynamic range. A key point in the design of RNS filter is the choice of proper moduli set. The dynamic range required for RNS is decided based on the values of filter coefficients and maximum possible output from the filter. The filter coefficients are taken with 14-bit accuracy. The first stage filter receives maximum of 4-bits from sigma-delta modulator as the input. The moduli set is selected to get sufficient dynamic range such that there is a unique representation for each possible value of filter output.

The index transform based multipliers are used to reduce the complexity of modular multipliers that are ideally suited for prime and powers of prime moduli [Radhakrishnan and Yuan, 1990]. In the selected moduli set, the prime moduli include 29, 31, 37, 43, 47 and 59, and the powers of prime moduli include 25 and 64. The modulus 64 is of the form 2^n so that including it in the moduli set simplifies the reverse converter [Radhakrishnan et al., 1999]. The modulo operations on $m_i = 64$, are easily implemented by normal binary operations limited to the least significant 5-bits. Moduli of the form $2^n - 1$ are also desirable as modulo addition is easily performed by n -bit binary adder with end-around carry [Soderstrand et al., 1986]. A filter channel corresponding to modulus ' m_i ' of the first stage is shown in Figure 3.13. A demultiplexer is used at the input to load filter coefficients sequentially for each mode or to distribute input through the register chain as shown in

Figure 3.13. The filter structure is made reconfigurable for WCDMA/WLANa using switch 'S' and multiplexer, leading to power saving. In each stage, the outputs from multipliers are combined using modulo adder trees. The filtered output corresponding to each mode is selected using a multiplexer.

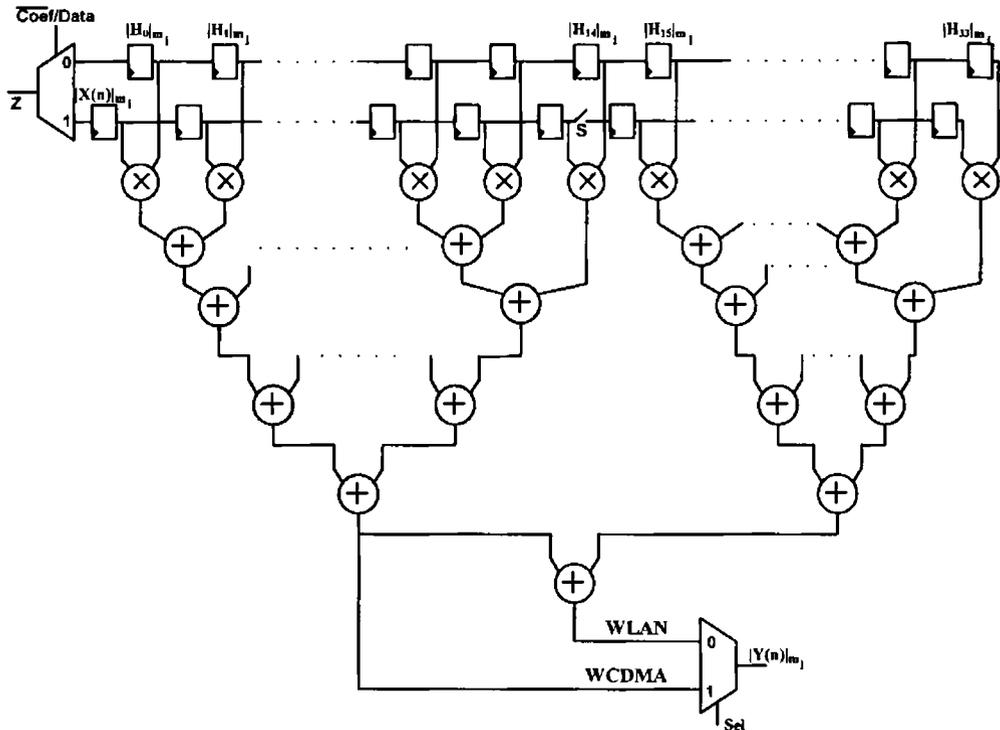


Figure 3.13 i^{th} filter channel of stage 1 programmable for WCDMA/WLANa

Modulo multiplication is performed by index calculus approach. In the selected moduli set of (25, 29, 31, 37, 43, 47, 59, 64), the moduli 29, 31, 37, 43, 47 and 59 are prime numbers and performs multiplication by index addition in the corresponding Galois field $GF(p)$. The primitive roots used for generating the Galois fields for these numbers are shown in Table 3.3. The modulus 25 is power of a prime number denoted as p^m , with $p = 5$, $m = 2$, primitive root $g = 2$ and $\phi(p^m) = 20$. So any integer X in this field is

represented as $X = \left| 2^\alpha 5^\beta \right|_{25}$, where $\alpha \in \{0,1,\dots,19\}$ and $\beta \in \{0,1\}$. Multiplication is performed by addition of α and β indices in the integer ring Z_p^m . The modulus 64 being power of 2 form integer rings of the form Z_2^m where each number is represented by a triplet index code $\langle \alpha, \beta, \gamma \rangle$. Here multiplication is done by normal binary addition for α , modulo 16 addition for β , and modulo 2 addition with an XOR gate for γ indices. When the residue digit becomes zero, as index can not be defined, extra logic is incorporated in the design for each modulus. As modulus 31 is of the form $(2^n - 1)$ and 64 is of the form 2^n , modulo multiplication can be performed more efficiently by combinational logic than using index calculus [Wang et al., 1996], [Adamidis and Vergos, 2007]. So, combinational circuits are implemented to perform modulo multiplication for these two channels.

Table 3.3 Primitive roots for the selected moduli set

Prime modulus (p)	Primitive root (g)
29	2
31	3
37	2
43	3
47	5
59	2

The simulation results and analysis of dual-mode decimation filter for WCDMA/WLAN standards using index calculus multiplier are presented in Section 6.6. The complete back end process is done, and the placed cell structure and routed view of the RNS decimation filter are also given.

3.6 Summary

The hardware architectures of FIR digital filters operating in RNS and traditional binary number system are presented. The RNS coding technique is attractive for FIR filters as it requires only multiplication and addition which are very fast operations in residue domain. While designing RNS filter care must be taken to choose the moduli set depending on the filter length and input word length so as to provide sufficient dynamic range avoiding overflow. The area overhead due to forward and reverse conversion seems to be compensated after a particular filter length as the rate of increase of area for RNS filter is less than that for traditional filter for each additional filter tap. The simulation results and performance analysis are given in Section 6.3.

Dual-mode RNS based reconfigurable decimation filters for WCDMA/WiMAX standards and WCDMA/WLANa standards are designed and implemented. The simulation results and analysis are given in Section 6.4 and 6.5 respectively. The forward converter is eliminated in the new filter by suitably selecting the moduli set. Multistage implementation for sampling rate conversion results in reduced hardware complexity and power consumption. Powering down or bypassing of the unused hardware in each mode of operation leads to further power saving. As the entire filter stages are implemented in RNS and are operating with the same moduli set, a reverse converter is needed only at the last stage output. Since these FIR filters operate in RNS domain, high speed operation with lesser pipelining is achieved due to its carry free operation on smaller residues in parallel channels. A programmable multistage RNS based decimation filter for WCDMA and WLANa standards using index calculus multiplier is also developed. The modulo multiplication is performed by index calculus approach to achieve

increased programmability required for multi-mode operation. The simulation results and analysis, with placed cell structure and routed view are presented in Section 6.6.

Chapter 4

RRNS-Convolutional Concatenated Coded OFDM Wireless Communication System with a Direct Analog-to-Residue Converter

A novel approach for direct analog-to-residue conversion is presented in this chapter using the most popular sigma-delta analog-to-digital converter. This converter provides high resolution, high conversion speed and a low cost for implementation. The non-positional nature of RNS makes it suitable for fault-tolerant architectures. The error detection and correction properties of Redundant Residue Number System (RRNS) are obtained by introducing few redundant moduli that are relatively prime to the non-redundant moduli. An RRNS-Convolutional concatenated coding (RCCC) scheme for OFDM wireless communication system is presented to improve the system performance under different operating conditions.

4.1 Direct Analog-to-Residue Converters

An important step in the Residue Number System (RNS) based signal processing is the conversion of signal into residue domain. The forward and reverse conversion circuitries are complex for a general moduli set and limits the applications of RNS. Many researchers have explored efficient methods for forward and reverse conversions [Radhakrishnan et al., 1999], [Ananda Mohan and Premkumar, 2007] [Srikanthan et al., 1998]. The analog-to-residue conversion usually involves two steps. The analog signal is first converted to digital signal by analog-to-digital converter, then to residue form by a binary-to-residue converter. Several implementations of this conversion have been presented for various goals; one of the implementations is by a direct conversion from an analog input to residue form. The direct analog-to-residue (A/R) converters operating at Nyquist rate available in literature are detailed in the following section.

A novel approach for analog-to-residue conversion is presented in this research using the most popular sigma-delta analog-to-digital converter (SD-ADC). In this approach, the front end is the same as in traditional SD-ADC that uses sigma-delta modulator with appropriate dynamic range and the filtering is done by a filter implemented using RNS arithmetic. Hence, the natural output of the filter is an RNS representation of the input signal.

4.1.1 Nyquist Rate Analog-to-Residue Converters

The existing Nyquist rate A/R converters include a multiple-residue flash converter, successive approximation based A/R converter and iterative flash A/R converter. Nyquist rate A/R converters are practically implemented

only up to 10-12 bits of resolution due to component matching and circuit nonidealities.

4.1.1.1 Multiple-Residue Flash A/R Converter

Mandyam and Stouraitis presented a direct analog-to-residue conversion that uses flash ADC, PLA, latches, code converter, buffers and XOR gates as shown in Figure 4.1 [Mandyam and Stouraitis, 1990]. A converter of n -bits resolution requires $(2^n - 1)$ comparators and 2^n resistors. Any number X can be represented as $X = B_i m_i + x_i$, where m_i is the modulus, B_i is the quotient which represents the base value in a flash converter and x_i is the residue. The comparators are grouped into (M / m_r) groups with m_r outputs in each group where M is the dynamic range and m_r is the largest value of the moduli set. The $m_r - 1$ output lines from each group are connected to a buffer which in turn are connected to a data bus. The comparator with the lowest threshold voltage in each group is the base-value comparator of that group. The outputs from the comparators correspond to the residues in the range 0 to $m_r - 1$ with 0 assigned to the output of the base-value comparator. Any analog input sample X belongs to one of the above groups. Hence the number of 1's in the output thermometer code from that group uniquely identifies the residue x_r .

A set of two input XOR gates are used to generate the enable signal for buffers. The XOR inputs are connected to the outputs of adjacent base-value comparators. So for a given input sample X , only one of the XOR outputs will be logic high and the corresponding buffer will be enabled. The output of the enabled buffer is pushed to a latch through the data bus which in turn addresses a PLA to generate the residue x_r . The XOR gate whose output is

group number in binary form generated using a code converter together with the number of 1's in the buffer output are used to uniquely identify the remaining residues.

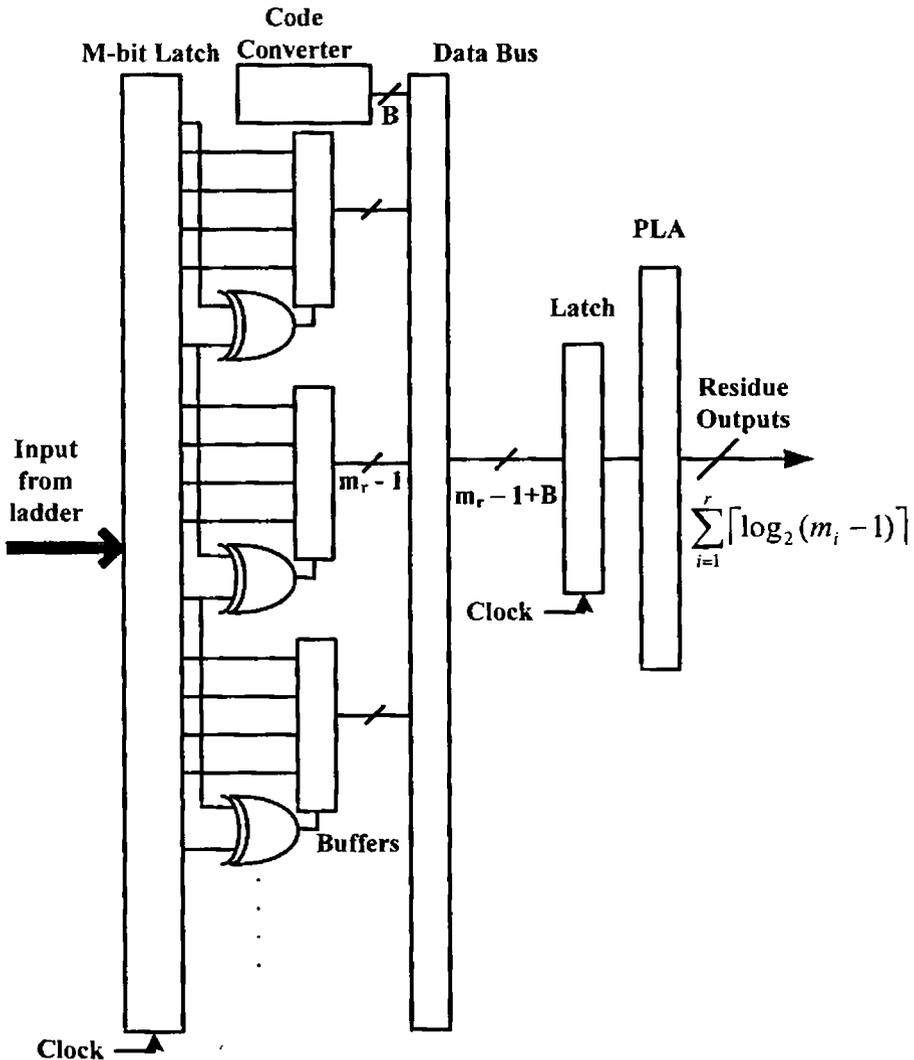


Figure 4.1 Multiple-residue flash converter

The size of the PLA used in the converter is

$$(B + m_r - 1) \times \sum_{i=1}^r \lceil \log_2(m_i - 1) \rceil \text{ bits, where } B = \left\lceil \log_2 \left(\frac{M}{m_r} - 1 \right) \right\rceil. \text{ The conversion}$$

is very fast and can be controlled with a clock. But, flash A/R converters with high resolution are impractical to construct as the number of elements grows exponentially with resolution 'n'. Flash A/R converters with only 8 - 10 bits of resolution are practically implemented.

4.1.1.2 Successive Approximation based A/R Converter

The A/R converter presented by Radhakrishnan et al. uses successive approximation ADC as the basic element [Radhakrishnan and Preethy, 1999]. The block diagram is shown in Figure 4.2. The converter consists of a cascade of two successive approximation ADCs, a few modulo adders and small look up tables. Here, the successive approximation ADC of the first stage is modified by replacing the comparator with a difference amplifier. Also, a weighting factor ' m_r ', equal to the value of the largest modulus, is applied to the DAC output and is fed back to this difference amplifier. The sampled analog input voltage ' X ' is applied to the other input of difference amplifier. The output voltage from difference amplifier is equivalent to $X - i * m_r$, where ' i ' is the value stored in the register. The size of register in the first stage is $k = \left\lceil \log_2 \left(\frac{M}{m_r} - 1 \right) \right\rceil$ bits. After identifying all the k bit positions, the output of the difference amplifier is $X - R * m_r$, where ' R ' is the value stored in the register.

When the first stage stops conversion the output of the difference amplifier is the voltage equivalent of $x_r = X \text{ mod } m_r$. This residue value ' x_r ' is converted to digital output using the second stage successive approximation ADC. The output register size of this converter is $l = \lceil \log_2 m_r \rceil$ bits. The remaining $(r - 1)$ residues are generated from the register content ' R ' of the

first stage ADC and the output register content ' x_r ' of the second stage ADC. The residue generation logic used is similar to that in Figure 3.4. The register content of first stage is partitioned into groups of l bits wide. Each group addresses small ROM look up tables that are programmed to produce the residues corresponding to the modulus m_j . These are combined using few modulo adders to generate the residue x_j . Total of $(r-1)$ similar stages are used to generate all the residues in parallel.

The second stage analog-to-digital conversion is delayed until the completion of the first stage. The total conversion time for generating the first residue is approximately $k+l+2$ clock cycles. The generation of remaining $(r-1)$ residues involve additional delay in ROMs and modulo adders. But majority of this time overlaps with the conversion time in second stage. So, all the residues are generated almost simultaneously.

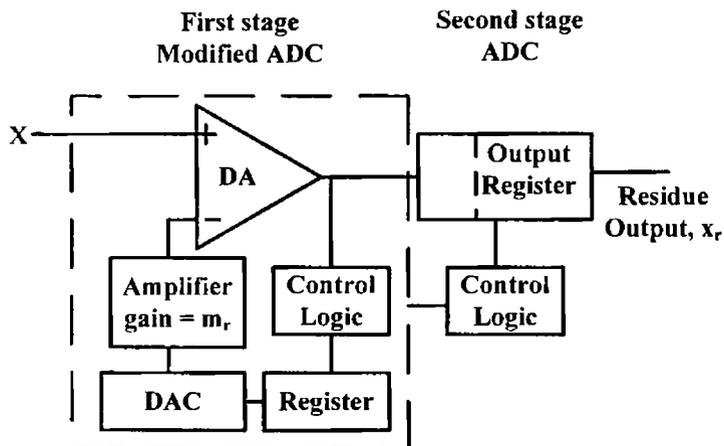


Figure 4.2 Successive approximation A/R converter

4.1.1.3 Iterative Subranging Flash A/R Converter

An iterative flash A/R converter is presented in Figure 4.3, which uses the principle of subranging to reduce the hardware complexity of flash ADCs

[Radhakrishnan and Preethy, 1998]. By iterating a flash converter for a moderate number of bits together with subranging, it is possible to attain a considerably better resolution with slightly extended clock cycles and significantly reduced hardware. Here, two flash converters are used. The first stage flash converter is used iteratively to find the quotient of ' R_r ' with respect to the largest modulus, m_r . The second stage flash converter converts the analog voltage equivalent to the residue ' x_r ' into binary form.

The analog input X is fed to the voltage divider of first flash converter FC1 to get a thermometer coded output corresponding to the m most significant bits (MSBs) of R_r . The output from the flash is converted to binary form using a latch and a PLA. This is stored as the m MSB bits of Register 1 using a demultiplexer. The binary output from Register 1 is converted to analog voltage using a DAC with gain m_r and fed to a difference amplifier. The difference amplifier produces the difference between this value and the analog input X . This difference is equivalent to the rest of the quotient and the residue value. This is further amplified by an amplifier with a gain of 2^{im} , where ' i ' is the iteration count. It is then fed back to FC1 to get the next m bits of the quotient. These are stored in the next m lower significant positions of Register 1. This process is repeated for p cycles where $p = \left\lceil \frac{B}{m} \right\rceil$ and

$B = \left\lceil \log_2 \left(\frac{2^n - 1}{m_r} \right) \right\rceil$. The contents of Register 1 represent the quotient R_r after

p cycles. The output of the difference amplifier is switched to the second flash converter FC2 to produce the residue x_r in binary form. The generation of remaining $(r-1)$ residues is done as in successive approximation A/R converter.

The conversion time for generating residue ' x_r ' is the sum of conversion times in both the flash converter stages. This time is given by $(p+1)$ extended clock cycles. The extended clock cycle includes the delays of DAC, amplifiers, analog and digital switches, comparators, latch and PLA. The generation of remaining residues requires additional delay. This is made overlapping with the second ADC, thereby generating all the residues almost simultaneously.

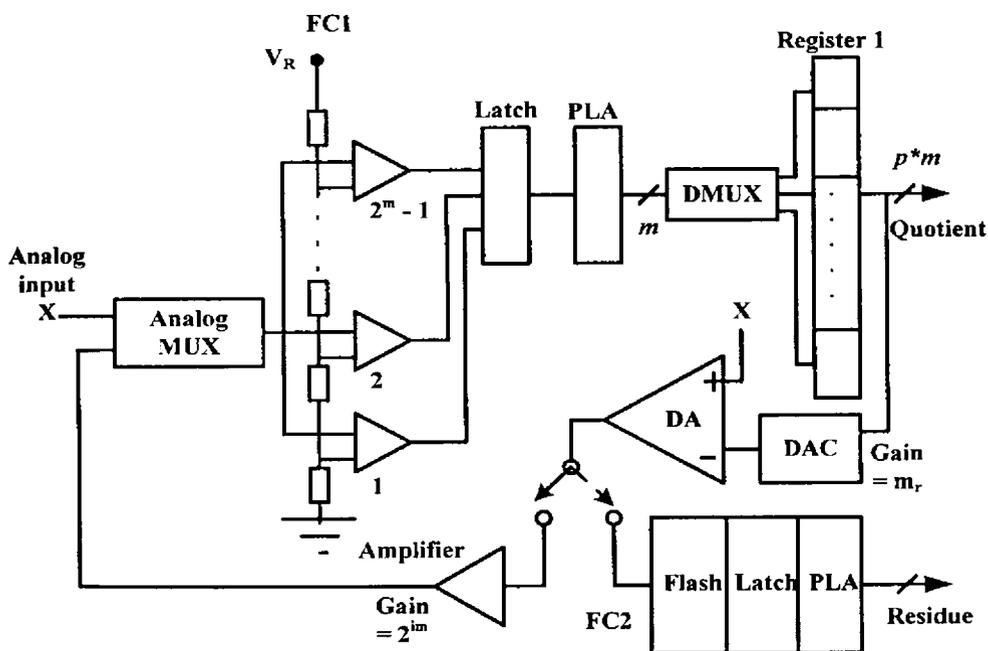


Figure 4.3 Iterative flash A/R converter

The Nyquist rate A/R converters are practically implemented only up to 10-12 bits of resolution due to component matching and circuit nonidealities. But there are many real world applications that require higher resolution than this. This motivates the development of a parallel analog-to-residue converter based on sigma-delta ADC. The new converter offers high resolutions of up to 20-bits.

4.1.2 A Novel Sigma-Delta based Parallel Analog-to-Residue Converter

The main motivations for using sigma-delta based architecture for the new A/R converter are the following: As majority of the circuitry in sigma-delta converters are digital, the performance will not drift significantly with time and temperature. An external sample and hold circuit is not required due to the high input sampling rate and low precision of the analog-to-digital conversion. The devices are inherently self-sampling and tracking. The background noise level which determines the signal to noise ratio (SNR) is independent of the input signal level.

The new A/R converter architecture based on sigma-delta based modulator is shown in Figure 4.4. The analog input is sampled at an oversampling rate much greater than Nyquist rate. The order of the modulator ' L ', the OSR ' M ' and the number of quantizer bits ' B ' are selected to meet the dynamic range requirements for various resolutions. The binary bits from sigma-delta ($\Sigma\Delta$) modulator are given to the following RNS based decimation filter. The residue digits are generated in parallel at the decimator outputs.

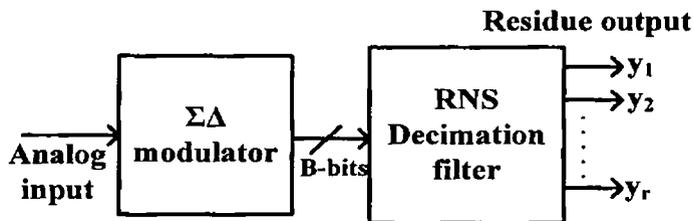


Figure 4.4 Sigma-delta based parallel A/R converter

4.1.2.1 Sigma-Delta Modulator

Sigma-delta modulator trades resolution in time for resolution in amplitude. Oversampling and noise shaping are the two key techniques on which the modulator relies. Oversampling reduces the baseband quantization

noise, and noise shaping moves quantization noise from the baseband to higher out-of-band frequencies. The oversampling and noise shaping techniques are combined to achieve superior resolution with relaxed requirements on analog hardware compared to Nyquist rate converters [Norsworthy et al., 1997].

An efficient way of implementing higher-order $\Sigma\Delta$ modulator is to cascade multiple lower order stages such that each stage processes the quantization noise of the previous stage. In cascaded or MASH topology, the outputs of each individual stage go to a digital error cancellation logic where the quantization noise of all stages except that of the last one is removed. The quantization noise of the remaining stage is filtered by the noise transfer function $(1 - z^{-1})^L$, where L is the modulator order of the overall $\Sigma\Delta$ modulator. A fourth order $\Sigma\Delta$ modulator can be implemented using two second order stages as shown in Figure 4.5. The main advantage of MASH architecture is the high degree of noise shaping without any stability problems. However, cascaded modulators require very good matching between analog and digital processing paths.

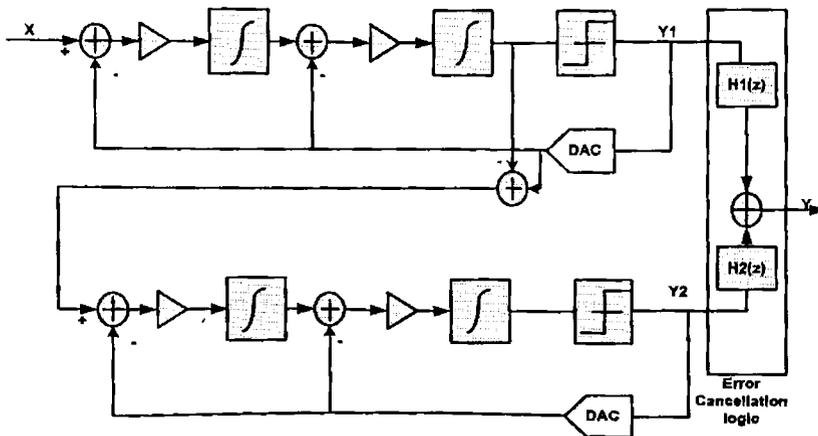


Figure 4.5 A 2-2 cascaded MASH architecture

4.1.2.2 RNS based Decimation Filter

The oversampling converters relax the requirements placed on analog circuitry at the expense of more complicated digital circuitry. This trade-off becomes more desirable for modern submicron technologies with low power supplies because the complicated high speed digital circuitry is more easily realizable in a lesser area. But the realization of high resolution analog circuitry is complicated by low power supply voltages and poor transistor output impedance. The digital decimation filter serves two purposes. It acts as antialiasing filter that removes the unwanted noise above the Nyquist band seen in the analog input spectrum to the $\Sigma\Delta$ modulator. So, it avoids aliasing into the baseband by the decimation process. The decimation filter also removes the out-of-band quantization noise produced by the $\Sigma\Delta$ modulator. Upon filtering, the output is resampled at the Nyquist rate. A strict linear phase characteristics is required for most digital audio data converters. Hence symmetric FIR filters are widely used for decimation filter implementations.

The decimation filter receives the output of $\Sigma\Delta$ modulator as its input. The decimation filter operates in the RNS domain defined by a proper moduli set that provides sufficient dynamic range avoiding overflow. The moduli set consists of relatively prime integers and are selected in such a way that the number of bits for representing each modulus is greater than the maximum number of bits from the modulator. This eliminates the need for a forward converter and the output of the modulator 'b' is directly mapped into the residue domain (x_1, x_2, \dots, x_r) , by a simple encoding as in (4.1).

$$\begin{aligned}
 x_i &= b, \forall i \text{ if } b \text{ is positive, and,} \\
 x_i &= m_i - |b|, \text{ for } i = 1, 2, \dots, r \text{ if } b \text{ is negative}
 \end{aligned}
 \tag{4.1}$$

The RNS based decimation filter is shown in Figure 4.6, where the MAC operations are performed in RNS domain. The structure of a particular modulo filter channel based on modulus ' m_i ' is shown in Figure 4.7, where \otimes and \oplus represent modulo multiplication and modulo addition respectively. The downsampler in each channel resamples the output at Nyquist rate. Here all the residues are generated in parallel at the filter outputs.

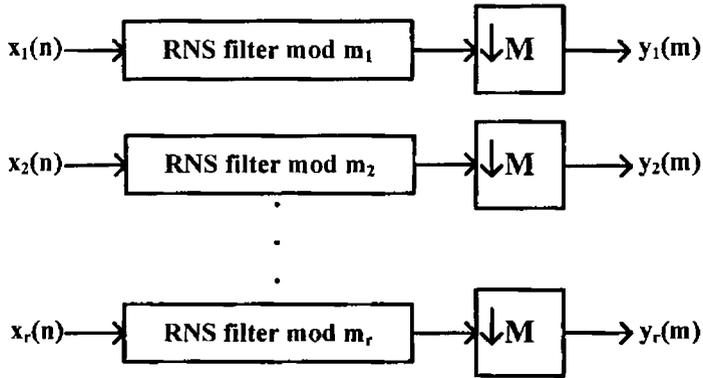


Figure 4.6 RNS based decimation filter for A/R converter

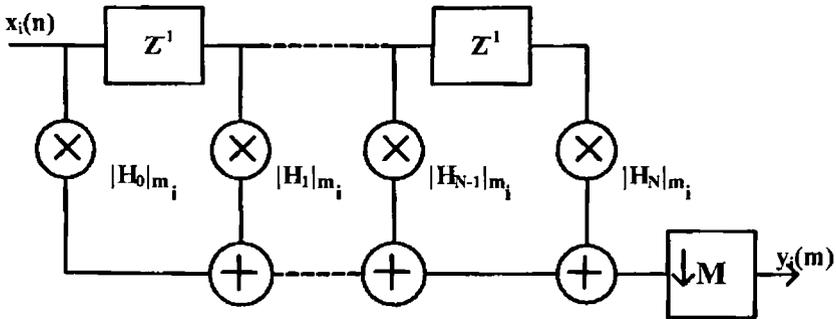


Figure 4.7 i^{th} modulo filter channel for A/R converter

The $\Sigma\Delta$ modulator complexity for A/R converter with a particular resolution is decided based on the dynamic range (DR) requirement. The dynamic range in dB of an ADC with n -bit resolution is given in (4.2). The theoretical DR for a $\Sigma\Delta$ modulator with order ' L ', oversampling ratio ' M ', and

number of quantizer bits ‘ B ’ is given in (4.3). Using (4.2) and (4.3), the required modulator order, oversampling ratio and the number of quantizer bits are easily calculated for a given resolution.

$$DR = 6.02 * n + 1.76 \quad (4.2)$$

$$DR = \frac{3}{2} \frac{2L+1}{\pi^{2L}} M^{2L+1} (2^B - 1)^2 \quad (4.3)$$

The A/R converter with 12-bits resolution requires a DR of 74 dB as given by (4.2). To meet the DR requirement, a fourth order sigma-delta modulator with a 2-bit quantizer and an OSR of 16 is selected using (4.3). A 2-2 cascaded MASH topology is used for the modulator. The 2-bit output from $\Sigma\Delta$ modulator is given to the RNS based decimation filter. The moduli set (16, 19, 23) gives a dynamic range of more than 12-bits for the RNS. Simulations are carried out for sigma-delta A/R converters of various resolutions. The simulation results and the complexity of the modulator and the decimation filter obtained for various resolutions are presented in Section 6.7.

4.2 RRNS-Convolutional encoded Concatenated Code for OFDM based Wireless Communication

The modern telecommunication industry demands higher capacity networks with high data rate. Orthogonal frequency division multiplexing (OFDM) is a promising technique for high data rate wireless communications at reasonable complexity in wireless channels. OFDM has been adopted for many types of wireless systems like wireless local area networks such as IEEE 802.11a, and digital audio/video broadcasting (DAB/DVB). A concatenated coding scheme that improves the performance of OFDM based wireless communications is presented in this research. It uses a Redundant Residue

Number System (RRNS) code as the outer code and a convolutional code as the inner code. The bit error rate (BER) performances of the communication system under different channel conditions are investigated. These include the effect of additive white Gaussian noise (AWGN), multipath delay spread, peak power clipping and frame start synchronization error.

4.2.1 OFDM Communication System

The increasing demand for broadband communication systems with a greater range of services like video conferencing, internet services and digital multimedia applications has promoted the development of orthogonal frequency division multiplexing (OFDM) based systems. The OFDM is a digital multicarrier modulation method which distributes the data over a large number of closely spaced orthogonal carriers. The spectrum of each carrier has null at the centre frequency of each of the other carriers in the system [Schulze and Luders, 2005], [Heiskala and Terry, 2001]. The available bandwidth is divided among the orthogonal carriers. Each carrier is then modulated by a low data rate stream with a conventional modulation scheme such as quadrature amplitude modulation (QAM) or quadrature phase shift keying (QPSK). OFDM provides high spectral efficiency by spacing the channels close together. This will not result in interference between the carriers, as they are orthogonal to each other. In a coded OFDM (COFDM) system, signals are coded before transmission for forward error correction (FEC). The efficiency in spectrum usage and robustness to multipath fading make COFDM as a popular scheme for wideband digital communication.

Several researchers have developed different coding schemes to improve the performance of multicarrier wireless communication systems. A multi-code direct sequence code division multiple access (DS-CDMA) system

based on RRNS as inner code and Reed-Solomon (RS) code as outer code is presented [Yang and Hanzo, 1999]. The performance of a DS-CDMA system over bursty communication channels and multipath environment is analyzed using a concatenated coding with convolutional code as outer code and RRNS as inner code [Madhukumar and Chin, 2000]. The design rules and general analytical upper bounds for parallel concatenated, serial concatenated, hybrid concatenated and self concatenated codes over AWGN and Rayleigh fading channels are presented [Divsalar and Pollara, 1997]. The suitability of OFDM as a modulation technique for wireless communication system is investigated in [Lawrey, 1997] in which a comparison with CDMA system is provided. A concatenated code for IEEE 802.11a system is presented where a block Hamming code joins with a convolutional code, to achieve better system performance under fixed power and bit error rate (BER) requirements [Tsai and Huang, 2005].

One of the major drawbacks of OFDM is that it generates signals with large amplitude variations resulting in high peak to average power ratio (PAPR). These large peaks increase the amount of intermodulation distortion resulting in an increase in the error rate [Schulze and Luders, 2005]. The system performance can be improved by minimizing the PAPR which allows a higher average power to be transmitted for a fixed peak power. A lot of research has been done that reduces the PAPR for OFDM based systems [Tarokh and Sadjadpour, 2003], [Guo and Hsu, 2006], [Tezeren, 2004]. A method to enhance the bandwidth efficiency of a multicarrier CDMA system by using RNS representation for information symbols combined with PSK/QAM modulation and orthogonal spreading is presented [Madhukumar and Chin, 2002]. Yang and Hanzo has done the performance evaluation of RNS based parallel communication scheme using orthogonal signaling with

ratio static test over AWGN channel and multipath fading channel [Yang and Hanzo, 2002a], [Yang and Hanzo, 2002b].

A concatenated coding scheme consisting of RRNS as outer code and convolutional code as inner code for OFDM based wireless communication system is presented here. The new coding scheme combines the error detection and correction properties of RRNS with convolutional codes. The RCCC scheme offers significant improvement in BER performance under different channel conditions. The performance of the system is analyzed for AWGN channel and multipath fading channel. The effect of frame start synchronization error and peak power clipping for PAPR reduction for the new coding scheme are also analyzed.

4.2.2 Error Detection and Correction with RRNS

The residue number system (RNS) is primarily used for high speed digital signal processing due to the modular carry free arithmetic operations. The nonweighted and nonpositional nature of residues offer fault tolerant properties to RNS. The lack of ordered significance among the residue digits allows the erroneous digit to be discarded without affecting the result, provided sufficient dynamic range is there in the reduced system to represent the result. The RNS is defined by a set of relatively prime integers (m_1, m_2, \dots, m_v) which are called the nonredundant moduli. Error detection and correction properties are introduced by inserting few redundant moduli. Thus redundant residue number system (RRNS) is defined by the moduli set $(m_1, m_2, \dots, m_v, m_{v+1}, \dots, m_u)$. The redundant moduli should be relatively prime to the nonredundant moduli and should satisfy the condition $(m_{v+1}, \dots, m_u) > \max(m_1, m_2, \dots, m_v)$. The total dynamic range of RRNS is given by $M_T = \prod_{i=1}^u m_i$. This

total range $[0, M_T)$ is divided into two adjacent intervals in terms of the ranges defined by the nonredundant and redundant moduli. The interval $[0, M)$ is called the *legitimate range*, where $M = \prod_{i=1}^v m_i$ and the interval $[M, M_T)$ is called the *illegitimate range*. In order for RRNS to have self checking, error detection and error correction properties, the information or data has to be constrained within the legitimate range. It has been shown that the RRNS with $(u - v)$ redundant moduli can detect $(u - v)$ errors and can correct up to $\lfloor (u - v)/2 \rfloor$ errors, where $\lfloor \cdot \rfloor$ denotes the integer part [Mandelbaum, 1972], [Yang and Hanzo, 2004].

Several researchers have studied error detection and correction properties of RRNS. Watson and Hastings described RRNS algorithms which can detect and correct single residue errors through a consistency checking procedure [Watson and Hastings, 1966]. Yau and Liu presented two error correcting algorithms, one for single residue error correction and the other for burst residue error correction [Yau and Liu, 1973]. This method eliminates the requirement of a correction table in Watson's method by performing iterative computations. But this method is slower than Watson's method. Barsi and Maestrini determined the necessary and sufficient conditions for the correction of single residue digit errors allowing minimal redundancy [Barsi and Maestrini, 1973]. Etzel and Jenkins presented filter simulation programs using RRNS with special emphasis on overflow detection, error correction and gradual system degradation in presence of recurring errors [Etzel and Jenkins, 1980]. Cosentino proposed a concurrent error detection model using three nonredundant channels and two redundant channels with single error correction capability [Cosentino, 1988]. Shenoy and Kumaresan proposed a base-extension procedure using Chinese remainder theorem (CRT) which is

faster than mixed radix conversion (MRC) method [Shenoy and Kumaresan, 1989]. Yang and Hanzo demonstrated the applications of RRNS codes in global communication systems to simplify the associated subsystems by unifying the entire encoding and decoding procedures [Yang and Hanzo, 2001].

An error correction scheme with RRNS is given in Figure 4.8 [Cosentino, 1988], [Preethy et al., 2001b]. Here, the binary number Z' is generated from the received nonredundant residue digits (z_1, z_2, \dots, z_v) using a reverse converter based on CRT. An auxiliary set of residues $|Z'|_{m_{v+1}}, |Z'|_{m_{v+2}}, \dots, |Z'|_{m_u}$ corresponding to the redundant channels are generated from the output Z' by forward conversion. The error syndrome for each redundant channel is calculated as in (4.4), by comparing the received redundant residue digit with the auxiliary residue generated.

$$S_{m_i} = z_{m_i} - |Z'|_{m_i} \quad \text{for } i = v + 1, \dots, u. \quad (4.4)$$

If all the error syndromes are zeros, then all the received residue digits are correct and hence there is no error present. If any one of the redundant residue channel is in error, the corresponding error syndrome is nonzero and the other syndromes are zeros. In such cases the output calculated using the nonredundant residues is correct. If there is an error in the nonredundant channel, all the syndromes are nonzeros. In this case a correction has to be applied to the output Z' . There is a unique error corresponding to each combination of the syndromes. So, the error correction can be done with the help of a look up table (LUT). The LUT is addressed by the syndrome values, and the size of the LUT required to store the correction factor is determined

using (4.5). The correction factor (CF) from the LUT is added to the output Z' from reverse converter to produce the correct output Z .

$$N = \left(2 \sum_{i=1}^v m_i - 1 \right) + \left(\sum_{i=v+1}^u m_i \right) - 1 \quad (4.5)$$

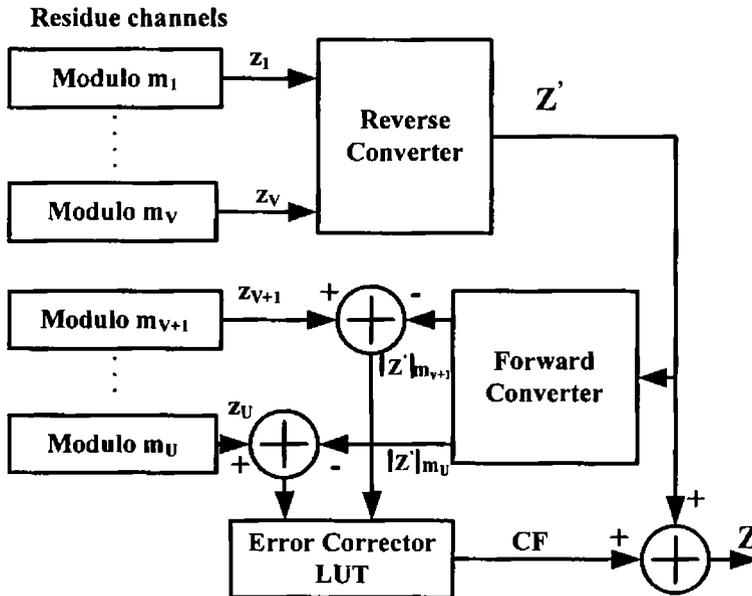


Figure 4.8 Principle of error correction with RRNS

4.2.3 System Description

The signal flow through a typical wireless digital communication system that includes concatenated forward error control coding, interleaving and deinterleaving, orthogonal digital modulation and channel impairments is illustrated. Concatenated coding is a good way to create long powerful codes with large coding gain and reduced decoding complexity by combining relatively simple channel codes [Sweeney, 2002]. The new RRNS-Convolutional concatenated coding corrects the errors using the outer RRNS

decoder that are not corrected by the inner decoder. Thus better BER performance is achieved by exploiting the properties of RRNS.

4.2.3.1 Transmitter Model

The functional block diagram shown in Figure 4.9 illustrates the transmitter section of the OFDM system with RCCC scheme. The analog input is directly converted to residue domain using sigma-delta based A/R converter of N -bit resolution. The moduli set of RRNS is selected in such a way that it offers redundancy and sufficient dynamic range for unique and unambiguous information representation. An information symbol X , is represented in RRNS as $(x_1, x_2, \dots, x_v, x_{v+1}, \dots, x_u)$ with respect to a moduli set $(m_1, m_2, \dots, m_v, m_{v+1}, \dots, m_u)$, where $x_i = X \text{ modulo } m_i$ for $i = 1$ to u . This is the outer RRNS coding. In this system, A/R converter with 8-bit resolution is considered and RRNS moduli set is selected as $(5, 7, 8, 9, 11)$ where $(5, 7, 8)$ forms the nonredundant moduli and $(9, 11)$ forms the redundant moduli. This allows detection of errors in two residue digits and can correct single residue digit errors.

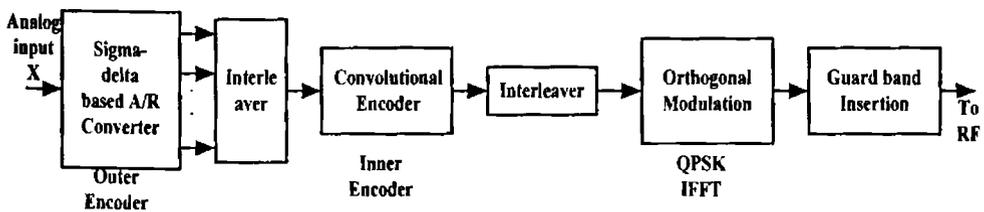


Figure 4.9 Block diagram of transmitter section

The residue digits are interleaved and then applied to a convolutional encoder for inner encoding. The new system uses industry standard $\frac{1}{2}$ rate convolutional encoder with constraint length of 7, as shown in Figure 4.10. The generator polynomials are: $g_1 = 133_8$ and $g_2 = 171_8$. The interleaved and rearranged data bits are mapped into signal constellation points according to

the type of modulation used. Differential QPSK is used as the modulation scheme for this work. Serial-to-parallel conversion is done for the modulated data. An Inverse Fast Fourier Transform (IFFT) is taken for implementing OFDM. The IFFT transforms the subcarriers from the frequency domain into the corresponding time domain. The OFDM signal is represented as in (4.6).

$$S(n) = \frac{A}{M} \sum_{i=0}^{M-1} x_i(n) \exp(2\pi f_i n) \text{ for } 0 \leq n, i < M \quad (4.6)$$

where A is the scaling factor, M is the total number of subcarriers, $x_i(n)$ is the n^{th} bit of the i^{th} data stream and $f_i = f_c + \frac{i}{T}$, for $i = 0, 1, \dots, M-1$. ' T ' is the symbol duration for the information sequence, and f_c is the centre frequency of the subcarriers. A guard band interval is inserted to avoid intersymbol interferences (ISI) and intercarrier interferences (ICI) caused by the multipath fading. Finally, the signal is transmitted after radio frequency up-conversion.

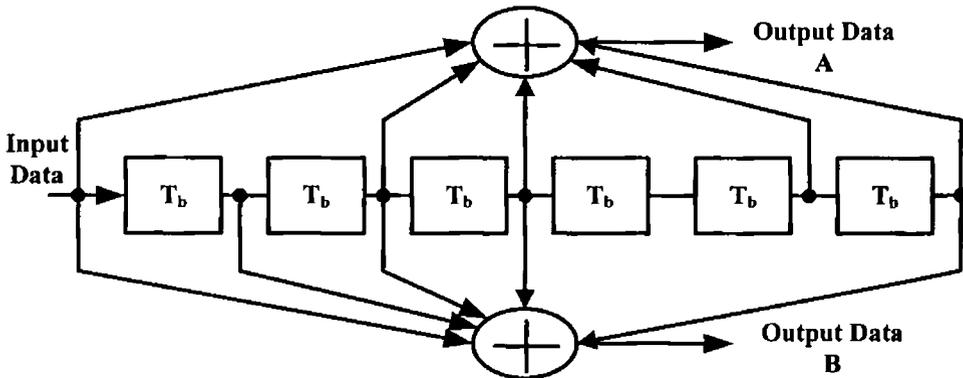


Figure 4.10 Convolutional encoder

4.2.3.2 Receiver Model

The channel attenuates the transmitted signals, delays it in time, and corrupts them by addition of Gaussian noise. The effects of multipath delay

spread are accounted by using a lowpass FIR filter model. The length of the filter represents the maximum delay spread, and the magnitude of filter coefficients represents the reflected signal amplitudes. The received signal can be represented as in (4.7), assuming P resolvable frequency selective paths for the multipath channel,

$$r(t) = \sum_{p=0}^{P-1} \alpha_p(t) s(t - \tau_p) + N(t) \quad (4.7)$$

Here, $N(t)$ represents a stationary zero-mean Gaussian random process with single sided power spectral density of N_0 , and α_p and τ_p are the complex valued channel gain and time delay of the p^{th} path respectively.

The functional block diagram of the receiver section is shown in Figure 4.11. The received signal is down-converted from radio frequency and synchronized with the symbol interval. The guard band, which is inserted for eliminating the ISI and ICI effects, is removed. The symbol constellations corresponding to the original transmitted spectrum are recovered by passing the signal through FFT. The resulting data are deinterleaved and channel decoded. The convolutional encoding applied to the data is decoded by Viterbi decoding. The output data is deinterleaved and given to RRNS decoder. The binary symbol is generated from the nonredundant residue digits using a reverse converter based on CRT. The error corrector LUT addressed by the error syndromes gives out a correction factor. This is added to the output of the reverse converter to get the corrected binary symbol. The system model uses two redundant moduli and three nonredundant moduli. So the size of the LUT becomes $[2 * \{ (5-1) + (7-1) + (8-1) \} + \{ 9 + 11 \} - 1] = 53$ address locations. Thus RRNS decoding corrects single residue digit errors that are not corrected by Viterbi decoding.

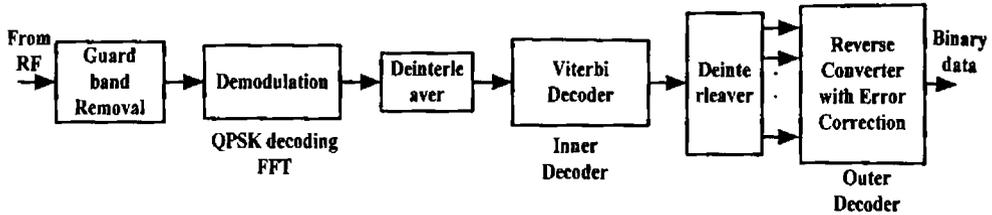


Figure 4.11 Block diagram of receiver section

The subcarriers in OFDM can add constructively and destructively. This creates the potential for a large variation in the signal power resulting in a large peak to average power ratio (PAPR). The PAPR is defined as in (4.8):

$$PAPR = \frac{\max_{t \in [0, T]} |S(t)|^2}{\varepsilon \{ |S(t)|^2 \}} \quad (4.8)$$

where $|S(t)|^2$ is the instantaneous power of the transmitted signal, T is the symbol duration and $\varepsilon\{\}$ indicates expectation. The large dynamic range of OFDM systems presents a particular challenge for the power amplifier (PA) design. The PA is required to operate in the linear region to minimize the amount of distortion and to reduce the amount of out-of-band energy generated by the transmitter. This means that OFDM needs to keep its average power low in order to accommodate the signal power peaks. It corresponds to lower output power for the majority of the signal in order to accommodate the infrequent peaks. However, lowering the average power affects the efficiency and range. Peak power clipping is a solution to reduce PAPR. The amount of peak clipping can be increased with a proper coding scheme without affecting the BER performance.

The simulation results that show the improved performance of the OFDM system with RCCC scheme under different operating conditions are given in Section 6.8.

4.3 Summary

A novel sigma-delta based parallel analog-to-residue converter is presented here. It exhibits superior performance over Nyquist rate A/R converters in terms of high resolution, high conversion speed and low cost implementation. The RNS based decimation filter channels generate all residues in parallel and hence the whole conversion operation becomes faster. The simulation results obtained for A/R converters of various resolutions are given in Section 6.7. The new A/R converter is of significant interest in high resolution and high speed data conversions for wideband wireless applications.

An RRNS-convolutional concatenated coding (RCCC) scheme for OFDM based wireless communication system is also presented. The concatenated code uses RRNS code as the outer code and convolutional code as the inner code. Errors that are not corrected at the receiver by Viterbi decoding will get corrected by the RRNS decoding due to the redundancy introduced. The RCCC scheme offers improved BER performance in presence of additive white Gaussian noise and multipath delay spread. This coding scheme makes the OFDM system more robust against multipath effects and timing errors. Also, the signal can be heavily clipped to reduce the PAPR without significant increase in BER for the RCCC OFDM system. The simulation results of the new system under different operating conditions are presented in Section 6.8.

Chapter 5

Easily Testable Circuits for MAC Units

This chapter presents an easily testable realization of multiply and accumulate (MAC) units using Positive Polarity Reed-Muller expressions (PPRMs). An exhaustive branching algorithm to implement any logic function in RM form using Reed-Muller Universal Logic Modules (RM-ULMs) is presented. This approach reduces the delay and hardware requirement for synthesizing logic functions. A Genetic Algorithm (GA) based exhaustive branching approach for combinational logic synthesis using ULMs is also presented. The search algorithm finds an implementation that uses only a particular ULM with minimum number of modules and levels for any function.

5.1 Reed-Muller Expressions

Every Boolean function can be expressed in the form of Reed-Muller (RM) expression using AND and XOR operators. The AND-XOR algebra forms a complete Boolean algebra. This representation has various advantages such as ease of complementing and testing [Reddy, 1972]. The RM representations may be shorter with a reduced number of product terms leading to smaller circuits on-chip over the conventional descriptions [Harking, 1990]. Several papers have been published discussing the design and minimization techniques for RM logic, derivation of various polarities, as well as conversion between RM and Boolean forms [Sasao, 1993a], [Miller and Thomson, 1995], [Varma and Trachtenberg, 1991].

The following are the basic theorems of the XOR operator. The XOR logic has double duality property which can be verified by applying inversion theorem and transposition theorem. This makes the XOR logic a very flexible system of logic.

$$\begin{aligned}
 \text{Basic theorems:} \quad & x \oplus x = 0 \\
 & x \oplus x' = 1 \\
 & x \oplus 0 = x \\
 & x \oplus 1 = x'
 \end{aligned} \tag{5.1}$$

$$\begin{aligned}
 \text{Inversion theorems:} \quad & (x \oplus y)' = x' \oplus y = x \oplus y' \\
 & x' \oplus y' = x \oplus y
 \end{aligned} \tag{5.2}$$

$$\text{Commutative law:} \quad x \oplus y = y \oplus x \tag{5.3}$$

$$\text{Associative law:} \quad (x \oplus y) \oplus z = x \oplus (y \oplus z) \tag{5.4}$$

$$\text{Distributive law:} \quad x(y \oplus z) = xy \oplus xz \tag{5.5}$$

$$\text{Disjunction theorem:} \quad \text{If } f = g \oplus h \text{ and } gh = 0, \text{ then } f = g + h \tag{5.6}$$

Transposition theorem: If $f = g \oplus h$, then $g = f \oplus h$ and $h = g \oplus f$ (5.7)

A function given in AND-OR form can be expressed in AND-XOR form using disjunction theorem. The function in terms of OR operators is expanded to minterm form initially, so that all the terms are disjoint. Then the OR operators can be directly replaced with XOR operators. As an example, the canonical form of a three variable function in AND-XOR logic is given in (5.8).

$$f(a,b,c) = \alpha_0 a'b'c' \oplus \alpha_1 a'b'c \oplus \alpha_2 a'bc' \oplus \alpha_3 a'bc \oplus \alpha_4 ab'c' \oplus \alpha_5 ab'c \oplus \alpha_6 abc' \oplus \alpha_7 abc \tag{5.8}$$

where $\alpha_i = 1$ or 0 depending on i^{th} minterm is present or not. An alternate canonical form is obtained as in (5.9) by expanding all complemented variables using the basic theorem $x' = x \oplus 1$.

$$f(a,b,c) = \beta_0 \oplus \beta_1 a \oplus \beta_2 b \oplus \beta_3 c \oplus \beta_4 ab \oplus \beta_5 ac \oplus \beta_6 bc \oplus \beta_7 abc \tag{5.9}$$

where $\beta_i = 1$ or 0 depending on i^{th} term is present or not. This can be generalized for any arbitrary number of variables. There are various classes of AND-XOR expressions as defined below [Sasao, 1993b] [Sasao, 1997].

Positive Polarity Reed-Muller Expression (PPRM)

When an arbitrary n -variable function is represented as in (5.10), it is called a positive polarity Reed-Muller (PPRM) expression.

$$f(x_1, x_2, \dots, x_n) = \beta_0 \oplus \beta_1 x_1 \oplus \beta_2 x_2 \oplus \dots \oplus \beta_n x_n \oplus \beta_{12} x_1 x_2 \oplus \beta_{13} x_1 x_3 \oplus \dots \oplus \beta_{n-1n} x_{n-1} x_n \oplus \dots \oplus \beta_{12\dots n} x_1 x_2 x_3 \dots x_n \tag{5.10}$$

For a given function the coefficients β 's are unique. Hence PPRM is a canonical representation. All the literals are positive for PPRM.

Fixed Polarity Reed-Muller Expression (FPRM)

The fixed polarity Reed-Muller (FPRM) expression allows only one polarity for each input variable. Each variable x_i in (5.9) can choose either of positive (x_i) or negative ($\overline{x_i}$) polarity. Thus, there are 2^n different sets of polarities for an n -variable function. There exists a unique set of coefficients $(\beta_0, \beta_1, \dots, \beta_{12\dots n})$, for a given function and for a given set of polarity. Thus FPRM is a canonical representation.

Generalized Reed-Muller Expression (GRM)

There is no restriction on the allowed polarities of input variables in generalized Reed-Muller (GRM) expressions. The variables can take both positive and negative polarities, but it does not allow the same set of variables in more than one product term. Each of the $n2^{n-1}$ literal in (5.9) can take two polarities. So there are $2^{n2^{n-1}}$ different sets of polarities for an n -variable function. As there exists a unique set of coefficients $(\beta_0, \beta_1, \dots, \beta_{12\dots n})$ for a given set of polarities, GRM forms a canonical expression for a logic function.

Exclusive OR Sum-of-Products Expressions (ESOP)

Arbitrary product terms combined by XOR operators is called exclusive OR sum-of-products (ESOP) expressions. It has no restrictions on the allowed polarities of variables or on the allowed product terms. It is the most general form of AND-XOR expression [Kalay et al., 2000]. The different Reed-Muller forms follow the relationship: $PPRM \subseteq FPRM \subseteq GRM \subseteq ESOP$.

5.2 Easily Testable Circuits

A Boolean function can be implemented in different circuit design methods. Each realization may require different number of test vectors. The logic function implementation in AND-XOR logic plays an important role in design-for-testability. Several researchers have explored various design techniques and testability issues of AND-XOR logic. Reddy showed that only $n + 4$ test vectors are needed to detect all single stuck-at faults in a PPRM network, where n is the number of input variables [Reddy, 1972]. Later, Saluja and Reddy extended the results for detection of multiple stuck-at faults [Saluja and Reddy, 1975]. But the realization uses a cascaded XOR structure where the propagation delay is large. Pradhan introduced a test method for multiple fault detection in ESOP circuits [Pradhan, 1978]. The testing of a multilevel two-input XOR tree is done by a test set consisting of four vectors irrespective of the depth of the tree [Rahaman et al., 2004]. The XOR tree propagates any single fault to the output [Dubrova and Muzio, 1996]. This property minimizes the number of test vectors needed for fault detection and simplifies the test pattern generation for RM circuits. Sasao presented a GRM implementation to detect multiple stuck-at faults where a XOR tree structure is used to reduce circuit propagation delay [Sasao, 1997]. Kalay et al. presented an ESOP implementation with a minimal universal test set of size $n + 6$ to detect all possible single stuck-at faults [Kalay et al., 2000]. A bit parallel multiplier over Galois field with a constant test set of length 8 to detect all the single stuck-at Faults is presented [Rahaman et al., 2007].

The adders and multipliers in MAC units of a FIR filter can be implemented in RM form by taking the easiness of testing into consideration.

The basic element of a MAC unit is a full adder whose sum and carry outputs are expressed in RM form as in (5.11).

$$\begin{aligned} s &= a \oplus b \oplus c_{in}, & \text{and} \\ c_{out} &= ab \oplus bc \oplus ac_{in} \end{aligned} \quad (5.11)$$

The present decimation filters designed for dual-mode operation has MAC units using adders and ROMs as the basic elements. The AND-XOR realizations of adders are obtained with full adders implemented in RM form. The simulation results using the test tool ATALANTA that show the number of test vectors and the test patterns needed to test the full adder and the adders of various sizes implemented in both AND-OR and AND-XOR forms are given in Section 6.9.

5.3 Combinational Logic Synthesis using Reed-Muller Universal Logic Modules

RM functions can be implemented using discrete components or more conveniently by Reed-Muller universal logic modules (RM-ULMs). An RM-ULM is a device with c -control inputs, 2^c data inputs and a single output $f(c)$ and is designated as RM-ULM(c). The behaviour of this module is described as in (5.12):

$$\begin{aligned} f(c) &= b_0 \oplus b_1 x_1 \oplus b_2 x_2 \oplus b_3 x_2 x_1 \oplus \dots \oplus b_{2^c-1} x_c x_{c-1} \dots x_1 \\ &= \bigoplus \sum_{i=0}^{2^c-1} b_i P_i \end{aligned} \quad (5.12)$$

where $b_i = 0$ or 1 , and the product term (or piterm) P_i is,

$$P_i = x_{i_c} x_{i_{c-1}} \dots x_{i_1} \quad \text{where } i = \sum_{j=0}^{c-1} 2^j x_j \quad (5.13)$$

x_{i_k} will be present in P_i if the k^{th} bit of binary representation for i is 1 . The logic symbol for RM-ULM(c) is shown in Figure 5.1.

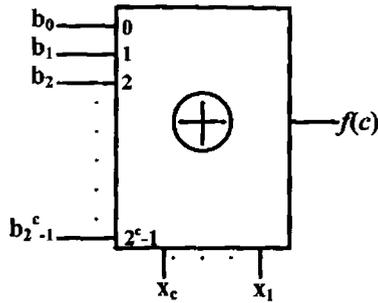


Figure 5.1 Logic symbol of RM-ULM(c)

VLSI implementations using only one type of modular building blocks can decrease system design and manufacturing cost. Circuit delay and cost can be reduced by using RM-ULMs connected in tree structure for functions in RM form. A tree network is very suitable for VLSI realization because of the uniform interconnection structure and the repeated use of identical modules.

The use of RM-ULM for realization of logic functions has already been explored by researchers. Programmed algorithms have been developed for optimization of number of modules at sub-system level in a tree network [Xu et al., 1993], [Almaini et al.1992]. The algorithm looks for possible cascade networks, and if it is not found a tree structure is implemented. Tan and Chia presented an alternate algorithm which performs similar optimization of fixed polarity Reed-Muller expansions (FPRM) with a reduced computation time [Tan and Chia, 1996]. The above algorithms do not explore all the possible branching options of the tree structure and hence the delay of the circuit synthesized may not be minimal.

In this research, further delay reduction is achieved by using a novel tree-structured exhaustive branching network using RM-ULM(1) for implementing a logic function given in positive polarity Reed-Muller (PPRM)

form. A logic function with n -variables can be implemented using 2^n-1 RM-ULM(1)s in n -levels by standard implementation. Any implementation using less than 2^n-1 number of modules and / or lesser number of levels can be considered as an improvement in cost and / or speed.

5.3.1 N-ary Exhaustive Branching Technique

For a given number of input variables n , there is a well-defined number of functions, which is equal to 2^{2^n} [Correia and Reis, 2001]. Standard implementation of a tree network requires n -levels to implement these functions. Xu presented a programmed algorithm to reduce the complexity of the network in terms of number of modules and levels. In his approach 1's, 0's, \hat{x}_i (where \hat{x}_i is a variable x_i or its complement x_i' , $1 \leq i \leq n$) or functions using any number of variables can be given to any data inputs of the RM-ULM. But the control inputs accept variables only. In this research, the performance is further improved by an exhaustive branching technique with \hat{x}_i or functions of two variables at control input. Since x_i' or functions are also given to control input, the utilization of all branching options are made possible. This decreases the number of levels, and hence the delay is reduced for any logic function implementation using RM-ULMs.

The first level (output stage) will have a single RM-ULM, the second level will have a maximum of $(2^c + c)$ RM-ULMs, where c is the number of control inputs ($c = 1$ in this case) and so on. In general, the maximum number of RM-ULMs in a level can be expressed as $(2^c + c)^{L-1}$ where L indicates the number of levels. The maximum number of RM-ULMs, N in the complete network having L levels is given in (5.14).

$$N = \sum_{x=1}^L (2^c + c)^{x-1} \quad (5.14)$$

A network with 1-level can realize functions up to 3 variables, since there are 3 inputs. By connecting x_i to the control input, the remaining \hat{x}_j variables ($j \neq i$) or constants (0 or 1) can be connected to each of the 2 data input lines. So there are 6 possible values for each data input line, resulting in 6^2 functions. Selecting 1 variable as control input from the total of 3 variables and its complements, can take $6C_1$ combinations. Out of 62 distinct functions implemented at level 1, 24 are 3 variable functions which require 3 levels in standard implementation.

Level 2 allows the implementation of functions having maximum of 9 variables, using 3 control lines and 6 data lines. Selecting 3 variables from the total of 9 variables and its complements, results in $18C_3 \times 3!$ combinations. The remaining 6 variables and its complements or constants (0 or 1) at 6 data lines give rise to 14^6 distinct functions with one combination at control input. In the tree structure given by Xu, at level 2 maximum number of variables possible is only 7, which result in 10^4 distinct functions with one combination at control input. The exhaustive branching approach increases the number of variables and functions that can be implemented in level 2. As the number of levels increases this difference becomes more and more significant, and more delay reduction can be achieved for functions with large number of variables. In general with L levels, the number of functions that can be implemented using RM-ULMs in the exhaustive branching method is $[2(y+1)]^y$ for one combination at control inputs. The number of combinations possible at control inputs is $\{2z^L C z^{L-1}\} \times \{z^{L-1}!\}$ where $y = (z^L - z^{L-1})$ and $z = (2^c + c)$. Maximum number of variables at level L is

$n_{max} = z^L$. For a given function if there are n dependent variables, the levels L required for implementation in this approach is given as $\lceil \log_{(2^c + c)} n \rceil \leq L \leq (n - 1)$, whereas in the tree structure L can be in the range $\lceil \log_{(2^c)} n \rceil \leq L \leq (n - 1)$. This clearly demonstrates a reduction in delay attained by the exhaustive branching technique over the implementation using tree structure.

5.3.2 Exhaustive Branching Algorithm

Behavior of an RM-ULM(1) can be expressed as $F_j \oplus F_s F_k$, where F_s , F_j and F_k are functions of t variables ($1 \leq t \leq n$). The number of variables of F_s , F_j and F_k varies according to the complexity of the function to be realized. The maximum number of variables in F_s , F_j or F_k determines the delay of the network. The network terminates when F_s , F_j and F_k are 1's, 0's or \hat{x}_i ($1 \leq i \leq n$). If all inputs except one terminate with a variable \hat{x}_i or a logical constant and only one input continues into the next level, a cascade is generated where a single module is used in each level. The new algorithm aims to identify \hat{x}_j or functions of 2 variables at each control input, that eliminate as many branches as possible and reduce the number of levels and modules required for implementation. The algorithm for any function given piterms (P_i), is as follows:

Exhaustive Branching Algorithm:

Step 1: Get the piterms in decimal, and the number of variables, n . Set level, $L = 1$, number of modules, $M = 1$.

Step 2: List the piterms in n -bit binary as a piterm table.

Step 3: Check whether any column in the table is all zeros. Eliminate the variable corresponding to that column and get the reduced piterm table.

Step 4: Get the reduced piterm tables for each variable x_i (one table for $x_i = 1$ and another table for $x_i = 0$) and find the x_i for which the reduced piterm tables correspond to constants (0 or 1) or \hat{x}_j ($j \neq i$) by checking the number of ones in each piterm table, c_I . If $c_I \leq 1$, terminate.

Step 5: For each x_i , check the following conditions:

- (i) Number of zeros \geq number of ones
- (ii) For each (1, 0) pair, the remaining bits are constants
- (iii) Number of such pairs is equal to 2
- (iv) One pair has remaining bits as all zeros and the other has ones in one column only.

Terminate if all the above conditions are satisfied as implementation is obtained with $(1 \oplus x_i)$ at the control input.

Step 6: $L = L + 1$, $M = M + 1$. Get the reduced piterm tables for each variable and find the x_i for which the following conditions are satisfied.

- (i) One reduced piterm table corresponds to a constant (0 or 1) or \hat{x}_j ($j \neq i$).
- (ii) The other reduced piterm table is a single module implementation by repeating the steps 4 & 5.

Step 7: Get reduced piterm tables for each possible $(1 \oplus x_i)$ (by checking conditions (i) & (ii) of step 5), and find the $(1 \oplus x_i)$ for which the conditions (i) & (ii) of step 6 are satisfied.

Step 8: $M = M + 1$. Get the reduced piterm tables for each variable, and find the x_i for which the reduced piterm tables are single module implementations by repeating the steps 4 & 5.

Step 9: Get reduced piterm tables for each possible $1 \oplus x_i$ (by checking conditions (i) & (ii) of step 5), and find that $1 \oplus x_i$ for which the reduced

piterm tables are single module implementations by repeating the steps 4 & 5.

Step 10: Get the reduced piterm tables for each possible $x_i x_j$, and find the $x_i x_j$ for which the conditions (i) & (ii) of step 6 are satisfied.

Step 11: Get the reduced piterm tables for each possible $(x_i \oplus x_j)$, and find the $(x_i \oplus x_j)$ for which the conditions (i) & (ii) of step 6 are satisfied.

Step 12: $M = M + 1$. Get the reduced piterm tables for each possible $x_i x_j$, and find the $x_i x_j$ for which both reduced piterm tables are single module implementations by repeating the steps 4 & 5.

Step 13: Get the reduced piterm tables for each possible $(x_i \oplus x_j)$, and find the $(x_i \oplus x_j)$ for which both reduced piterm tables are single module implementations by repeating the steps 4 & 5.

The synthesis results obtained for various combinational logic functions with RM-ULMs using the exhaustive branching algorithm are given in Section 6.10.

5.4 Genetic Algorithm based Approach for Combinational Logic Synthesis

An evolutionary approach based on genetic algorithm (GA) is used as the main engine to synthesize logic functions. GA has been widely used as a search technique which mimics the natural process of Evolution and Darwin's principle of "*Survival of the Fittest*". In the last decade the use of GA for the design of digital circuits has led to a novel area of research in evolutionary design called evolvable hardware. Evolvable hardware is capable of realizing optimized circuits beyond those of conventional design of logic circuits.

The use of genetic algorithm for realization of logic functions has already been explored by researchers. John Koza has used genetic algorithm to

design combinational circuits using AND, OR and NOT logic gates; but his emphasis has been on generating functional circuits rather than optimizing them [Koza, 1992]. Coello et al. presented a computer program that automatically generates high quality circuit designs using five possible types of gates (AND, NOT, OR, XOR and WIRE) that reduces the use of gates other than WIRE [Coello et al., 1996]. Miller and Thompson applied GA to minimize FPRM expansions and ESOP expansions [Miller and Thompson, 1995]. Evolutionary algorithms applied for the design of arithmetic circuits is also presented [Fogarty et al., 1998]. Torresen presented an evolutionary method in order to solve complex problems by applying divide and conquer method [Torresen, 1998]. An evolutionary approach to synthesize combinational circuits using different sets of gates of varying complexity was done [Reis and Machado, 2003]. In all these cases different types of gates were used to synthesize the function. However, the use of the different types of gates may not be realistic in VLSI systems design where the emphasis is to reduce the manufacturing cost rather than the number of components used. VLSI implementations using single type of modular building blocks can reduce the system design and manufacturing cost. Aguirre and Coello presented genetic programming approach to synthesize Boolean functions using multiplexers [Aguirre et al., 1999], [Aguirre and Coello, 2004].

5.4.1 Universal Logic Modules (ULMs) for Logic Synthesis

The algorithm uses NAND gate, NOR gate, multiplexer and Reed-Muller module as ULMs for realizing any logic function specified as minterms. Multiplexer implements the function in AND-OR form and the behaviour of a multiplexer with n -select inputs, 2^n data inputs and a single output F is given by (5.15):

$$F = \sum_{i=0}^{2^n-1} x_i m_i(s) \quad (5.15)$$

where $m_i(s)$ is the i^{th} minterm of the n -select variables and x_i is the i^{th} data input. Reed-Muller expressions are more advantageous than conventional expressions for XOR intensive applications such as error detection, arithmetic circuits etc. This AND-XOR representation has various advantages such as ease of complementing and testing. It may require lesser number of product terms leading to smaller circuits on-chip than the conventional implementations. RM functions can be implemented using discrete components or more conveniently by RM-ULMs. The behaviour of RM-ULM(c) with c -control inputs, 2^c data inputs and a single output $f(c)$ is described as in (5.12). More specifically, the ULMs considered in this research are 3-input NOR gate, 3-input NAND gate, single control line multiplexer (2:1 MUX) and a single control line Reed-Muller ULM (RM-ULM(1)), each having 3-inputs and 1-output. A tree network is very suitable for VLSI realization because of the uniform interconnection structure and the repeated use of identical modules. Following this line of research, a GA-based evolutionary synthesis of combinational circuits using appropriate ULMs is presented here.

5.4.2 GA based Approach for Logic Synthesis

GA-based synthesis of combinational circuits specified by minterms using universal logic modules is considered here. The implementation is in the form of a tree-structured exhaustive branching network using single type of ULMs. The measure of circuit optimality is defined in terms of total number of ULMs used and the number of levels required. The algorithm searches

for the type of ULM to be used for realizing the circuit with minimum number of modules and levels. Thus the resulting implementation will have a reduced delay and/or power compared to those using other ULMs.

GA is a population based approach in which solution to a problem is encoded in the form of string of characters called chromosome [Goldberg, 1989]. The first aspect of this problem is encoding of solutions. Each circuit is encoded in the form $\langle input1 \rangle \langle input2 \rangle \langle input3 \rangle$ where ' $input\ i$ ' represents the input to each of the 3-input ULM. A chromosome is formed with as many triplets of this kind as needed for realizing a function. The number of bits of the chromosome depends on the total number of input combinations possible for each module. The initial population of circuits is generated at random and the algorithm searches for a solution among them. "Fitness" value of a chromosome tells how "good" the chromosome is. These initial chromosomes are evaluated using a fitness function, and a fitness value ' f_i ' is assigned for each chromosome.

The three different genetic operators used are reproduction, crossover and mutation. *Reproduction* is a process in which individual strings are copied from the old population to the new population according to their fitness function values f_i . The reproduction operator may be implemented in algorithmic form in a number of ways such as, Roulette wheel selection, Boltzman selection, Tournament selection, Rank selection etc. This research uses a biased Roulette Wheel with slots sized in proportion to its fitness value. The better the fitness value of the chromosome, the greater the chances that it will be selected. However it is not guaranteed that the fittest member goes to the next generation. For the *crossover* operator, the strings in the new

population are grouped together into pairs at random. A crossover point is randomly selected, and then a single point crossover is performed among pairs. After a crossover is performed, the resulting solution may fall into a local optimum. Hence some genes of the child chromosome are randomly changed by *Mutation*. It is done by random alteration of the value of a string position. In binary coding, this simply means changing a 1 to a 0 and vice versa. When creating a new population by crossover and mutation, the best chromosome may be lost. Hence Elitism, which is a method for copying the best chromosome to the new population prior to crossover and mutation, increases the performance of GA.

The new algorithm works on the principle of genetic algorithm and is implemented level by level. The steps involved are as follows:

Step 1: Get the minterms and number of variables, and convert it to a truth table. Set level $L = 1$, and number of modules $M = 1$.

Step 2: Select a suitable encoding for the chromosome and generate an initial random population.

Step 3: Select a particular type of ULM. Set the number of iterations to N .

Step 4: Compute the fitness function for each chromosome. Assign $N = N - 1$.

Step 5: If fitness value $f_i = 1$, the objective is fulfilled and terminate. Else, generate an intermediate population using Roulette wheel, and apply crossover and mutation.

Step 6: If $N \neq 0$, repeat from step 4, for the same ULM, else repeat from step 3 until all the ULMs are considered.

Step 7: Assign $M = M + 1$, and check whether all possible branching options are considered at the current level. If No, repeat from step 2, else assign $L = L + 1$, and search for a solution at the next higher level.

The search at a particular level continues for an optimum number of generations, and if no solution is found another type of ULM is considered. If the objective is not attained with any type of ULM, the search moves on to the next higher modular level. After getting a solution with fitness value 1, further optimization is done by checking whether there are modules that implement the same sub-functions in the network. The synthesis results obtained for various functions using GA based approach are demonstrated in Section 6.11.

5.5 Summary

The Boolean functions implemented in the form of Reed-Muller expressions provide ease of testability. Adders of different sizes are implemented in both RM form and conventional AND-OR form. The test patterns and the fault free responses for detecting single stuck-at faults are found out using the test tool ATALANTA. The test results show that the test set size for 100% fault coverage in RM circuit is less than that for AND-OR circuit as shown in Section 6.9. The MAC units described basically consist of adders and ROM modules. Hence, easily testable MAC units for a filter structure can be obtained by implementing adders in RM form.

An exhaustive branching algorithm for the synthesis of RM-ULM tree network is presented. The delivered network has reduction in delay and complexity in terms of number of modules, compared to the existing implementations. The logic synthesis results obtained for various functions using the exhaustive branched algorithm are given in Section 6.10. By suitable selection of variables, its complements or functions as control inputs, the number of modules and delay are reduced. Theoretically, the algorithm can handle any number of variables for any completely specified logic

function. The computation time is not always directly proportional to the number of variables, but this increases with the complexity of the function to be realized. Since the topology of the delivered network is that of a tree, VLSI implementation of this network requires very few extra works in routing algorithms to redesign or for circuit layout.

A genetic algorithm-based logic synthesis using exhaustive branched ULM network is also presented. The algorithm searches and finds an appropriate ULM so that the evolved network has minimum number of modules and levels. This optimization in turn results in reduction of power consumption and delay. The synthesis results obtained using the GA based approach for various functions are given in Section 6.11.

Chapter 6

Simulation Results and Analysis

This chapter presents the simulation results of various new reconfigurable architectures and circuits designed for wireless transceivers as part of this research. This include Multi-standard Decimation filter Design Toolbox, simulation results of Polyphase non-recursive comb decimators, RNS based dual-mode decimation filters reconfigurable for WCDMA/WiMAX and WCDMA/WLANa standards, sigma-delta based analog-to-residue converters, RRNS-convolutional concatenated coding scheme for OFDM communication system, and easily testable realization of MAC units using RM form. Also, The combinational logic synthesis results obtained using a new exhaustive branching algorithm and a GA based approach are presented. The salient features of the new system over other existing systems are compared and the results are tabulated.

6.1 Decimation Filter Implementation and Analysis

The individual stages of multistage decimation filter for a new 'Multi-standard decimation filter design toolbox' are designed to minimize hardware complexity as well as computational effort. For each stage, passband edge is same as 80% of the channel bandwidth, and stopband edge is selected to reduce the filter complexity while meeting the overall standard specification. The first stage, being a CIC filter, reduces the hardware as it consists of only adders and registers. But it exhibits passband droop and insufficient attenuation in the stopband. So the following filters are designed to compensate for the droop and to meet the overall filter specification for a particular standard. For GSM and WCDMA the second stage is selected as a halfband filter which has almost half the coefficients as 'zero'. The complexity of halfband filter is reduced by allowing a symmetrical transition band about $F_s/4$, where F_s is the sampling frequency at halfband input. So the stopband edge is relaxed for halfband filter. The third stage is a FIR filter that performs decimation as well as droop compensation in the passband. A three stage decimation filter is implemented for GSM and WCDMA. For WLANa, WLANb, WLANg and WiMAX decimation filter is implemented in two stages. The first stage is a CIC and the second stage is a CIC droop compensation FIR filter. The last stage of a decimation filter always has passband and stopband edges to meet the standard specification. Usually it will be more complex compared to the preceding stages, but operates at a reduced sampling frequency. Table 6.1 shows the decimation filter implementation details such as the type of filter used, the decimation factors, and the number of filter coefficients for each stage of all the six standards.

Table 6.1 Decimation filter implementation results for multiple standards

Standards	Modulator quantizer bits	OSR	Filter structure	Decimation factor	Filter length/No. of Sections
GSM	1	128	CIC	32	3
			Halfband	2	11
			FIR	2	101
WCDMA	1	16	CIC	4	4
			Halfband	2	19
			FIR	2	48
WLANa	4	8	CIC	4	9
			CIC compensation FIR	2	32
WLANb	4	12	CIC	4	7
			CIC compensation FIR	3	38
WLANg	4	12	CIC	4	6
			CIC compensation FIR	3	38
WiMAX	4	8	CIC	4	4
			CIC compensation FIR	2	36

The user can select required wireless communication standard, and obtain the corresponding multistage decimation filter implementation using this toolbox. The toolbox will help the user or design engineer to perform a quick design and analysis of decimation filters for multiple standards without doing extensive calculation of the underlying methods. The tool provides the user with all necessary details of decimation filter designed for the selected standard including filter coefficients, frequency response, pole-zero plot etc. The multistage decimation filter implementation reduces the hardware complexity and computational effort while meeting the standard requirements. The OSR for each standard is selected to get the required dynamic range with a particular sigma-delta modulator order and number of quantizer bits. CIC is

used as the first stage which is a simple structure consisting of only adders and registers. Using a halfband filter, with almost half the coefficients ‘zero’ in the next stage, provides further reduction in filter complexity. The last stage is a complex FIR filter which meets the overall standard specification but it operates at a reduced sampling frequency. The reduction in number of coefficients in each filter stage promises better synthesis results in terms of circuit compactness and power dissipation.

6.2 Simulation Results and Analysis of Polyphase Non-recursive Comb Decimation Filter

The polyphase implementation is compared mainly with the recursive or Hogenauer CIC [Hogenauer, 1981] and the non-recursive CIC implementations [Gao et al., 1999]. The filter architectures are defined using VHDL codes and functional simulation is performed using *ModelSim*. The filter responses obtained with the three implementations are identical.

Power, speed and area analysis is done with the *Synopsys design compiler* using 0.18 μm , 1.8V CMOS technology. Synthesis is done for 4th order CIC filter ($k = 4$) using three different architectures, and a comparison of performance is done for different decimation factors as $R = 64, 128$ and 256. In all the implementations differential delay, ‘ M ’ is assumed as ‘1’. Table 6.2 shows the power comparison for the three implementations with an input word length, $B_{in} = 4$. The synthesis results show that for a 4th order CIC with decimation factor of $R = 64$, the polyphase implementation has about 70.02 % and 36.93% of power saving compared to the corresponding Hogenauer CIC and non-recursive implementations respectively. The power comparison plot is given in Figure 6.1

Table 6.2 Total dynamic power consumption for CIC architectures

Filter type	Dynamic power consumption in mW		
	R = 64	R =128	R = 256
Hogenuer CIC	4.6950	5.3245	5.9968
Non-recursive CIC	2.2318	2.5398	2.8412
Polyphase CIC	1.4077	1.6129	1.815

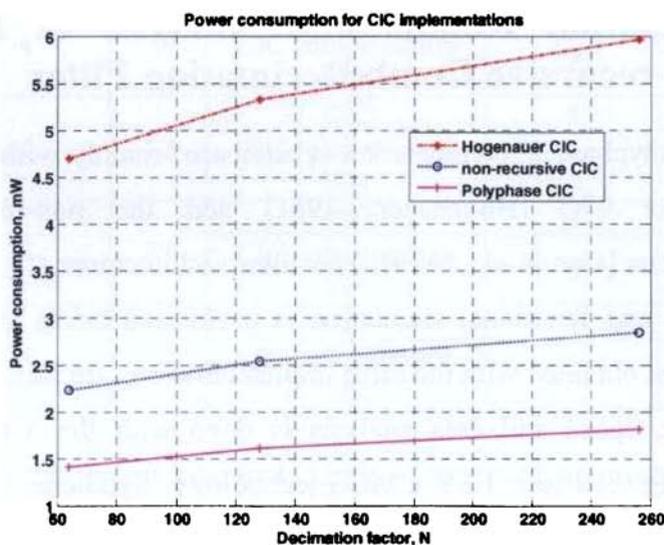
**Figure 6.1** Power consumption for CIC architectures with $k = 4$ and $B_{in} = 4$

Table 6.3 shows the maximum speed at which each implementation can be operated. It is observed that polyphase structure is about 7 times faster than Hogenuer CIC and 3.7 times faster than the non-recursive structure. The speed up is due to the smaller register sizes for the initial stages and the reduced sampling frequency of operation for the final stages. Thus it is well suited for higher data rate applications.

Table 6.3 Highest operating frequency for CIC architectures with $k = 4$,
 $R = 64$ and $B_{in} = 4$

Filter type	Highest operating frequency, MHz
Hogenuer CIC	64.7
Non-recursive CIC	122.6
Polyphase CIC	463.1

The total area required for filter implementation has been found to be more for polyphase CIC and non-recursive implementation than that for recursive structure. The increase in area requirement for polyphase decomposition is due to the additional adder required for multiplication and the extra decimator switch required in each stage. The synthesis results obtained for 4th order CIC filter with $B_{in} = 4$ are given in Table 6.4. It shows the area requirement of each CIC structure for different decimation factors as $R = 64, 128$ and 256 . The area comparison plot is given in Figure 6.2.

Table 6.4 Area requirement for CIC architectures

Filter type	Total area occupied, μm^2		
	$N = 64$	$N = 128$	$N = 256$
Hogenuer CIC	30946	35367	39788
Non-recursive CIC	44832	59053	75202
Polyphase CIC	56554	75004	96079

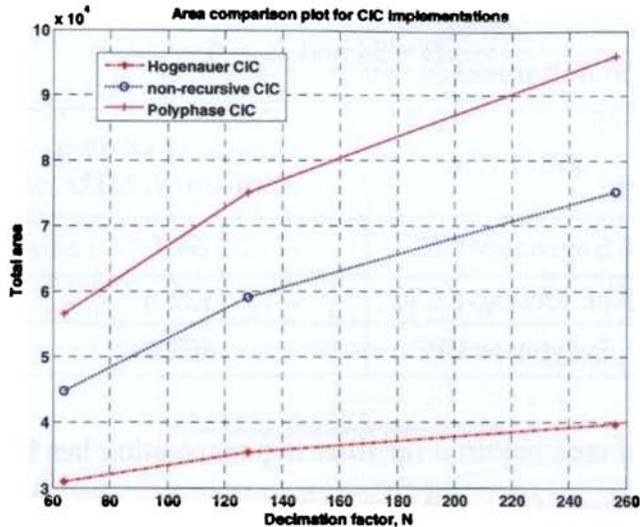


Figure 6.2 Area requirement for CIC architectures with $k = 4$ and $B_m = 4$

The polyphase decomposition of non-recursive structure has the advantages of low power consumption and high speed operation compared to the recursive and non-recursive implementations. Low power consumption is achieved due to the fact that the word length is small for the initial stages which operate at high sampling rate, and as the word length increases for the subsequent stages the sampling rate is decreasing. Also, the computational complexity per input sample is reduced for each stage of polyphase structure than that for the non-recursive implementation. The maximum speed of operation of the polyphase structure is improved by the smaller word length of the first stage compared with that for recursive structure. As the first stage is operating at half the sampling rate and as parallel processing is done with polyphase decomposition, further speed improvement is obtained compared to the non-recursive implementation. The area requirement seems to be high for polyphase and non-recursive CICs than that for recursive structure. So the designer has to select the architecture of CIC based on the system

requirements of power consumption, speed of operation and silicon area. Due to the low power and high speed operation, the polyphase CIC filters find applications in digital radio receivers, wireless communication systems, digital RF/IF signal processing and many others.

6.3 Performance Analysis of FIR Filter Implementation: RNS Versus Traditional

The FIR filter architectures are designed for traditional and RNS domain operations. To compare the filtering operation, both are simulated for 64 taps with a passband and stopband frequency of 500 Hz and 1400 Hz respectively. The frequency response of the filters is shown in Figure 6.3. The original signal consists of 500 Hz message signal along with 3000 Hz noise. The power spectral density (PSD) of two types of filtered outputs are shown in Figures 6.4. The noise attenuation is exactly the same in both cases.

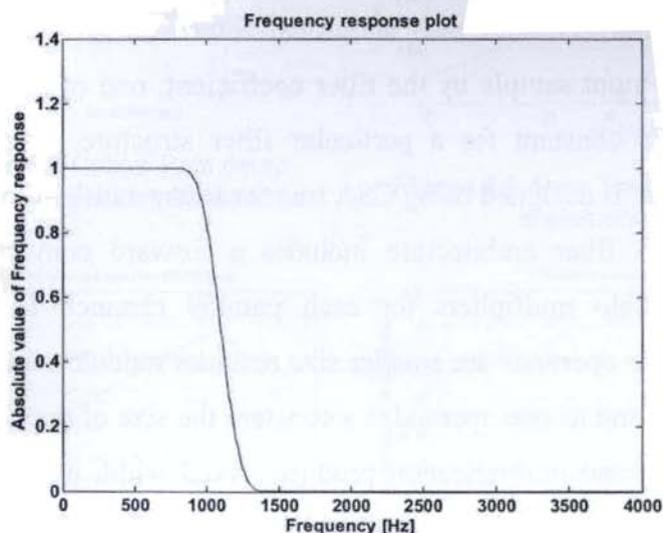


Figure 6.3 Frequency response of FIR filter

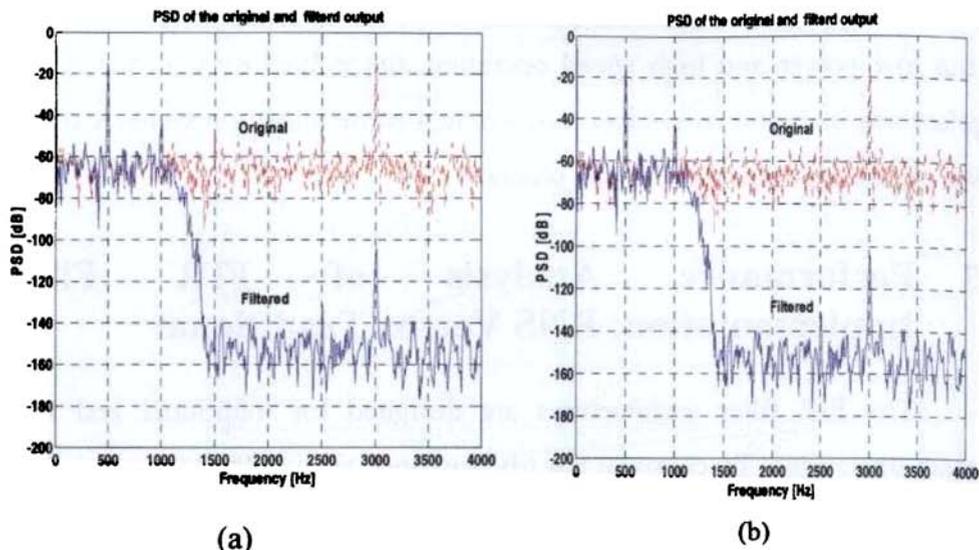


Figure 6.4 PSD plot of original and filtered output (a) Traditional (b) RNS

Hardware analysis is done with the logic synthesis tool *Leonardo Spectrum* from Mentor Graphics Corporation, using ASIC library. The traditional filter implementation uses carry propagate adders whose size increases progressively with each additional filter tap. As the multiplier is multiplying the input sample by the filter coefficient, one of the operands of the multiplier is constant for a particular filter structure. Accordingly a reduced hardware is designed using CSA tree for faster multiplication.

The RNS filter architecture includes a forward converter, modulo adders and modulo multipliers for each parallel channel, and a reverse converter. As the operands are smaller size residues modulo multiplier based on LUT is used, and as one operand is a constant the size of the LUT is small. Modulo addition and multiplication produces fixed width output so that for each additional filter tap the rate at which the hardware increases is less compared to the traditional filter.

The graphs shown in Figures 6.5 and 6.6 give a comparison of critical path delay and area respectively for a traditional and RNS filter compared to that of a full adder. Figures 6.7 and 6.8 demonstrate the speed up factor and area requirement for RNS filter compared to traditional filter as the number of filter taps increases, assuming 6-bit binary input. The graphs indicate that RNS filter becomes more than three times faster, and requires only 60% or less area than the corresponding traditional filter implementation as the number of filter taps increases above 32.

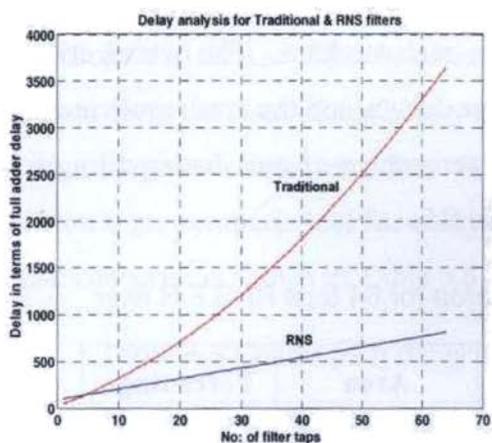


Figure 6.5 Critical Path delay: Traditional Vs RNS implementation

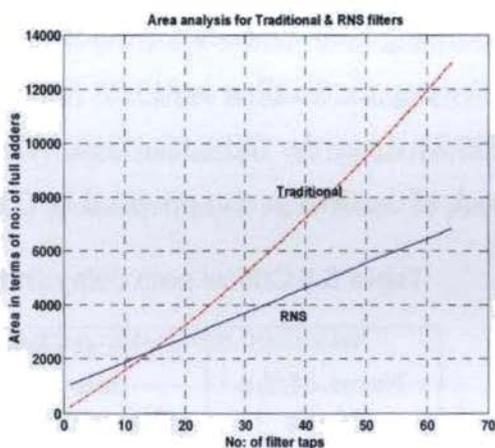


Figure 6.6 Area: Traditional Vs RNS implementation

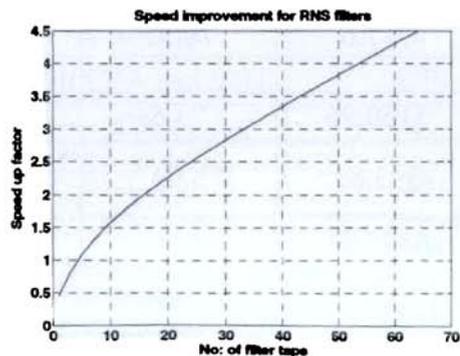


Figure 6.7 Speed up factor for RNS filter Vs traditional filter

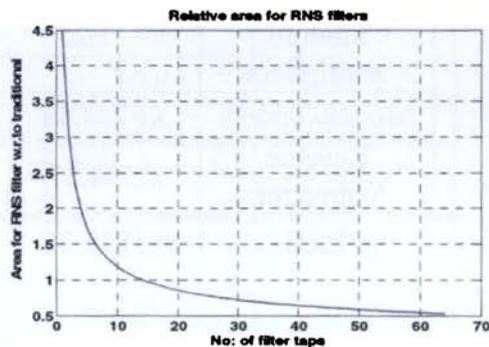


Figure 6.8 Percentage area improvement of RNS filter Vs traditional

The synthesis results obtained using Leonardo Spectrum logic synthesis tool for a 64 taps RNS FIR filter with an input word length of 6-bits, and passband and stopband frequencies of 800Hz and 1400Hz, is shown in Table 3.1. The critical path delay and area for each block of the filter is normalized with respect to a full adder critical path delay of 1.98 ns and area of $38\mu\text{m}^2$. The moduli set are selected as (23, 25, 27, 29, 31, 32) which provide 29 bit dynamic range with 446623200 unique integer representations. So, the range of negative and positive numbers represented by the selected RNS is from -223311600 to +223311599. The filter has six parallel channels processing 5-bit residues corresponding to each modulus. The overall critical path delay for the filter is 813.28 full adder delays, and the area requirement is 6825.81 times the full adder area. The percentage of area required by each block of the filter is also provided in Table 6.5.

Table 6.5 Critical path delay and area for 64 taps RNS FIR filter

Name of the block	Critical path delay (Normalized w.r.to FA)	Area (Normalized w.r.to FA)	Percentage area of each block (%)
Forward converter	13.28	603.42	8.84
LUTs for modulo multipliers	1.25	717.6	10.51
Modulo adders	724.48	5160.96	75.6
Reverse converter	74.27	343.83	5.04
Total	813.28	6825.81	100

The forward converter consists of ROMs that store the residues for each field of the binary number which are then combined using modulo adder

tree structure. As the largest value in the selected moduli set is 32, the least significant 5 bits of the binary number directly gives the residue value for the modulus 32. To produce the remaining residue digits for the other 5 channels the area consumed is 603.42 times the full adder area. The reverse converter is implemented with 6 ROMs and three levels CSA tree [Radhakrishnan et al., 1999]. A 5-bit binary adder and a LUT are used for converting the most significant field from CSA tree. The output is combined with the remaining output field of the CSA tree using another CSA and CPAs for modulo correction. Finally, a vector multiplexer is used to select the required 24-bit output from the CPAs. This is concatenated with the 5-bits of modulo 32 residue to form the final 29-bit binary number. Then the number is converted to a sign magnitude representation. The synthesized results using Leonardo Spectrum logic synthesis tool for delay and area required in each block of this reverse converter is shown in Table 6.6.

Table 6.6 Critical path delay and area of reverse converter

Name of the block	Critical path delay (Normalized w.r.t. FA)	Area (Normalized w.r.t. FA)	Percentage area of each block (%)
ROMs (6 Nos)	6.6	63.16	18.37
CSA tree	3	99	28.79
5 bit CPA and LUT	6.75	10.94	3.18
CSA	1	24	6.98
CSA and two CPAs	26	74	21.52
Vector multiplexer	0.69	15.73	4.58
Sign Magnitude converter	30.23	57	16.58
Total	74.27	343.83	100

The critical path delay and area required for a traditional filter having the same specifications using 17-bits wide filter coefficients and an input word length of 6-bits are shown in Table 6.7. As the filter coefficients and input samples are signed numbers, signed addition and multiplication are considered. Traditional filter implementation has a critical path delay of 3650.72 times the full adder delay and area of 12983.6 times full adder area. This implies the RNS filter operates 4 times faster and requires only half the area than the corresponding traditional implementation.

Table 6.7 Critical path delay and area for 64 taps traditional FIR filter

Name of the block	Critical path delay (Normalized w.r.to FA)	Area (Normalized w.r.to FA)	Percentage area of each block (%)
CSA tree based multipliers	20	6110	47.06
Ripple carry adders (signed addition)	3630.72	6873.6	52.94
Total	3650.72	12983.6	100

The speed and area comparison graphs shows that RNS filter is more than 3 times faster and requires only less than 60% of area than that for the corresponding traditional filter, when the filter length is increased above 32 taps for an input word length of 6-bits. Such compact and high speed real-time digital filters find applications in radar, communications and image processing systems.

6.4 Simulation Results and analysis of Dual-mode Decimation Filter for WCDMA/WiMAX

The input sampling frequency is 61.44 MHz for WCDMA and is downsampled to the data rate of 3.84 Mchips/s in three stages. The cascaded two stage filter structure downsamples the input sampling frequency of 133.632 MHz for WiMAX to the data rate of 16.704 Msymbols/s. The decimation filter responses obtained for WCDMA and WiMAX, satisfying the standard specifications, are shown in Figure 6.9 and 6.10 respectively. Filter responses of each stage and the cascaded overall responses are shown.

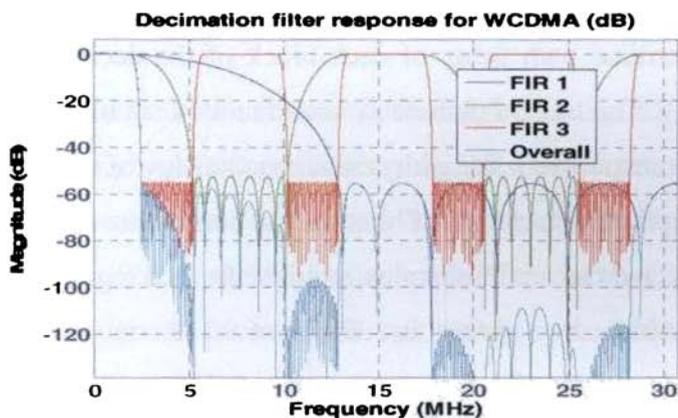


Figure 6.9 Filter responses for WCDMA mode in WCDMA/WiMAX decimator

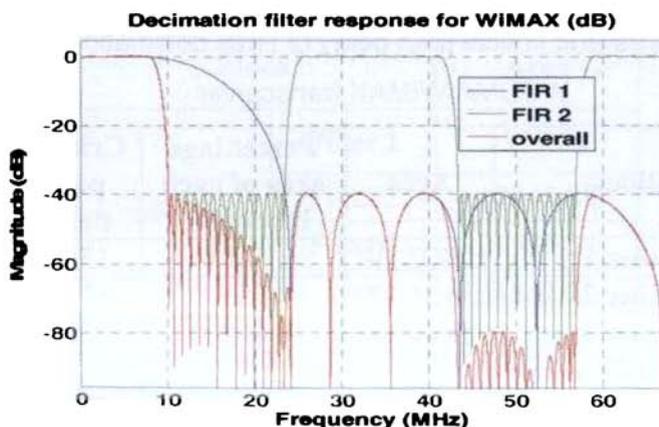


Figure 6.10 Filter responses for WiMAX mode in WCDMA/WiMAX decimator

The RNS moduli set (25, 29, 31, 37, 43, 47, 59, 64), consisting of 8 relatively prime integers of 5-bits and 6-bits lengths implements the filters without overflow. It permits filter coefficients of 14-bit accuracy and input word length of 4-bits from $\Sigma\Delta$ modulator. The hardware synthesis is done with *Leonardo Spectrum* logic synthesis tool from Mentor Graphics. In order to operate first stage filter at 133.632 MHz, pipelining is done after every two modulo adders to meet the critical path delay. Three stage pipelining is done to meet the critical path delay for second filter which operates at maximum frequency of 33.408 MHz. The third stage is used only for WCDMA mode at a frequency of 7.68 MHz and no pipelining is required. The total area requirement and critical path delay of each block of the decimation filter is shown in Table 6.8. The critical path delay and area for each block of the filter is normalized with respect to a full adder critical path delay of 0.45 ns and area of $38\mu\text{m}^2$. The area requirement of the decimation filter in single mode WCDMA receiver and the additional area required for making it adaptable for dual-mode operation are given in Table 6.9. It is observed that programmability is achieved at the expense of 24% of additional area compared to single mode WCDMA receiver.

Table 6.8 Area and critical path delay of RNS decimation filter for WCDMA/WiMAX transceiver

Block	Area	Percentage area of each block (%)	Critical path delay
Filter 1	2407.29	17.13	15.39
Filter 2	4925.37	35.1	61.17
Filter 3	5840.18	41.6	244.29
Multiplexers	29.47	0.002	0.69
Reverse converter	852.82	6.07	79.12
Total area	14055.13	100	

Table 6.9 Area requirement for programmability

Type of transceiver		Area	Percentage area (%)
Single mode WCDMA	Filter 1	2220.44	75.8
	Filter 2	1748.3	
	Filter 3	5840.18	
	Reverse converter	852.82	
	Total	10661.74	
Dual-mode Transceiver		14055.13	100
Additional area for programmability		3393.39	24.2

Table 6.10 reports the characteristics of decimation filter implemented in traditional number system performing signed multiplication and addition. RNS filter implementation offers about 64% saving of area. Pipelining done as in the RNS filter will not meet the critical path delay for traditional case. Here pipelining is to be done in the multipliers as well as in the adder chain.

Table 6.10 Area requirement for WCDMA/WiMAX decimation filter:

Traditional Vs RNS

Block		Area
Traditional	Filter 1	1379.83
	Filter 2	12945.88
	Filter 3	24536.54
	Total	38862.25
RNS		14055.13

6.5 Simulation Results and analysis of Dual-mode Decimation Filter for WCDMA/WLANa

The input sampling frequency is 61.44 MHz for WCDMA and is downsampled to the data rate of 3.84 Mchips/s in three stages. The cascaded two stage filter structure downsamples input sampling frequency of 96 MHz for WLANa to the data rate of 12 Msymbols/s. The overall decimation filter responses obtained for WCDMA and WLANa, satisfying the standard specifications, are shown in Figure 6.11 and 6.12 respectively.

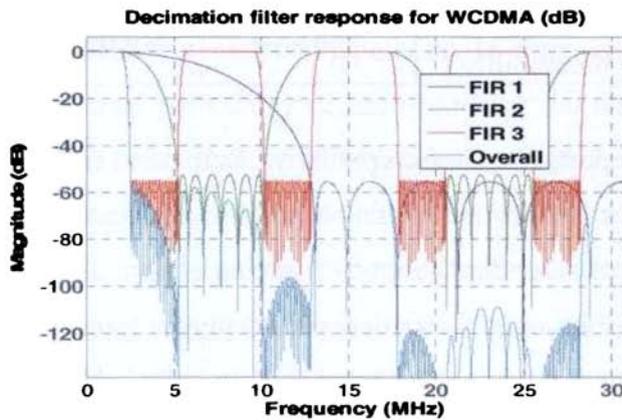


Figure 6.11 Filter responses for WCDMA mode in WCDMA/WLANa decimator

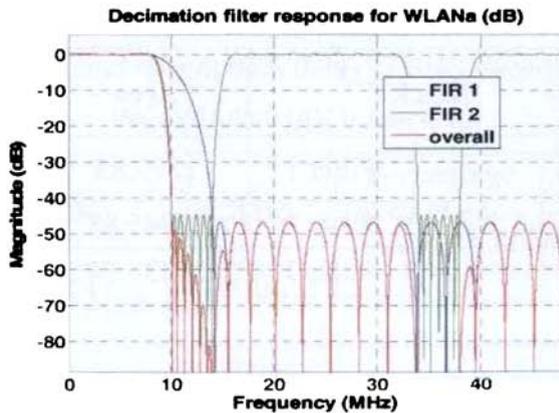


Figure 6.12 Filter responses for WLANa mode in WCDMA/WLANa decimator

The hardware synthesis is done with *Leonardo Spectrum* logic synthesis tool from Mentor Graphics. In order to operate first stage filter at 96 MHz, pipelining is done after every three modulo adders to meet the critical path delay. Two stage pipelining is done to meet the critical path delay for second filter which operates at maximum frequency of 24 MHz. The third stage is used only for WCDMA mode at a frequency of 7.68 MHz and no pipelining is required. The total area requirement and critical path delay of each block of the decimation filter is shown in Table 6.11.

Table 6.11 Area and critical path delay of RNS decimation filter for WCDMA/WLANa transceiver

Block	Area	Area of each block (%)	Critical path delay
Filter 1	5240.13	32.9	21.88
Filter 2	3981.09	24.96	87.33
Filter 3	5840.18	36.63	244.29
Multiplexers	29.47	0.002	0.69
Reverse converter	852.82	5.4	79.12
Total area	15943.69	100	

The area requirement of the decimation filter in single mode WCDMA receiver and the additional area required for making it adaptable for dual-mode operation are given in Table 6.12. It is observed that programmability is achieved at the expense of 33% of additional area compared to single mode WCDMA receiver. Table 6.13 reports the characteristics of decimation filter implemented in traditional number system performing signed multiplication and addition. RNS filter implementation offers about 61% saving in area. Pipelining is to be done in the multipliers as well as in the adder chain to meet the critical path delay for traditional case.

Table 6.12 Area requirement for programmability

Type of transceiver		Area	Percentage area (%)
Single mode WCDMA	Filter 1	2220.44	66.9
	Filter 2	1748.3	
	Filter 3	5840.18	
	Reverse converter	852.82	
	Total	10661.74	
Dual-mode Transceiver		15943.69	100
Additional area for programmability		5281.95	33.1

Table 6.13 Area requirement for WCDMA/WLANa decimation filter:

Traditional Vs RNS

Block		Area
Traditional	Filter 1	3940.89
	Filter 2	11074.45
	Filter 3	25762.62
	Total	40777.96
RNS		15943.69

6.6 Implementation of Programmable Decimation Filter using Index Calculus Multipliers

The programmable decimation filter architecture for WCDMA/WLANa standards is defined by VHDL code and the functional verification is performed using *ModelSim*. The hardware synthesis is done with *Synopsys design compiler*. The area requirement and critical path delay of each block of the RNS decimation filter is shown in Table 6.14. The critical path delay and area for each block of the filter is normalized with respect to a full adder critical path delay of 0.33 ns and area of $76\mu\text{m}^2$. The percentage

area requirement for each block of the RNS decimation filter is shown in Figure 6.13.

Table 6.14 Area, critical path delay and dynamic power dissipation for RNS decimation filter

Block	Area	Critical path delay
Filter 1	15983.5	58.95
Filter 2	12200.1	52.41
Filter 3	17875.2	58.95
Reverse converter	852.18	79.12
Total area	46910.9	
Dynamic power dissipation	479.1387 mW	

The area requirement of the decimation filter in single mode WCDMA receiver and the additional area required for making it adaptable for dual-mode operation are given in Table 6.15. It is observed that programmability is achieved at the expense of 34% of additional area compared to single mode WCDMA receiver.

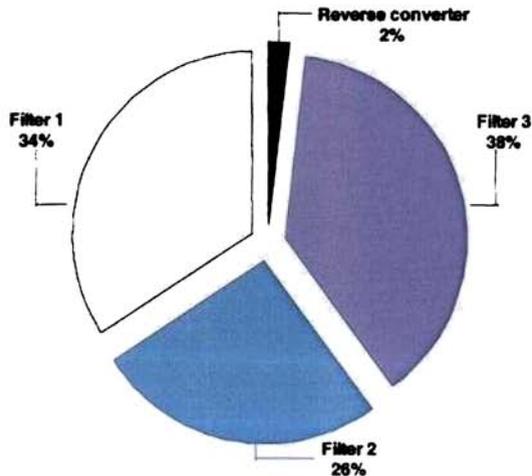


Figure 6.13 Area requirements for RNS decimation filter

Table 6.15 Area requirement for programmability

Type of transceiver		Area	Percentage area (%)
Single mode WCDMA	Filter 1	6780.8	65.7
	Filter 2	5368	
	Filter 3	17823.04	
	Reverse converter	852.18	
	Total	30824.02	
Dual-mode transceiver		46910.9	100
Additional area for programmability		16086.88	34.3

In order to operate the first stage of RNS filter at 96 MHz, two-stage pipelining is done to meet the critical path delay. The second and third stages are operating at downsampled frequencies of 24 MHz and 7.68 MHz respectively. They do not require pipelining due to the fast MAC operations in RNS domain. Table 6.16 reports the characteristics of decimation filter implemented in traditional binary number system performing signed multiplication and addition. RNS filter implementation requires only 87% of area with respect to the traditional implementation. The dynamic power dissipation for the RNS based dual-mode decimation filter is 28.4% less than that for traditional case. The inherent delay for each stage of traditional implementation is more, compared to the RNS implementation. The first stage of RNS decimation filter operates 2.6 times faster compared to the traditional implementation. Similarly, the second and third stages of RNS decimation filter operate 5 and 7.4 times faster than the traditional filter. The pipelining used for the RNS filter will not meet the critical path delay for traditional case. Hence, pipelining is required in the multipliers as well as in the adder chain of all the stages for traditional implementation to meet the critical path delay.

Table 6.16 Area, critical path delay and dynamic power dissipation for traditional decimation filter

Block	Area	Critical path delay
Filter 1	2966.59	153.68
Filter 2	14504.41	259.58
Filter 3	36165.2	437.91
Total area	53636.2	
Dynamic power dissipation	669.621 mW	

To evaluate the design techniques, the new architecture is implemented using RTL synthesizable VHDL code. Also the design is synthesized with *ArtisanTM* 0.18 μm and $V_{DD}=1.8\text{V}$ technology using Synopsys design compiler tools. The back end process, place and route, are done using *Cadence EncounterTM* tool set. The placed cell structure and routed design for the RNS decimation filter is shown in Figure 6.14 and 6.15 respectively.

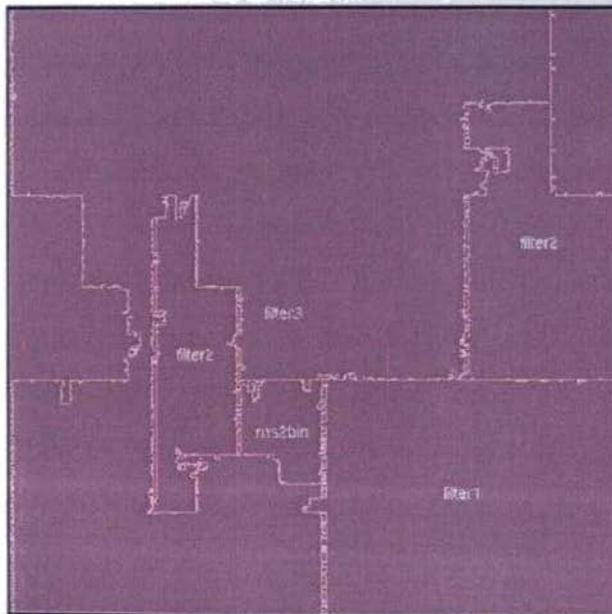


Figure 6.14 Placed cell structure for RNS based decimation filter



Figure 6.15 Routed view of RNS decimation filter

6.7 Simulation Results for Sigma-Delta based Parallel Analog-to-Residue Converter

The sigma-delta based parallel A/R converter is simulated for various resolutions. The $\Sigma\Delta$ modulator complexity for A/R converter with various resolutions is shown in Table 6.17. The RNS moduli set is chosen to provide sufficient dynamic range for the number system. The simulations are performed using the MATLAB[®] *Simulink models*.

The simulation result for A/R converter with 12-bits resolution shows a DR of 77.9 dB for the modulator with a 2-2 cascaded MASH topology. The A/R converter is designed to operate in the voice band with 20 KHz bandwidth. The decimation filter is designed to provide 40dB attenuation in the stopband. The output signal spectrum obtained after digital lowpass filtering in the decimator for an input signal at 5 KHz is shown in Figure 6.16. The output spectrum at Nyquist rate obtained after downsampling is shown in Figure 6.17.

Table 6.17 Sigma-delta modulator complexity for A/R converters of various resolutions

Resolution (No. of bits)	Sigma-delta modulator complexity			RNS moduli
	OSR (M)	Order (L)	Quantizer bits (B)	
12	16	4	2	[16 19 23]
14	16	4	3	[17 31 32]
16	16	5	4	[17 19 31 32]
20	32	5	4	[17 19 21 31 32]

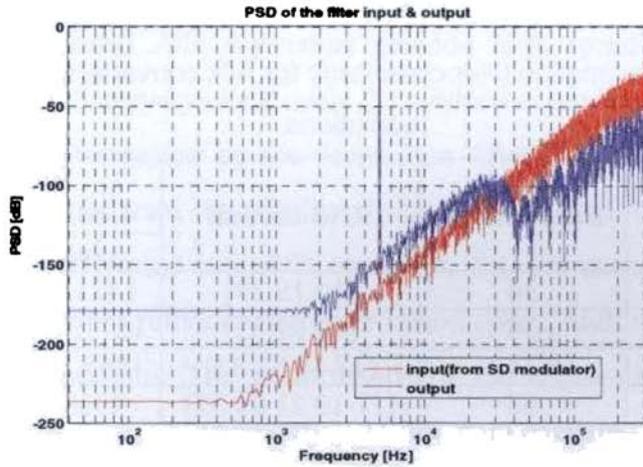


Figure 6.16 Power spectral density (PSD) plot for filter input and output

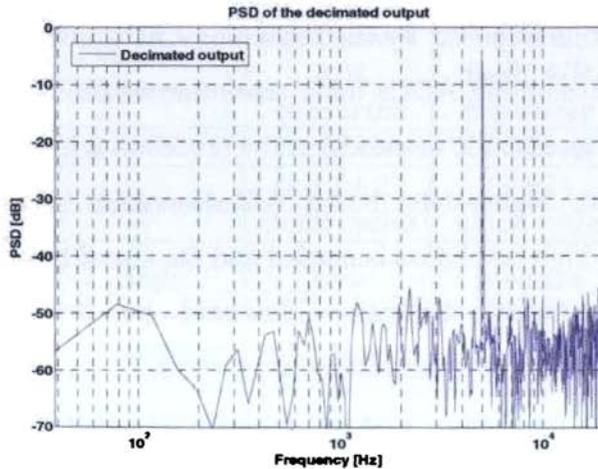


Figure 6.17 PSD plot for decimation filter output at Nyquist rate

The complexity of digital decimation filter for A/R converters with various resolutions in terms of filter order, implementation area and critical path delay are given in Table 6.18. Remez Parks-McClellan optimal equiripple FIR filter is chosen for the implementation. The hardware synthesis is done with the logic synthesis tool *Leonardo Spectrum* from Mentor Graphics Corporation, using ASIC library. The critical path delay and area for each filter is normalized with respect to a full adder (FA) critical path delay of 0.45 ns and area of 38 μm^2 .

Table 6.18 Decimation filter complexity for A/R converters with various resolutions

Resolution (bits)	Filter order	RNS moduli	Filter complexity	
			Area	Delay
12	34	[16 19 23]	1421.9	37.17
14	34	[17 31 32]	1271.02	37.42
16	34	[17 19 31 32]	1865.12	37.42
20	68	[17 19 21 31 32]	4939	43.28

The A/R converters based on Nyquist rate ADCs are suitable for data conversions in systems where the conversion process is constrained by bandwidth limitations imposed by the technology in which the converter is implemented. Oversampling ADCs trade resolution in time for resolution in amplitude in order to ease the demands on the precision with which the signal is to be quantized. Sigma-delta based A/R converter is well suited for applications where high resolution is needed and the signal bandwidth is much less than the bandwidth limitations imposed by the implementation technology. Nyquist rate A/R converters are practically implemented only up to 10-12 bits of resolution due to component matching and circuit nonidealities. The $\Sigma\Delta$ modulator does not require stringent component

matching, and hence sigma-delta based A/R converters with high resolutions of up to 20-bits are practically realizable. The analog part of sigma-delta based A/R converter is relatively simple, and a low cost implementation is possible unlike the Nyquist rate counterparts. The conversion speed of successive approximation based A/R converter is $k + l + 2$ clock cycles, where ' k ' and ' l ' are the register size of the first and second stages. For the iterative flash A/R converter, the conversion speed is $p + 1$ extended clock cycles, where ' p ' is the number of iterations in which the first flash stage performs conversion. The sigma-delta based A/R converters provide high speed conversion at oversampling rate. For every clock cycle the modulator produces output which is filtered and downsampled by the decimation filter to produce residues at Nyquist rate. The hardware complexity of sigma-delta based A/R converter is more than that of Nyquist rate A/R converter. So, the new A/R converter based on $\Sigma\Delta$ modulator provides high conversion speed, high resolution and a low cost implementation. The performance of the sigma-delta based A/R converter is compared with the Nyquist rate A/R converters in Table 6.19.

Table 6.19 Performance comparison of A/R converters

Feature	Flash A/R converter	SAR based A/R converter	Iterative flash A/R converter	Sigma-delta based A/R converter
Resolution	8-10 bits	10-12 bits	10-12 bits	Upto 20 bits
Conversion speed	1 clock cycle	$(k + l + 2)$ clock cycles	$(p + 1)$ clock cycle	1 clock cycle
Hardware complexity	High	Low	Medium	Medium
Cost of implementation	High	Medium	Medium	Low

6.8 Simulation Results and Performance Analysis of RRNS-Convolutional Concatenated Coding for OFDM System

The performance of the OFDM communication system using RCCC scheme is evaluated under different operating conditions. The simulation results show that the RCCC scheme offers significant improvement in BER performance for the OFDM system. The communication model is implemented in MATLAB[®]. The OFDM system is simulated with 800 subcarriers using differential QPSK modulation scheme and with the FFT and IFFT sizes of 2048 points. The simulations are carried out to evaluate the system performance under additive white Gaussian noise and multipath fading effects. The PAPR reduction by peak clipping under different clip compression ratios is found out for this coding scheme. The BER performance with cyclic prefixed guard band for different frame start synchronization errors is also analyzed.

6.8.1 Additive White Gaussian Noise Tolerance

The channel adds zero-mean Gaussian noise to the transmitted signal and the BER performances of uncoded, convolutional coded, and concatenated coded system are obtained. The simulations are carried out by varying the signal to noise ratio (SNR), and the BER values are plotted against the channel SNR for different cases as shown in Figure 6.18. The simulation results show that the RCCC scheme offers a coding gain of about 4dB at BER of 10^{-2} . This system can tolerate SNR of greater than 8 – 10 dB with QPSK modulation and RRNS-Convolutional concatenated coding scheme.

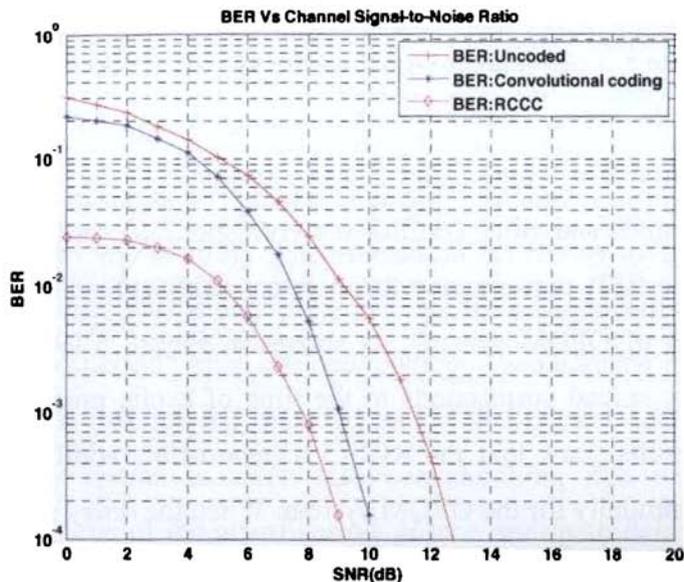


Figure 6.18 BER versus SNR for uncoded, convolutional coded and RCCC OFDM system

6.8.2 Multipath Delay Spread Immunity

One of the important properties of OFDM is its robustness to multipath delay spread. This is achieved by distributing the digitally encoded symbols over several orthogonal subcarriers in order to reduce the symbol rates. In a frequency-selective multipath fading channel, the base pulses of the original OFDM signal and the delayed version of the signal are no longer orthogonal. This leads to severe ISI as the orthogonality of the signals is lost. To address this problem, a guard interval is inserted between OFDM symbols.

In this research, the OFDM system uses cyclic prefix as guard band where the last 256 samples are copied and inserted in front of the symbol. Now a multipath reflection that stays within the guard interval will not cause interference problems. For a channel bandwidth of 1.25 MHz, 256 samples as the guard period correspond to a reflected signal with an additional path length of 61.4 km. The simulation is carried out for a multipath signal containing

single reflected signal which is 3 dB weaker than the direct signal. It is sufficient to take a 3 dB weaker signal as the signals weaker than this do not cause measurable errors. The multipath modeling is done by using a lowpass FIR filter function. The length of the filter corresponds to the delay in terms of number of samples and filter coefficients correspond to the reflected signal amplitudes. The BER performance for different delay spreads is obtained for the three types of OFDM systems as shown in Figure 6.19. The tolerable multipath delay spread corresponds to the time of cyclic prefix of the guard period. The results show that the RCCC scheme offers additional multipath delay spread immunity for the OFDM system. When the delay spread is longer than the guard period, the BER increases rapidly due to the increased ISI. But the RCCC scheme causes the BER to increase at a lesser rate compared to the other two schemes.

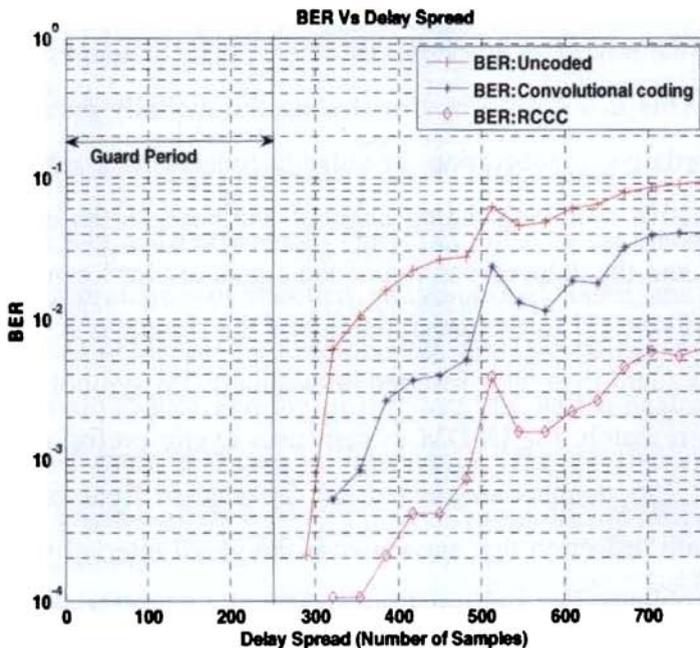


Figure 6.19 BER versus multipath delay spread for uncoded, convolutional coded and RCCC OFDM system

6.8.3 Effect of Frame Synchronization Errors

The cyclic prefix guard band insertion provides tolerance to frame start time error as well. The BER performance of the system for different timing errors specified in terms of number of samples is shown in Figure 6.20. The results show that the starting synchronization errors up to the guard band period are tolerable. This is due to the fact that the orthogonality is maintained during the guard period. Also, the new coding scheme keeps the BER of the system less than that for the uncoded and convolutional coded systems. If multipath delay spread is taken into consideration, this will reduce the effective stable time of the guard period. Hence multipath delay spread leads to reduced timing error tolerance. But the RCCC scheme offers better timing error tolerance in presence of multipath signals for a particular BER.

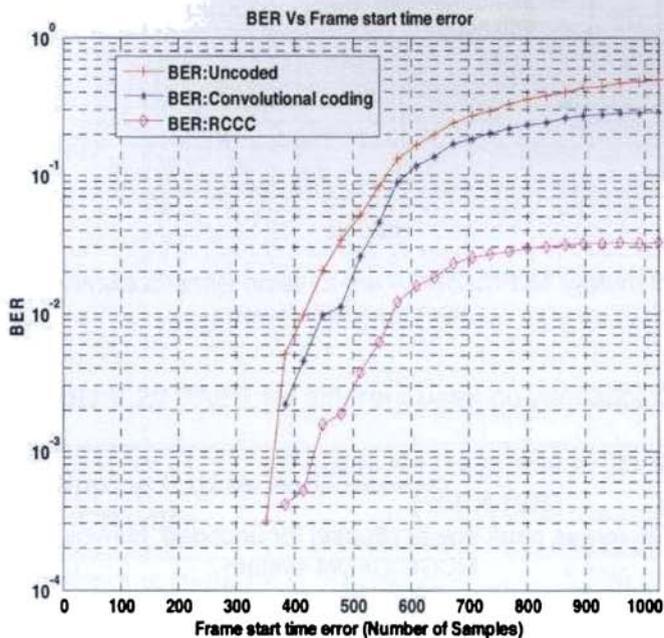


Figure 6.20 BER versus frame start time error for uncoded, convolutional coded and RCCC OFDM system

6.8.4 Peak Power Clipping for PAPR Reduction

The signal peak at the transmitter is clipped to reduce the PAPR value without much increase in the BER. As the clipping level is increased the PAPR reduces, but the BER is increased. The BER performance for different clip compression ratios in dB is shown in Figure 6.21, where the clip compression ratio (CR) is defined as the ratio of the peak power of the signal before clipping to the peak power of the clipped signal. The RCCC coding scheme allows the signal to clip heavily without significant increase in BER. The results show that the system can operate at a BER of 10^{-3} with a clip compression ratio of 15 dB.

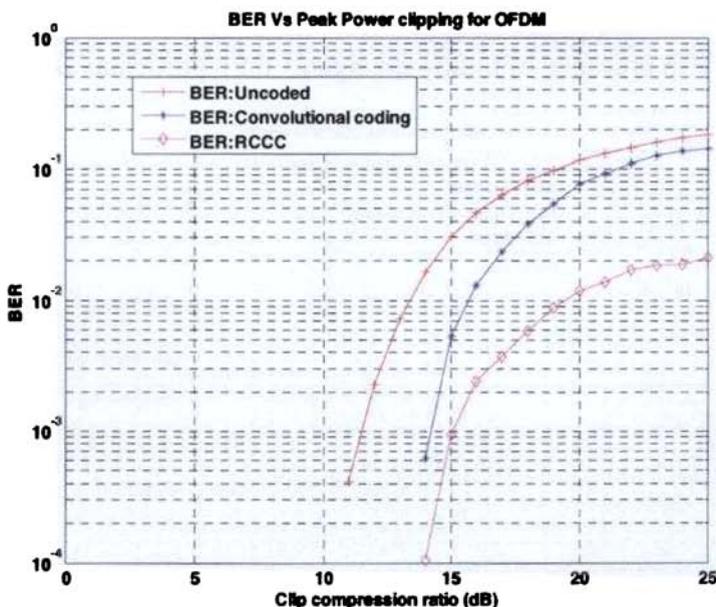


Figure 6.21 BER versus peak power clipping for uncoded, convolutional coded and RCCC OFDM system

The BER performance of the system for different clip compression ratios with varying amount of channel noise is shown in Figure 6.22. For high value of CR, more signal amplitude is clipped resulting in high BER. Hence as

CR is increased, the required SNR to achieve the same BER performance is increased. This is due to the increased probability of existence of OFDM signal amplitudes higher than the clipping level. The PAPR values for different clip compression ratios are shown in Table 6.20. The performance of the system for CR = 2 dB is very close to the no clipping performance. There is a trade off between BER performance and PAPR reduction.

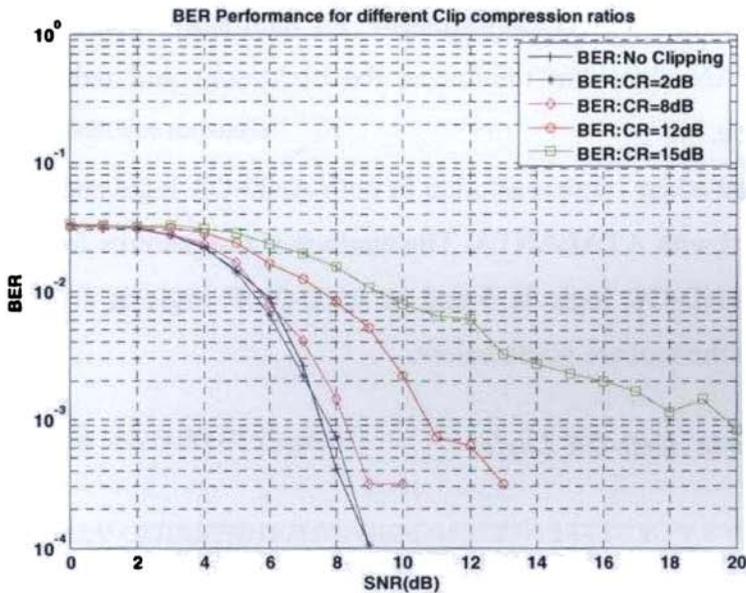


Figure 6.22 BER versus channel noise of the RCCC OFDM system for different peak power clipping levels

Table 6.20 PAPR for different peak power clipping

Clipping Ratio CR (dB)	Peak to Average Power Ratio, PAPR (dB)	
	Maximum	Average
No clipping (0 dB)	9.1417	5.5961
2 dB	9.126	5.5905
5 dB	8.7477	5.4783
8 dB	8.3789	4.8693
10 dB	8.2788	4.1188
12 dB	7.8339	3.3014

6.9 Simulation Results for Easily Testable MAC Units

The test pattern generation tool ATALANTA developed by the Virginia Polytechnic and State University is used to generate test vectors for various designs. ATALANTA is an automatic test pattern generator (ATPG) for stuck-at faults in combinational circuits. It employs the ‘fan-out oriented test generation (FAN) algorithm’ for test pattern generation, and ‘the parallel pattern single fault propagation technique’ for fault simulation [Lee and Ha, 1993]. The FAN algorithm minimizes the backtracks and reduces the test generation time.

The full adder implementation in both AND-OR logic and AND-XOR logic are tested with ATALANTA. The number of test patterns for 100% fault coverage in AND-OR logic is 6 and in AND-XOR logic is 4. The outputs obtained from the tool are given below.

A) Full Adder in AND-OR Logic

***** SUMMARY OF TEST PATTERN GENERATION RESULTS *****

1. Circuit structure

Name of the circuit	: fa_andor
Number of primary inputs	: 3
Number of primary outputs	: 2
Number of gates	: 7
Level of the circuit	: 3

2. ATPG parameters

Test pattern generation mode	: RPT + DTPG + TC
Limit of random patterns (packets)	: 16
Backtrack limit	: 10
Initial random number generator seed	: 1216695906
Test pattern compaction mode	: REVERSE + SHUFFLE
Limit of suffling compaction	: 2
Number of shuffles	: 4

3. Test pattern generation results

Number of test patterns before compaction	: 8
---	-----

Number of test patterns after compaction	: 6
Fault coverage	: 100.000 %
Number of collapsed faults	: 28
Number of identified redundant faults	: 0
Number of aborted faults	: 0
Total number of backtrackings	: 0
 4. Memory used	 : 10276 Kbytes
 5. CPU time	
Initialization	: 0.267 secs
Fault simulation	: 0.000 secs
FAN	: 0.000 secs
Total	: 0.267 secs

Test patterns and fault free responses:

```

1: 001 10
2: 101 01
3: 010 10
4: 110 01
5: 011 01
6: 100 10

```

B) Full Adder in AND-XOR Logic

***** SUMMARY OF TEST PATTERN GENERATION RESULTS *****

1. Circuit structure	
Name of the circuit	: fa_andxor
Number of primary inputs	: 3
Number of primary outputs	: 2
Number of gates	: 7
Level of the circuit	: 3
2. ATPG parameters	
Test pattern generation mode	: RPT + DTPG + TC
Limit of random patterns (packets)	: 16
Backtrack limit	: 10
Initial random number generator seed	: 1216697675
Test pattern compaction mode	: REVERSE + SHUFFLE
Limit of suffling compaction	: 2
Number of shuffles	: 4
3. Test pattern generation results	
Number of test patterns before compaction	: 7

Number of test patterns after compaction : 4
 Fault coverage : 100.000 %
 Number of collapsed faults : 32
 Number of identified redundant faults : 0
 Number of aborted faults : 0
 Total number of backtrackings : 0

4. Memory used : 10276 Kbytes

5. CPU time

Initialization : 0.000 secs
 Fault simulation : 0.000 secs
 FAN : 0.000 secs
 Total : 0.000 secs

Test patterns and fault free responses:

1: 110 01
 2: 011 01
 3: 101 01
 4: 001 10

Similarly, the ripple carry adders using full adders implemented in AND-OR logic and AND-XOR logic are tested with the tool ATALANTA. The number of test patterns for 100% fault coverage in both logics is given in Table 6.21.

Table 6.21 Number of test patterns for various adders in AND-OR and AND-XOR logic

Circuit Structure	Number of Test Patterns	
	AND-OR Logic	AND-XOR Logic
Full adder	6	4
2-bit Adder	8	5
3-bit Adder	9	6
4-bit Adder	11	7
5-bit Adder	12	8
6-bit Adder	13	9

6.10 Combinational Logic Synthesis Results using Exhaustive Branching Algorithm

The RM-ULM logic synthesis results obtained for various functions using the exhaustive branching algorithm is demonstrated in the following examples.

Example 1:

Implementation of 4-variable function $F = \oplus \Sigma (13, 14)$

The delivered network has 3 modules using only 2 levels in this approach, as shown in the Figure 6.23, while in the tree implementation [Xu et al., 1993], [Tan and Chia, 1996] the synthesized network will have 3 modules in 3 levels as shown in Figure 6.24.

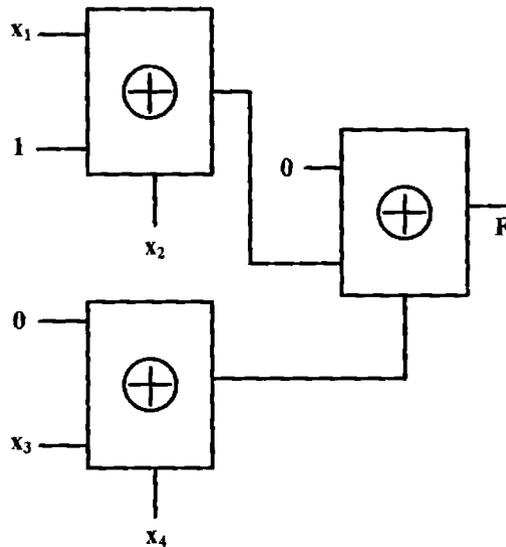


Figure 6.23 Exhaustive branched implementation for $F = \oplus \Sigma (13, 14)$

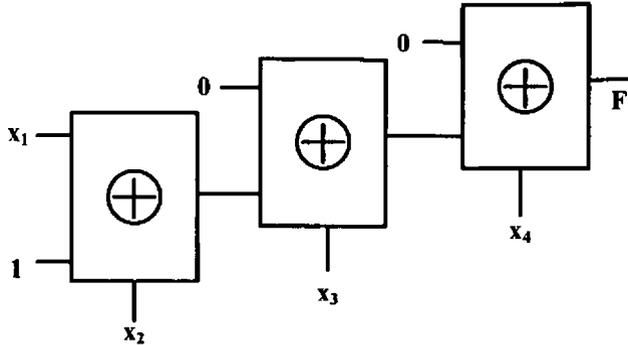


Figure 6.24 Tree implementation for $F = \oplus \Sigma(13, 14)$

Example 2:

Implementation of the 4-variable function, $F = \oplus \Sigma(5, 6, 9, 10)$

The implementation has 3 modules using only 2 levels in this approach, as shown in Figure 6.25, while the tree implementation will have 4 modules in 3 levels or a minimum of 3 modules in 3 levels as shown in Figure 6.26. In the above two examples a reduction in delay is found using the new approach.

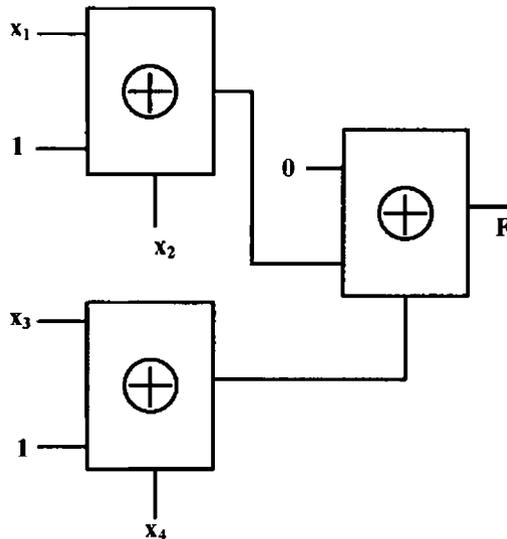


Figure 6.25 Exhaustive branched implementation for $F = \oplus \Sigma(5, 6, 9, 10)$

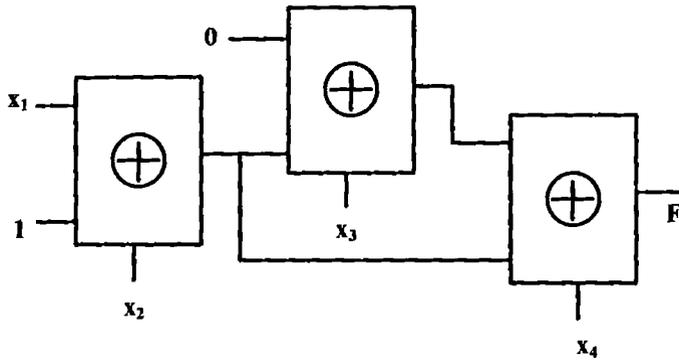


Figure 6.26 Tree implementation for $F = \oplus \Sigma(5, 6, 9, 10)$

Example 3:

Implementation of the 3-variable function, $F = \oplus \Sigma(0, 1, 2, 4, 6)$

The delivered network has only 1 module using 1 level in the new approach as in Figure 6.27, whereas the tree implementation requires 2 modules in 2 levels. One possible implementation is shown in Figure 6.28. This example clearly indicates the reduction in delay and number of modules.

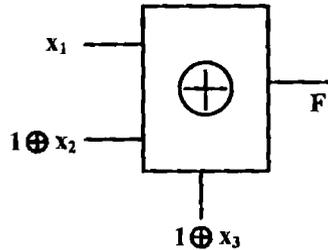


Figure 6.27 Exhaustive branched implementation for $F = \oplus \Sigma(0, 1, 2, 4, 6)$

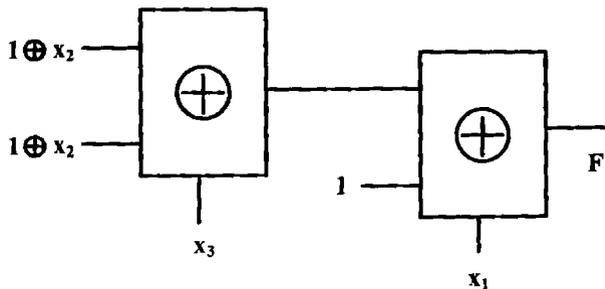


Figure 6.28 Tree implementation for $F = \oplus \Sigma(0, 1, 2, 4, 6)$

Example 4:

Implementation of the 3-variable function, $F = \oplus \Sigma(0, 2, 3, 4, 5)$

The delivered network using the new approach is same as that of the tree implementation with 2 modules in 2 levels as shown in Figure 6.29.

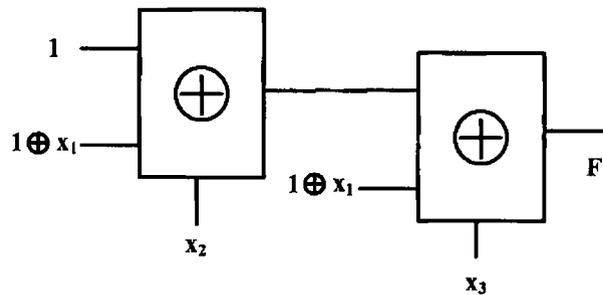


Figure 6.29 Exhaustive branched and tree implementation for $F = \oplus \Sigma(0, 2, 3, 4, 5)$

Simulation is done for 2, 3 and 4-variable functions up to 2 levels. Table 6.22 shows the reduction in delay and/or hardware for certain functions. The number of levels and modules are indicated by L and M respectively. The reduction in number of modules required will lead to reduced power consumption.

Table 6.22 Comparison in terms of delay and hardware for standard, tree and exhaustive branched implementations

Functions	Standard Implementation L/M	Tree Implementation L/M	Exhaustive Branched Implementation L/M
$F = \oplus \Sigma(13, 14)$	4 / 15	3 / 3	2 / 3
$F = \oplus \Sigma(5, 6, 9, 10)$	4 / 15	3 / 3	2 / 3
$F = \oplus \Sigma(0, 1, 2, 4, 6)$	3 / 7	2 / 2	1 / 1
$F = \oplus \Sigma(0, 2, 3, 4, 5)$	3 / 7	2 / 2	2 / 2

6.11 GA based Combinational Logic Synthesis Results using ULMs

Implementations obtained by the GA based approach for various functions are presented here. Sometimes it is possible to get solutions that use same number of modules and levels with different ULMs. The order of preference of ULMs can be given as input to the program that helps to select a particular solution for an application. Both 'true' and 'complementary' inputs are assumed to be available for all implementations.

Example 1:

Implementation of 4-variable function $F = \sum m(6, 7, 8, 12, 13, 14, 15)$

The GA selects an appropriate ULM and finds an optimum solution. The delivered network consists of 3 multiplexers using only 2 levels in this approach, as shown in Figure 6.30. There is no solution found with less than 3 modules and 2 levels with any other ULMs. This example shows that the algorithm finds a solution which takes minimum area and delay.

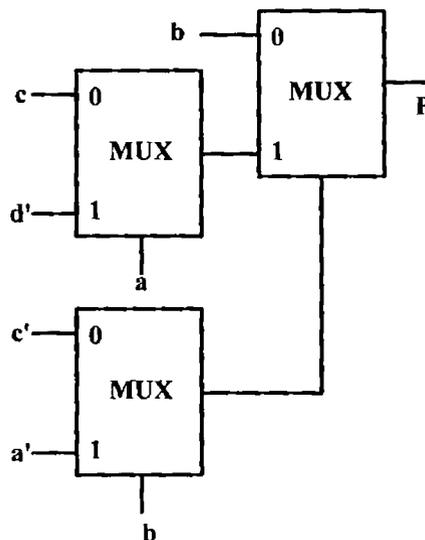


Figure 6.30 GA implementation for $F = \sum m(6, 7, 8, 12, 13, 14, 15)$

Example 2:

Implementation of a 3-variable function, $F = \sum m(1, 2, 4)$, with preference given to RM-ULM.

The delivered network consists of 3 RM-ULMs in 2 levels. A solution is possible using 3 multiplexers in 2 levels also. Since the preference is set to RM-ULM, the delivered network is as given in Figure 6.31. Whereas the NAND and NOR implementations require more number of modules and levels than this.

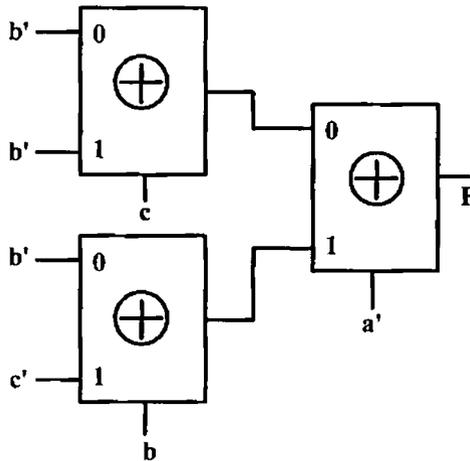


Figure 6.31 GA implementation for $F = \sum m(1, 2, 4)$

Example 3:

Implementation of a 4-variable function, $F = \sum m(1, 2, 3, 4, 5, 6, 7)$

Here the implementations obtained by GA for all types of ULMs are given for comparison. The optimum network is delivered using NOR gate as the design unit and is shown in Figure 6.32. The implementation obtained with NAND as the design unit uses 4 modules in 2 levels and is given in Figure 6.33. The delivered networks with MUXs and RM-ULMs are shown in Figure 6.34 and 6.35 respectively, and both use 3 modules in 2 levels. The graph shown in

Figure 6.36 gives a comparison of number of modules and levels required for implementing the function, $F = \sum m(1, 2, 3, 4, 5, 6, 7)$ with various ULMs.

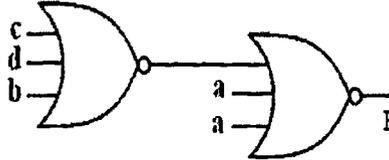


Figure 6.32 GA implementation for $F = \sum m(1, 2, 3, 4, 5, 6, 7)$ using NOR gates

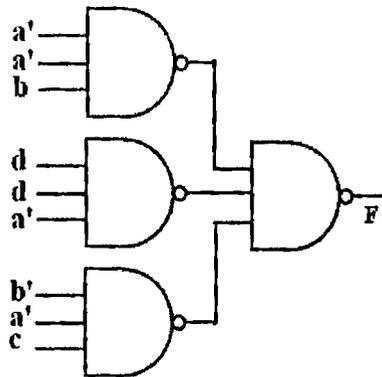


Figure 6.33 GA implementation for $F = \sum m(1, 2, 3, 4, 5, 6, 7)$ using NAND gates

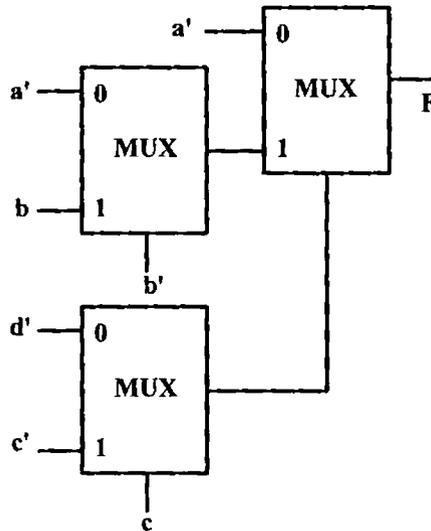


Figure 6.34 GA implementation for $F = \sum m(1, 2, 3, 4, 5, 6, 7)$ using multiplexers

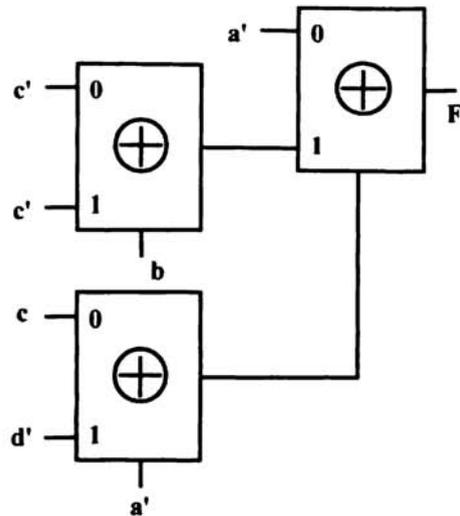


Figure 6.35 GA implementation for $F = \sum m(1, 2, 3, 4, 5, 6, 7)$ using RM-ULMs

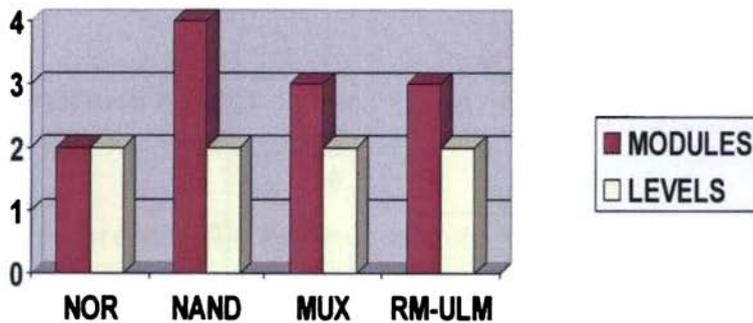


Figure 6.36 Comparison in terms of number of modules and levels required for implementing $F = \sum m(1, 2, 3, 4, 5, 6, 7)$ with various ULMs

Example 4:

The GA implementation obtained for a three variable function, $F = \sum m(7)$ using various ULMs and the comparison in terms of number of modules and levels are shown in Figure 6.37 and 6.38 respectively.

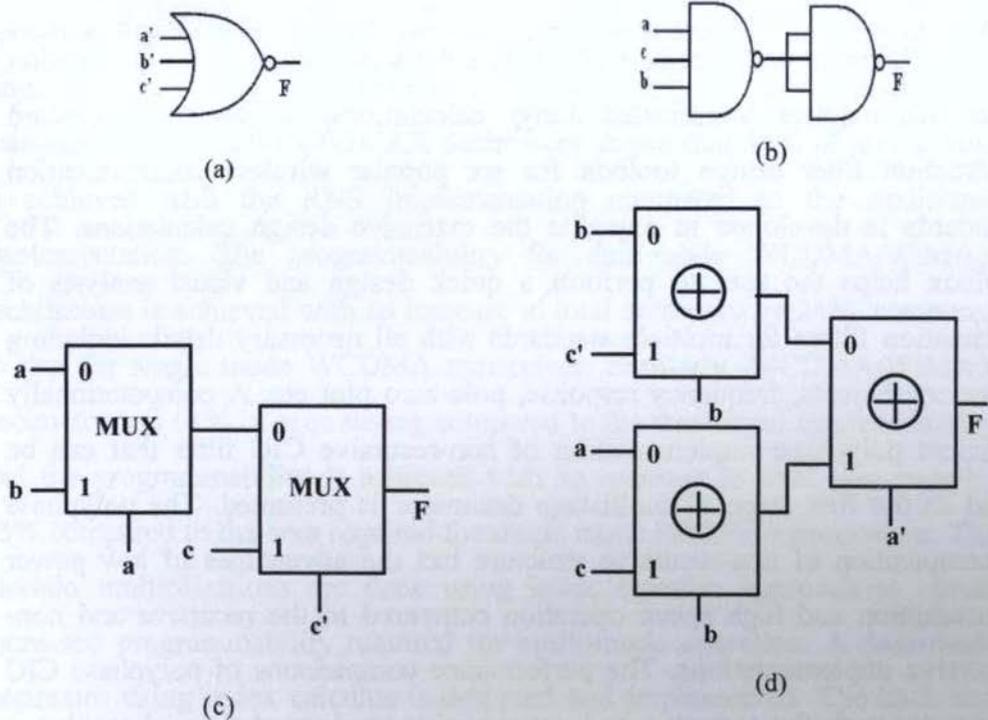


Figure 6.37 GA implementation for $F = \sum m(7)$ using (a) NOR (b) NAND (c) MUX (d) RM-ULM

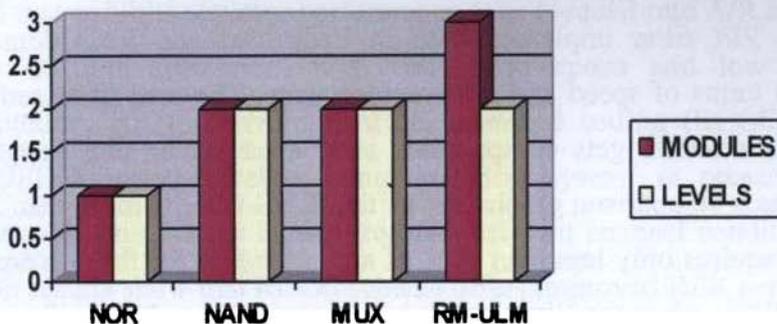


Figure 6.38 Comparison in terms of number of modules and levels required for implementing $F = \sum m(7)$ using NOR, NAND, MUX, and RM-ULM

6.12 Summary

The most complex part of a sigma-delta modulator is the decimation filter that requires complicated design calculations. A new multi-standard decimation filter design toolbox for six popular wireless communication standards is developed to expedite the extensive design calculations. The toolbox helps the user to perform a quick design and visual analysis of decimation filters for multiple standards with all necessary details including filter coefficients, frequency response, pole-zero plot etc. A computationally efficient polyphase implementation of non-recursive CIC filter that can be used as the first stage of multistage decimator is presented. The polyphase decomposition of non-recursive structure has the advantages of low power consumption and high speed operation compared to the recursive and non-recursive implementations. The performance comparisons of polyphase CIC decimator with the recursive and nonrecursive implementations show that it has higher speed of operation, lower power consumption and more area requirement. So the designer can trade and select the CIC architecture based on the overall system requirements.

The FIR filter implementation in traditional and RNS domain are analyzed in terms of speed and area requirements. The area overhead due to conversion circuitries gets compensated after a particular filter length. The speed and area comparison graphs shows that RNS filter is more than 3 times faster and requires only less than 60% of area than that for the corresponding traditional filter, when the filter length is increased above 32 taps for an input word length of 6-bits. The speed-up and area reduction obtained in RNS domain is utilized to implement multi-mode decimation filters. Dual-mode decimation filters programmable for WCDMA/WiMAX and

WCDMA/WLANa standards are designed and implemented. High speed operation is achieved in RNS implementation with lesser pipelining in each stage compared to the traditional implementation. The performance comparison of WCDMA/WiMAX decimators shows that 64% of area saving is achieved with the RNS implementation compared to the traditional implementation. The programmability for dual-mode WCDMA/WiMAX architecture is achieved with an increase in total area only by 24%, compared to that for single mode WCDMA transceiver. Similarly, WCDMA/WiMAX decimator has 61% of area saving compared to the traditional implementation and the programmability is achieved with an increase in total area only by 33% compared to the area required for single mode WCDMA transceiver. The modulo multiplications are done using index calculus approach to obtain increased programmability required for multi-mode operation. A dual-mode decimator using index calculus is designed and implemented. The back end process is done and the placed cell structures as well as the routed view are taken.

A novel sigma-delta based parallel analog-to-residue converter is presented that exhibits superior performance over Nyquist rate A/R converters in terms of high resolution, high conversion speed and low cost for implementation. RRNS-convolutional concatenated coding (RCCC) scheme for an OFDM based wireless communication system is presented. The performance of this system is analyzed for different channel conditions. The simulation results show that RCCC scheme offers improved BER performance and the system can tolerate AWGN with $\text{SNR} > 8 - 10$ dB. The guard band insertion with cyclic prefixing of last 256 samples provides tolerance to multipath delay spread and frame start synchronization errors. The RCCC scheme makes the OFDM system more robust against multipath effects and

timing errors. Also, the signal can be heavily clipped to reduce the PAPR without significant increase in BER for the RCCC OFDM system. The simulation result shows that the system can operate at a BER of 10^{-3} with a clip compression ratio of 15 dB. The performance analysis shows that the RCCC is suitable for OFDM as it improves the tolerance of system to channel noise, multipath effects, timing errors and peak power clipping.

Easily testable MAC units for the filters are presented using RM form for realization. The number of test vectors required for 100% fault coverage of single stuck-at faults is found using the test tool ATALANTA. The test results show a reduction in number of test vectors for adders of various sizes implemented in RM form. Also, combinational logic synthesis results obtained using exhaustive branching technique and GA based approach are presented. The synthesis results show that the new algorithms achieve better implementation in terms of reduction in number of modules and delay.

Chapter 7

Conclusions and Suggestions for Further Work

The conclusions drawn from the design and implementation of various architectures for a high performance wireless communication system are presented in this chapter. Suggestions for further work in this field are also presented.

7.1 Conclusions

Research on efficient design methods and architectures for a high performance wireless transceiver is presented as part of this thesis. Multi-standard wireless transceivers that facilitate towards *global roaming* are considered in this research. Sigma-delta analog to digital converters (SD-ADCs) are used in multi-standard transceivers to adapt to the requirements of different standards. The oversampling data converters relax the requirements of analog circuitry at the expense of more complicated digital circuitry. It takes the advantage of today's VLSI technology tailored for high-speed/high-density digital circuits rather than accurate analog circuits. This is done by performing majority of the conversion processes in digital domain. The most complex part of a high resolution sigma-delta converter is the decimation filter. The design optimization techniques for the digital decimation filter are considered as part of this research.

The new multi-standard decimation filter design toolbox developed for six popular communication standards helps the user to expedite complicated design calculations and enables a visual analysis. The multistage implementation consisting of a CIC filter as the first stage followed by a halfband and/or droop compensating FIR filter achieved reduction in hardware complexity and computational effort. A computationally efficient polyphase implementation of non-recursive comb decimation filter is presented that has high speed and low power consumption compared to the recursive and non-recursive implementations. So, the polyphase implementation is suitable for high data rate wireless transceivers.

The performance evaluation of FIR filters operating in RNS domain shows that the area overhead due to forward and reverse conversion is

compensated after a particular filter length. The speed-up and area reduction of RNS implementation is utilized in the programmable decimation filters for multi-standard transceivers. Dual-mode decimators reconfigurable for WCDMA/WiMAX and WCDMA/WLANa standards are designed and implemented. The reconfigurable decimation filters operating in RNS domain offer high speed operation with lesser area requirement and lower dynamic power consumption compared to the traditional implementation.

A novel sigma-delta based parallel analog-to-residue converter that reduces the complexity involved in RNS conversion circuitry is presented. It exhibits superior performance over the existing Nyquist rate A/R converters in terms of high resolution, high conversion speed, medium hardware complexity and low cost for implementation. The BER performance of a communication system operating in RNS domain is analysed by modeling an OFDM system. The RRNS-Convolutional concatenated coding (RCCC) scheme provides improved BER performance under different operating conditions by exploiting the error detection and correction properties of RRNS. The simulation results show that RCCC scheme offers improved BER performance in presence of additive white Gaussian noise and multipath delay spread. The guard band insertion with cyclic prefixing provides tolerance to multipath delay spread and frame start synchronization errors. This coding scheme makes the OFDM system more robust against multipath effects and timing errors. Also, the signal can be heavily clipped to reduce the PAPR without significant increase in BER for the RCCC OFDM system. Hence, RCCC is an efficient scheme for forward error correction as it improves the tolerance of system to channel noise, multipath effects, timing errors and peak power clipping.

The testability property of RM form together with XOR intensive nature of arithmetic circuits is utilized for the building up of easily testable

MAC units. The MAC units described in this research use adders and ROM cells as the basic building blocks. Hence, the overall testability is improved by using adders implemented in RM form as the basic elements. The simulation result using the test tool ATALANTA validates that the size of test set is small for RM implementations than that for AND-OR structures. An algorithm for combinational logic synthesis using RM-ULMs that does exhaustive branching is presented. The simulation results prove that the algorithm attains reduction in number of modules and levels for implementing logic functions. Also, a GA based logic synthesis using appropriate ULMs is presented. The search algorithm finds a solution with a particular ULM that requires minimum number of levels and modules.

The simulation results and performance analysis show that the comprehensive design approaches and reconfigurable architectures presented in this research are in good agreement with the requirements of new generation portable communication systems. Hence, these design techniques and circuits are dependable alternatives that could be used for high performance wireless applications.

7.2 Suggestions for Further Work

The problems for further investigations in continuation with the present work are listed below.

- Further optimization is possible for decimation filter implementation by employing coefficient optimization and common subexpression elimination techniques. Several modulo multiplier designs are available in literature. Performance of RNS filters with other types of multipliers could be evaluated to get the optimum one.

- VLSI signal processing systems are prone to transient errors due to the scaling down of feature size and power supply voltages, to achieve high density and low power dissipation. Fault tolerant decimation filter implementation could be accomplished by including few redundant moduli to the present work.
- Triple-mode decimation filter implementation is a promising extension of the dual-mode decimator developed in this research. The modulo multipliers based on index calculus become more suitable as the number of modes increases. Triple-mode decimator programmable for GSM/WCDMA/WLAN standards could be considered for implementation.
- A real implementation and performance evaluation could be done for the sigma-delta based analog-to-residue converter using Spice simulation.
- Further analysis of OFDM transmitter/receiver chain with various combinations of other coding techniques could be performed.
- Cognitive radio is a wireless communication device that is designed to intelligently detect whether a particular segment of radio spectrum is currently in use. Also, it uses the unused spectrum dynamically without interfering the transmissions of authorized users. The current wireless communication technology and the increasing demand for spectrum gives cognitive radio an essential role in the future spectrum efficient communications. High speed and high resolution ADC is a major design challenge for a cognitive radio. Hence sigma-delta ADCs could be efficiently applied for cognitive radio platforms.

REFERENCES

- [1] D. Adamidis and H.T. Vergos, "RNS multiplications / sum-of-squares units", *IET Computers and Digital Techniques*, vol. 1, No. 1, pp. 38-48, January 2007.
- [2] A.H. Aguirre, C.A.C. Coello and B.P. Buckles, "A genetic programming approach to logic function synthesis by means of multiplexers", *Proceedings of First NASA/DOD workshop on Evolvable Hardware*, IEEE Computer Society Press, Los Alanitos, California, pp. 46-53, July 1999.
- [3] A.H. Aguirre and C.A.C. Coello "Using genetic programming and multiplexers for the synthesis of logic circuits", *Engineering Optimization*, Vol. 36, No. 4, pp. 491-511, August 2004.
- [4] G. Alia and E. Martinelli, "A VLSI modulo m multiplier", *IEEE Transactions on Computers*, Vol. 40, No. 7, pp. 873-878, July 1991.
- [5] P.E. Allen and D.R. Holberg, *CMOS Analog Circuit Design*, Oxford University Press, New York, Second Edition, 2002.
- [6] A.E.A. Almaini, J.F. Miller and L Xu, "Automated synthesis of digital multiplexer networks", *IEE Proceedings Computers and Digital Techniques*, Vol. 139, No. 4, pp. 329-334, July 1992.
- [7] S. D' Amico, M. De Matteis and A. Baschiroto, "A 6.4mW, 4.9nV/ $\sqrt{\text{Hz}}$, 24dBm IIP3 VGA for a multi-standard (WLAN, UMTS, GSM and Bluetooth) receiver", *32nd European Solid-State Circuits Conference*, pp. 82-85, September 2006.
- [8] P.V. Ananda Mohan and A.B. Premkumar, "RNS-to-binary converters for two four-moduli sets $\{2^n - 1, 2^n, 2^n + 1, 2^{n+1} - 1\}$ and $\{2^n - 1, 2^n, 2^n + 1, 2^{n+1} + 1\}$ ", *IEEE Transactions on Circuits and Systems - I*, Vol. 54, No. 6, pp. 1245-1254, June 2007.

- [9] C.J. Barrett, "Low-power decimation filter design for multi-standard transceiver applications", *Master of Science Thesis in Electrical Engineering*, University of California, Berkeley, 1997.
- [10] F. Barsi and P. Maestrini, "Error correcting properties of redundant residue number systems", *IEEE Transactions on Computers*, Vol. C-22, No. 3, pp. 307-315, March 1973.
- [11] G.L. Bernocchi, G.C. Cardarilli, A.D. Re, A. Nannarelli and M. Re, "Low-power adaptive filter based on RNS components", *IEEE International Symposium on Circuits and Systems*, pp. 3211-3214, May 2007.
- [12] J.L. Beuchat, "Some modular adders and multipliers for filed programmable gate arrays", *Proceedings of International Parallel and Distributed Processing Symposium*, France, April 2003.
- [13] V. Bobin and D. Radhakrishnan, "A VLSI residue arithmetic multiplier with fault detection capability", *Proceedings of IEEE International Conference on Computer Design: VLSI in Computers and Processors*, pp. 348-351, October 1989.
- [14] J.C. Candy, "Decimation for Sigma Delta Modulation", *IEEE Transactions on Communications*, Vol. COM-34, No. 1, pp. 72-76, January 1986.
- [15] G.C. Cardarilli, R. Lojacono, G. Martinelli and M. Salerno, "Structurally passive digital filters in residue number system", *IEEE Transactions on Circuits and Systems*, Vol. 35, No. 2, pp.149-158, February 1988.
- [16] G.C. Cardarilli, A. Del Re, A. Nannarelli, and M. Re, "Low-power implementation of polyphase filters in quadratic residue number system", *Proceedings of IEEE International Symposium on Circuits and Systems*, Vol. 2, pp. II - 725-8, May 2004.
- [17] G.C. Cardarilli, A. Del Re, A. Nannarelli and M. Re, "Low power and low leakage implementation of RNS FIR filters", *Proceedings of 39th Asilomar Conference on Signals, Systems, and Computers*, pp. 1620-1624, October 2005.

- [18] Di Claudio E., Piazza F., and Orlandi G., "Fast combinatorial RNS processors for DSP applications", *IEEE Transactions on Computers*, Vol. 44, pp. 624-633, 1995.
- [19] C.A. Coello, A.D. Christiansen and A.H. Aguirre, "Using genetic algorithms to design combinational logic circuits", *Intelligent Engineering through Artificial Neural Networks*. Vol. 6, pp. 391-396, 1996.
- [20] V.P. Correia and A.I. Reis, "Classifying n - input Boolean functions", *VII Workshop IBERCHIP 2001*, Montevideo, IWS 2001.
- [21] R.J. Cosentino, "Fault tolerance in a systolic residue arithmetic processor array", *IEEE Transactions on Computers*, Vol. 37, No. 7, pp. 886-890, July 1988.
- [22] D. Divsalar and F. Pollara, "Serial and hybrid concatenated codes with applications", *Proceedings of International Symposium on Turbo Codes*, Brest, France, pp. 80-88, 1997.
- [23] E.V. Dubrova and J.C. Muzio, "Testability of generalized multiple-valued Reed-Muller circuits", *Proceedings of the 26th International Symposium on Multiple-Valued Logic*, Spain, pp.56-61, May 1996.
- [24] M.H. Etzel and W.K. Jenkins, "Redundant residue number systems for error detection and correction in digital filters", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. ASSP-28, No. 5, pp. 538-545, October 1980.
- [25] T.C. Fogarty, J.F. Miller and P. Thomson, "Evolving digital logic circuits on Xilinx 6000 family FPGAs", *Soft Computing in Engineering Design and Manufacturing*, P.K. Chawdhary, R. Roy and R.K. Pant (eds.), Springer-Verlag, London, pp. 299-305, 1998.
- [26] S. Foo, P. Moss, T. Norton and D. Stafford, "Fifth order sigma delta modulator with decimation", *Proceedings of the Thirty-Sixth Southeastern Symposium on System Theory*, Atlanta, pp. 522-526, March 2004.
- [27] Y. Gao, L. Jia, J. Isoaho and H. Tenhunen, "A comparison design of comb decimators for sigma-delta analog-to-digital converters", *Analog Integrated*

- Circuits and Signal Processing*, 22, Kluwer academic publishers, pp. 51-60, 1999.
- [28] Y. Gao, L. Jia and H. Tenhunen, "A fifth-order comb decimation filter for multi-standard transceiver applications", *ISCAS 2000 - IEEE International Symposium on Circuits and Systems*, Switzerland, Vol.3, pp. 89-92, May 2000.
- [29] A. Ghazel, L. Naviner and K. Grati, "Design of down-sampling processors for radio communications", *Analog Integrated Circuits and Signal Processing*, 36, Kluwer academic publishers, pp. 31-38, 2003.
- [30] D.E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley Professional, Canada, January 1989.
- [31] P. Gray and R. Meyer, "Future directions in silicon ICs for RF personal communications," *Proceedings, 1995 Custom Integrated Circuits Conference*, pp. 83-90, May 1995.
- [32] D.H. Guo and C.Y. Hsu, "The economical PAPR minization scheme for combinative coding technique applied OFDM communication system", *Analog Integrated Circuits and Signal Processing*, vol.46, pp.139-144, February 2006.
- [33] O. Gursoy, O. Saglamdemir, M. Aktan, S. Talay and G. Dunder, "Low-power decimation filter architectures for sigma-delta ADCs", *4th International Conference on Electrical and Electronics Engineering*, Bursa, Turkey, December 2005.
- [34] G.H. Hardy and E.M. Wright, *An Introduction to the Theory of Numbers*, UK, Oxford Press, 1979.
- [35] B. Harking, "Efficient algorithm for canonical Reed-Muller expansions of Boolean functions", *IEE Proceedings Computers and Digital Techniques*, Vol. 137, No. 5, September 1990.
- [36] J. Heiskala and J. Terry, *OFDM Wireless LANs: A Theoretical and Practical Guide*, Sams Publishers, ISBN: 0672321572, 2001.

- [37] A.A. Hiasat, "New efficient structure for a modular multiplier for RNS", *IEEE Trans. on Computers*, Vol. 49, No. 2, pp. 170-174, February 2000.
- [38] E.B. Hogenauer, "An economical class of digital filters for decimation and interpolation", *IEEE Transactions on Acoustic, Speech and Signal Processing*, Vol. ASSP-29, No. 2, pp. 155-162, April 1981.
- [39] W.K. Jenkins and B.J. Leon, "The use of residue number systems in the design of finite impulse response digital filters", *IEEE Transactions on Circuits and Systems*, Vol. 24, No. 4, pp. 191-201, April 1977.
- [40] U. Kalay, D.V. Hall and M.A. Perkowski, "A minimal universal test set for self-test of EXOR-sum-of-products circuits", *IEEE Transactions on Computers*, Vol. 49, No. 3, pp. 709-716, March 2000.
- [41] M. Kim and S. Lee, "Design of dual-mode digital down converter for WCDMA and cdma2000", *ETRI Journal*, Vol.26, No.6, Dec. 2004, pp.555-559.
- [42] N. Koblitz, *A course in number theory and cryptography*, 2nd Edition, Springer – Verlag, New York, 1994.
- [43] J.R. Koza, *Genetic Programming: On the Programming of Computers by means of Natural Selection*, MIT Press, 1992.
- [44] M. Laddomada, "Comb-based decimation filters for sigma-delta A/D converters: Novel schemes and comparisons", *IEEE Transactions on signal processing*, Vol. 55, No. 5, pp. 1769-1779, May 2007.
- [45] E. Lawrey, "The suitability of OFDM as a modulation technique for wireless telecommunications, with a CDMA comparison", *BE Thesis, James Cook University*, October 1997.
- [46] H.K. Lee and D.S. Ha, "On the generation of test patterns for combinational circuits," Technical Report No. 12_93, Dept of Electrical Eng., Virginia Polytechnic Institute and State University.
- [47] S.F. Li and J. Wetherrell, "A compact low-power decimation filter for sigma-delta modulators", *Proceedings of IEEE International Conference on Acoustics*,

Speech, and Signal Processing, Turkey (ICASSP '00), Vol. 6, pp. 3223-3226, June 2000.

- [48] W. Li, J. Liu, J. Wang, C. Zhang and W. Guo, "An efficient digital IF down-converter for dual-mode WCDMA/EDGE receiver based on software radio", *IEEE 6th CAS Symposium on Emerging Technologies: Mobile and Wireless Communications*, China, pp. 713-716, May 31-June 2, 2004.
- [49] A.S. Madhukumar and F. Chin, "Performance of a residue number system based DS-CDMA system over bursty communication channels", *Proceedings of IEEE Vehicular Technology Conference (VTS-Fall VTC 2000)*, Vol.5, pp.2433-2440, 2000.
- [50] A.S. Madhukumar and F. Chin, "Residue number system-based multicarrier CDMA for high-speed broadband wireless access", *IEEE Transactions on Broadcasting*, Vol. 48, No. 1, pp. 46-52, March 2002.
- [51] M.N. Mahesh and M. Mehendale, "Low power realization of residue number system based FIR filters", *Proceedings of the 13th International Conference on VLSI Design*, 2000.
- [52] D. Mandelbaum, "Error correction in residue arithmetic", *IEEE Transactions on Computers*, Vol. C-21, No. 6, pp. 538-545, June 1972.
- [53] S. Mandyam and T. Stouraitis, "Efficient analog-to-residue conversion schemes", *Proceedings of IEEE International Symposium on Circuits and Systems*, New Orleans, LA, pp. 2885-2888, May 1990.
- [54] U. Meyer-Baese, *Digital signal processing with field programmable gate arrays*, Springer-verlag Berlin Heidelberg, New York, 2001.
- [55] J.F. Miller and P. Thomson, "Combinational and sequential logic optimization using Genetic Algorithms", *Proceedings of the First IEE/IEEE International Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications*, London, England, IEE Conference Publication No. 414, pp. 34-38, September 1995.

- [56] S.R. Norsworthy, R. Schreier and G.C. Temes, *Delta-Sigma Data Converters, Theory, Design, and Simulation*, Piscataway, NJ: IEEE Press, 1997.
- [57] V. Paliouras and T. Stouraitis, "Area-time performance of VLSI FIR filter architectures based on residue arithmetic", *Proceedings of the 23rd EUROMICRO Conference 'New Frontiers of Information Technology'*, Hungary, pp. 576-583, September 1997.
- [58] M.S. Palma, T.K. Sarkar and D. Sengupta, "A chronology of developments of wireless communication and electronics from 1921 to 1940", *Proceedings of IEEE Antennas and Propagation Society International Symposium*, Boston, MA, USA, Vol. 1, pp. 6-9, July 2001.
- [59] B. Parhami and C.Y. Huang, "Optimal look up schemes for VLSI implementation of input/output conversions and other residue number operations," in *VLSI Signal Processing VII*, J. Rabaey, P. M. Chau and Eldon, eds., IEEE Press, New York, 1994.
- [60] B. Parhami, *Computer Arithmetic: Algorithms and Hardware Designs*, Oxford University Press, New York, 2000.
- [61] D.K. Pradhan, "Universal test sets for multiple fault detection in AND-EXOR arrays", *IEEE Transactions on Computers*, Vol. 27, No. 2, pp. 181-187, February 1978.
- [62] A.P. Preethy and D. Radhakrishnan, "A 36-bit balanced moduli MAC architecture", *Proceedings of IEEE Midwest Symposium on Circuits and Systems*, New Mexico, USA, pp. 380-383, August 1999.
- [63] A.P. Preethy, D. Radhakrishnan and A.Omondi, "A high performance RNS multiply-accumulate unit", *11th Great Lakes symposium on VLSI*, USA, pp.145-148, March 2001a.
- [64] A.P. Preethy, D. Radhakrishnan and A. Omondi, "Fault-tolerance scheme for an RNS MAC: performance and cost analysis", *Proceedings of IEEE International*

Symposium on Circuits and Systems (ISCAS 2001), Sydney, Australia, Vol. 2, pp. 717-720, May 2001b.

- [65] S. Rabii and B.A. Wooley, *The Design of Low-Voltage, Low-Power Sigma-Delta Modulators*, The Springer International Series in Engineering and Computer Science, First Edition, 1998.
- [66] D. Radhakrishnan and Y. Yuan, "A fast RNS Galois field multiplier", *IEEE International Symposium on Circuits and Systems*, LA, USA, Vol.4, pp. 2909-2912, May 1990.
- [67] D. Radhakrishnan, "Modulo multipliers using polynomial rings", *IEE Proceedings Circuits, Devices and Systems*, Vol. 145, No.6, December 1998.
- [68] D. Radhakrishnan and A.P. Preethy, "A direct analog-to-residue converter", *IEEE Region 10 International Conference on Global Connectivity in Energy, Computer, Communication and Control (TENCON1998)*, Vol.2, pp.336-339, 1998.
- [69] D. Radhakrishnan and A.P. Preethy, "A parallel approach to direct analog-to-residue conversion", *Information Processing Letters*, Vol. 69, No. 5, pp. 249-252, March 1999.
- [70] D. Radhakrishnan, T. Srikanthan and J. Mathew, "Using the 2ⁿ property to implement an efficient general purpose residue-to-binary converter", *Proceedings SCS '99*, Iasi, Romania, pp. 183-186, July 1999.
- [71] H. Rahaman, D.K. Das and B.B. Bhattacharya, "Testable design of GRM network with EXOR-tree for detecting stuck-at and bridging faults", *Proceedings of the 2004 Asia and South Pacific Design Automation Conference*, pp. 224-229, January 2004.
- [72] H. Rahaman, J. Mathew and D.K. Pradhan, "Constant function independent test set for fault detection in bit parallel multipliers in GF(2^m)", *20th International Conference on VLSI Design held jointly with 6th International Conference on Embedded Systems*, Bangalore, India, pp. 479-484, January 2007.

- [73] J. Ramirez, A. Garcia, U. M-Baese and A. Lloris, "Fast RNS FPL-based communications receiver design and implementation", *FPL 2002*, LNCS 2438, pp. 472-481, September 2002.
- [74] J. Ramirez, A. Garcia, S.L. Buedo, and A. Lloris, "RNS-enabled digital signal processor design", *IEE Electronics Letters*, 2002, Vol.38, pp. 266–268.
- [75] S.M. Reddy, "Easily testable realizations for logic functions", *IEEE Transactions on Computers*, Vol. C-21, No. 11, pp. 1183-1188, November 1972.
- [76] C. Reis and J.A.T. Machado "An Evolutionary Approach to the Synthesis of Combinational Circuits", *Proceedings of IEEE International Conference on Computational Cybernetics Siófok*, Hungary, August 2003.
- [77] K.K. Saluja and S.M. Reddy, "Fault detecting test sets for Reed-Muller canonic networks", *IEEE Transactions on Computers*, Vol. C-24, pp. 995-998, October 1975.
- [78] T. Sasao, "Logic Synthesis with EXOR gates", *Logic Synthesis and Optimization*, T. Sasao, ed. Kluwer Academic Publishers, London, 1993a.
- [79] T. Sasao, "AND-EXOR expressions and their optimization", *Logic Synthesis and Optimization*, T. Sasao, ed. Kluwer Academic Publishers, London, 1993b.
- [80] T. Sasao, "Easily testable realizations for generalized Reed-Muller expressions", *IEEE Transactions on Computers*, Vol. 46, No. 6, pp. 709-716, June 1997.
- [81] H. Schulze and C. Luders, *Theory and Applications of OFDM and CDMA: Wideband Wireless Communications*, John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester, West Sussex PO19 8SQ, England, 2005.
- [82] Shahana T. K., Rekha K. James, Babita R. Jose, K. Poulouse Jacob and Sreela Sasi, "Performance Analysis of FIR Digital Filter Design: RNS Versus Traditional", *7th IEEE International Symposium on Communications and Information Technologies (ISCIT 2007)*, Sydney, Australia, pp. 1-5, 16-19 October 2007.

- [83] Shahana T. K., Babita R. Jose, Rekha K. James, K. Poulouse Jacob and Sreela Sasi, "RNS based Programmable Multi-mode Decimation Filter for WCDMA and WiMAX", *IEEE 67th Vehicular Technology Conference: VTC2008-Spring*, Singapore, pp.1831-1835, 11–14 May 2008a.
- [84] Shahana T. K., Babita R. Jose, Rekha K. James, K. Poulouse Jacob and Sreela Sasi, "Dual-Mode RNS based Programmable Decimation Filter for WCDMA and WLANa", *IEEE International Symposium on Circuits and Systems (ISCAS 2008)*, Washington, USA, pp. 952-955, 18-21 May 2008.
- [85] F. Sheikh and S. Masud, "Efficient sample rate conversion for multi-standard software defined radios", *IEEE International Conference on Acoustics, Speech and Signal Processing*, HI, pp. II-329 – II-332, April 2007.
- [86] A.P. Shenoy and R. Kumaresan, "Fast base extension using a redundant modulus in RNS", *IEEE Transactions on Computers*, Vol. 38, No. 2, pp. 292-297, February 1989.
- [87] M.A. Soderstrand, "A high-speed low-cost recursive digital filter using residue number arithmetic", *Proceedings of the IEEE*, Vol. 65, No. 7, pp. 1065-67, July 1977.
- [88] M.A. Soderstrand, W.K. Jenkins, G.A. Jullien, and F.J. Taylor, *Residue Number System Arithmetic: Modern Applications in Digital Signal Processing*, (IEEE Press, New York, 1986).
- [89] M.A. Soderstrand and R.A. Escott, "VLSI implementation in multiple-valued logic of an FIR digital filter using residue number system arithmetic", *IEEE Transactions on Circuits and Systems*, Vol. 33, No. 1, pp. 5-25, January 1986.
- [90] D. Soudris, M. Perakis, X. Mizas, V. Mardiris, K. Katis, C. Dre et. al., "Low power design of a multi-mode transceiver", *Proceedings of IEEE International Symposium on Circuits and Systems*, Geneva, Vol. 2, pp. 721-724, 2000.

- [91] T. Srikanthan, M. Bhardwaj and C.T. Clarke, "Area-time-efficient VLSI residue-to-binary converters", *IEE Proceedings on Computers and Digital Techniques*, Vol. 145, No. 3, pp. 229-235, May 1998.
- [92] P. Sweeney, *Error Control Coding: From Theory to Practice*, John Wiley & Sons Ltd., Baffins Lane, Chichester, West Sussex PO19 1UD, England, 2002.
- [93] N.W. Szabo and R.I. Tanaka, *Residue Number Arithmetic and Its Application to Computer Technology*, McGraw Hill, New York, 1967.
- [94] E.C. Tan and C.Y. Chia, "Alternative algorithm for optimization of Reed-Muller universal logic module networks", *IEE Proceedings Computers and Digital Techniques*, Vol. 143, No. 6, pp. 385-390, November 1996.
- [95] Ze Tao and S. Signell, "Multi-standard delta-sigma decimation filter design", *IEEE Asia Pacific Conference on Circuits and Systems (APCCAS 2006)*, Singapore, pp. 1212-1215, December 2006.
- [96] B.Tarokh and H.R. Sadjadpour, "Construction of OFDM M-QAM sequences with low peak-to-average power ratio", *IEEE Transactions on Communications*, Vol.51, No.1, pp. 25-28, January 2003.
- [97] J.L. Tecpanecatl-Xihuitl, Ashok Kumar and M.A. Bayoumi, "Low complexity decimation filter for multistandard digital receivers", *IEEE International Symposium on Circuits and Systems*, Vol. 1, pp. 552- 555, May 2005.
- [98] S.U. Tezeren, "Reed-Muller codes in error detection in wireless adhoc networks", *M.S. Thesis, Naval Postgraduate School*, Monterey, California, March 2004.
- [99] J. Torresen, "A divide-and-conquer approach to evolvable hardware", *Proceedings of the Second International Conference on Evolvable Hardware*, pp. 57-65, 1998.
- [100] C.S. Tsai and B.C. Huang, "Concatenated codes design for OFDM based wireless local area networks", *Third international working conference on Performance Modelling and Evaluation of Heterogeneous Networks (HET-NETs)*, West Yorkshire, U.K, July 2005.

- [101]D. Varma and E.A. Trachtenberg, "Computation of Reed Muller expansions of incompletely specified Boolean functions from reduced representations", *IEE Proceedings Computers and Digital Techniques*, Vol. 138, No. 2, pp. 85-92, March 1991.
- [102]Z. Wang, G.A. Jullien and W.C. Miller, "An algorithm for multiplication modulo $2^N - 1$ ", *Proc. 39th IEEE Midwest Symposium on Circuits and Systems*, pp. 1301-1304, 1996.
- [103]R.W. Watson and C.W. Hastings, "Self-checked computation using residue arithmetic", *Proceedings of the IEEE*, Vol. 54, No. 12, pp. 1920-1931, December 1966.
- [104]A. Xotta, A. Gerosa and A. Neviani, "A multi-mode $\Sigma\Delta$ analog-to-digital converter for GSM, UMTS and WLAN," *IEEE International Symposium on Circuits and Systems*, vol.3, pp. 2551-2554, May 2005.
- [105]L. Xu, A.E.A. Almaini and J.F. Miller, L. McKenzie, "Reed-Muller universal logic module networks", *IEE Proceedings Computers and Digital Techniques*, Vol. 140, No. 2, pp. 105-108, March 1993.
- [106]H.K. Yang and W.M. Snelgrove, "High speed polyphase CIC decimation filters", *IEEE International Symposium on Circuits and Systems (ISCAS '96)*, GA, USA, Vol. 2, pp. 229-232, May 1996.
- [107]L.L Yang and L Hanzo, "Residue number system based multiple code DS-CDMA systems", *Proceedings of IEEE 49th Vehicular Technology Conference*, Houston, USA, Vol.2, pp. 1450-1454, May 1999.
- [108]L.L. Yang and L. Hanzo, "Redundant residue number system based error correction codes", *Proceedings of IEEE VTS 54th Vehicular Technology Conference*, Atlantic City, USA, Vol.3, pp.1472-1476, 2001.
- [109]L.L Yang and L Hanzo, "A residue number system based parallel communication scheme using orthogonal signaling: Part I- System outline", *IEEE Transactions on Vehicular Technology*, vol.51, No.6, pp.1534-1546, November 2002a.

- [110] L.L. Yang and L. Hanzo, "A residue number system based parallel communication scheme using orthogonal signaling: Part II- Multipath fading channels", *IEEE Transactions on Vehicular Technology*, vol.51, No.6, pp.1547-1559, November 2002b.
- [111] L.L. Yang and L. Hanzo, "Coding theory and performance of redundant residue number system codes", 2004. URL: http://www-mobile.ecs.soton.ac.uk/lly/papers/RRNS_code.pdf
- [112] S.S.S. Yau and Y.C. Liu, "Error correction in redundant residue number system", *IEEE Transactions on Computers*, Vol. C-22, No. 1, pp. 5-11, January 1973.
- [113] L. Zhang, V. Nadig and M. Ismail, "A high order multi-bit $\Sigma\Delta$ modulator for multi-standard wireless receiver", *IEEE International Midwest Symposium on Circuits and Systems*, pp. III-379-III-382, 2004.

LIST OF PUBLICATIONS OF THE AUTHOR

International Journals:

1. **Shahana T. K.**, Babita R. Jose, Rekha K. James, K. Poulouse Jacob and Sreela Sasi, “RNS based Programmable Decimation Filter for Multi-Standard Wireless Transceivers”, *ECTI Transaction on Electrical Engineering, Electronics and Communications*, ECTI – Transaction Journal, Vol. 6, No. 2, pp. 57-66, 2008.
2. **Shahana T. K.**, Babita R. Jose, Rekha K. James, K. Poulouse Jacob and Sreela Sasi, “A Toolbox Approach to Decimation Filter Design for Multi-Standard Wireless Transceivers”, *IETECH Journal of Communication Techniques, International Engineering and Technology Publications*, Vol.2, No.3, pp. 181-188, 2008.
3. Babita R. J., **Shahana T. K.**, P. Mythili and J. Mathew, “Sigma-Delta Analog to Digital Converter for WLAN with RNS based Decimation Filter”, *IETECH Journal of Information Systems, International Engineering and Technology Publications*, Vol. 2, No. 2, pp. 68-75, 2008.

Journal Papers Communicated:

4. **Shahana T. K.**, Babita R. Jose, K. Poulouse Jacob and Sreela Sasi, “A Novel Sigma-Delta based Parallel Analog-to-Residue Converter”, *Paper Communicated to International Journal of Electronics*. Taylor and Francis Ltd.
5. **Shahana T. K.**, Babita R. Jose, K. Poulouse Jacob and Sreela Sasi, “Decimation Filter Design Toolbox for Multi-standard Wireless Transceivers”, *Paper Communicated to Wireless Networks: The Journal of Mobile Communication, Computation and Information*, Springer Netherlands.

International Conferences:

6. **Shahana T. K.**, Babita R. Jose, Rekha K. James, K. Poulouse Jacob and Sreela Sasi, "RRNS-Convolutional encoded Concatenated Code for OFDM based Wireless Communication", *Accepted for 16th IEEE International Conference on Networks (ICON 2008)*, New Delhi, India, 12-14 December 2008.
7. **Shahana T. K.**, Babita R. Jose, Rekha K. James, K. Poulouse Jacob and Sreela Sasi, "Dual-Mode RNS based Programmable Decimation Filter for WCDMA and WLANa", *IEEE International Symposium on Circuits and Systems (ISCAS 2008)*, Washington, USA, pp. 952-955, 18-21 May 2008.
8. **Shahana T. K.**, Babita R. Jose, Rekha K. James, K. Poulouse Jacob and Sreela Sasi, "RNS based Programmable Multi-mode Decimation Filter for WCDMA and WiMAX", *IEEE 67th Vehicular Technology Conference: VTC2008-Spring*, Singapore, pp.1831-1835, 11-14 May 2008.
9. **Shahana T. K.**, Rekha K. James, Babita R. Jose, K. Poulouse Jacob and Sreela Sasi, "Polyphase Implementation of Non-recursive Comb Decimators for Sigma-Delta A/D Converters", *IEEE International Conference on Electron Devices and Solid-State Circuits (EDSSC 2007)*, Taiwan, pp. 825-828, 20-22 December 2007.
10. **Shahana T. K.**, Babita R. Jose, Rekha K. James, K. Poulouse Jacob and Sreela Sasi, "GUI Based Decimation Filter Design Tool For Multi-Standard Wireless Transceivers", *IET International Conference on Information and Communication Technology in Electrical Sciences (ICTES 2007)*, Chennai, India, pp. 600-605, 20-22 December 2007.
11. Babita R. J., **Shahana T. K.** and P. Mythili, "Wideband Low-Distortion Sigma-Delta ADC for WLAN with RNS based Decimation Filter", *IET International Conference on Information and Communication Technology in Electrical Sciences (ICTES 2007)*, Chennai, India, pp. 546-552, 20-22 December 2007.

12. **Shahana T. K.**, Rekha K. James, Babita R. Jose, K. Poulouse Jacob and Sreela Sasi, "Performance Analysis of FIR Digital Filter Design: RNS Versus Traditional", *7th IEEE International Symposium on Communications and Information Technologies (ISCIT 2007)*, Sydney, Australia, pp. 1-5, 16-19 October 2007.
13. **Shahana T. K.**, Rekha K. James, K. Poulouse Jacob, Sreela Sasi, "Genetic Algorithm-based Combinational Logic Synthesis using Universal Logic Modules", *ESA'07 - The 2007 International Conference on Embedded Systems and Applications, WORLDCOMP 2007*, Las Vegas, Nevada, USA, pp. 210-215, 25-28 June 2007.
14. **Shahana T. K.**, Rekha K. James, Poulouse Jacob, and Sreela Sasi, "Automated Synthesis of Delay-Reduced Reed-Muller Universal Logic Module Networks", *Proceedings of 23rd IEEE Norchip Conference*, Oulu, Finland, pp. 90-93, 21-22 November 2005.

National Conference:

15. **Shahana T. K.**, Babita R. Jose, Rekha K. James, K. Poulouse Jacob and Sreela Sasi, "Performance Evaluation of RNS based Decimation Filter for Wideband Wireless Transceivers", *National Conference on Broadband Technologies (Broadband 08)*, organized by MBCET Trivandrum in association with IEEE Kerala section and CSI Trivandrum chapter, March 2008.

INDEX

A		Convolutional encoder	101, 102
Additive white Gaussian noise	95, 97, 154	Cyclic prefix	154 -157
Aliasing	25, 28, 92	D	
Analog cellular age	4	Decimation filter	13, 14, 19
Analog-to-digital converter	5, 9	Single stage	24, 25
Nyquist rate ADC	10	Multistage	24, 25
Oversampling ADC	10	Difference amplifier	17
Analog-to-residue converter	83	Differential QPSK	102
AND-XOR algebra	109	Digital cellular age	4
Canonical form	110, 111	Digital signal processing	5, 22, 42, 44, 97
Antialias filter	7, 10, 92	Direct conversion homodyne receiver	7, 8, 21, 22, 65
B		Downsampler	20, 24, 40, 93
Bit error rate, BER	95, 154 -159	Droop compensation	29, 34
Blocking and interference profile	19, 30, 66	Dual-mode decimation filter	63
Buffer	84, 85	WCDMA/WiMAX	65
C		Simulation results	141
Carrier to noise ratio	19, 31	WCDMA/WLANa	69
Carry propagate adder	57, 60, 61	Simulation results	144
Carry save adder	57, 61	E	
Cascaded integrator comb	26 - 28, 42, 43	Easily testable circuits	112
Chinese Remainder Theorem	54, 61, 99	Simulation results	160
Chromosome	122, 123	Error cancellation logic	91
Clip compression ratio	154, 158, 159	Error detection and correction	97, 98
Cognitive radio	180	Error syndrome	99, 103
Concatenated coding	94, 97, 100	Exclusive OR sum-of-products	111
Constraint length	101	Exhaustive branching technique	115
		Algorithm	117-119
		Synthesis results	163
		Evolvable hardware	119

F

Fast Fourier Transform	103, 154
Finite impulse response filter	24, 55
Performance analysis	135
Fixed polarity Reed-Muller	111
Flash ADC	84, 87
Forward converter	57, 58, 64, 68
Forward error correction	95
Frame start synchronization errors	95, 97, 154, 157
Full adder	113, 137, 138, 139, 146, 152, 160, 161, 162

G

GA based logic synthesis	119
Algorithm	123
Synthesis results	167
GA operators	
Reproduction	122
Crossover	122
Mutation	123
Galois field	72, 76
Generalized Reed-Muller	111
Global roaming	5
GSM	20, 23, 27, 28
Guard band	102, 103, 153, 155, 157
GUI	32, 33, 35
GUIDE	20, 32

H

High speed cellular age	4
Halfband filter	28, 29
Hogenauer CIC filter	26, 42

I

Illegitimate range	98
--------------------	----

Image rejection	6
Index calculus	72, 73, 74, 77, 78
Intercarrier interferences	102, 103
Intersymbol interferences	102, 103, 155, 156
Inverse fast Fourier transform	102, 154
Iterative subranging flash A/R converter	87

L

Legitimate range	98
Local oscillator	6, 7, 21
Look up table	55, 60, 68, 86, 87, 99
Loop filter	5, 11
Low cost moduli	54
Low IF receiver	8

M

Mixed radix conversion	54, 99
Modulo addition	57, 59
Modulo multiplication	57, 59, 68, 74, 76
Multipath delay spread	95, 103, 155
Multipath fading channel	97, 155
Multiple residue flash A/R converter	84
Multiply and accumulate	51, 55, 60, 67, 71, 93, 112, 113, 160
Multistage noise shaping	22, 23, 65, 69, 91, 150
Multi-standard	5 - 9, 14, 19 - 22, 63, 64, 172

N

Noise shaping	11, 13, 14, 19, 90, 91
Noise transfer function	13, 22, 91

Non-recursive CIC filter 43
 Nonredundant moduli 97, 101, 103
 Nyquist rate A/R converter 83, 89
 Nyquist theorem 10

O

OFDM 4, 5, 94, 95,
 96, 97, 101, 102, 104, 154
 Out-of-band noise 11, 13
 Oversampling ratio 5, 10, 19, 22, 30

P

Peak power clipping 95, 97, 104, 158
 Peak to average power ratio 96, 104
 Pitem 113, 117, 118, 119
 Pole-zero plot 37, 38, 39, 130
 Polyphase decomposition 42, 45, 46
 Polyphase non-recursive CIC 44
 Simulation results 131
 Positive polarity Reed-Muller
 110, 111, 112, 114
 Primitive root 72, 74, 76, 77
 Programmable decimation filter
 64, 65, 67, 70, 74

Q

QAM 95, 96
 QPSK 95, 102
 Quantization noise 11, 19, 42, 91
 Quantizer 5, 11, 12, 13

R

Radio frequency 5, 6, 7, 9
 Receiver model 102
 Reconfigurable sigma-delta ADC 22
 Recursive CIC filter 42, 43

Redundant moduli 97, 98, 101, 103
 Reed-Muller logic 109, 121
 Register growth 42, 43
 Residue Number System 51
 RNS basics 51
 RNS arithmetic 52, 54
 Dynamic range 52, 57, 75
 RNS moduli 54
 Reverse converter 57, 61, 68
 RM-ULM 113, 114, 115, 121
 RRNS-convolutional concatenated
 coding 94, 100

S

Sampling rate conversion 25, 40
 Sigma-delta ADC 5, 6, 11,
 12, 19, 22, 83
 Sigma-delta modulator 12, 21, 83, 90
 Dynamic range 22, 23, 30
 DR equation 23
 Simulink models 150
 Signal to noise ratio
 13, 19, 90, 154, 159
 Signal transfer function 13
 Single stuck-at faults 112, 174
 Software defined radio 5, 19
 Successive approximation
 83, 86, 87, 88
 Superheterodyne receiver 6, 7

T

Testability 112, 124
 Toolbox 20, 23, 30, 32
 Transmitter model 101
 Tree network 114, 115, 124
 Triple-mode decimation filter 180
 Triplet index code 72, 77

U			
Uncoded OFDM	154 – 158	Wideband IF double conversion receiver	8, 9
Universal logic module	113, 120, 167	WiMAX	4, 20, 23, 31, 32
Up-conversion	102	Wireless standards	14, 20
		WLANa, b, g	14, 20, 23, 31, 32
		X	
		XOR gate	84
		XOR logic	109
V			
VHDL code	131, 146, 149		
Video conferencing	95		
Viterbi decoding	103, 105		
VLSI	11, 20, 43, 60, 114, 120, 125		
		Z	
		Zero-IF receiver	7
		Z-plane	37
W			
WCDMA	20, 23, 31, 32, 64		