FPGA PERFORMANCE ANALYSIS OF DIFFERENT FORWARD ERROR CORRECTION CODES FOR 5G COMMUNICATION SYSTEM

A Thesis submitted to the

UPES

For the Award of

Doctor of Philosophy

in

Electronics Engineering

By

Aakanksha Devrari

April 2024

SUPERVISOR (s)

Dr. Adesh Kumar

Department of Electrical & Electronics Engineering

School of Advanced Engineering (SoAE)

UPES

Dehradun- 248007: Uttarakhand

April 2024

FPGA PERFORMANCE ANALYSIS OF DIFFERENT FORWARD ERROR CORRECTION CODES FOR 5G COMMUNICATION SYSTEM

A Thesis submitted to the

UPES

For the Award of

Doctor of Philosophy

in

Electronics Engineering

By

Aakanksha Devrari

(SAP ID 500080559)

April 2024

SUPERVISOR

Dr. Adesh Kumar

Professor

Department of Electrical & Electronics Engineering

Department of Electrical & Electronics Engineering

School of Advanced Engineering (SoAE)

UPES

Dehradun- 248007: Uttarakhand

April 2024

DECLARATION

I declare that the thesis entitled **"FPGA Performance Analysis of Different Forward Error Correction codes for 5G Communication System"** has been prepared by me under the guidance of **Dr. Adesh Kumar**, Professor, Department of Electrical & Electronics Engineering, School of Advanced Engineering, UPES. No part of this thesis has formed the basis for the award of any degree or fellowship previously.

Aakanksha Devrari
Department of Electrical & Electronics Engineering
School of Advanced Engineering,
UPES
Dehradun-248007, Uttarakhand

Date: April 2024

UPES
UNIVERSITY OF TOMORROW

Great Place To Work. Certified FEB 2023 – FEB 2024 INDIA

CERTIFICATE

I certify that **Aakanksha Devrari** has prepared her thesis entitled **"FPGA Performance Analysis of Different Forward Error Correction Codes for 5G Communication System"** for the award of Ph.D degree of the UPES, under my guidance. She has carried out work at the School of Advanced Engineering, UPES.

Supervisor

Dr. Adesh Kumar

Professor

Department of Electrical & Electronics Engineering

School of Advanced Engineering,

UPES, Dehradun -248007, Uttarakhand

Date: 22 April 2024

ABSTRACT

The fifth generation (5G) of mobile communication is a new universal wireless standard that permits the formation of an original form of the network, which associates everything and everyone globally, comprising objects, gadgets, and machines. The 5G wireless technology has been experienced to provide fast data speeds, low latency, wideband spectrum full connectivity, availability, massive network capacity, and a more reliable user experience. Higher efficiency and performance must agree for innovative and user experiences in industrial relations. Higher data rates of up to 20 Gigabits per second (Gbps) highest data rates, and 100+ Megabits per second (Mbps) normal data rates are possible with 5G networks the research work provides the direction to the 5G technology, network evolution, generation networks, existing cell networks-challenges, problems, 5G potential areas, standardization and network requirements.

The problem statement of this research work is “The evaluation of different channel coding techniques based on latency, complexity, reliability, and flexibility as per the needs of the user in machine type communication”. Hence, the evaluation of different channel coding methods mainly focuses on fulfilling the requirements of users. At present, the channel coding methods for 5G mobile communication are LDPC, turbo, and polar codes [5, 8]. These channel coding schemes need to be analyzed from a hardware utilization point of view to ensure the user requirements in the real-time world. Extensive research is done on the use of Low Density Parity Check (LDPC), turbo, and polar encoders and decoders using different algorithms. The comparative performance analysis of these coding-based encoder and decoder architectures on FPGA platform will be the new research work and can be used further for the device-to-device and machine-to-machine communication as in 5G communication system. The Research work is focused on the design and FPGA implementation of the turbo, LDPC, and polar encoder and decoder hardware chip, performing simulation and estimating the comparative performance. The following FPGA parameters are used to support the design: number of flip-flops, number of slices, number

of I/O Blocks, number of LUTs, memory usage, maximum frequency, and combinational delay, minimum and maximum time of clock, and power.

An overview of the different channel coding techniques is also discussed such as Low-LDPC, turbo codes, and polar codes. In multiuser wireless communication contexts like the spread spectrum, several codes with strong correlation properties are used. Because shifting processes can generate a large number of coding schemes with controlled operations, various error detection and correction codes, like gold codes, LDPC codes, and turbo codes are appropriate for such communication systems. The linear feedback shift register (LFSR) is used for generating pseudo-random numbers, error controlling, and high-speed operations in hardware. LFSR deftly adjusts and feeds back the output of a traditional shift register into its input to make the function cycle through a variety of patterns endlessly. LFSRs are mostly used for error detection and correction in cyclic redundancy codes (CRC).

In the past ten years, one of the most significant advancements that have been made in the field of channel coding is known as the polar code. Polar codes have incredible performance enough to persuade the third-generation partnership project (3GPP) to implement their use for control channel activities in one of the enhanced mobile broadband possibilities for the fifth generation. The entirety of the industry has not yet decided on the channel coding that will be used for the various use cases of 5G, like ultra-reliable low latency communication (URLLC) and massive machine type communication (MMTC) and ultra-reliable low latency communication (URLLC). The following analysis concentrates on the channel coding techniques, and more precisely for the URLLC use case of 5G New Radio (5G-NR), it also examines how well polar codes operate in this environment. An investigation of polar codes utilizing a wide range of performance factors is carried out to satisfy the prerequisites of the URLLC scenario. The extremely reliable polar code has a superior error-correcting capability, and when combined with its low computational complexity and decoding delay, it becomes a serious rival in the battle for URLLC channel coding.

Over the past ten years, polar codes have captured people's interest recently in both the academic community and the business world. Because of this, 3GPP concluded that polar codes would be the best option for the channel coding scheme that would be used in the standardization process for the 5^{th} generation of wireless networks. The difficulties associated with developing a family of polar codes that are suitable for meeting the requirements of 5G systems, such as the rate flexibility and short decoding time is demonstrated. The next generation wireless systems will utilize polar codes, which are error-correcting codes with a high capacity and were consequently authorized for use in this generation (5G). The algorithms used for decoding polar codes have progressed in several different ways over time, striking multiple trades between complexity and error correction performance. Successive cancellation list (SCL) along with its several offshoots make up a powerful and extensively researched family of algorithms that are always being developed further.

The coding techniques for wireless communication must fulfill the system requirements in terms of hardware while the device and machine communication is taking place. The turbo codes are providing a good coding gain which is close to Shannon’s limit, whereas LDPC codes can provide error corrected data when the channel is noisy. The primary requirement of the channel coding method in a 5G mobile communication system for the transmission of a short length message is the flexibility. The flexibility is directly related to its hardware and software implementation. The second requirement of the channel coding methods in a 5G mobile communication system is the complexity. Evaluation of algorithmic complexity in different channel coding techniques is very hard. The third requirement of the channel coding method in a 5G mobile communication system is the latency. Latency means the time taken by the message bits to pass through the entire communication system while considering different channel coding techniques. Finally, the last user requirement is the reliability which means the techniques needs to be reliable for any given message length sequence.

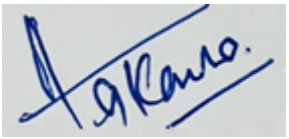
The research work presents the performance evaluation of LDPC, turbo, and polar codes in terms of hardware architecture. The hardware chip of encoder

and decoder for turbo, LDPC, and polar is designed using Xilinx ISE 14.7 software, which is targeted for Virtex- 5 FPGA. The FPGA hardware complexity is examined in terms of FPGA performance parameters like flip flops, LUTs, IoB, and slices utilization. The performance of these coding techniques is also analyzed in terms of timing information parameters like minimum duration, minimum and maximum time of the clock signal, and path delay, etc. The research work is supportive for 4G and 5G mobile systems in device-to-device communication. The novelty of the work is that the design is scalable and can be extended based on the requirements of the systems which is synthesized and experimentally verified on a Virtex 5 FPGA board. The concept of this design is programmable and can be extended to n-bit based on the applications. Concurrently, implementation mechanism can have a significant impact on the area occupations and delay values that are produced. There is currently no comprehensive research that takes into consideration all of these parameters that look at 5G's priorities.

ACKNOWLEDGEMENT

I would like to express my unrestrained appreciation to my thesis internal supervisor Dr. Adesh Kumar, Professor, School of Advanced Engineering, UPES, Dehradun, Uttarakhand for their invaluable and diligent guidance, patience, immense knowledge, also kind and continuous encouragement during my Ph.D. study and related research. I am grateful for their contributions of time, ideas, and motivation, which have provided a good base for my research work. I could not have imagined having better advisors and mentors for my Ph.D. study.

My sincere thanks also goes to Prof. Piyush Kuchhal, Cluster Head, Electrical Cluster, SoAE, UPES, Dehradun for his support and suggestions on my Ph.D. topic. The meetings and conversations were vital in inspiring me to think outside the box, from multiple perspectives to form a comprehensive and objective critique. Lastly, I would like to thank my family members for their love, encouragement, immense trust, and support throughout my Ph.D. study and research work and my life in general. Without their blessings, I would not be able to complete this research work effectively.



(Aakanksha Devrari)

Table of Contents

List of Tables

List of Figures

List of Abbreviations

Notation	Description
$R_{A, B}(\tau)$	Cross-Correlation Function
'τ'	Time Lag between the Functions
*A**	Complex Conjugate of Source A
$r_{xx}(T)$	Autocorrelation Function
x(n-T)	Shifted Version
L	Length of the sequence, Decoding Routes
<	Less Than
>	Greater Than
X	Times
f	Frequency
c	Speed of Light
Xn	Time Domain Signal
Xk	Frequency Domain Signal
Nsc	Sub Carrier
K	Sub carrier, Register, Time samples
S(n)	Inverses DFT Output
h(n)	Transfer Function of the Channel
r(n)	Received Signal
N	Frequency Samples, Clock Cycles, Cell Number
K	Input Bits
N	Sub Symbol, AWGN, Code Length
d(k,n)	Transmitted Signal
a	Information Flow, Number of Users
d	Symbols
x	Encoded Data Sequence
s	Output Data of the Filter
y	'N' Point Generated FFT Sequence
Pa	User Transmitted Power

Xa	Transmitted Symbols
η	Spectral Efficiency
R	Rate of Transmission
W	Bandwidth
m	Memory Blocks
t	Time Period
T	Time delay, Constraint length
Δt	Time Instance Value
q	Symbol
ff	Flip Flop
D	States of Shift Register
g(x)	Generator Polynomial
X_0	Final Term in Polynomial
J	Register
M	Switches
$g_1(X)$	Primitive Polynomial
$g_2(X)$	Primitive Polynomial
C	Channel Capacity
B	Channel Bandwidth
S/N	Signal to Noise ratio
H	Parity Check Matrix
wc	Column Weight
wr	Row Weight
CN	Check Node
VN	Variable Node
G	Generator Matrix
Cw	Code Word
di	Binary Message Inputs
S(n)	States of Turbo Encoder
α	Forward Matrix Value
β	Backward Matrix Value
pe1	Parity Bits
pe2	Parity Bits

μ	Scaling Factor
W	Channels
Wvec	Vectored Channels
X	Binary Discrete Memory Less Channel
Z	Output of the Channel
Y	Input of the Channel
X(z—y)	Transition Probability of the Channel
J(X)	Symmetric Channel Capacity
A(X)	Bhattacharyya Parameter
G⊗n	Kronecker Product of Matrix G
F	Frozen Set
δ	Probability Value
u	Input Vector Value
Qc	Frozen Bits
Lui	Log Likelihood Ratio Value
α'_k	Forward State Matric
β'_k	Backward State Matric
γ'_k	Branch Matric
s_k	Trellis Diagram States at Time 'k'
L_a	Priori LLR
d_{in}	Methodical Bit to the Realization of Extrinsic Information
P_{k1}	Bit for Parity Check

Acronyms	**Description**
ADC	Application Delivery Controller
AGU	Address Generator Unit
AMPS	Advanced Mobile Phone System
ASICs	Application-Specific Integrated Circuits
ATC2	Agilent Trace Core 2
AWGN	Additive White Gaussian Noise
BER	Bit Error Rate

BF	Beam Forming
BLER	Block Error Rate
BP	Belief Propagation
BPA	Belief Propagation Algorithm
BPSK	Binary Phase-Shift Keying
BSCAN	Boundary Scan
CCSDS	Consultative Committee for Space Data Systems
CDMA	Code-Division Multiple Access
CDRT	Coordinated Direct and Relay Transmission
CEPT	European Conference of Postal and Telecommunications
CoMP	Compare Communications
CoMP	Coordinated Multipoint
CP	Cyclic Prefix
CPU	Central Processing Unit
CRC	Cyclic Redundancy Codes
CRC	Cyclic Redundancy Check
CRNOMA	Cognitive Radio
CRSC	Circular Recursive Systematic Constituent
CS	Compare and Select
CSI	Channel State Information
CU	Control Unit
C/A	Coarse/Acquisition
DAC	Digital-to-Analog Converter
DE	Density Evolution
DF	Decoders and Forwarders
DGP	Data Generation & Packetization
DSSS	Direct Sequence Spread Spectrum
DVB-C2	Digital Video Broadcasting–Cable 2
DVB- RCS	Digital Video Broadcasting - Return Channel
DVB-S2	Digital Video Broadcasting via Satellite dB
D2D	Device to Device
ECC	Error Control Codes
EDGE	Enhanced Data rates for GSM Evolution

eMBB	Enhanced Mobile Broadband
FBMC	Filter Bank Multi-Carrier
FCC	Federal Communications Commission
FDMA	Frequency Division Multiple Access
FEC	Forward Error Correction
FER	Frame Error Rate
FFs	Flip-flops
FFT	Fast Fourier Transform
FIFO	First Input First Output
FPGA	Field Programmable Gate Array
Gbps	Gigabits Per Second
GFDM	Generalized Frequency Division Multiplexing
GHz	Gigahertz
GSM	Global System for Mobile Communication
HARQ	Hybrid Automatic Repeat Request
HBC	Human Bound Communication
HD	High Definition
HetNets	Heterogeneous Networks
HPF	High-Pass Filter
HSDPA	High-Speed Downlink Packet Access
HSS	Home Subscriber Server
IBA	Integrated Bus Analyzer IP cores
ICI	Inter Cell Interference
ICON	Integrated Controller
IDFT	Inverse Discrete Fourier Transform
IFFT	Inverse Fast Fourier Transform
ILA	Integrated Logic Analyzer
ILM	Inter Leaver Memory
IMT	International Mobile Telecommunications
IoB	Input Output Block
IoT	Internet of Things
IOV	Internet of Vehicles
IP	Internet Protocol

ISI	Inter Symbol Interference
IS-95	Interim Standard 95
ITU	International Telecommunication Union
IUI	Inter User Interference
I/O	Input Output
JTAG	Joint Test Action Group
Kbps	Kilobits Per Second
LCD	Liquid Crystal Display
LDPC	Low-Density Parity Check
LED	Light Emitting Diode
LFSR	Linear Feedback Shift Register
LLR	Likelihood Ratio
LOC	Location Constraints
LOS	Line of Sight
LTE	Long Term Evolution
LUTs	Lookup Tables
Mbps	Megabits Per Second
MCM	Multicarrier Modulation
MGS	Metal Gear Solid
MHz	Megahertz
MIMO	Multi In Multi Out
MIMO-NOMA	Multi-Input Multi-Output Non-Orthogonal Multiple Access
Mm	Millimeter
MMTC	Massive Machine Type Communication
MSPA	Minimum sum Posterior Algorithm
MTS	Mobile Telephone Service
MU-MIMO	Multi-user MIMO
M2M	Mobile to Mobile
NGD	Negative Group Delay
NOMA	Non-Orthogonal Multiple Access
NR	New Radio
NTT	Nippon Telegraph and Telephone Corporation

OCDMA	Optical CDMA
OCDMA	Optical Code Division Multiple Access
OFDM	Orthogonal Frequency Division Multiplexing
OFDMA	Orthogonal Frequency-Division Multiple Access
OMA	Orthogonal Multiple Access
OOB	Out of Band
OP	Outage Probability
OTFS	Orthogonal Time Frequency And Space
PAPR	Peak-to-Average Power Ratio High-Power Amplifier
PC–CA	Parity Check Polar Codes
PD-NOMA	Power Domain NOMA
PEs	Processing Elements
PISO	Parallel-In/ Serial-Out
PN	Pseudo Noise
PRBS	Pseudorandom Binary Sequence
PSTN	Public Switched Telephone Network
PTT	Push to Talk
QAM	Quadrature Amplitude Modulation
QKD	Quantum Key Distribution
QPSK	Quadrature Phase Shift Keying
QR	Quick Response
R	Reed-Solomon
RAG	Reverse Address Generator
RAM	Random Access Memory
RANs	Radio Access Networks
RB-f-OFDM	Resource block f-OFDM
RCC	Radio Common Carrier
RISIC	Residual ISI cancellation
RRC	Root Raised Cosine
RTL	Register Transfer Logic
SCL	Successive Cancellation List
SDMA	Spatial Division Multiple Access
SHA	Shannon’s Theoretical Limit

SIC	Successive Interference Cancellation
SISO	Single-Input and Single-Output
SMS	Short Messaging Service
SNR	Signal to Noise Ratio
SOA	Semiconductor Optical Amplifier
SPA	Sum Posterior Algorithm
SSCL	Simplified Successive Cancellation List
STA	Static Timing Analysis
SU-MIMO	Single-user MIMO
S/P	Serial to Parallel
TD SCDMA	Time Division Synchronous Code Division Multiple Access
TD-SCDMA	Time-Division Synchronous CDMA
TU	Two-User
UCF	User Constraints File
UE	User Equipment
UFMC	Universal Filtered Multi-Carrier Modulation.
URLLC	Ultra Reliable Low Latency Communication
UMTS	Universal Mobile Telecommunication System
UMTS W-CDMA	Universal Mobile Telecommunications System Wideband Code Division Multiple Access
VGA	Video Graphics Array
VHDL	Very High-Speed Integrated Circuit Hardware Description Language
VIO	Virtual Input Output
VLC	Visible Light Communication
V2V	Vehicle-to-Vehicle
W-CDMA	Wideband Code Division Multiple Access
WI	Work Item
Wi-Fi	Wireless Fidelity
WiMAX	Worldwide Interoperability for Microwave Access
W-OFDM	Windowed OFDM
WWWW	World Wide Wireless Web

XST	Xilinx Synthesis Technology
ZF-BF	Zero-Forcing Beamforming
1G	First Generation
2D	Two-Dimensional
3G	Third Generation
3GPP	3rd Generation Partnership Project
3GPP TSG	3rd Generation Partnership Project Technical Specification Groups
4G	Fourth Generation
4G-RAT	4G Radio Access Technologies
5G	Fifth Generation

Units	**Description**
GB/s	Gigabytes per Second
Kbps	Kilobits or Kilobytes per Second
Mbps	Megabits per Second
KHz	Kilohertz
MHz	Megahertz
µs	Micro Second
mm	Millimetre
nm	Nanometer
mW	Milliwatt
V	Volt
K	1000
ms	Millisecond
ns	Nanosecond
KB	Kilobyte
Gbps	Gigabit per Second

CHAPTER 1

INTRODUCTION

The chapter explains the introduction to 5G wireless communication, motivation, and the need for research. The organization of the thesis is also discussed at the end of the chapter.

1.1 Fifth Generation Networks

The objective of next generation 5G wireless systems [1] is to provide remarkably high data rates normally in the gigabit per second range, incredibly low latency, gradually improved base station capacity, and considerably upgraded quality of service when compared to 4G LTE networks. Conventional cellular networks [2] are already undergoing considerable stress because of the growing number of smart devices, interactive media software, and the rapid growth of wireless multimedia data demand and use. This is already straining them. Most of the problems that cell networks are having now are expected to be resolved using 5G wireless systems that will have better capacity, data rate, latency, and quality of service. In this survey, a detailed look at how wireless networks are moving toward 5G networks is given.

RANs have undergone a lot of changes in the way they are built. Based on Marconi's pioneering demonstration, Shannon's 1948 information theory was used to potentially generate wireless communication systems [5]. The 1G analog wireless cellular connectivity has been used to enable mobile telephony in the 1980s before being phased out in the early 1990s and replaced by a 2G digital cellular network. Due to its digitalization, 2G may be used to deliver encrypted and data services alongside customary voice services, like short messaging. W-CDMA, CDMA 2000, TD-SCDMA, and WiMAX were all used in the early twenty-first century to enable multiple data services like mobile television, video calls, and internet access [6].

OFDM, IP, and MIMO antenna framework technology were combined to reach optimal mobile data transmission in 4G-LTE networks launched in 2009 [4]. 4G has been a massive commercial and technological success. Due to the

prevalence of tablets and smartphones, mobile communication systems are becoming popular, providing a significant quantity of data throughput in 4G networks and communication systems.

A large percentage of operators and manufacturers will use the 3GPP 5G new radio benchmark for dense urban areas in the initial deployment stage of 5G networks [7]. The corresponding 5G network uses a frequency range of 2-6 GHz for operating. Despite the delayed densification of the network, millimeter-wave and massive MIMO methods are extensively used in 5G networks. Slicing the network is a technique that is used in a few 5G mission-critical solutions [14].

5G is very well suited for high-definition video services, Internet Protocol television, mobile data transmission over long distances, and basic virtual augmented reality services. Indoor and data services in densely populated areas will remain the 5G era's primary focus. The majority of businesses and device manufacturers are still in the early stages of 5G network deployment. A fully functional 5G network offers three service options for a variety of application scenarios: MMTC [102], eMBB, and URLLC. Supply and demand are the two most important variables. As per the supply side, a few such technologies continue to need experimental validation and extensive field testing [8]. They are currently not widely used due to their high cost and limited functionality. On the consumer side, some innovative communication and collaboration technologies are only capable of supporting a confined number of services and devices. Although 5G shows a steady development strategy that enables it to provide substantially more improved services than 4G, it does not comprise any revolutionary technology.

A thorough review of energy cognizance and cost savings to cut down on the increased network energy consumption and cost of operation is done. Mobile wireless communication systems that were mostly voice-based have been around for more than a few decades. People have been using these systems for a long time now [98]. Table 1.1 depicts the generation in wireless communication networks. Throughout the world, wireless communication has been moving forward at a steady pace over the last few decades, with the

development of second, third, and 4G wireless networks becoming more common. When digital modulation, operating frequency reuse, and packet-based

Table 1.1 Generations in wireless communication networks [9]

Properties	1G	2G	3G	4G	5G
Multiplexing	FDMA	TDMA FDMA	CDMA	OFDMA	OFDMA
Standards	AMPS, PTT, MTS	CDMS, GSM, EDGE, IS-95	IMT 2000[58] TD SCDMA UMTS	LTE, LTE-A, WIMAX	WWWW 5G-NR
Core network	PSTN	PSTN	Packet N/W	Internet	Internet of Things
Maximum frequency	894 MHz	1900 MHz	2100 MHz	6 GHz	90 GHz
period	1980-1990	1990-2000	2000-2010	2010-2020	2020-2030
Maximum rate	2.4 Kbps	144 Kbps	2 Mbps	1Gbps	35.46 Gbps

Internet became more prevalent, the physical layer technologies like W-CDMA, MIMO, and HARQ gained prominence, and are becoming more advanced [10]. In addition, with the rise of smart devices, today's all IP based 4G LTE networks are becoming more and more common in people's homes and places of work. In the end, there will be a new type of mobile multimedia app that is more focused on what the user wants, like mobile teleconferences, streaming video, and e-healthcare, as well as online games. Wireless operators are making more money thanks to new apps that not only meet the needs of the people who use them but also give them new business opportunities, allowing them to make more money.

1.2 Existing Cell Networks: Problems and Challenges

As per the International Telecommunication Union, the wireless network statistics in terms of global mobile traffic increased nearly by 70% in 2014. In

total, there is only 26 percent of all mobile devices are smartphones [15]. They make up 88 percent of all mobile data traffic, but only 26 percent of all mobile devices are smartphones. Ever since 2012, traffic from videos has made up more than half of overall mobile traffic around the world. New applications for improved reality like IoT, D2D communication, Internet of vehicles, healthcare, M2M communication, and financial technology are also being tried out by researchers. It is very hard for current 4G LTE cellular systems to keep up with this huge and fast rise in data use and connectivity.

Assuming a theoretical downlink data rate of 150 Mbps, outdated LTE systems with 22 MIMO can only stream full HD video at a rate of 4 Mbps at the same time in the downlink. As standard LTE networks [12] were initially built to support only about 600 RCC connectivity users in each cell unit, M2M communications, and IoT are expected to support thousands of associated devices within a single LTE cell [55]. To increase capacity and data rates, the LTE cellular network is looking into a lot of different things, like MIMO, CoMP transmission, HetNets, and multiple antennas, among other things. In the long run, though, it's not likely that the current rate of traffic growth will keep up [54]. Thus, it is very important to be able to handle the huge growth in both users and traffic volume in mobile broadband communications.

The mm-wave is a new frontline in the radio spectrum that is very exciting. Wireless communication is limited by the efficiency of the spectral signal and the amount of bandwidth that can be used [32]. It is also based on the size of the cell. When it comes to physical layer technology, Shannon’s capacity limit has already been reached. The system bandwidth has not been looked at yet.

There is some doubt about whether the sub-mm-wave band can handle the rapidly growing amount of mobile traffic and connectivity [16]. As a result, when high-frequency bandwidth comes into play, wireless communications will not be able to help. The key to developing the next generation 5G networks is to look into the mm-wave band, that uses a high-frequency range between 3 to 300 GHz. There are records from the past that show that collision-avoidance radars were initially used in the mm-wave spectrum. 59 to 64 GHz and 81 to 86 GHz, respectively, were opened by the FCC of the United States in December

for wireless and peer-to-peer communications, which do not need to be paid for. Many different things have already been done with the mm-wave bands, like radio telescopes and radar. Fig.1.1 shows that only the 3 GHz to 300 GHz band from which mm-waves can be used for communication isn't good for that. Even a minor portion of the available millimeter-wave spectrum could provide a hundred times the capacity and the data rate of today's cell phone spectrum. Finally, the accessibility of a huge portion of the mm-wave spectrum opens up a whole new realm for upcoming wireless technology that is restricted by frequency band constraints [33].

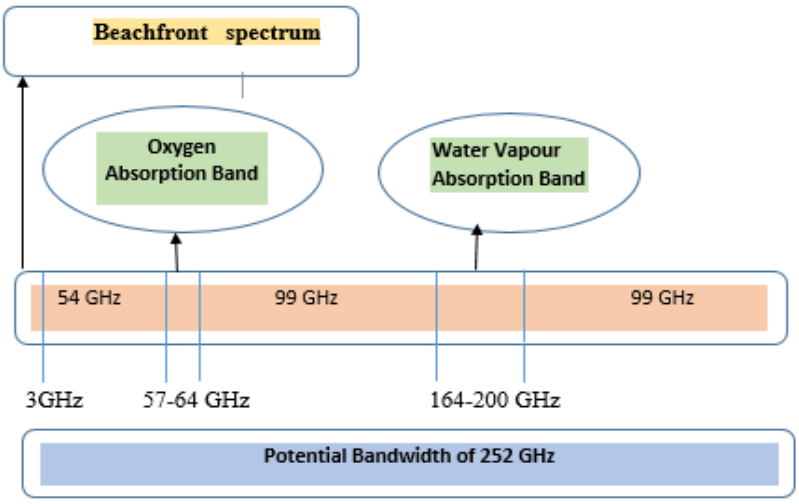


Fig.1.1 Available Spectrum in mm-wave ranging 3 ~ 300 GHz.

1.3 5G Potential and Motivation

The 5G is expected to be the catalyst for the next big change in wireless communications because of the combination of millimeter-wave band access, hyper-connectivity visualization, and new application-oriented [71]. Fig.1.2 shows how 5G wireless communications will allow for huge increases in data rates, connectivity, coverage, and bandwidth while also cutting round trip latency and energy utilization by several orders of magnitude. There will be a lot of different kinds of autonomous devices connected to the Internet of Things when Samsung's 5G vision comes to fruition. As shown in Table 1.2, there are a lot of different views on what 5G needs from both industry and academia [60].

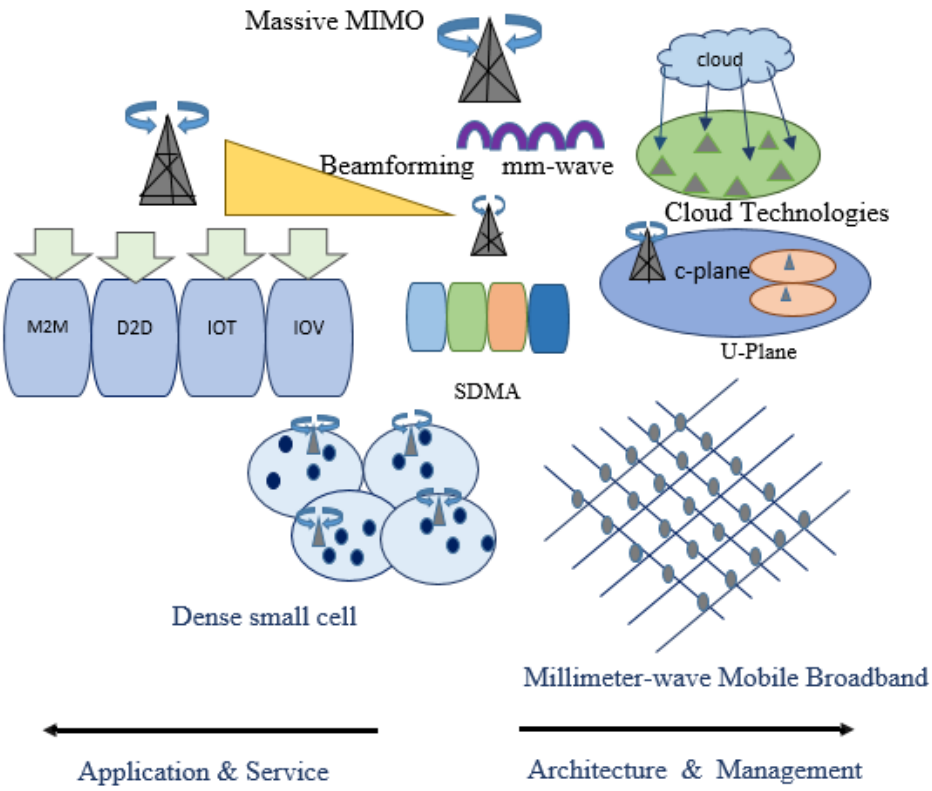


Fig.1.2. Next Generation 5G Wireless Systems [12]

Table 1.2. Industry and its 5G perspective globally [15]

Industry	**Perspective**
Nokia	Channel models Internet & Augmented Reality
Docomo 5G	Wireless connectivity Enriched content and Extensive approach
5G Forum	Heterogeneous network 5G commercialization
5G NOW [43, 50, 51]	Frame concept UFMC Orthogonality
Samsung Electronics	Cloud computing IoT[94]
Qualcomm	User Experience Novel services
Ericson[29]	Sustainable and affordable
Huawei	Deployment services Massive connectivity & capacity Diverse applications

1.4 5G Standardization

A group called ITU decides how much spectrum will be used for 5G amenities in the first place. Regional regulatory bodies, like the FCC and the CEPT, play a big role in the International Telecommunication Union's decision-making process [52]. Following the allocation of spectrum for a specific service, ITU designates a specific allocation for specific technology and sets the desires of that technology. The 3GPP and IEEE, as well as other internal standardization groups, set the rules for how the technology works [56]. Fig.1.3 shows a timeline for the development of 5G as set out by the International Telecommunication Union and the 3GPP [65]. Eight main requirements for the next generation 5G systems have been found by combining research projects from both industry and academia [44]. These requirements are as follows:

(i) There is a real-world data rate of 1 to 10 Gbps, a lot more than the 150 Mbps that traditional LTE networks could theoretically reach.

(ii) 4G has a round trip time of 10 milliseconds, which is almost 10 times slower than 5G.

(iii) To do this, a lot of connected devices need to be able to run at high speeds for a long time in a specific area.

(iv) To make the IoT a reality, new 5G networks should connect a lot of devices at the same time.

(v) In the future, 5G envisions a network that is almost always available, even when there is a lot of traffic.

(vi) Near-complete exposure for "anytime, anywhere" connectivity which means 5G wireless networks must cover all users, no matter what is their existence.

(vii) Standardization groups are already taking into account the development of green technology. Almost 90% less energy is used.

(viii) This is going to become even more important because of the high data rates and wide availability of 5G wireless.

In 5G networks, which are still in the initial stages of growth, having more battery life is very important. Wireless companies, universities, and research groups have started working together on different parts of 5G wireless systems

with the eight requirements above in mind. Table 1.2 shows the 5G visions of well-known wireless companies and operators from around the world. According to Ericsson, the development of 5G should start in a way that is well-matched with 4G LTE networks which are going to keep the services running on the equivalent carrier frequency that was used by old-fashioned devices.

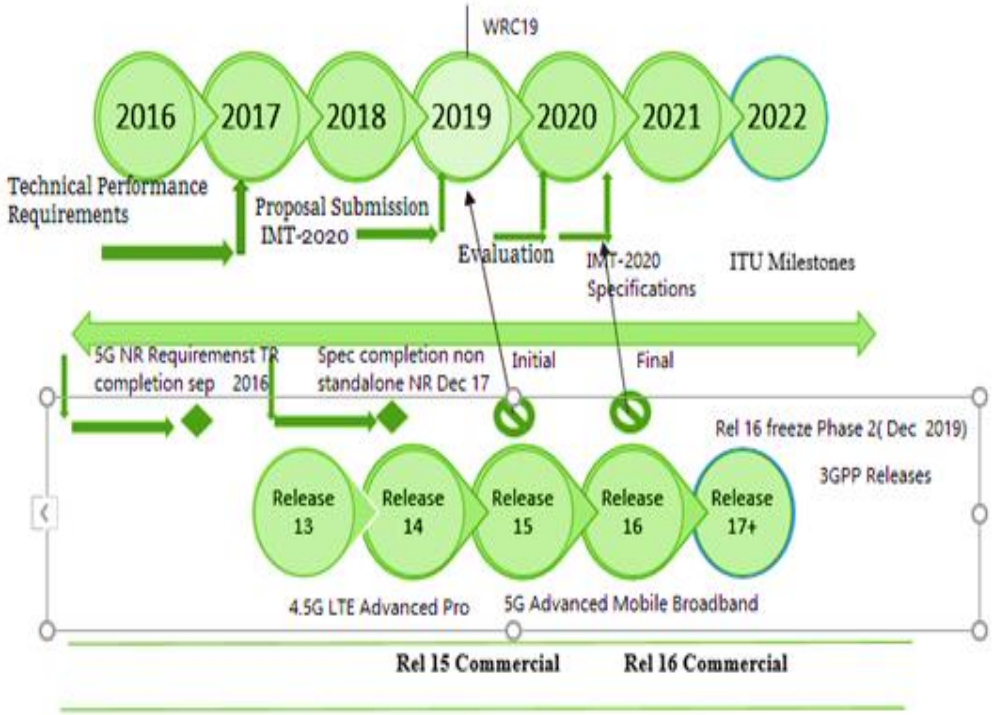


Fig.1.3 5G timeline concerning ITU-R and 3GPP [22]

1.5 5G Networks Requirements

5G is significant not just because it can manage millions of users at ultrafast speeds, but it can change the lives of people all around the world. These technology improvements can help make life easier. Beginning 4G to 5G, data rate requirements demands to be about 1000 times faster. Data traffic must be increased by a factor of ten thousand to meet the needs of everyone. At least 100 megabits per second, the edge rate or 5% rate can be used, up to one gigabit per second. The 5G network must handle a round trip latency of about 1 millisecond. During the 5G system, it is expected that there will be a 10 to 100-fold escalation in the number of things to be connected. The energy efficiency must be at least 100 times better. When it comes to capacity, reliability, access

technology, and cost, Table 1.3 lists 5G needs in terms of reliability, latency, cost, energy consumption, and access technology.

Table 1.3. 5G requirements of the industry

Performance parameters for 5G Requirements			
Capacity	**Reliability & Latency**	**Cost & Energy consumption**	**Access Technology**
Dense urban areas and indoor >10 Gbps	Air < 1-millisecond Bidirectional	Energy efficient 90 percent	Radio access technology
Sub urban areas >100 Mbps	End to end < 5 milliseconds	Long battery life of around 10 years for mobile to a mobile communication	IP based networks
Rural areas >10 Mbps	Distributed storage and capacity	Low-price data services	4G compatible
Per node 10 to 100 connections	99.99 percent	Hardware cost is very low	Co-operation in terms of inter-services and spectrum sharing
Easy Handovers	Tracking at high speed & high mobile operations	Low-cost network infrastructure.	Wireless networks are heterogeneously interoperable.

Table 1.4. Current global 5G standards

Brands	**NTT DOCOMO [95]**	**METIS**	**NOKIA**	**ACADEMIA**	**SAMSUNG**
Data Rate	100 x	100 x	10 x	1000 x	50 x
Traffic	-	>1000x	10000x	-	-
Latency	<1ms	<5x	<1ms	<1ms	<1ms
Cost	-	-	-	-	10x
Connectivity	100x	100x	100x	100x	10x
Energy	-	10x longer life	10 years M2M battery life	<100x	-

The academic community has several ambitious goals, whereas the manufacturing sector has more realistic goals. These sectors are on the verge of agreeing on 5G requirements. The achievement of high spectral efficiency is among the most important 5G requirements since it allows us to considerably reduce the cost per bit. Table 1.4 depicts the current global 5G standards. Channel coding and modulation are critical 5G physical layer components that must be present for 5G systems to achieve spectrum utilization [45]. To obtain higher spectrum utilization, multi-terminal encoding-decoding in data transmission, and to decrease the transmission power, channel estimation, and modulation for 5G networks must be improved.

1.6 5G Modulation Techniques

LTE Advanced is a 4G mobile system that is presently deployed internationally. It was primarily designed to provide high-bandwidth connectivity to devices that require strict alignments, such as tablets and smartphones. Apart from an increase in mobile data usage, the expansion and improvement of services like the Internet of Things, M2M, and V2V connectivity will place new traffic loads on networks, prompting the deployment of fifth-generation cellular networks to fulfill these new demands [17]. The basic mobile communication objectives are increasing day by day based on future needs such as data rates, additional capacity, increased mobility, reduced latency, and better quality [20]. Numerous modulation schemes, which include UFMC, FBMC, and GFDM, are suggested for future 5G mobile networks. To address the shortcomings of the 4G wireless network infrastructure, 5G techniques are introduced [81]. These include novel multi-carrier schemes. The goal of this article is to contrast the most promising candidates, as well as to serve as a primer on multicarrier waveforms in general. The study focuses on comprehending 5G modulation schemes such as UFMC, FBMC, and GFDM, as well as OFDM.

1.6.1 Multiple Carrier Techniques

The enhancement of spectral efficiency in mobile communication networks entails enhancing the amount of data transmitted while maintaining the high data transmission rate over a narrow frequency range, necessitating the

development of new data transmission techniques. Single-carrier modulation is a data transfer technique that transmits the entire data stream over a single carrier [18]. These techniques are recommended because they reduce battery consumption while increasing coverage. Complex equalizers are necessary to accomplish spectral efficiency in the single-frequency carrier technique. Complementary, to this multicarrier modulation divides a single wideband carrier into countless subcarriers that do not intersect or overlap with one another. Wireless communication systems must employ multicarrier modulation to enhance the amount of data transmitted [19]. Multicarrier techniques, like FDM, benefit from the fact that they're less vulnerable to channel disruptions than single-carrier techniques.

It introduces less interference between symbols while retaining the higher data transmission rate as a single carrier system. Fig.1.4 highlights the difference in spectra between both the non-overlapping and overlapping techniques. For many years, OFDM has been a broadly used multicarrier modulation scheme in 4G systems such as LTE or LTE advanced and Wi-Fi. Due to the rectangular shape of the pulses, it has some critical limitations, such as frequency outflow or a high out-of-band signal [34]. While using the CP minimizes spectral efficiency, the demand for synchronization and strict frequency to maintain the carrier orthogonally essentially guarantees that intercarrier interference is kept to a minimum. Due to these disadvantages, multiple possible candidate schemes, such as the UFMC, FBMC, and GFDM, have been thoroughly researched to overcome such drawbacks.

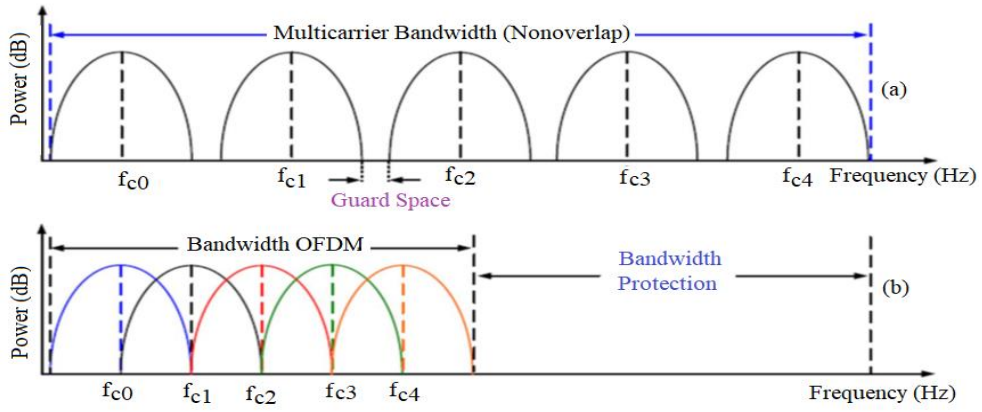


Fig.1.4 Multicarrier communication behavior **(a)** non-overlapping **(b)** overlapping OFDM

The fundamental designs for the next generation of systems, as well as two new multicarrier techniques, are discussed. FBMC method is one such example [66]. Another technique UFMC is examined in contrast to OFDM when synchronization errors are introduced [13]. In comparison to OFDM, which splits a carrier into sub-bands, UFMC splits a carrier into sub-bands, with each subband further partitioned into subcarriers, resulting in a smaller filter length due to the subband grouping methodology. In OFDM, the entire carrier is filtered, whereas, in UFMC, each sub-band is filtered individually, resulting in reduced side lobe levels from the outer sub-band. When synchronization errors occur, the UFMC technique outperforms the OFDM scheme [90].

1.6.1.1 OFDM

Multipath propagation is a wireless communication system effect caused by scattering, reflection, and diffraction that has a massive effect on radio channels or in the frequency selective fading channels [99]. As a consequence, the received signal not just includes a direct LOS, but also a huge proportion of diffracted and reflected signals with varying time delays. This results in Inter symbol Interference (ISI). ISI induced by the frequency selective fading channels can be lowered by using the multicarrier technique, a parallel data transmission technique. OFDM is a subcategory of FDM where no guard bands exist between adjacent sub-carriers, unlike FDM.

OFDM is a very popular multiple carrier modulation scheme in 4G systems and the most popular technique in 3G wireless communication. OFDM is a type of modulation that builds on the concept of single subcarrier modulation by engaging various sub-carriers inside a single channel. OFDM uses a huge number of closely spread orthogonal sub-carriers spaced parallel, instead of a single subcarrier broadcasting a higher stream of data. A traditional digital modulation scheme like 16 QAM, QPSK, or other similar schemes is used for modulating each subcarrier at a lower symbol rate, yielding a total of 256 subcarriers. OFDM systems utilize a cyclic prefix before every symbol to avoid ISI induced by the propagation channel.

The serial-to-parallel converter afterward tends to take a block of symbols based on the frequency of the signal and blends each symbol with one of the sub-carriers by changing the phase as well as amplitude of the signal. Following the completion of the addition stage, data representing the in-phase and quadrature elements as a function of time are obtained. The additional steps apply for processing from a frequency function to a duration function. To accomplish this, IFFT is used, with the time-domain signal obtained from the transform's output [22].

Before being broadcast, this can be converted to digital format, filtered, and mixed with radio frequencies. In a multi-path environment, the receiver receives multiple copies of the bit stream, each with a separate time of arrival. When these two sine waves are combined, they generate a sine wave with the same frequency but a varying phase and amplitude than the original sine wave. To decrease symbol interference, the final section of each symbol is attached to the front of the symbol, a technique known as cyclic prefixing. The schematic diagram of an OFDM transceiver is depicted using Fig.1.5.

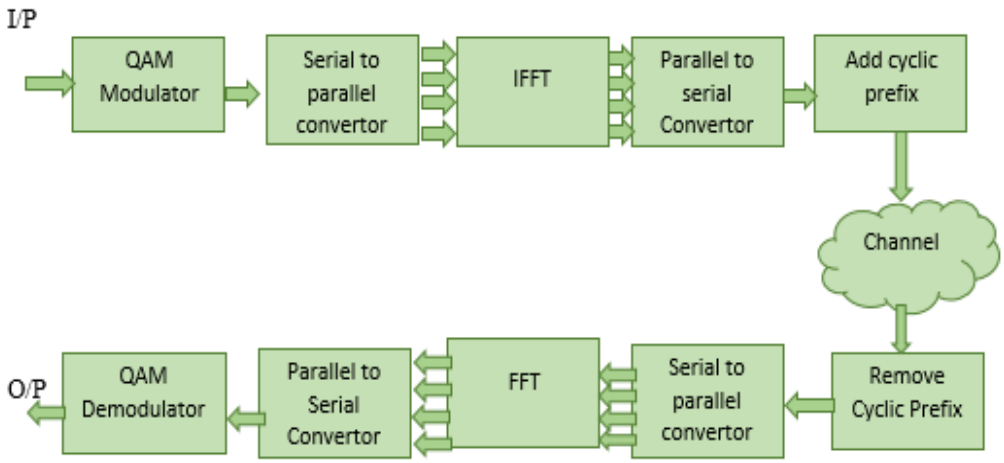


Fig.1.5 OFDM transceiver system

The S/P convertor block in the OFDM transmitter is used to transform a serial sequence of bits into parallel data blocks, which is the first step in the transmission process. Using an IFFT technique, parallel QAM symbols are transformed into subcarriers, and the time domain base signal (X_n) is achieved

from the frequency domain signal (X_k). The following equ.1.1 can be used to calculate the OFDM base signal with N_{sc} subcarriers,

$$x_n = \frac{1}{N_{sc}} \sum_{k=0}^{N_{sc}-1} X_k e^{j2\pi nk/N_{sc}} \quad k = 0, 1, \ldots . N-1 \qquad (1.1)$$

(Xn) denotes the number of OFDM symbols transmitted. (Xk) represents the frequency waveform for the k^{th} subcarrier. The P/S conversion block converts the parallel stream to a serial stream after the IFFT operation and then incorporates a CP block, which is a duplicate of the tail of the symbol that was positioned in the early part to minimize the influence of inter-symbol interference. An AWGN channel is a channel type that is frequently used to estimate the background noise in a channel. To improve performance, the receiver employs a reversed structure when compared to the OFDM transmitter. The OFDM receiver first samples the incoming signal before it is filtered and converted to a baseband. Following that, the data is subjected to the FFT to improve the phase and amplitude of each subcarrier. When it comes to meeting the earlier-mentioned necessities for the future 5G cellular network, OFDM, on the other hand, has several shortcomings [40]. One of the drawbacks of OFDM is that its rectangular pulse shape results in a high PAPR and frequency outflow. It is also extremely complex to frequency and time offsets, necessitating synchronization between users to remove interference. While using the cyclic prefix, spectral efficiency loss is observed. To remove these limits, several alternatives, including UFMC, GFDM, and FBMC, have been explored in recent years.

1.6.1.2 SC-FDMA

It is one of the category of FDMA schemes. Just like a linearly pre-coded OFDMA technique, SC-FDMA also adds the DFT procedure before the conventional OFDMA process [27]. It has received considerable attention as a practical alternative to OFDMA, mainly in uplink communication systems, where a low value of PAPR provides substantial benefits to the mobile terminals in terms of communication efficiency, energy, and amplification cost. This technique has also gained prominence as a feasible downlink connectivity

alternative to OFDMA. The high PAPR value of the signal, in particular, is a significant source of concern because these large peaks cause significant degradation in the performance when a signal moves across a non-linear HPA. In comparison to OFDM, SC-FDMA has a low PAPR value, due to which it is best suited for uplink transmission. Fig.1.6 presents the block diagram of the OFDM and SC-FDMA transceiver system. The use of DFT and subcarrier mapping in the transmitter, as well as subcarrier de-mapping in the receiver, is the core difference between SC-FDMA and OFDM.

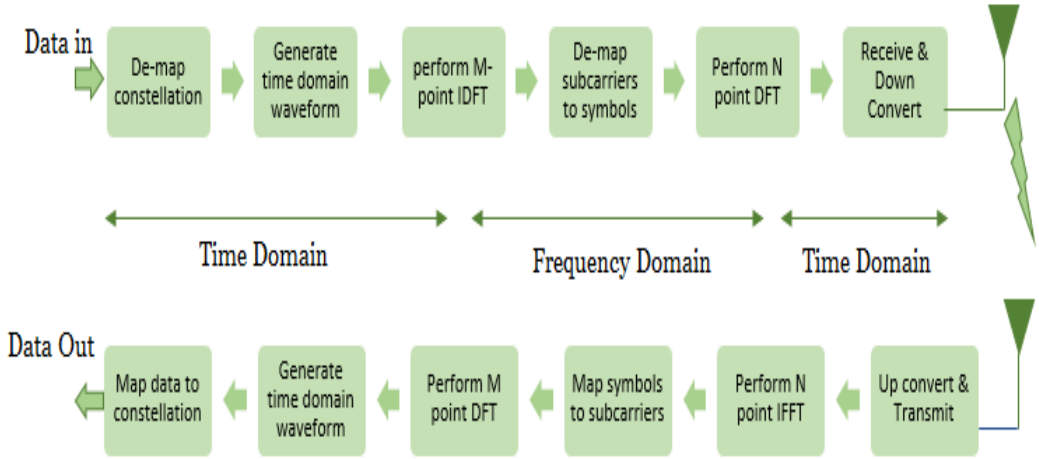


Fig.1.6 Block diagram of SC- FDMA and OFDM

The SC-FDMA system includes an extra DFT module before the IFFT module in the transmitter chain, as well as an IDFT module in the receiver chain. If compared to the OFDM system, the PAPR of the SC-FDMA is usually lower. These systems are less vulnerable to frequency offset. It is widely used in the transceiver path of LTE subscriber terminals, while an OFDMA variant is used in downlink (or receive path of LTE subscriber).

1.6.2 5G Modulations Techniques Based on Pulse-Shaping Methods

Subcarrier-based filtering, also known as pulse shaping, is a process for minimizing out-of-band leakage. In general, it is acknowledged that the frequency and time widths of pulses cannot be minimized concurrently. As a result, pulse-shaping waveforms are currently inactive in both the frequency and time domain of the signal to maintain a high SE. In comparison to the

conventional OFDM, the pulse-shaped modulation transceiver structure is significantly more complex. The sections that follow go over the fundamentals of two common pulse shaping modulations FBMC and GFDM.

1.6.2.1 FBMC

FBMC is a category of MCM that is used to boost the bandwidth consumption of cognitive radio network systems and frequency spectrum access techniques [105]. The FBMC technique is frequently used in optical fiber communication, wireless connectivity, CR systems, multiple access networks, and other applications. FBMC is a variant of OFDM with some of its basic structure enhanced. OFDM has a very low spectral efficiency and a very low data rate. Furthermore, the inclusion of the cyclic prefix results in high power consumption and low performance.

The use of a cyclic prefix is not required in FBMC. As a result, when compared to the OFDM technique, FBMC has high performance while consuming very little power. The FBMC is primarily used to simultaneously transmit multiple data packets over a variety of carrier frequencies. As a result, OFDM has a high value of signaling overhead and a low spectral efficiency. To avoid this problem, the FBMC employs a non-orthogonal filter bank technique to maximize spectral efficiency while minimizing signal overhead. When comparing FBMC to OFDM, the system complexity is significantly higher. To address this issue, a significant structure is introduced to minimize area usage and implementation time. FBMC employs a multiplier-less FIR filter with a shift and adds the multiplication method at the filtering end. As a result, the most recent FBMC structure is less complex. To minimize memory size and propagation delay, a new inverse FFT has been introduced [28]. Furthermore, advanced modulation schemes are implemented into the FBMC methodology to overcome the trade-off between power and speed underlying the FBMC design. Fig.1.7 depicts the FBMC in its current configuration, which involves parallel to serial converter, QAM modulation, polyphase filter, and Radix 2 inverse FFT. QAM modulation is used to enhance the carrier signal that transfers the signal during data transmission.

Fig.1.7 FBMC transmitter block diagram

During frequency-to-time domain conversion, the Radix 2 IFFT with 128 points is used. This 128-bit data is reduced to 8-bit data by employing a parallel-to-serial converter. Last but not least, the polyphase filter is used to eliminate the FBMC transmitter's side lobes. As a result, a clear carrier output is predicted. The digital QAM modulation technique, as shown in Fig.1.8, is composed of several components, such as DGP, QAM-TC, pulse shaping FIR filter, and symbol mapping. The entire data set is divided into an arbitrary number of packets in the DGP, with the bits of each packet grouped for symbol mapping. DGP generates the preamble bit and data bit using both the controller and the data source. The scrambling operation is used to construct packets when the data source is scrambled. The input data bit-stream is transformed into a 6-bit integer by grouping bits at a sampling rate of 1/6 of the sampling rate required by the symbol mapper. Symbol mapping means the process of converting DGP bits into QAM symbols. Up-sampling and pulse shaping of symbols is accomplished using an interpolating root-raised cosine (RRC) filter for data transmission.

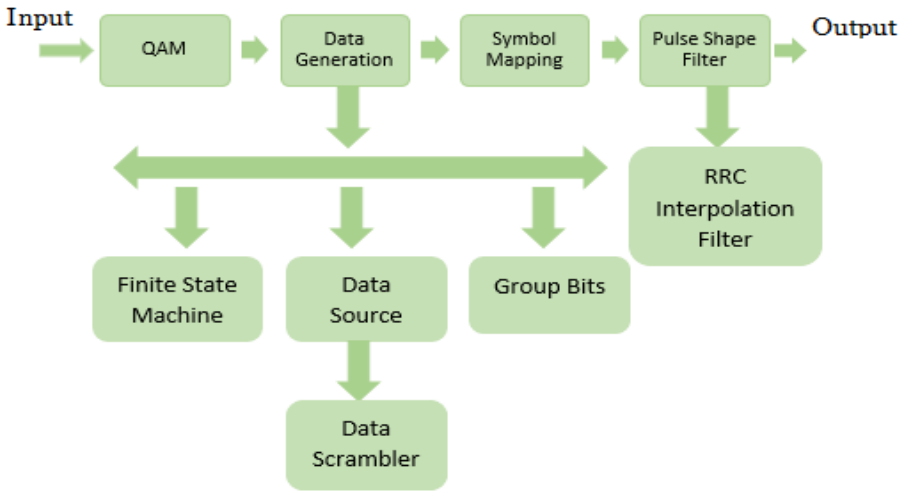


Fig.1.8 Digital QAM modulation system

1.6.2.1 GFDM

Fig. 1.9 depicts the GFDM block diagram. OFDM and SC-FDMA are two subsets of GFDM. S(n) denotes the output of the IDFT performed to the precoded data sequence, h(n) represents the transfer function of the channel, whereas r(n) is the received signal which is subjected to noise across the channel. GFDM differs from FBMC in that it uses circular shifted filters to perform pulse shaping rather than linear filters. Even if orthogonality is completely dropped, OOB outflow can be minimized by wisely selecting the circular filter. The number of time samples (K) and frequency samples (N) are changed for GFDM blocks depending on the type of application. The transmitted signal of a GFDM system can be represented as

$$f(m) = \sum_{n=0}^{N-1} \sum_{k=0}^{K-1} d^{(k,n)}, g(m - nk)\ e^{j2\pi mk/K} \qquad (1.2)$$

$d^{(k,n)}$ is the transmit signal for $0 \leq m \leq K_{N-1}$, at sub symbol 'n'.

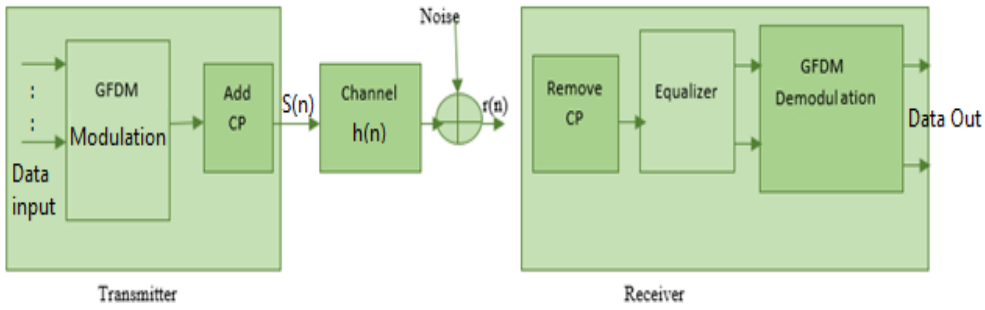


Fig.1.9 GFDM trans-receiver system

Matrix operations can be used to express modulation and demodulation processes in the same way that traditional OFDM does. The traditional OFDM IDFT and DFT matrices are overtaken in GFDM by some specific matrices that correlate to the GFDM signal modulation technique, respectively. The GFDM transceiver structure, on the other hand, differs significantly from the traditional OFDM transceiver structure. Other pulse-shaped modulations, like pulse-shaped OFDM and QAM FBMC, are proposed as alternatives to the two previously mentioned for 5G networks.

In general, pulse shaping modulations attempt to confine and communicate signals to a narrow bandwidth and, as a result, lessen OOB leakage, allowing them to function to some extent in asynchronous circumstances using a narrow guard band. Likewise, FBMC achieves real-domain orthogonality by the use of OQAM, which reduces the need for guard interval and interference cancellation, saving the cost. Due to their low propagation delay, GFDM's circular shifted filters eliminate the time domain's long tail of linear filters. Another advantage of GFDM is that it is easily compatible with technologies that employ MIMO [31]. Sub-band filtering-based modulations are discussed. Sub-band filtering is another method for reducing OOB leakage.

1.6.3 Sub-band Filtering-Based on 5G Modulation Technique

1.6.3.1 Universal Filtered Multicarrier (UFMC)

UFMC is a multiple carrier modulation method that has a combination of the best characteristics of FBMC and OFDM. The UFMC waveform filters a group of carriers by utilizing a frequency domain and is used in conjunction with a post-filtering technique [113]. The overall bandwidth in UFMC is first distributed into sub bands, which are then further divided. Each subband is made up of several subcarriers that are used to transmit data. Unlike FBMC, UFMC utilizes a cluster of subcarrier modulation to minimize OOB. As compared to FBMC, the sub-carrier grouping lessens the complexity of the filter and the time required to perform modulation operations. The QAM modulator is used in UFMC. It has improved time and frequency synchronization, as well as spectral efficiency, because of the non-existence of CP and the reduction in OOB discharge caused by the filtering operation.

Fig.1.10 depicts the functional diagram of the UFMC transceiver. UFMC is a novel technique that is being considered for use in the 5G communication network. According to the block diagram, the high-bits-per-second input data is transformed into multiple bits with lower data rates. Subbands can be created from the entire group of subcarriers (N) in which each subband consists of fixed subcarriers.

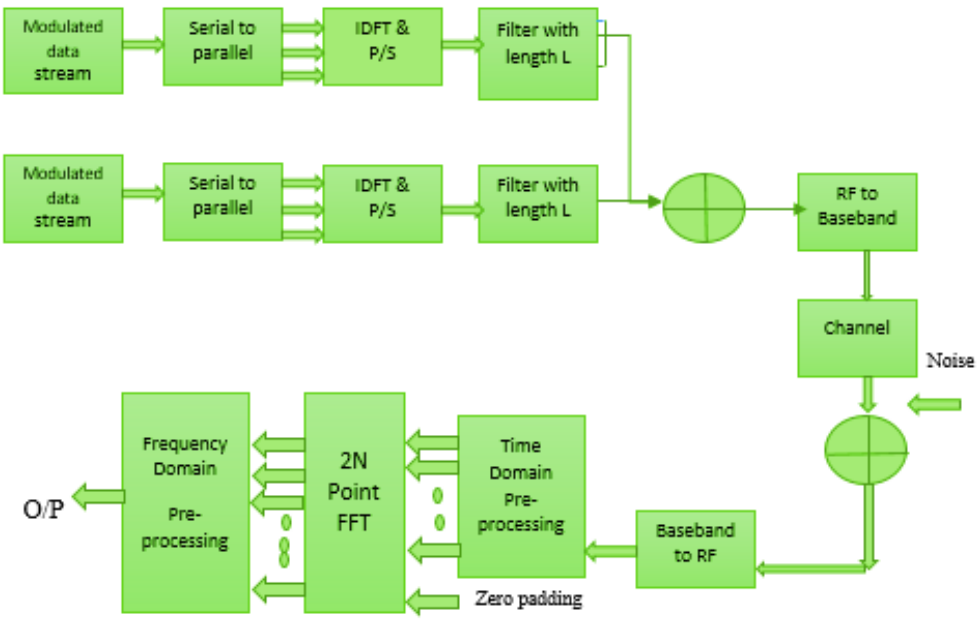


Fig.1.10 Block diagram of UFMC transceiver system

Filtering is used to decrease the spectral emissions which occur outside of the band. The output of IFFT is filtered with the help of a Chebyshev window utilizing a parameterized side lobe attenuation for each sub band. Demodulation is accomplished by performing an FFT on the time signal from the modulation side that has been pre-processed for filtering the S/P conversion. Whereas, the FFT converts time domain received data into the frequency domain data. The equalizer can be used to achieve an equalization of the combined effects of sub-band filtering and channel. The original data can be recovered by using the symbol de-mapper, which converts symbols to bits. Because the cyclic prefix in UFMC technology can be prohibited, it has higher spectral consumption than OFDM technology.

Other technologies are more sensitive to errors in frequency and time-shift estimation than UFMC technology. The block-wise filtering technique makes the system more adaptable and can be used to remove the main disadvantages of OFDM. The filter's weighting window length is a critical factor that controls the features of communication systems based on UFMC. With fewer side lobes, the amount of involvement on adjacent subcarriers decreases. The UFMC makes use of a length ‘L’ Dolph-Chebyshev filter. The filter length is affected

by the size of the sub-band, which can be calculated as the number of carriers observed in the sub-band.

1.6.3.2 Filtered-OFDM

The structure of F-OFDM is similar to UFMC, but the receiver structure is different [117]. The key distinction is that F-OFDM currently uses a CP and it enables retained ISI. This section discusses a system model of a single-band - OFDM system and the impacts of its visualization. Fig.1.11 demonstrates the information flow ‘a through a QAM mapper to generate symbols from a two-valued complex constellation. Here ‘x’ denotes the encoded data sequence generated after adding cyclic redundancy to the symbols, ‘s’ denotes the output data of the filter, ‘r’ denotes the received sequence from the channel which is subjected to matched filter as an input, and ‘y’ denotes the N point generated FFT sequence

As a result, rather than the decimation technique, a Match Filter is used at the receiver. Additionally, the down-sampling process can be performed before DFT operation, which reduces the complexity as CP minimizes the majority of distortion produced by the tail of the filter. This allows for a longer filter length than is possible with UFMC, as well as an improved attenuation effect. The degradation in performance by the residual interference is mitigated with the help of effective channel coding. Furthermore, unlike UFMC, the subcarrier spacing and channel length in f-OFDM transmissions do not have to be the same for different users.

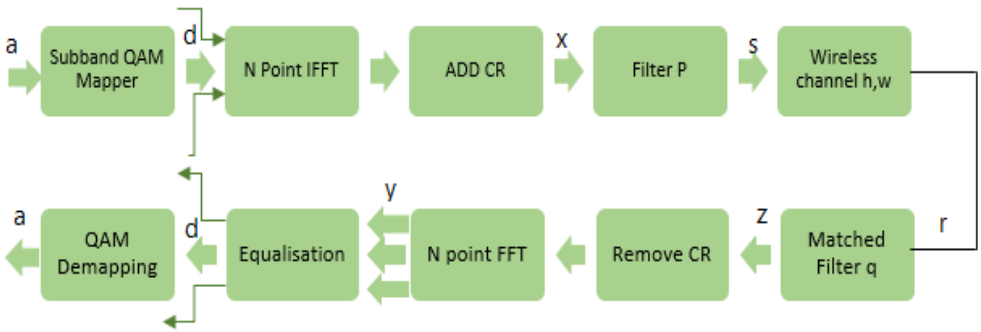


Fig.1.11 Block diagram of the F-OFDM transceiver [63]

Because it is easy to implement and is used in a variety of applications with a variety of parameters, the soft truncated sinc-filter is mostly used in f-OFDM. As a result, when it comes to frequency multiplexing, f-OFDM is extremely versatile. Other sub-band filtering-based modulations have been proposed in addition to UFMC and f-OFDM. Sub-band filtering modulation schemes can significantly decrease OOB leakage and accomplish improved performance when compared to conventional OFDM modulation. RB-f-OFDM uses filters that are based on the resource block rather than just the entire band of users as in conventional f-OFDM.

1.6.4 Advanced Modulation Techniques

1.6.4.1 SP-OFDM

Fig.1.12 depicts an SP-OFDM diagram. It is made up of IDFT and DFT, a spectral pre-coder, and an iterative detector, as shown in the diagram. S(n) denotes the output of the IDFT performed to the precoded data sequence, h(n) signifies the transfer function of the channel, and r(n) is the received signal which is subjected to noise across the channel. When compared to traditional OFDM, a rank-deficient matrix is typically used to pre-code the data symbols mapped onto the subcarriers to estimate the signal into a designated low-dimensional subspace, allowing the pre-coded signal to be in high order. Even though precoding OFDM signals with a rank-deficient matrix reduces channel capacity, the OOB outflow of OFDM can be considerably decreased at the expense of small dimensions.

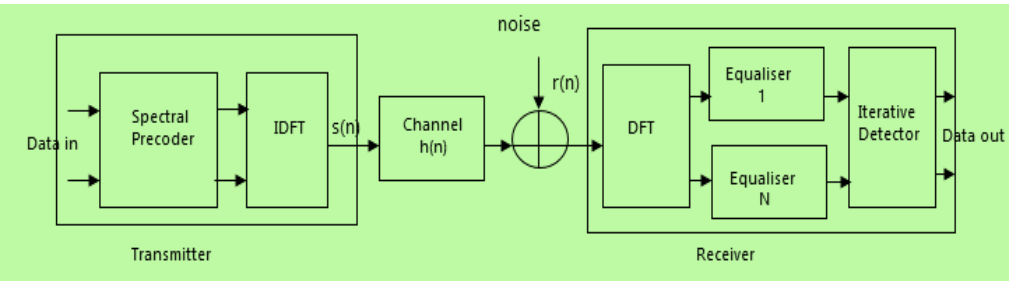


Fig.1.12 Block diagram of SP-OFDM

When comparing SP-OFDM to filter-based modulations, the following three benefits are apparent:

The ISI induced by the filter tail can be eliminated without using the filtering techniques. As a result, the time spent countering multipath on wireless channels is lowered, and the SE of the wireless channels is upgraded. SP-OFDM can seamlessly notch well-opted frequencies in fragmented bands without using multiple narrow sub-band filters. Merging, filtering, and precoding methods can additionally improve performance.

In these two implementations, the iterative detector and spectral precoder have been modified using an associated inverse transform module and 2D symplectic transform [33]. To determine the corresponding data in the time-frequency domain, a 2D symplectic Transform can be used. The calculated data can afterward be sent to its destination using a time-frequency-domain modulation. Subsequently, a 2D symplectic Fourier transform is comparatively inaccessible from the time-frequency domain technique, it can be combined with pulse shaping and subband filtering to further minimize leakage in an optical transmission system.

When a mobile device travels quickly, the channel experiences rapid fading time. As a result, channel parameters must be estimated and tracked regularly, resulting in a significant increase in resource costs. Furthermore, most of modulations are designed with the assumption that channels within a symbol block are constant. When traveling at high speeds, additional interference is introduced, degrading the system's performance.

A variation of this technique known as OTFS can be used in MIMO systems to approximate the CSI of different antennas [41]. When equated to the overall system scale, the doppler and delay dispersions are quite insignificant. The channel can be demonstrated compactly and stably in the delay doppler domain. As a result of which, the channel causes the pilots to spread locally. As an outcome of the pilots' channel spreading, different pilots can evaluate the CSI in MIMO systems in a restricted area of the delay doppler plane. Several other modulation techniques are proposed, like W-OFDM, which employs windowing to handle the disruption between the adjacent OFDM symbols [42].

1.7 Advanced Multiple-Access Techniques

This section focuses on multiple access techniques like OMA and NOMA techniques [69]. To be more specific, various types of modulation schemes that could be used for OMA and their comparison in terms of BER, OOB leakage, and spectral efficiency are analyzed. Following that, a variety of NOMA candidates is focused on such as code-domain NOMA, power-domain NOMA, and multiplexing of NOMA in multiple domains [53]. The opportunities and challenges in the field of multiple access and modulation for 5G networks using this investigation can be identified. 5G wireless networks have piqued the interest of many academics and industry professionals in recent years [35]. The 3GPP believes that 5G networks will support three main applications like eMBB, mMTC, and URLLC [107]. Furthermore, enhanced vehicle-to-communications is one of the critical facilities which must be supported by 5G networks. As a result, these scenarios necessitate challenges such as improved spectral efficiency, massive connectivity, and high system throughput while designing 5G network architectures [36]. Novel multiple access and modulation schemes are currently being investigated to meet these new requirements.

OFDM has proven to be effective in 4G and 5G networks. OFDM is capable of combating wireless channel delay spread using simple detection methods when using an appropriate CP, therefore it has become a standard choice for the present broadband transmission applications. On the other hand, traditional OFDM is incapable of fulfilling the new 5G network requirements. In mMTC, sensor nodes typically transmit different data asynchronously, whereas OFDM requires users synchronization to avoid significant interference between adjacent sub-bands.

To address some new demands in 5G networks, numerous modulation schemes are suggested, including pulse shaping, precoding, and filtering all of which are intended to reduce OOB leakage in OFDM signals. One of the direct methods for reducing OOB leakage is filtering. Also, by using a correctly designed filter, stop-band leakage can be suppressed to a large limit. Pulse shaping is subcarrier-based filtering that decreases the subcarrier overlaps. Nevertheless, it typically has a long tail in the time domain because of the Heisenberg-Gabor

uncertainty principle. To further reduce leakage, data can be transmitted using precoding before OFDM modulation is applied to the data stream. In addition to the approaches described above, some new modulation techniques are specifically designed for 5G wireless networks.

Novel modulation methodologies for lowering multi-user interference in OMA networks, such as NOMA, can be developed for 5G networks. While some reviews cover both OMA and NOMA, this analysis gives a more comprehensive explanation of different NOMA and OMA schemes [39]. Fig.1.13 depicts the classification of OMA and NOMA additionally; the performance and characteristics of various approaches and methodologies are shown in this section. NOMA enables the full utilization of limited spectrum systems to facilitate more users, resulting in the capacity rise of 5G networks, despite the addition of complexity and interference at the receiver.

Different modulation methods can be used in combination with OMA in 5G networks. TDMA and FDMA were used in 2G systems, CDMA in 3G systems, and OFDMA in 4G systems. OMA is the fundamental basis for all present and previous wireless networks. These systems employ resource blocks divided orthogonally into code, frequency, and time domains contributing to increased interference among both the adjacent blocks and relatively simple detection of signals [70].

Due to the smaller resource blocks, OMA promotes an inadequate number of users, thereby limiting the capacity and spectral efficiency of the existing network infrastructure. Numerous NOMA schemes are suggested for 5G networks to sustain a maximum number of users and implementations belonging to a wide variety of classes. By multiplexing within time, frequency, or code domains, the NOMA method, adds a new dimension to the communication system. In other words, it can be viewed as an "add-on" that has the prospect of seamlessly integrating with established MA methods [62]. NOMA is based on the multiplexing of the code domain or power domain for preserving more users in the same block.

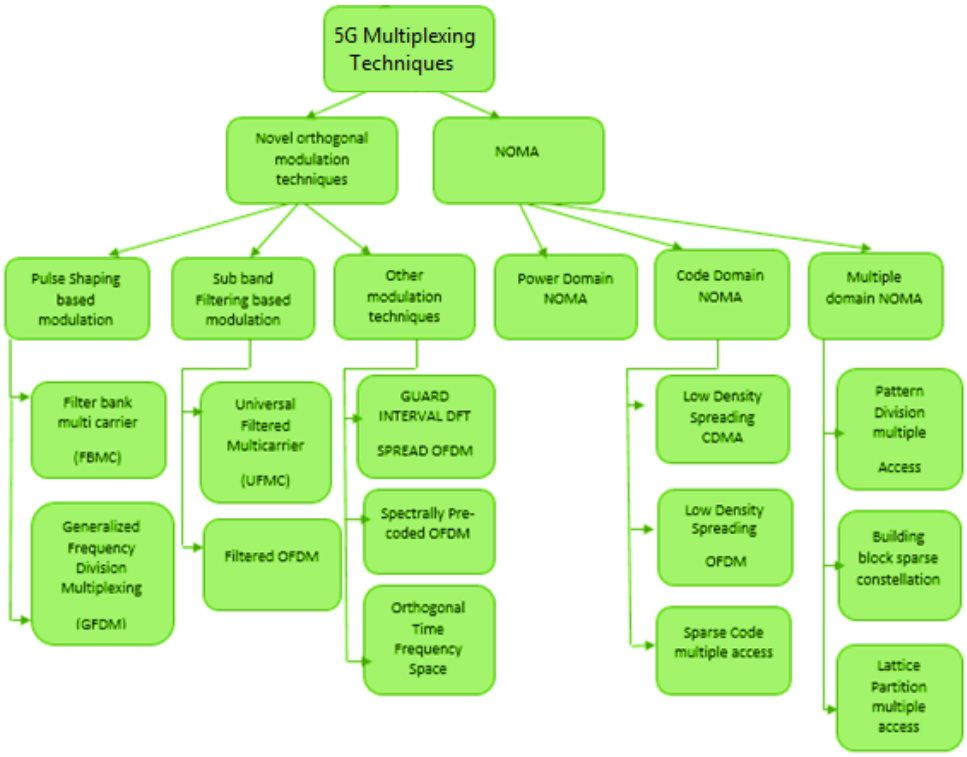


Fig.1.13 Novel multiplexing techniques in 5G networks

Incorporating these techniques with OMA enables effective management of OOB leakage in 5G networks. However, several issues remain unresolved in this field. IoT is a possible application for f-OFDM. Because the subbands in this scenario are narrow, the noise produced by a short CP can reduce the performance of the system significantly and should be considered during the detection process. Additional processing techniques, like SIC, are used to improve detection performance. RISIC may be advantageous. Current subband filtering designs, like soft truncated filtering in f-OFDM and Dolph-Chebyshev filtering in UFMC, have fixed lengths and are not scalable [63]. Leakage levels, filter lengths, and other parameters will be different depending on the user and the application scenario. Frequency and time dispersions are two independent variables that cannot be minimized simultaneously which raises questions about whether or not to design an effective prototype filter to balance both parameters. Multi-carrier-based modulation contenders, like FBMC and UFMC, have a high value of PAPR, just like traditional OFDM modulation candidates. This should be done to increase the power amplifier's efficiency while decreasing the PAPR. Traditional PAPR reduction methods are known to introduce distortions by

degrading system performance. The question arises of extending the PAPR reduction methods which is intriguing and needs to be investigated further.

1.7.1 NOMA

NOMA was proposed by NTT DoCoMo as a multiple access control mechanism for the 5G mobile systems [111]. NOMA is a non-orthogonal multiplexing in which multiplexing of users occurs in the power domain. Using SIC, signals from multiple users are mapped at the transmitter end after which they are separated at the receiver end. The NOMA presented by NTT DoCoMo is commonly stated as PD-NOMA [82]. Over the past several decades, the multiple access scheme is recognized as a significant technology in differentiating every generation of wireless communication systems. OMA is extensively used in wireless systems. Since the very 1G mobile communications network has persisted to be used in 4G systems. As part of these orthogonal multiplexing schemes, users are orthogonally multiplexed all over, time, frequency, or code domains, enabling mixed signals to be split at the receiver while maintaining low complexity. It can efficiently reduce IUI while requiring a low complexity. However, its spectral efficiency must be enhanced further to fulfill the standards of the 5G wireless mobile network, for which INTT DoCoMo proposed a NOMA scheme for downlink [112].

The inclusion of NOMA as a proposal in 3GPP LTE Release 13 marks the beginning of its standardization (3GPP 2014) [115]. Based on the study of multiuser superposition a primary downlink variety of NOMA is mandated and made operational (3GPP 2015a). Although NOMA was originally suggested as a downlink method, research has demonstrated that it can also be used in the uplink, and normalization of the uplink NOMA is presently being considered (3GPP, NTT-DOCOMO 2016). The signal from the user having maximum transmitted power is to be decoded first, while the signals from other users are considered noise. After sensing and decoding the signal with the maximum transmitted power, the signal component associated with that user is deducted from the received signal to make additional users more detectable.

It is to be observed that the first detected node is generally subjected to the highest amount of inter-user interference, and thus first detected user's detection error is transferred to the other users. As a result, sufficient power should be provided to the initial user. The enhanced version of NOMA from two to multiple users is very simple and clear, and the following subsection discusses the related work on NOMA power allocation in detail [83]. Fig.1.14 highlights the difference between NOMA and OMA when two users use the same base station. The diagram illustrates that the NOMA system has a small probability of failure. However, the use of NOMA necessarily requires the use of a more complicated transmitter and receiver to minimize interfering signals. A power-domain NOMA is usually effective when a single resource block is shared by a small number of users. Multiple access interference becomes more extreme once the number of customers multiplexed in the power domain rises, resulting in the NOMA's performance deterioration [91]. The following are the benefits of NOMA [123]

(i) Increased Spectral Efficiency- When compared to other technologies, NOMA can provide high spectral efficiency. This benefit of NOMA over OMA can be clarified both mathematically and in terms of information theory. As proven by NOMA the best capacity of the downlink channel on the transmitter side can be attained by SC in the transmitter end and SIC in the receiver end. OMA, on the other hand, is unable to do so.

(ii) Second, unlike OMA, the enabled NOMA users are not purely limited to the accuracy of scheduling and availability of resources. As a result, NOMA can also handle the issues of massive connectivity.

(iii) In terms of multiplexing, the NOMA transmitter relies less on real-time CSI, requiring feedback signaling from the UE. As a result, regardless of UE mobility or signaling latency in a given scenario, it is reasonable to expect a robust performance gain. The third member of the OMA family is NOMA [103]. To encourage heterogeneous connectivity and higher throughput, NOMA is used for effective interference mitigation [84].

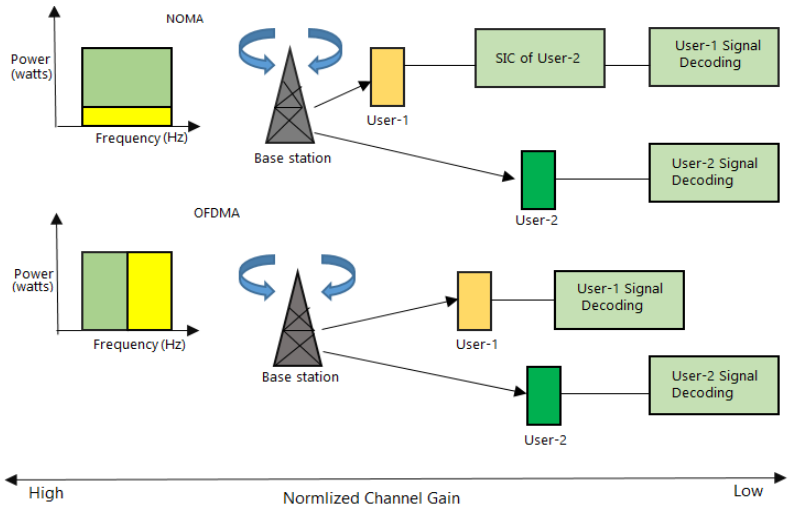


Fig.1.14 Basic downlink NOMA and OMA (OFDMA) Structure [121]

The following are some of NOMA's most important characteristics [130]:

(i) NOMA offers high SE as it allows users to utilize every resource block, resulting in a higher SE for the system.

(ii) Second, because NOMA allows sharing of a single resource block among multiple users, it can boost seamless connectivity for billion devices. This characteristic is critical in the Internet of Things instances where users require very low data rates and there are many users.

(iii) In NOMA, BS does not need an ideal uplink CSI and it is referred to as inadequate channel feedback.

Other than the received signal strength, there is no need to include anything else in the channel feedback. The NOMA uplink eradicates the need for users to send scheduling requests to the BS, which is typically needed in OMA. As an outcome, transmission latency is reduced significantly.

1.7.1.1 NOMA in the Power-Domain

NOMA with power domain is among the potential MA schemes with scalability for 5G networks. The downlink DFT of NOMA is 1024 bytes in size, each user consumes 36 sub-carriers, and the guard band consumes 12 sub-carriers for the ITU Vehicular. In the 3GPP-LTE-A network, a variant of NOMA dubbed MUST is presented [131]. It has been illustrated that implementing NOMA increases system capacity while also enhancing user experience. Recently,

3GPP LTE Release 14 approved the new work item summarizing downlink multiple-user superposition transmission for LTE, to recognize the techniques required to enable downlink intra-cell multiple-user superposition communication [85]. The NOMA can be understood with the help of basic principles including cooperative NOMA, NOMA's power allocation, and NOMA based on multiple antennae. Unlike, user detection in MIMO or CDMA, which typically requires multiple receiver observation, power domain NOMA systems require only one observation [49]. The resulting signal at the base station, most notably during NOMA uplink transmission, is expressed as

$$Z = \sum_{a=1}^{A} h_a \sqrt{Pa\ xa} + n \qquad (1.3)$$

where 'P_a' and 'x_a' denote the a^{th} user transmission power and transmission symbols, respectively, 'n' represents an AWGN with variance 2, and 'a' signifies the number of users sharing a^{th} user's resource block. To allow SIC at the receiver, it is significant to modify the transmission power 'P_a' for every user to make sure that users with large transmission power can be accurately detected. When the BS receiver calls up SIC, the user having the highest CSI is decrypted, followed by the user with the second-highest CSI. The process will be repeated until the received signal is depleted of the corresponding signal component. The SIC receiver, as illustrated in the diagram, continues to operate in decreasing order of signal strength.

The transmission power values for different NOMA users are generally different due to the differences in channel conditions. The downlink communication of NOMA for two users is demonstrated using Fig.1.15. In this figure users utilizing the same resource block are differentiated by their power levels and the transmission is constrained by a total power constraint. Users are denoted by UE1 and UE2. To be precise, the BS transmits a super-imposed signal that includes the two users' signals. Unlike old power allocation schemes like water filling, it assigns minimum power to the users having higher downlink CSI to make sure fairness and diversity in all domains.

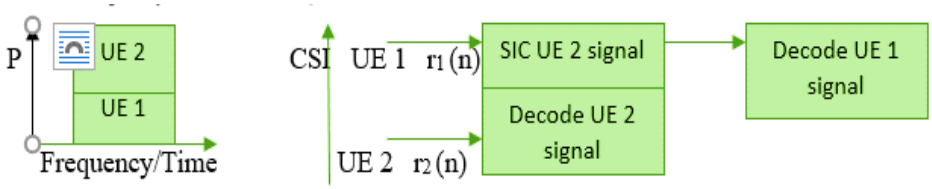


Fig.1.15 Downlink power-domain NOMA [132]

Additionally, [74] illustrates that MIMO-NOMA outshines MIMO-OMA in terms of ergodic aggregate capacity and aggregate channel capacity. Additionally, [101] suggested two coordinated beamforming methods for addressing ICI between two cells-based MIMO-NOMA networks, where two base stations significantly improve their beamforming vectors in concert to increase cell-edge user data rates. Table 1.5 depicts the power allocation in NOMA. This methodology is extended to networks with any number of cells and used to characterize the number of users aided by the proposed system in multi-cell MIMO networks [124].

Table 1.5 Allocation of power in NOMA

Allocation Strategy w.r.t power	**Complexity**	**Equality**	**Reference**
Weighted priority to users	Low	Low	(124)
Priority-based on Equality	High	High	(103), (125), (62)
A trade-off between Equality and throughput	-	-	(80)-(98)
To increase the user rate	Low	Medium	(39)-(90)
Power allocation to weak users	Low	Medium	(122)

Table 1.6 summarizes the current scenario in the field of multiple antenna-based NOMA, where the abbreviations ‘BF’ denote beamforming, ‘OP’ denotes outage probability, ‘TU’ denoted two-user, and ‘MU’ denotes multi-user cases, respectively.

Table 1.6. Multiple Antenna Based on NOMA

Framework	Method	Metrics	Features	Ref.
SU MIMO	Deposition using QR	OP	Channel difference	[127]
SU-MIMO	Based on Beamforming	Ergodic capacity	Optimal power allocation	[101]
MU-MIMO	Alignment of signals	OP	Higher diversity Gain	[130]
MU-MIMO	Precoding in two stages	Sum rate + OP	Feedback of 1 bit	[128]
MU-MIMO	Random BF	Sum rate	Perfect CSI is not required	[72]
MU-MIMO	ZF-BF	Rate gain + OP	Unavailability of CSI at the transmitter end.	[131]
MU-MIMO	ZF-BF	Min-max Rate	MIMO-NOMA fairness	[125]

1.7.1.2 NOMA in a Cooperative Setting

When contrasted to users closer to the base station, cell edge users typically experience low data rates and low received signal power. To increase cell-edge transmission rates, transmission (and reception) strategies such as relaying and CoMP communication have been widely used [86]. The basic concept of relay-assisted NOMA uses users with greater CSI as decoders and forwarders (DF) or amplifiers as relays to increase the user transmission rates. [88] proposes a co-operative NOMA model where the 'M' user uses 'M' time slots in exchange for cooperating. The conventional non-co-operative NOMA system must be implemented during the first time slot. The user having the highest chance of success serves as the relay for the next user with the second chance of success during the next time slot. Therefore, the user with the m^{th} best CSI acts as a relay for the next user with the succeeding CSI to increase the rate OF transmission. CoMP transmission, which involves multiple base station cell-edge users at the same time, can improve cell-edge user performance. [125] introduced NOMA in CoMP, using alamouti code to promote a cell edge user. [126] confirmed the

usefulness of NOMA in CoMP systems in comparison to initially demonstrated traditional joint transmission NOMA. [128] explored a CDRT method in which the BS interacts with a nearby user and relay at the same time, referencing NOMA and interacting with a far-end user using the relay in the subsequent time slots. The primary challenge is removed by utilizing a NOMA inherent property that permits a receiver to attain additional information, like other user signals, for the cancellation of interference. Table 1.7 summarizes the current research on NOMA in data transmission, where "OP" denotes the likelihood of an outage.

Table 1.7 Cooperative NOMA

Methods	**Parameters**	**Features**	**References**
DF	Ergodic rate + OP	Maximum gain	[132]
DF	Throughput+ OP	Powered delay	[123]
CoMP	Sum rate	Superposition coding	[68]
DF	Sum rate	2-stage power allocation	[123]
CoMP+DF	Ergodic rate + OP	Same diversity order of the same stream	[67]
AF	Ergodic rate + OP	Multiple relay antenna	[69]
DF	OP	2 stage relays	[133]

1.7.1.3 NOMA in the Influence of Cognitive Radio Technology

In the field of wireless communications, cognitive radio [75] is suggested, which enables un-licensed users to utilize a licensed spectrum till the time unacceptable interference is induced to the licensed users. [76,77] authors examine how to control sensing overhead and CR throughput. Spectrum sensing is a CR-enabling technique. If an un-licensed user is positioned at a considerable

distance from a licensed user interruption between them will be minimal in most cases. As a result, an adaptive NOMA scheme inspired by CR-NOMA is proposed, which differs from the conventional method having a fixed allocation of power. Using NOMA-based underlay CR networks, authors [78] have investigated outage probability and generated an expression for the likelihood of an outage. [79] also developed power control methods for CR-NOMA to improve the sum rate, energy efficiency, and other performance metrics.

1.8 Channel Coding Methods in 5G

High spectral efficiency is critical for 5G and satellite communications networks because it facilitates significant cost reductions per bit. To achieve high spectral efficiency, channel modulation, and channel coding is required, both of which are critical components of the physical layer. In general, this efficiency is attained by combining a higher-order modulation with a low code rate and higher SNR. The transmitted power of practical wireless networks is limited. While using high-order modulation it is essential to be selective. As a result, a different category of channel coding scheme is required for the embedded version of 5G and satellite communications [92]. 5G specifications are studied, in terms of its spectral efficiency and error correction capability.

A significant increase in demand for higher data rates across a range of wireless systems is observed. To achieve higher data rates, a variety of design parameters should be considered, including complexity, bandwidth, latency, and energy consumption. These parameters can occasionally devolve into a constraint condition. Cellular communication methods have shifted over time, from 1G to 5G, and have enabled us to access a plethora of services. 1G prioritized voice services, after that 2G enhanced voice services and added support for text messages, later low-cost mobile internet, and 3G integrated voice services came into the picture, further 4G gives higher mobile multimedia services, and eventually, 5G requires increased capacity, lower latency, and greater consistency [80].

With the addition of turbo and LDPC codes, the speed of error correction methods increased significantly from 2G to 3G. It signaled the beginning of a

paradigm shift. On the other hand, the error correction methods used in 3G and 4G networks are nearly identical. Due to the additional problems in 5G, such as interference in heterogeneous networks and significantly increased spectral efficiency an even larger leap from 4G to 5G is to be expected. The DVB-C2 standard came into existence in 2008 [64]. It has a spectral efficiency value of 10.8 bits/s/Hz and is capable of modulating from 16QAM to 4096 QAM at code rate values ranging from 1/2 to 9/10. Obtaining such spectral efficiencies with a reasonable amount of effort under real-world operating conditions is challenging. In comparison, a higher-order modulation scheme would deteriorate the performance of the system due to its susceptibility to wireless fading channels, prevalent in a functional wireless communication system. Today's modern cellular and satellite communication technology makes use of LDPC and Turbo codes, among other technologies. These techniques are suboptimal when viewed from the perspective of information theory. As a result, alternate strategies must be explored. For 5G a new error channel coding method is required, which is currently being researched. The term spectral efficiency (*Ƞ)* means the rate at which data is transmitted for a given bandwidth. It is given as:

$$Ƞ = \frac{R}{W}\ (bits/s/Hz) \qquad (1.4)$$

The connection between spectral efficiency and SNR is defined by Shannon. in the following way to ensure reliable transmission

$$Ƞ < log_2(1+SNR) \qquad (1.5)$$

As depicted in equ. 1.5, spectral efficiency increases as SNR increases. In general, high SE is obtained by combining a higher-order modulation having a low code rate with a high signal-to-noise ratio. However, transmitted power is limited in practical wireless communication networks and is represented as:

$$SNR > 2^{Ƞ} - 1 \qquad (1.6)$$

A modulation technique with a low code rate is only relevant if the channel condition is favorable. The term efficiency is commonly used in cellular systems to describe how effectively the channel capacity is used. The spectral efficiency

of systems has improved as the technology moved from 1G to 4G, as presented in Table 1.8.

Table 1.8. Spectral efficiency of cellular systems [109]

Generations	**First Generation (AMPS)**	**Second Generation (GSM)**	**Third Generation (CDMA 2000) EV-DO**	**Fourth Generation (Long-term evolution)**
Max link spectral efficiency	0.001	0.52	2.5	4.08 (Single in single out)

As outlined in Table 1.8, spectral efficiencies have significantly increased between 1G and 4G frequencies. There was a noticeable increase in speed from 1G to 2G. Techniques for encoding and modulation significantly contributed. Additionally, MIMO methods are essential in 4G systems even though they enable the transmission of more data in less time. The importance of MIMO techniques is highlighted even more in 5G systems. To attain a spectral efficiency that is beyond 5 bits/s/Hz, a new type of channel coding scheme is required. Various error correction coding methods for 5G are discussed.

1.8.1 Turbo and Duo binary Turbo Codes

It was possible to obtain a very small error probability near the Shannon limit using turbo codes. Turbo codes maximize the efficient free distance [46] and optimize the load distribution between codewords when Eb/N0 is less [45]. The drawback of the conventional turbo codes is that these fail to reach the error floor at a BER value of 10^{-5} because of an inadequate minimum Hamming distance. Several solutions include developing a high-quality interleaved data structure, increasing memory capacity, and utilizing non-binary codes like 3D turbo codes or duo-binary turbo codes.

In this segment, duo-binary turbo codes [46], are seen as an applicant for 5G and satellite channels. Duo-binary codes are created by concatenating two CRSC codes, each having two inputs. It encodes all its data bits in a pairwise manner. CRSC encoders do not require tail bits to function properly [47]. Discusses the numerous benefits of duo-binary turbo codes, including enhanced convergence, increased minimum distances, decreased sensitivity, decreased

latency, and decoder robustness. Due to the slight complexity difference between both the MAP algorithm and its simplest variant, it is especially simple to implement large numbers of decoders for duo-binary turbo codes. The most important aspect of it is that it pairs extremely well with a high-order modulation [48]. Table 1.9 shows the standard specifications for turbo and duo-binary codes in terms of code rate, termination, and polynomial length. These codes have already been included in several standards.

Table 1.9 Turbo codes and duo-binary turbo codes

Standard	Category of Turbo code	Termination	Code Rate	Polynomial
3G (CDMA 2000 & UMTS)	Turbo codes	Tail bits	1/2, 1/3, 1/4	13, 15, 17
4G (WiMAX)	Duo- binary Turbo codes	Circular	1/2~7/8	13,15
CCSDS	Turbo codes	Tail bits	1/2, 1/3, 1/4 & 1/6	22, 33, 25, 37
DVB-RCS	Duo-binary Turbo codes	Circular	1/3 ~ 6/7	13, 15

1.8.2 Non-binary LDPC

R. Gallager [23] introduced the idea of LDPC codes. Later, Mackay and Neal discovered the LDPC codes, and they gained widespread attention. This was because the transistor era had just begun, and the hardware system at the time was incapable of supporting the ambiguity of LDPC encoding and decoding. Following the implementation of turbo codes in 1993, numerous researchers sought to understand its potential while developing a new error correction algorithm.

In 1996, Mackay and Neal developed a linear block code that possessed most of the same properties as turbo codes, including iterative decoding, large block length, and randomness. It was discovered that the new codes are virtually identical to those used by Gallager in his LDPC codes. Luby later introduced irregular LDPC codes [45] in 1998, which are a simplification of Gallager's LDPC codes and represent a significant advancement. At the moment, the most

impactful error control codes readily accessible are the irregular LDPC codes. LDPC codes offer numerous advantages over turbo codes. To begin, there is no reason to include a random interleaved. Second, it outshines the competitiveness in terms of error floor and error rate per block. Third, despite requiring a significantly larger number of iterations, iterative decoding of these codes is a simplified operation. The primary advantage is that it is not patent-protected. Reed Solomon are the non-binary LDPC codes [6, 45], and they can contain non-zero H matrix elements. Iterative BP is capable of achieving high performance for non-binary LDPC codes with medium code lengths and a high order modulation greater than 16 QAM. The complexity of decoding continues to be a major barrier to its commercialization.

1.8.3 Additional Potential Channel Codes

Two additional possible channel codes are discussed in this section. The first is called sparse regression codes, and the latter is called Polar Codes. Both are discussed in greater detail below. Sparse Regression Codes [3] are the other category of code used for channel coding and Gaussian multi-terminal source, which is constructed using a statistical framework with HD linear regression. These are used in the growth of multi-terminal Gaussian source and channel coding. These are compression and transmission codes that have rate optimal and low complexity. Their capacity for cooperative communication makes them ideal.

1.8.4 Polar Code

Arikan [3] defined polar codes as a subclass of linear block codes which were the first to be demonstrated while achieving channel code criteria. These codes revolutionize the channel coding space by taking a different approach to code construction than traditional codes, making them the newest contenders in the 5G. Additionally, the simplicity of the encoder combined with the simplicity of the SC decoder makes polar codes appealing for a variety of 5G applications. These codes increase the channel capacity when used in combination with the Kronecker product. Polar codes exist in systematic form. These codes have a recursive framework and no error floor, which makes them perfect for high-

density implementations. These codes are recognized through the theory of channel polarization, in which ‘N’ duplicates of any ordinary channel represented as ‘W’ having capacity I(W) are converted into extreme channels with the symmetric capacity of either ‘1’ or ‘0’. When ‘N’ increases, these extreme channels become polar codes. These ‘N’ polarised channels are either perfectly reliable channels with a maximum capacity of ‘1’ or extremely noisy that propagate only random noise. After that encoder passes ‘K’ information bits over consistent channels while freezing the residual N–K bits and transmitting them over un-reliable channels.

The basic polar decoder is an SC decoder, that can achieve Shannon's capacity with a low level of complexity. Numerous optimized SC decoding algorithms are proposed [24, 25] to improve the finite length of polar code. These decoders are explored for 5G research because they employ ‘L’ concurrent decoding paths and thus perform better than MAP decoders. It is observed that cyclic redundancy CA-Polar decoding with SCL can outperform LDPC and LTE turbo code. The PC-CA code is preferred for low-latency uses. Therefore, it is considered for the URLLC scenario [24]. PC CRC-aided codes make use of PC bits, distributed CRC, and maximize the profits of assistant bits. Fig.1.16 depicts the block diagram of a PC-PA polar code where ‘K’ denotes transmitted data bits, K’ denotes the Assistant bits formed using data bits, CRC and PC, and ‘N’ denotes the encoded data bits as an output of the polar encoder. The CRC bits are evenly distributed throughout a block which is used for the detection process. PC bits, on the other hand, are distributed in both reliable as well as un-reliable locations, which aids in early termination and error correction.

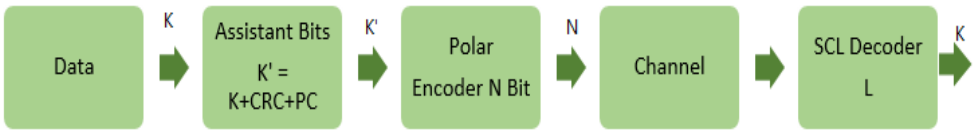


Fig.1.16. PC–CA polar code block diagram.

Both PC and CRC bits, which are used in conjunction with the SCL decoder, aid in refining the performance of the polar code. However, due to its early

intervention, PC bits improve the distance spectrum as well as provide supplementary coding gain.

1.9 Compatibility of Polar Code for 5G-URLLC

The final section discusses all possible channel coding schemes for the URLLC scenario. This portion discusses several highly probable reasons why polar codes are appropriate for the URLLC use case. Huawei is the primary leader in introducing polar codes as the channel coding technique for 5G field trials. It achieved a downlink speed of 27 Gbps in October 2016 [93]. Additionally, at 3GPP meetings, polar codes were chosen to replace deep-rooted turbo codes for the 5G eMBB scenario. Numerous studies are being conducted on channel coding techniques for the URLLC case, to find out the best option amongst all. Polar codes are considered the most attractive candidate amongst all in this competition. This is because of superior error correction capability for small information block lengths at lower code rates, as well as the ease with which code can be shortened and punctured [94].

Polar codes achieve better coding gain and improved spectral efficiency by requiring less SNR than other candidate codes with a similar error rate. For URLLC's reliability constraint, an FEC code with no error floor and a greater coding gain are to be preferred. The reliability of any transmission is evaluated if the error-free performance lies below BLER of 10^{-5}. LDPC and turbo code demonstrate the occurrence of an error floor with smaller block lengths at around 10^{-5} BLER. These codes exhibit error-free performance [43, 45], making them more consistent than other FEC contenders for achieving the desired value of SNR. The power consumption of a decoder in a circuit is proportional to its computational complexity. Polar decoders consume minimal power as it uses low-complexity SC-based decoding algorithms [46]. Polar code consumes up to 20 times less power than turbo codes with equivalent complexity. Polar code meets the low latency requirements of R1-164040, R1-1611692, 3GPP TSG, 3GPP TSG RAN WG1, and RAN WG1[6].

1.10 Motivation

Future wireless networks, including those of the 5G, serve a wide variety of novel use cases and applications that have distinct performance characteristics. In this regard, the ITU-R reached a consensus in September 2015 on their perception of IMT-2020, which highlighted three major application scenarios for 5G technology (i) URLLC (ii) mMTC, and (iii) eMBB. In this respect, the ITU-R made a deal in September 2015 on their dream for IMT-2020 and beyond networks. As a direct consequence of these specialized uses, new 5G technologies have been driven to be developed to provide a universal connection to all of the many applications. It is not accurate to refer to the 5G-NR as only an upgrade to the 4G-RAT, as this is an entirely new technology.

The 5G New Radio Access Technology, also known as 5G-NRAT, combines the benefits of multiple new technologies into a single system. These technologies include the most recent heterogeneous network architecture, the majority of new frequency bands in GHz with enormous bandwidth, millimeter wave communication, Internet of Things (IoT), massive MIMO, and mMTC. In other words, 5G-NRAT is the result of the combined influence of multiple new technologies. In addition, selecting a channel coding scheme that is both effective and efficient is a crucial step for any mobile system. The channel coding schemes for 5G-NR have extra encounters to enable and motivate a diverse range of use cases and new applications that are currently under development. As a consequence of this, for 5G scenarios, it is recommended that channel coding approaches undergo a paradigm shift. But many of the applications that will use 5G in the future haven't been tested yet with cutting-edge channel codes like turbo and LDPC codes. These are important parts of conventional 3G as well as 4G systems amongst LTE and UMTS. Industry and academia are working together to build a channel coding system that is both efficient and powerful and that can fulfill the needs of all of the different 5G applications. Over the past ten years, the polar code has emerged as a notable development in the field of channel coding, which is noteworthy in light of the ongoing search for an effective coding system for future generations.

In the battle to develop algorithms for 5G channel coding, the Polar codes developed by Erdal Arikan are seen as a serious competitor. A determination was made in 3GPP Release-15, regarding the channel coding schemes for the 5G eMBB development. As per this decision, the tail-biting convolutional code was replaced by polar codes. In RAN 86 and 87 sessions, the 3GPP concluded that in the 5G-eMBB scenario, LDPC codes are going to be utilized for usage in data channels, whereas polar codes are going to be utilized in the control channel. The polar codes are additionally accessible via the physical broadcast channel. The channel coding for the two use cases that are still outstanding URLLC and mMTC has not yet been determined. This is a research topic that is available to anyone, and the 3GPP has started contributing to the RAN 85 standard. As a result, it is reasonable to assume that the polar code will be the most potential candidate for the URLLC among other FEC candidate contenders. In the next part, channel coding design issues that are presented by the URLLC situation are outlined. These challenges are difficult to understand because this scenario has two unpleasant expectations: (i) very high levels of reliability, and (ii) very low levels of latency. These codes have great potential to deliver excellent error correction performance, with excellent reliability and a very low latency of one millisecond. In addition, polar code has a superior coding gain, which means it has no error floor, and also has a low complexity in terms of decoding, making it more suitable for its application in URLLC cases.

Besides this, the data rate requirements for URLLC are the same as those for control channels, and in the eMBB scenario, polar code seems to be the winner for control channel coding. Arkan [1] was the one who first presented the capacity-achieving codes known as the polar family. Both academic institutions and private businesses have seen steady increases in the amount of time and attention they devote to researching polar codes over the past decade. As part of the 3rd generation partnership project, which is working on standardizing 5G wireless systems, polar codes have been given rights to be used in channel coding for both uplink and downlink controlling in eMBB. Some of the new communication infrastructures that 5G wants to bring in are URLLC and

mMTC, which could use polar codes. Channel dependability value calculation for each encoded bit is necessary for the building of a polar code.

This identification method is successful once the SNR and code length are established. As the 5G framework forecasts a wide variety of code lengths, speeds, and channel conditions, it is difficult to have a unique dependability vector for every parameter. As a direct result of this, a lot of work has gone into making polar codes that are easy to make, have simple descriptions, and have high error correction performance over a wide range of code and channel variables. It would be beneficial for the research community to incorporate them in error-correction performance evaluations as well as the design of encoders and decoders given their imminent widespread application. There is a potential for significant speed and complexity overhead in both the encoding and decoding operations, and the performance of the decoder is inextricably linked to the characteristics of the polar code. They were able to successfully increase their audience by including 5G compliance in their work, which tends to be centered on the installation of hardware and software systems. A document known as an industry specification details how a group of companies that compete with one another will provide a service.

Because of this agreement, different manufacturers can design goods that are compatible with one another. Standard details are typically the outcome of a negotiation between different businesses. A patchwork of approaches whose combination generates acceptable performance is the product of never-ending discussions and disputes between conflicting agendas. A step-by-step procedure is given for polar code encoding, starting with code concatenation and moving on to interleaving functions, rate-matching schemes, and polar code sub-channel allocation. The process starts with code concatenation and finally ended with sub-channel allocation and rate-matching schemes that are specific to polar codes. The main purpose of the work is to give a complete guide to understanding and using 5G compatible polar encoding.

These codes can achieve the required bandwidth in a memoryless channel even if the code length ‘N’ is becoming closer to infinity. The process of standardizing 5G communications is putting a primary emphasis on increased

throughput, decreased power consumption, and improved error-correction performance. For example, the primary objective of machine-to-machine communications in 5G is to connect a wide range of devices at a scale that is larger than that of the most bandwidth-intensive application fields in 3G and 4G [3], all while staying within a limited power budget. As a consequence of this, it is necessary to devise methods of encoding and decoding that are both reliable and effective.

1.11 Thesis Organization

The thesis work contains seven chapters, and it is structured in the following manner:

Chapter 1 The chapter gives a brief introduction to the fifth-generation networks, existing challenges and problems in the cellular networks, 5G potential and motivation, 5G standardization, 5G modulation techniques and 5G channel coding methods, and compatibility of polar codes in 5G.

Chapter 2 The chapter discusses the literature survey with the help of different research papers in the field of 5G cellular systems, problem statements, and the research objectives. It also discusses the findings of the literature survey to carry out further research in the relevant field and development.

Chapter 3 The chapter discusses the concept of the linear feedback shift register and gold codes in wireless communication. It gives a brief detail about the basic concept of spread sequence i.e PN sequence, its properties, sequence generation using LFSR, ways of implementing LFSR including Gaolis and Fibonacci, the concept of serial and parallel LFSR, maximum length sequence generation for different input lengths, implementation of gold code using a 10-bit linear feedback shift register, a design methodology for implementing gold code using LFSR for a 5-bit and 10-bit input sequence.

Chapter 4 The chapter explains the forward error communication system using a block diagram, the concept of LDPC codes, LDPC encoder and decoder, design methodology for LDPC codes, hardware architecture, the concept of

turbo codes, turbo encoding, and decoding procedure, design methodology for turbo codes.

Chapter 5 The chapter includes an introduction to the methodology, language for designing, software tools, simulation description, and design synthesis, as well as an explanation of the project design flow using FPGA. It also covers the entire functional simulation and logic verification environment.

Chapter 6 The chapter covers the RTL view of various functional modules, simulation waveforms, FPGA device utilization summary of the turbo, LDPC, polar encoder & decoder, and the basic building modules of all the modules. The chapter also details the FPGA synthesis, experimental verification, and real-time signal processing in Chip-scope Pro-Analyzer. The chip performance is assessed based on different FPGA parameters and the comparative analysis is of all three encoders and decoders carried out to estimate the optimal solution for the 5G communication system.

Chapter 7 This chapter gives the concluding remark and the future scope.

CHAPTER 2

LITERATURE REVIEW

The chapter discusses the literature survey done with the help of different research papers in the field of 5G cellular systems and channel coding. It also discusses the findings of the literature survey to carry out further research in the relevant field and development.

2.1 Detailed Survey of Research Papers

The findings of the different standard research papers contributed by different authors is discussed below.

Turbo codes are utilized in the 4G mobile infrastructures, in deep space satellite communication, and in numerous applications that challenge inventors to achieve dependable data transmission over latency or bandwidth inhibited using transmission links in the presence of noise. The hardware application of a serial turbo decoder architecture [83] was done on FPGA for LTE and Wi-MAX communication systems. An ILM was used for LTE systems that utilized comparators and subtractors instead of multipliers and dividers. This unique interleaver overcomes the need for memory utilization as compared to the parallel architecture approach. The algorithm was proposed for the scheme of the interleaver [84] for turbo decoder architecture. The error correction performance, as well as latency, was improved in comparison to the sequential decoder by the integration of the algorithm. The algorithm involves the parallel architecture structure, which is highly efficient about the area, and BER thus, improving the throughput respectively. VHDL was used for the application of different error control coding [85] encoders and decoders on FPGA.

The error control codes are an important area of discussion in telecommunication scenario since they provide efficient ways of error correction and error detection. The analysis of different channel coding

techniques [86], was proposed for the 5G mobile systems based on BPSK modulation for AWGN channels such as polar, turbo, and LDPC. In terms of computational complexity, polar codes proved to be the best of all. A turbo decoder architecture was presented based on the Log Map algorithm [87] using a parallel processor to meet the requirements of long-term evolution communication standards. It is observed that this structure gives better performance concerning BER. The synthesis results show the device utilization such as the count values of slices and 4 input LUTs is less using this algorithm, making the design cost-effective. Based on the study and discussion on the forward error correction methods [88] like turbo codes and conventional coding, it is observed that turbo codes depict the best theoretical bound in terms of efficiency and channel capacity. In recent times these codes are preferred in several applications such as CDMA-2000 and 3G cellular networks such as UMTS. The Log MAP algorithm is used for turbo decoding [89] in the Wi-Max communication system. The lookup table (LUT) based Log MAP algorithm was proposed for turbo decoder structure in LTE wireless communication system [90] that reduces the logic resource utilization by 15% in comparison to the conventional design approach. The implementation of a concurrent turbo decoder [91] with an inverse address originator scheme was proposed for future broadcasting communication systems. This address-generation scheme reduces the iteration processing time which further reduces latency. Xilinx integrated logic analyzer (ILA) tool is used to check the validity of the turbo decoder, BLER, and latency. A new design approach for the realization of a turbo decoder on [92] FPGA with a RAG to reduce the latency and complexity. To reduce memory issues a contention-free interleaver is designed and a clock gating technique is used to improve power dissipation.

The turbo decoder base was proposed to get high throughput. This was achieved using a processor in which the algorithm has accelerator units and highly controlled software. As the need for configurable memory interfacing is required memory interfacing with a switch fuse unit is used instead of fixed memory interfacing. The parallel decoder [93] based on configurable interleaved networking was used for a long-term evolution standard. A multi-parallelism interleaver approach [94] was designed to reduce computational

complexity. The performance loss due to parallel structure is compensated using path metrics [95]. The proposed design is a competitive solution to recent work in terms of high reliability and efficiency.

The related work presents the turbo decoder design and FPGA implementation with different aspects. They have implemented the LTE turbo decoder with the Max-log MAP technique on Virtex-6 XC6VLX75T with a clock frequency of 270 MHz, the number of slices 4108, and LUTs 6310. The work is also done on the VHDL-based design to implement a turbo decoder with a Log-MAP technique that follows the decoding iterations but not analyzed FPGA resource utilization. The performance of the turbo codec is also estimated with the log-MAP algorithm on CPU and FPGA and further utilization of the FPGA for LTE. The work is not done on the latest FPGA that provides fast switching, minimum FPGA hardware parameters utilization, memory, and delay. The work investigates the performance of the turbo encoder and decoder by designing the hardware chip using a simple architecture with the integration of the log MAP algorithm on the same chip. Xilinx simulation for the data verification is done, and a comparison of hardware and timing parameters on the latest supporting FPGA is presented. The originality of the work is that the encoder and decoder design supports greater frequency with optimal timing and hardware FPGA resources.

The cutting-edge optical memory technology discussed focuses on the Semiconductor optical amplifier (SOA)-assisted ultrafast nonlinear interferometer [114]and can read, and understand variable-length packets at 20 Gb/s without reversing the data. This memory can read and write variable-length packets without inverting the data. Many topologies of coherent OCDMA [115] related to homodyne and heterodyne detections were used to determine whether or not the effects of multiuser interferences have an impact on the BER performance of the structure that makes use of a unipolar spreading code. The author assumed the Gaussian distributed multiple co-channel interferences while releasing the SNR of coherent homodyne OCDMA systems. These systems might make use of an injection-locking phase modulation strategy or an external phase modulation approach. The OFDM technology [116] consists of FFT and IFFT at the transmitter and receiver end respectively.

LFSR has been used for the implementation of such modules at the hardware level in which variable FFT [118] and different configurations of FFT are used. The channel capacity of the MIMO OFDM [119] also depends on such operations in which LFSR operations do not require side operations. The spread spectrum method used by CDMA uses a PN sequence. A very important issue with the spread spectrum technique is the near-far effect [120]. An OCDMA coding [121] was proposed for an indoor VLC locating system by the author. The proper receiver is used to analyze both unipolar and bipolar codes. For bipolar code analysis, an HPF is utilized. When using a code length that is the same for both sorts of codes, the minimum, and maximum distance errors are identical. The findings suggest that bipolar codes include several characteristics that render them more ideally suited for use in an indoor localization system. The optical CDMA system based on duo-binary signaling and LDPC coding [122] is used.

The carrier-hopping prime code and code weight, together with their shifted versions, are utilized in the system to generate signature sequences. Once the channel quality is further strengthened, the application of LDPC coding seems to have the ability to substantially raise the overall system performance. This might be accomplished by increasing the amount of data that is encoded in each bit. Quantum multiple access [123] was used which is a reliable method for integrating several single photon channels. An add-drop multiplexer is used to inject and extract a single photon in an optical fiber connection that carries quantum bits of several different users. In addition, spreading dispersed noise in the channel at the receiver end. Both of these factors contributed to a reduction in noise and an improvement in SNR value. The architecture that was proposed illustrates an increased performance of CDMA-based QKD networks by utilizing a single photon rather than amplifying and modulating the signal. The author proposed a concept of a Hidden Markov model-based spread spectrum known as a HSS along with a DSSS system [124]. HSS stands for Hidden Spread Spectrum. The capacity of HSS to produce chip sequences at random for every information bit renders autocorrelation-based attacks that have previously been used against the DSSS. HSS continues to appreciate the benefits of processing gain in the same manner that DSSS does. An effective

digital HBC transceiver [125] hardware design is suggested by the author in compliance with IEEE 802.15.6 standard. This was done to get around some of the problems that are inherent in the standards for RF communication, such as signal leakage, power consumption, and on-body antennas.

For both transmitter and receiver modules, the proposal makes use of a frequency-selective digital transmission mechanism. To investigate the available design resources, several families of FPGA are utilized. The performance analysis of three different CDMA coding systems [126] based on M-sequence, random binary sequence, and gold code is presented. The duration of the CDMA coding and the sampling rate of the device both have the potential to affect the resolution as well as the detectability of a lot of targets for multi-agent radar systems. The BER [127] is used to measure how effective the modified spreading code proves using the characteristics of optimized and non-optimized spreading codes. It has been discovered that the method proposed as optimized dynamic logistic map-based DSSS, is good than both orthogonal spreading code and non-optimized spreading code. Orthogonal optimization offers an improvement of 33% over static optimization while dynamic optimization offers an improvement of 56% over orthogonal optimization. Because of this, the proposed technology is successfully utilized in a wireless communication system.

The simulation was done to increase the data throughput of the IEEE 802.15.4 protocol by making use of a gold code sequence [128]. The results of the simulation demonstrate that conventional IEEE 802.15.4 PN sequences are outperformed by gold sequences with a spreading factor of 31/5 when it comes to increasing data throughput. Monte Carlo simulation results reveal that the gold set, which consists of spreading factors of 63/6, 127/7, and 511/9, can complete data transfer even when chip defects reach unprecedented lows. It is used to generate and acquire C/A *g*lobal positioning system GPS signals [129] for the relationships that exist between the various C/A gold codes. Because there are fewer XOR functions utilized in the suggested design, it requires less time to complete and makes use of fewer resources than conventional ones. In addition to this, it can be manufactured at a lower cost and can be applied to the

process of acquiring all C/A-Gold GPS satellite signals. The new MGS code producer [130] is used to design a gold sequence generator that makes use of discrete D flip-flops as its primary logic component. The findings of several simulations of different sequences indicate that the newly developed MGS has some benefits over gold sequences. This novel sequence is confirmed using some crucial parameters which depict that it is more acceptable and practically achievable. The pipelined LDPC decoder [149] was used for WiMAX standard 802.16e. This pipelined decoder architecture used a high-end synthesis tool to reduce validation time. The architecture used the minimum throughput requirement for large-scale integration approaches. Also, the use of 8-bit fixed point arithmetic adds extra precision to the design and delivers better performance concerning bit error rate. The Belief propagation and log-likelihood ratio-based demodulation methods were used [145] for LDPC codes to Ruby transform-based decoder structure so that the computational complexity is minimized.

The simulation results for both algorithms using the AWGN channel are obtained w.r.t BER. The Monte Carlo-based evaluation method was used for theoretical analysis. It was found that the combined approach of these two algorithms reduces the hardware complexity and improves the SNR. A new scheme for the construction of LDPC codes based on a progressive edge growth mechanism [136]. To achieve optimum a pipelined structure and memory organization using a single port bank was considered at the time of code construction. The hardware efficiency should be maximized using code constraints for quasi-cyclic LDPC code. The design was implemented in FPGA and throughput was experienced to be increased from 39 to 110% using pipelined structure. The generic behavioral model was detailed for generating LDPC decoders [150]. Decoders are needed in applications that require reliable and fast data transmission. On the other hand, coded RTL architecture provided good performance, but the critical path of IP design was slowed down.

The LDPC decoders are implemented using Xilinx Vivado HLS. It was observed that the hardware complexity is reduced by ten times and the throughput is increased by 1.5 times. The model performed similarly to

handcrafted RTL structures. The parallel architecture consists of multiple single input single output elements [150] that offered low latency for turbo decoding. This structure is compared with the conventional sequential architectures. For concurrent execution of the design, SISO-based parallel interleaver and related algorithms were presented. The parallel architecture reduced the latency by 20 times and increases the throughput by six times in comparison to other sequential decoders. The implementation of linear, convolution and cyclic codes was done on FPGA [133] using VHDL. Error control codes is used for error detection and correction in the noisy channel. It is achieved by the addition of redundant bits to the data at the transmitter end, and the correlation of these redundant bits is used at the receiver end for error detection and error correction. Multiple channel coding methods were suggested [126] such as LDPC, systematic convolutional, polar, turbo, and non-systematic codes, using the BPSK modulation method in machine-type communication for the 5G wireless system. Two different categories of Quasi Cyclic LDPC code were used [150] out of which one is binary and the other is non-binary, and their construction is based upon finite field subgroups. These codes provide better performance in the case of the AWGN channel using an iterative decoding algorithm. These codes utilized larger minimum distances in comparison to the finite geometry LDPC codes and it balances the performance in terms of decoding complexity and error using an iterative decoding process. These codes provide burst errors in such an environment, which was capable of replacing Reed Solomon codes by offering large coding gains at same code length and code rate. The combination of the Log-MAP decoding algorithm and the LUT-Log-MAP algorithm is called the LUT-Nor-Log-MAP algorithm [150].

The algorithm was used to perform normalization functions for the SISO decoder unit. In simulation results, it has been observed that resource utilization can be saved by 2.1 % using the LUT-Nor-Log-MAP algorithm. Also, the decoder using the LUT-Nor-Log-MAP algorithm depicted a gain of 0.25~0.5 dB. Finally, the design of the turbo decoder provided a throughput of 36 Mbit/s that can be achieved using synthesizing on the Cyclone IV FPGA platform. The implementation of LDPC decoder architecture on FPGA [143] for the application of DVB-S2. This algorithm was applied systematically in such a

manner that supported 360 functional units of the design. Moreover, the synthesis of the LDPC decoder was done by targeting two FPGAs XC6VLX240T and XC2VP100. The partition and shift methods were used for LDPC codes [140] to shift and split the check nodes and bit nodes into subsets and then connect these nodes into subsets. A theorem is derived to prevent the cycles harmful in LDPC decoding. The simulation results of these codes depict a good response in terms of bit error rate over EPR 4 channels. The flexible designs based on an open platform were used [124] that required the parallel computing method to decode the data based on an iterative minimum sum algorithm for a (3,6) regular LDPC code with variable codeword length. The decoding algorithm supports parallelism for which Altera offline compiler version 15.1 is required. This algorithm is tested on Altera Stratix VGXA7 FPGA hardware. This design gives an effective throughput of around 68.22 Mbps for a 2048 length (3, 6) regular LDPC code at a clock frequency of around 163.88 MHz, and for a length of 1024 (3, 6) LDPC code it gives a throughput of 54.8 Mbps by showing an improvement of 7 Mbps. The design and implementation of the turbo decoder is done using Verilog [125]. The design uses a max-log algorithm to reduce the number of iterations and provided early termination due to which power consumption was reduced. To provide early termination sign difference ratio factor was used.

The concept of clock gating was used to provide better power efficiency. The design was implemented and tested on Virtex-4 and Virtex-5 FPGA. The performance of convolution codes was analyzed in the presence of a Rayleigh fading channel [127] with a long constraint length. As the sequential decoding cannot be applied below a threshold value of SNR. Therefore, the selected SNR per bit information is 5.7 dB. Moreover, the process of uniform quantization is chosen in such a way that it can provide negligible loss. For this design, the finite interleaving method is used to an interleaver depth of (50 × 50) with a Doppler frequency of 0.01. The suggested scheme performs well wr.t bit error rate when compared to the turbo codes.

LDPC codes were used [134] where construction of the parity check matrix is done using the sub-matrix structure. In this design, the inverse of the matrix is

replaced by the multiplication of the sparse matrices. The replacement of matrices can result in efficient encoding that further reduces computational complexity. The simulation results showed an increase in throughput up to 1 Gbps. A quasi-cyclic LDPC code based on Euclidian geometry (8176, 7154) is used for high-speed data rate. The architecture incorporated a non-uniform quantization method and algorithmic transformation-based critical path reduction to reduce memory size. The parallel decoders are proposed in the design to enhance throughput. The design was implemented on Xilinx Virtex-II 6000, FPGA. The FPGA synthesis has proved that the design can achieve maximum throughput of 172 Mbps in 15 iterations. The 3rd generation partnership LTE standard [137] is based on the parallel turbo coding scheme that supports 188 block-size data. To reduce the hardware complexity of interleaver in turbo codes, a multistage network based on permutation polynomials with the address generator is used. An optimized decoding method is suggested to support high parallelism and enhance the system's performance. A radix-2 and radix-4 recursion based add compare select unit is proposed for the selection of block size that cannot be split by 16. The design is implemented on 130 nm CMOS technology that consumes 4.02 mm^2 core area and 1.81 architecture efficiency while achieving 384.3 Mbps peak throughput having 5.5 iterations at a clock speed of 290 MHz. An optimal design was presented [140] for irregular LDPC code that utilized a shaping method with the specified block length. It has been seen that LDPC codes are beating turbo codes in terms of computation cost and effective performance.

The aim of the research work is considered based on the hardware realization of turbo and LDPC-based encoders and decoders. The comparative performance analysis of these coding-based encoder and decoder architectures on the FPGA platform will be the new research work so that LDPC coding-based hardware can be used further for the M2M and D2D as in the 5G communication system. The research work focuses on the design and FPGA implementation of the turbo and LDPC coding-based hardware chip, performs simulation, and estimates the comparative performance.

Quite a few different strategies for fast designing frozen sets with as little complexity as possible have been put forward [144]. In the beginning, Arkan suggested including Bhattacharyya parameter tracking in addition to Monte Carlo simulation to determine the validity and dependability of bit channels. This method is known as density evolution (DE), which was first proposed in [135] and later refined in [146] and has the potential to theoretically guarantee estimation accuracy, albeit at a significant expense in terms of computing.

[137] Proposed a method for estimating the bit-channel reliability of AWGN channels which is based on the Gaussian approximation of density evolution. This method, which has been given the name the DE/GA method, can produce accurate results while maintaining a low level of complexity. On-the-fly design, on the other hand, results in a large rise in coding delay, which is incompatible with the needs of 5G. Current research on partial reliability sequence by the polarization effect has opened new possibilities for producing a universal reliability sequence regardless of the channel conditions. Because of some studies and extensive simulation, the 5G standardization is forced to propose a single universal reliability sequence. This sequence serves as a starting point for deriving the one-of-a-kind for each polar code that was taken into consideration in 5G.

[121] gave a proposal for the native polar code decoding technique, which was referred to as SC. It is possible to think of it as a depth-first binary tree having a left branch on its priority. The leaf node of the tree represents 'N' bits that need to be calculated, and the root node contains soft information about the code bits that have been received. The complexity of the decoding method is given as $O.(N\log_2(N))$. Fig.depicts the decoding structure using a tree for a (8,4) polar code that is represented in Fig.The leaf nodes of the decoding tree are shown in black, indicating information bits, while the nodes that are shown in white indicate frozen bits.

In [122], a list-based decoding method is presented as a possible solution to the problem. The goal is to enable several SC decoders to perform their jobs concurrently while preserving several possible codeword candidates, often known as pathways. A path metric is then generated for each candidate,

allowing less likely candidates to be rejected to restrict the number of paths. Each time when a leaf node encounters, a particular bit is predicted as both ‘1’ and ‘0’, essentially the number of code words is doubled. SCL considerably enhances the performance at modest code lengths, particularly if concatenated with an outside code such as CRC. However, this improvement somehow increases the design complexity. SCL based on CRC- assistance is utilized as a starting point in error-correction performance tests for 5G, notably with a list size of 8. The longer the list is, the more complicated it is to implement SCL, even though the algorithm becomes more effective as the list gets longer. To reduce the level of complexity, a great number of potential solutions both software and hardware have been proposed for decoders. Because of the divided SCL [124] and its evolution at various stages of the SC tree, these can reduce the memory consumed for the higher stage. SC stack decoding algorithm makes use of a priority queue to expand only the candidate with the highest likelihood. Adaptive SCL decoding recommends expanding list sizes in the event of failure decoding, whereas [129] provides a hardware decoder with a configurable list size.

According to the definition in [61], channel polarisation is a change that starts with ‘N’ duplicates of a binary discrete memoryless channel and makes ‘N’ synthetic bit channels. The newly made synthetic channels are "polarised," which indicates that each channel can send a single bit with a distinct level of reliability and in other words, with a variety of ways of being correctly decoded. The process is repeated till all of the new synthetic channels have been "polarised." When the value of ‘N’ is sufficiently high, the mutual information value of these channels will either be near 0 (which will result in utterly noisy channels) or close to 1, which will result in channels that have extremely high capacities. Polar codes make it possible to generate polarised channels with a low level of complexity, and the proportion of noiseless channels that can be constructed with them approaches the potential offered by the original Binary Discrete Memoryless Channels.

It is essential to retrieve sub-sequences for codes with shorter lengths actively from the universal reliability sequence, which results in reducing the amount of

space needed to store the sequence. This sequence is comprised of 1024-bit-channel indices that are sorted with the help of reliability. This sequence can be used to construct the frozen set of polar codes with a length of less than or equal to 1024, regardless of the channel conditions. This recursive dependability structure offers a significant improvement in the designing of polar codes and is likely to be the most enduring legacy left behind by the process of standardization. This impressive achievement was accomplished by including distance features in the construction of short polar codes [149], employing list decoders, and adding assistant bits into the code [13]. All of these factors contributed to the success of this endeavor.

[131] proposed SC decoding algorithm for decoding the polar codes. This methodology may be conceptualized as a binary tree search, used to decipher polar codes. When the code length is short, this strategy has a high decoding delay and poor error correction capability, but when the code length is long, it achieves the best results possible. The SCL algorithm is recommended in [74] to increase its ability to fix mistakes. This algorithm works with a list of possible 'L' codewords. Also, a CRC is added to the polar code to help choose the right candidate after the SCL decoding procedure. CRC-assisted SCL is better at fixing errors, but this came at the expense of making the algorithm harder to understand and taking longer to work. LLR values were used to describe a hardware implementation of SCL that was discussed. Decoding algorithms like SSCL and fast-successive cancellation lists SCL are proposed to cut down on latency and improve performance. These algorithms use recognizing bit patterns to trim back the SC decoding tree to minimize the bit estimation number, with little or no loss in its error correction routine. When compared to standard SSCL, Fast-SSCL, and SCL schemes these algorithms can significantly cut down on the number of iteration steps needed for decoding a single codeword [77].

[83] proposed a PSCL decoder as a solution to the high computational cost of SCL decoders. This decoder consumes significantly less area and has a negligible error correction capability when compared with conventional SCL decoders. A solution is addressed in [58] for the high computational cost of SCL decoders. Successive cancellation list-dependent decoders are probably one of

the leading ways to meet the speed and throughput requirements of 5G error correction. Although the vast bulk of recent work on polar code decoder systems has concentrated on boosting the throughput and area occupancy, very little work is done on reducing the amount of power that is consumed [29, 32]. The machine-to-machine connected devices that make up a substantial portion of the market are mobile end platforms that get their power from batteries and small-scale energy harvesting electronics; to function, these devices must have an extremely low power/energy consumption [11]. The research work provides a complete analysis of the efficiency of polar code SCL-based decoders concerning the FER, the amount of space they occupy, and the amount of power or energy they use.

[141] Polar codes are an outstanding development in the channel coding sector and stand to be a good competitor in the FEC coding techniques for the 5G communication system. Moreover, polar code is assumed to be a control channels coding scheme for 5G-eMBB over PBCH. Authors have investigated multiple coding possibilities for a 5G-URLLC scenario. To suit the requirements for low latency and ultra-high reliability, the error-free capability with low coding rates and limited range block lengths is studied. Polar and LDPC codes are evaluated in terms of decoding complexity, reliability, latency, error floor, and BLER performance. Simulation results depict that the BLER performance of polar codes outperforms LDPC codes for low code rates and small block lengths. It is also demonstrated that these codes are error-free and more stable when compared to LDPC code at any given signal-to-noise ratio.

[142] In this work, the author has outlined the encoding process of polar code inside the 5th generation wireless systems, providing users a friendly description to understand the encoding aspect in 5G systems. This encoding process illustrates the positive efforts of the 3GPP standardization to fulfill the numerous specifications for the eMBB control channels. However, new decoding architectures can be designed to optimize the decoding complexity and increase the error performance.

[143] Authors have surveyed and assessed the performance of LDPC and polar code as the contender in channel coding techniques for 5G-mMTCs. The

authors simulated polar codes with SC and SCL decoding techniques. LDPC code based on MSPA and SPA min-sum decoding for mMTC systems is also considered. Channel coding in mMTC plays a significant role in low latency, small energy usage, and generating fast throughput. Polar codes using SCL show great results when compared to other coding methods used to achieve BLER 10^{-4}. These codes produce a gain value of 8.3 dB to 10 dB and Eb/N0 with a difference of 1 dB to 3 dB.

[144] Authors have examined the SCL-based polar code decoder w.r.t. energy consumption, power consumption, and error correction capability. SSCL, Fast-SSCL, and SCL have the same error correction capability, whereas PSCL undergoes small FER loss. It is observed that the investigated polar code decoders have equivalent error correction capability just like LDPC codes for WiMAX. Also, all four decoder structures are compared in terms of energy usage, area, and power for implementations, their trade-offs are analyzed. While comparing WiMAX-based LDPC architectures with SCL-based decoders, it is concluded that polar code decoders yield reduced area, energy, and power consumption making them more appropriate for prospective 5G communication.

[145] Authors have inspected polar codes with small durations and low rates for the 5G eMBB control channel. It is seen that for every code length value, there exists a certain rate at which polar codes attain the optimum efficiency in terms of power. A CRC selection technique is provided to maximize the error correction capability of polar codes. It is observed that CRC is mostly effective for codes of long durations and higher rates. For codes of low speeds and small lengths, SCL without CRC gives a satisfactory performance. CRC effect on the speed of fast-SSCL and SSCL decoders is also analyzed. It is depicted that Fast-SSCL delivers significant improvement in terms of latency. For codes with low rates and smaller lengths, SSCL proves to be a suitable choice as it gives the same speed just like fast-SSCL. [146] The authors explored the construction of polar code for distance improvement and assess their distance spectrum. A heuristic approach is suggested to optimize the list decoding performance over a particular range of dimensions k and list size L. In addition, a polar code

design based on dynamically frozen bits and “low-weight bits” is proposed, which gives superior performance when compared with BCH-polar codes.

[147] The authors proposed an alteration to the traditional polar code for using bit posterior and successive cancellation decoders. The development of unfrozen bits LLR distributions at the time of iterative bit posterior decoding is used to categorize the most susceptible unfrozen bit channels. The unfrozen bit channels are replaced with the most reliable bit channels. The bit-swapping architecture based on LLR enhances the performance of the code using BP decoding.

[148] Channel codes for the 5G NR systems are a fresh topic of research and polar codes are a significant advancement in this area. The research work investigates multiple channel coding contenders for the 5GNR scenarios specifically for URLLC and mMTC with different code rates and different block lengths. It is observed that the polar codes outperform other codes for almost every code rate and at every block length. In the case of polar codes, there is also no error floor. As a result, Polar code appears to be the ideal choice for 5G mMTC and URLLC scenarios with lower code rates and shorter block lengths. Although polar codes have demonstrated their potential, the low decoding efficiency for small block lengths remains an unresolved issue. 5G channel coding is a hot topic for many outstanding challenges in the coming years.

2.2 Research Gaps

Extensive research is done on the use of turbo and LDPC coding-based encoders and decoders using different algorithms. The comparative performance analysis of these coding-based encoder and decoder architectures on the FPGA platform will be the new research work so that LPDC coding-based hardware can be used further for M2M and D2D communication as in 5G communication systems. The research work focuses on the design and FPGA implementation of the turbo and LDPC coding-based hardware chip, performs simulation, and estimates the comparative performance. Potential research areas in wireless communication systems are-

(i) Massive MIMO for 5G mobile information system.

(ii) Channel coding methods like LDPC, Turbo, and Polar codes.

(iii) Channel modeling for 5G mobile communication.

(iv) Resource allocation and protocol design for 5G system.

(v) Mobile systems using 5G cellular network capability.

(vi) Measurement and testing of 5G mobile system.

2.3 Problem Statement

The problem statement of this research work is "Investigation of the various channel coding strategies based on reliability, complexity, flexibility, and latency in machine-type communication." Low latency, high communication reliability, low complexity, and better flexibility are the primary demand for machine-type communication for 5G channel coding schemes. As a result, the evaluation of various channel coding techniques is primarily based on meeting customer demands in machine-type communication. The channel coding methods currently available for 5G communication systems are LDPC, Turbo, and Polar codes [5, 8]. These channel coding schemes need to be analyzed from a hardware utilization point of view to ensure the user requirements in the real-time world.

2.4 Objectives

The following objectives are formed based on the identified research gaps:

(i) Designing and FPGA implementation of Polar encoder and decoder for 5G communication system.

(ii) Verify FPGA performance with different hardware and timing parameters.

(iii) Comparative FPGA performance evaluation of turbo, LDPC, and polar codes to estimate optimal solution.

2.5 Research Methodology

The methodology and FPGA design flow for the entire design is depicted in Fig.10 The Following FPGA parameters are utilized to support the design.

(i) Number of flip-flops

(ii) Number of slices

(iii) Number of I/O blocks

(iv) Number of LUTs

(v) Memory usage

(vi) Maximum frequency

(vii) Minimum & Maximum time before clock

(viii) Power

(ix) Combinational delay

The chip design involves the RTL design and functional simulation, as well as an estimate of the hardware and timing parameters. The Xilinx ISE 14.7 software is used for designing and putting things into action. FPGA synthesis is done so that different test cases can be used to see how well the design works. The modeling, simulation, chip design, FPGA implementation, and validation parameters are as follows:

Synthesis*:* The process of optimizing logic and simulating how it will work is called "synthesis." RTL depicts the chip view that is retrieved after modeling the chip. It lists all of the inputs and outputs in the chip.

VHDL Compilation*:* It depends on how the chip is made, whether it is made using the bottom-up or top-down approach. It entirely depends on the software used for simulation. Xilinx ISE design suite 14.7 is the simulation tool that is used in this study.

VHDL Analysis*:* It is based on the simulation environment that was used to test the functionality of the chip.

Device Utilization*:* This shows the hardware that is utilized in the chip. Device hardware usually involves the number of logic gates, decoders, input/output buffers, flip flops, multiplexers, latches, etc.

Delay Time Calculation and Timing Report: It gives the delay information, minimum clock period value, maximum output time duration, and minimum input arrival time duration.

Design Constriants	•Design the LDPC, Turbo and polar Encoder Decoder with predefined length of message input.
Modeling Approach	•The design adheres to VHDL-based modelling and design. In Xilinx ISE 14.7, it is possible to have a data flow model, a behavioural model, and a structural model.
RTL Analysis of Enoder and Decoder	•The RTL diagram shows the pins on the encoder and decoder chip. The input and output pins, as well as their directions, are examined.
Functional Simulation	•The built-in simulator performs functional simulation of various test cases and inputs for all designed chips.
Test Cases Analysis	•The various cases and test samples are used to examine the output waveforms with varying time delays and input to output nodes.
FPGA Synthesis	•FPGA synthesis is used to secure the FPGA pins in the Virtex-5 FPGA before applying logic placement, routing, and burning the programme in FPGA.
Parameters Analysis	•Analyse hardware parameters of FPGA like IoBs, flip-flops, slices, LUTs, memory, and timing parameters.
Comparative Analysis	•Evaluate the performance of Turbo , LDPC Codes and polar codes.

Fig.2.1 Methodology for FPGA synthesis

FPGA Synthesis: The real-time verification of the designed chip is carried out in FPGA to verify the simulated data with all test cases.

Comparative Parameters Analysis: The performance of the chip is estimated using device hardware and timing analysis-based parameters on FPGA. The comparison is carried out to estimate the optimal solution.

CHAPTER 3

WIRELESS COMMUNICATION AND CODING

CDMA [18] using DSSS is the most significant wireless communication technology. According to industry estimates, CDMA networks have approximately ten times the capacity of analog networks and significantly more than GSM networks or TDMA. CDMA provides carriers and users with several other benefits, including higher-quality audio and video transmission, expanded coverage, and enhanced security [26]. A gold code generally known to be a gold binary sequence is used in GPS satellite radio navigation and telecommunication technologies such as optical, wire, radio, and electromagnetic systems.

Robert Gold has given this name to the gold codes [49, 56]. When numerous devices broadcast using the same frequency band, gold codes produce a relatively small cross-correlation [26] proving to be advantageous. There are $(2^n + 1)$ sequences in a set of gold code, which exists for a period of $(2^n - 1)$. LFSR plays an important role during the generation of maximum length based on the 'n' number of bits in the register. The two maximum length sequences are grouped with the $(2^n + 1)$ XOR gates, which constitute some set of $(2^n + 1)$ gold code sequences. LFSR [62] consists of 'm' successive 2-state memory blocks or flip-flops that are synchronized using a clock.

In synchrony with the input clock pulses, all memory units change states at the same time, and the data of each memory block is shifted to the next block with the arrival of each clock pulse. If there is no signal in the first delay element, throughout this operation, all memory units will be empty at the end of 'm' shifts. A feedback loop is required to transform the shift register's content from a delay unit to the sequence generator [63]. The feedback loop provides the contents of a particular memory unit referred to as taps of the XOR operation [69] based on the modulo 2 adder operation. This way, a new term is calculated based on some of the previous 'm' terms as input.

In the field of cryptography and encoding, ultrafast communications present a new set of obstacles. Because of their capacity to function at extremely high speeds, optical logic gates may be appealing components for these applications. The current bitwise logic record speed of 40 Gb/s could be extended to bitwise logic rates of hundreds of Gb/s by careful design of the nonlinear optical switching elements. However, due to issues in synchronization and fan-out, high-speed optical computers will be limited to applications requiring low logic gate counts for the predictable future. Using optical logic to create a linear feedback shift register would limit the system to 40 identical taps. Thus, making the shift register extremely exposed to a sparse matrix.

The most often used technique for raising the rate of generating PN sequences is to multiplex several lower rates of PN sequences together. While multiplexing enables the system to generate PN sequences at any desired rate, one must carefully consider what needs exist in addition to the rate requirement [59]. If 128 PN sequences with period 't' are generated at 1 Gbps and then multiplexed, the result will be a PN sequence with a period of 128t at 100 Gbps. An important parameter in the security of PN sequences is the relationship between the available sequences and typical observation time and their period. If the rate of communications is increased from 1 Gbps to 100 Gbps, a demand of a hundredfold increase in the duration [61] is required. PN sequences are generated deterministically yet have the qualities of sequences generated randomly. In effect, PN sequences attempt to appear identical to the output received by tossing a fair coin and recording '1' for each 'head', and '0' for each tailed outcome. PN sequences should have an almost equal amount of '0' and '1', balanced runs. For example, 101 and 110 must be equally recurrent and have a long periodicity. The PN sequences can be employed in a variety of applications that need data generation based on their random behavior. To encrypt the data stream, one can XOR a binary data stream and a binary PN sequence together. To an observer unfamiliar with the PN sequence, the resulting bit-stream will disguise the original plaintext. However, a user who is aware of the PN sequence can XOR the encrypted stream and extract the plaintext. PN sequences can also be employed to encode data, for example, in spread-spectrum applications when white-noise signals are desired for

transmission. LFSR is one of the simplest methods for creating PN sequences. At each clock cycle in an FSR, the bit in the register's last cell is retrieved for output, all remaining bits are moved down one cell, and a new bit is stored in the first cell. The new bit is computed from the bits in the FSR using a function that is linear in the case of LFSRs but may be arbitrarily chosen in the case of LFSRs. Each cell containing the data utilized to generate the new bit is said to have a tap. The contents of the cells obtained via taps are combined according to a feedback function, referred to as the characteristic polynomial in D-transform notation. There are various difficulties associated with rapidly creating PN sequences. The work develops and implements a gold code using LFSR for a 10-bit data sequence.

3.1 Spreading Sequence

These approaches require distributing the necessary bandwidth to transport the data, which may appear counterintuitive at first. Benefits of bandwidth expansion include the conversion of a narrowband signal to a broadband signal and resistance to narrowband interference [74]. The primary sequences are based on the pseudo-noise (PN) sequence, Kasami's sequence, Gold codes, and Walsh Hadamard codes.

3.1.1 PN Sequences

It is a periodic sequence in binary [49] with a waveform that resembles noise. Frequently, the codes are constructed using feedback shift registers and are generated using a deterministic method that begins with a seed value. The process is deterministic, resulting in a non-random integer sequence. However, with a sound approach, the generated sequences will pass numerous plausible randomness tests. Pseudorandom numbers or m sequences are used to describe this type of number. A PN generator is often made with XOR gates and shift registers generally called a LFSR. The following requirements must be fulfilled by a PN code:

(i) These sequences must be made up of two consecutively leveled numbers.
(ii) To facilitate code synchronization, these codes must have a sharp auto-correlation peak (1 chip wide) and a low cross-correlation value.

(iii) With less cross-correlation, the system can support a greater number of users. This criterion holds for both complete and partial code correlation. There is a good chance that the two codes will only partially correlate with each other because of the random nature of the data.

(iv) The final condition ensures that the code's divisions by ones and zeros are balanced.

3.1.2 Properties of PN Sequences

Correlation: The correlation property of a code is very important while designing CDMA codes. This is because these can affect not only the multiple access interferences, which are caused by interference from other users but also the self-interference caused by multipath propagation. Correlation [47] is a term that refers to the process of determining how similar two pieces of data occur. It can be defined in two ways i.e. a negative integer between -1 and 1. The other digit indicates the degree to which the two items are connected. The first one is determined by the cross-correlation property of the candidate within the same family. The other two are determined by the autocorrelation property, the different versions of the same code that have been time-shifted together. Table 3.1 depicts the correlation values and their interpretation.

Table 3.1 Interpretation of correlation values

Value of correlation	**Explanation**
1	Both sequences exactly match each other
0	No similarity between the sequence
-1	Both sequences are mirror images of each other.

Auto-Correlation: It is used to define the relationship between two variables. The autocorrelation of a signal x(n) can be determined by examining how well it can distinguish itself from all time-shifted forms of itself. Equ. (3.1) defines the autocorrelation function $r_{xx}(T)$ between $x(n)$ and its shifted version $x(n-T)$ applied to a finite discrete signal, where 'T' denotes the time delay between two sequences and 'L' denotes the length of the sequence. This phenomenon is

known as autocorrelation [56]. It is the similarity between two sequences that contain all of their phase shifts. Correlations that do not have a phase shift of zero should have a Correlation value close to zero.

$$r_{xx}(T)=\sum_{n=0}^{L} x(n)\, x(n-T)(\text{unnormalized}) \qquad (3.1)$$

The above equ. (3.1) gives three different types of interpretation. Positivity is defined as the inability of a signal to be distinguished from a version of the original signal that has been time-shifted. A signal that is distinct from the original signal has a negative value of correlation. A signal with zero correlation is orthogonal to itself. A data word with a period of seven bits at a time t = 0 is considered as shown in Table 3.2. For a time instance value Δt = 1 to 6, there exist only six time-shifted duplicates of the word. When each bit of the original time instance t = 0 is compared to the time-shifted replica, several agreements (X) and disagreements (Y) are discovered. When these are subtracted, a measure of the degree to which the two words match is obtained. The sequence "0001101" shows an excellent autocorrelation property, as demonstrated by the fact that in any time-shifted variant of itself, the correlation value is significantly different. Table 3.3 demonstrates that the sequence "0111011" does not have a strong autocorrelation property. Because there are some instances where a match is rejected (correlation value = - 1) and others where the time-shifted copy matches at a correlation value = 3.

Table 3.2. Example 1 of autocorrelation

Sequence	**Time-shift**	**(X)**	**(Y)**	**(X-Y)**
0001101	Δt = 0	7	0	7
1000110	Δt = 1	3	4	-1
0100011	Δt = 2	3	4	-1
1010001	Δt = 3	3	4	-1
1101000	Δt = 4	3	4	-1
0110100	Δt = 5	3	4	-1
0011010	Δt = 6	3	4	-1

Table 3.3 Example 2 of autocorrelation

Sequence	Time-shift	(X)	(Y)	(X-Y)
0111011	$\Delta t = 0$	7	0	7
1011101	$\Delta t = 1$	3	4	-1
1101110	$\Delta t = 2$	3	4	-1
0110111	$\Delta t = 3$	5	2	3
1011011	$\Delta t = 4$	5	2	3
1101101	$\Delta t = 5$	3	4	-1
1110110	$\Delta t = 6$	3	4	-1

Cross-Correlation*:* Instead of comparing the shifted copy of the sequence to itself, this scenario compares two sequences, and the degree of cross-correlation obtained by matching one sequence to another random sequence. The relationship between two sources 'A' and 'B' is described as a cross-correlation. Equ. (3.2) denotes the cross-correlation function $R_{A,B}(\tau)$, where 'τ' denotes the time lag between the functions, and *A** denotes the complex conjugate of source 'A'.

$$R_{A,B}(\tau) = \sum_{-\infty}^{\infty} A * (\tau)\, B(m + \tau) \quad \text{where } \tau = 0, N, 2N \tag{3.2}$$

3.2 Non-Interfering User Signal Codes

When user signals are sent over a CDMA network, a unique PN code is assigned that serves as a "key" in the receiver. The sequences of the same length outperform others in terms of cross-correlation. PN sequences to be used for coding should be orthogonal to each other. However, this is not always the case. Maximal length sequences, also known as m-sequences, are a small subset of potential approximated m-Sequences. A Gold code is formed by combining two

distinct sequences using an XOR gate. Selecting PN sequences with a low cross-correlation value is critical for mitigating the effects of co-channel interference, it is also referred to as co-channel noise. Although much research has been conducted on tiny cross-correlation PN sequences, the code sets developed by R. Gold, Walsh, and Kasami are still used today in the systems such as UMTS W-CDMA and IS-95. PN generators and LFSRs are frequently specified in technical system specifications as codes that are created using two polynomials of degree 'n'.

3.3 Linear Feedback Shift Register

The LFSR is a shift register in which some of its outputs are grouped in XOR configurations to create a feedback channel [26].

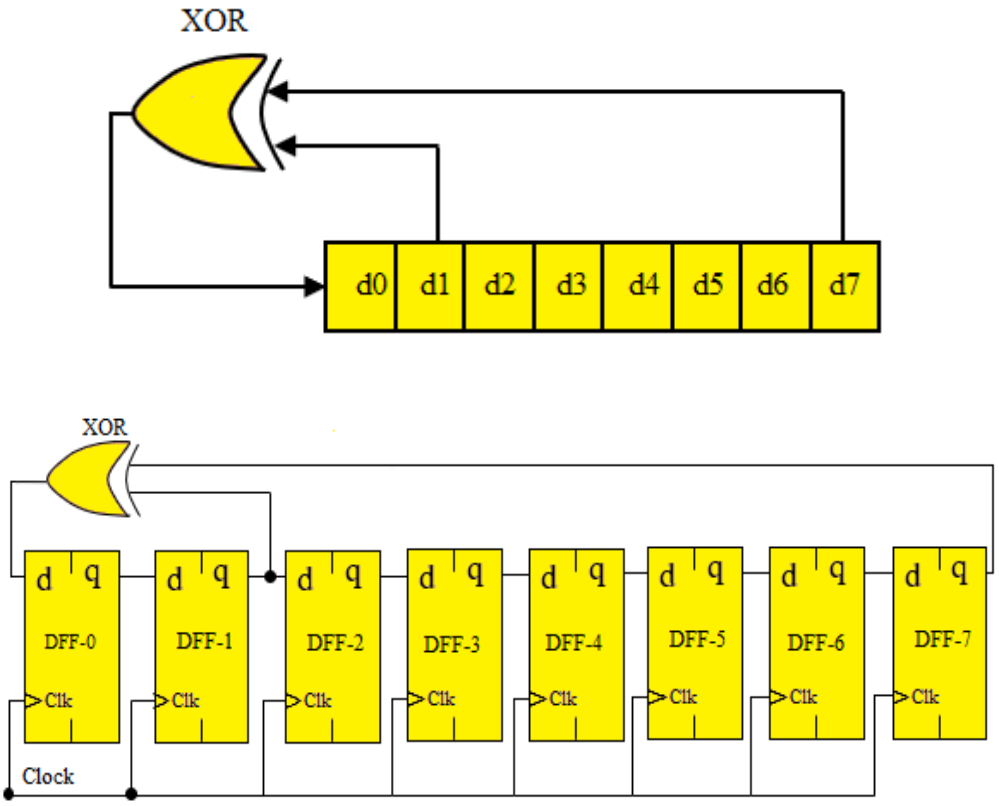
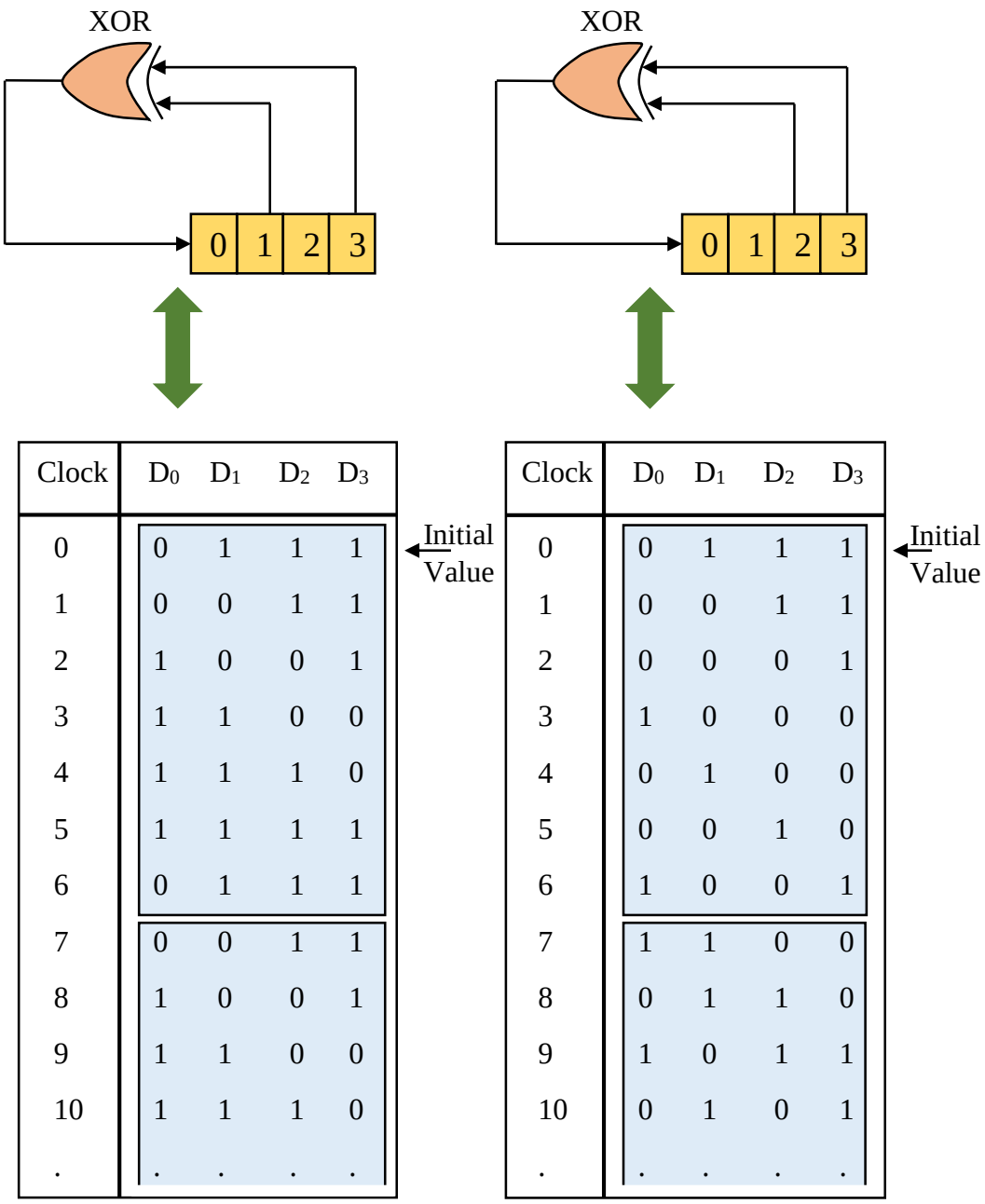


Fig.3.1. The circuit diagram and symbol of LFSR (8-bit)



Clock	D_0	D_1	D_2	D_3
0	0	1	1	1
1	0	0	1	1
2	1	0	0	1
3	1	1	0	0
4	1	1	1	0
5	1	1	1	1
6	0	1	1	1
7	0	0	1	1
8	1	0	0	1
9	1	1	0	0
10	1	1	1	0
.	.	.	.	.

Clock	D_0	D_1	D_2	D_3
0	0	1	1	1
1	0	0	1	1
2	0	0	0	1
3	1	0	0	0
4	0	1	0	0
5	0	0	1	0
6	1	0	0	1
7	1	1	0	0
8	0	1	1	0
9	1	0	1	1
10	0	1	0	1
.	.	.	.	.

Fig.3.2 Different tap selections and their comparison

LFSRs are widely utilized as pseudorandom pattern generators to produce a random assortment of '1' and '0'. High-toggle-rate configurations are generated while in LFSR test mode are excellent for producing high-fault coverage in communication devices. LFSR are easy to generate and can be used in multiple ways based on the system need and memory [128]. The logic is checked using a behavioral model from the design perspective. The most common category of an LFSR is shown in Fig.3.1. The 8-bit LFSR is a shift register having feedback

from two or more nodes or taps, in the register chain. In this case, the taps are at bit-1, and bit-7 position, which is written as (1,7). The All-register elements have the same input for the clock, which is left out of the symbol to make it easier to understand. The DFF0, DFF1, DFF3, DFF4, DFF5, DFF6, and DFF7 denote the 8-bit data processing using D flip-flop respectively based on the synchronized clock signal. The LFSR out is configured based on the XOR or XNOR operation of tap bits. The rest of the bits operate normally like a shift register. The feedback function and the taps selection decide the order of the LFSR value. Fig.3.2 depicts two 4-bit XOR-based LFSRs having different feedback taps. The first LFSR has feedback taps at (1,3) and the second LFSR has feedback taps at (2,3). Initially, both LFSRs start with the same value i.e. (0111), later when the clock pulses are added, the sequences start to change quickly because their taps are different. An LFSR will cycle throughout a loop with only a few values if certain conditions are met. The LFSRs with maximal length as these pass through all the possible values (except all 0s) before going back to their starting values.

3.3.1 Seeding an LFSR

A strange thing about an XOR-based LFSR is that if it reaches the all-zero value, it will keep shifting all-zero values for as long as it wants. The fact that XORs are utilized as the sum constituent of the easiest form of the half adder allows us to see the XOR function as a component of the digital addition operation. The shift registers subsequent bits and the overall contents are compared using XOR logic to control the input bit of an LFSR. Similarly, for an XNOR-based LFSR, it will keep shifting all the '1' values. Each register bit can start with either a logic '0' or a logic '1', so the prohibited value wakes up the LFSR. Therefore, the seed value is to be put at the beginning of an LFSR [129]. To load a seed value, registers having reset or set inputs can be utilized. Some reset inputs and set inputs of the registers can be coupled to the same control signal. When the control signal turns on, the LFSR is loaded with a seed value that is hard-wired into it. In some situations, it proves to help change the seed value. This can be done by adding a multiplexer to the input of the LFSR as shown in Fig.3.3.

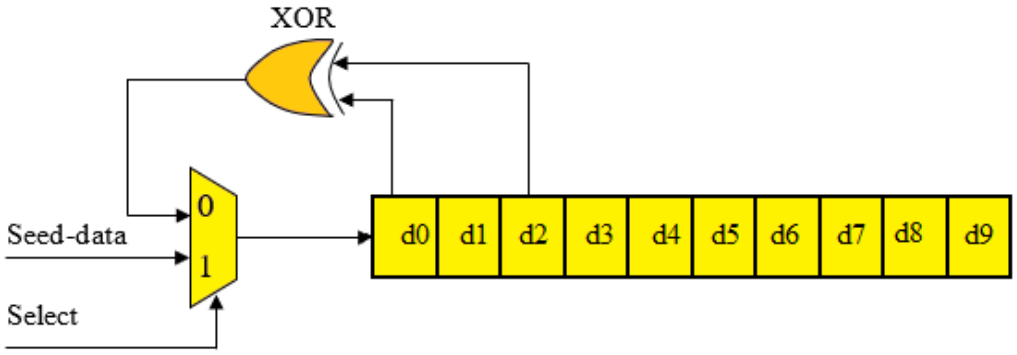


Fig.3.3 Seed values loading for a 10-bit LFSR.

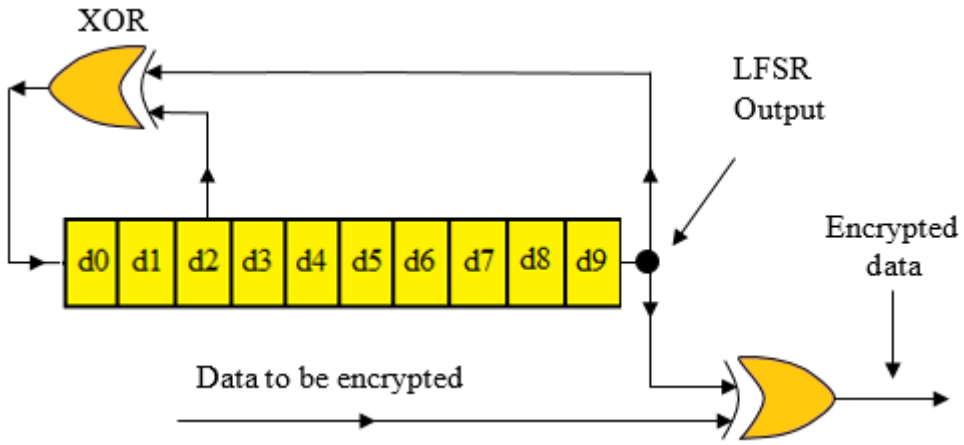


Fig.3.4 Encryption of data using 10-bit LFSR.

When the data input is chosen, the multiplexer acts like a normal shift register, and it can load any seed value. Once a seed value is loaded, feedback paths are chosen, and the device goes back to the LFSR mode. Using the strange order of values in an LFSR, data can be encrypted and decrypted. By XORing the bit stream with the output of an LFSR, one can encrypt data as depicted in Fig.3.4. One can decrypt a stream of the encrypted data by XORing them with the outcome of a similar LFSR. This method of encryption is very simple yet not very safe, but more cost-efficient and may be useful in some situations. It is applicable in secured communication such as wireless data communication and telecommunication switching.

3.4 Gold Code Sequence

The first use of gold code generators started late back in 1967 and is attributed to "R. Gold." The data generated by the two LFSRs can be combined using Modulo 2 additions to generate a set of tiny correlation PN codes. This process produces a sequence of gold codes that have a correlation behavior. A gold code sequence is obtained by multiplying the two sequential PN sequences. It shows the property of being consistent and has a low cross-correlation between two codes. The chosen pair of gold code sequences are generated by the temporary shift of one PN sequence. Minimum cross-correlation analysis is performed on the chosen pair of the gold code sequence, having the lowest cross-correlation. An m-sequence represented by an N-dimensional binary vector is considered to sample every ‘q’ symbol in sequence 'a' to generate an entirely new sequence a'. This procedure will be repeated until sufficient samples are gathered to generate an a' sequence of length ‘N’. If a GCD for a(n,q) is 1, then an ‘m’ sequence with a period of ‘N’ exists. This means that the sequences ‘n’ and ‘q’ have nothing in common except the number ‘1’.Two pairs of sequences are generally considered. These are both sequences with gold sequence characteristics: a' = a[q], a'' = a[q]. All the ‘n’ values excluding (0,4,8,12,...so on), are odd. Therefore, ‘q’ is (2k+1) or (2, 2k, -2k+1) for some values of ‘k’.

$$\text{GCD (n,k)} = \begin{cases} 1, & \text{for } n \; odd \\ 2, & \text{for } n \; mod \; 4 = 2 \end{cases} \qquad (3.3)$$

| R | 2(n+1)/2 + 1 limits the cross-correlation of a preferred gold sequence for a shift register of length ‘n’, and it limits the cross-correlation of a preferred gold sequence for a shift register of length 2. The logic used in the gold sequence generator simulation is as follows:

(i) These sequences act as a program component and begin with a fill or "seed." The LFSR algorithm is used to generate m-sequences, which are then used as program components. The PN Sequence generators will produce two signals, ‘pn_1’ and ‘pn_2’, and these two sequences can be used to get the signals.

(ii) Gold code Sequence is generated by XORing the previous step's outputs.

The data generated by the two LFSRs can be combined using Modulo 2 additions to generate a set of small correlation PN codes. This process produces a sequence of gold codes that have a correlation behavior [104]. A gold code sequence is obtained by multiplying the two sequential PN sequences. It has the advantage of being consistent and having a low cross-correlation between two codes. The chosen pair of gold code sequences are generated by the temporary shift of one PN sequence. Minimum cross-correlation analysis is performed on the chosen pair of the gold code sequence, having the lowest cross-correlation. Cross-correlation can be helpful for Gold code study. Let us consider

M = Number of the inputs bits of the Gold generator

L= Length of the sequences generated for the input Gold vector **G**

The sequence number and sampled and quantized at Q^{th} symbol.

The resultant Gold vector **G' = G[Q]**

The concept of common factor can be applied to estimate the maximum performance using generated common divisor (GCD), Then GCD (M, Q) = 1. Where M = Message input bits and Q and quantized data bits. The sequence can be divided into even and odd sequences, Then the correlation can be established using the following equations [30].

$$|R_c| \leq 2^{\left(\frac{M+1}{2}\right)} + 1 \qquad for\ odd\ sequences\ \ \mathrm{GCD}\ (\mathrm{M},\mathrm{Q}) = 1 \qquad (3.4)$$

$$|R_c| \leq 2^{\left(\frac{M+2}{2}\right)} + 2 \qquad for\ even\ sequences\ \ \mathrm{GCD}\ (\mathrm{M},\mathrm{Q}) = 2 \qquad (3.5)$$

The following even sequence and fixed and quantized even samples can follow a sampled sequence. For example, sampled sequence 0^{th}, 4^{th}, 8^{th}, 12^{th}, $16^{th.}$ …. etc. are considered as M mod 4 ≠0, and other values as M mod 4 = 2 is applicable to establish cross-correlation using LFSR shift register. These sequences act as a program component and begin with a fill or seeding the bits The LFSR algorithm is used to generate m-Sequences, which are then used as program components. The PN sequence generators will produce two signals against first

and second inputs and two sequences can be used to get the final output. Gold-Code Sequence is generated by XORing the previous step outputs. Table 3.4 lists the LFSR maximal-length sequence with 10-bit input data “1101100110”.

Table 3.4 LFSR maximal length sequence with 10-bit input data “1101100110”

State	LFSR -1 data with (3, 10)	Output LFSR-1	LFSR -2 data with Tap (2,3, 6,8,9,10)	Output LFSR-2	Gold Sequence
0	1101100110	0	1101100110	0	0
1	0110110011	1	0110110011	1	0
2	1010011001	1	1000010010	0	1
3	1100001100	0	0100001001	1	1
4	0110000110	0	1001001111	1	1
5	0011000011	1	1111101100	0	1
6	1000100001	1	0111110110	1	0
7	1101010000	0	0011111011	1	1
8	0110101000	0	1010110110	0	0
9	0011010100	0	0101011011	1	1
10	0001101010	0	1001100110	0	0
11	0000110101	1	0100110011	1	0
12	1001011010	0	1001010010	0	0
13	0100101101	1	0100101001	1	0
14	1011010110	0	1001011111	1	1

15	0101101011	1	1111100100	0	1
16	1011110101	1	0111110010	0	1
17	1100111010	0	0011111001	1	1
18	0110011101	1	1010110111	1	0
19	1010001110	0	1110010000	0	0
20	0101000111	1	0111001000	0	1
21	1011100011	1	0011100100	0	1
22	1100110001	1	0001110010	0	1
23	1111011000	0	0000111001	1	1
24	0111101100	0	1011010111	1	1
25	0011110110	0	1110100000	0	0
26	0001111011	1	0111010000	0	1
27	1001111101	1	0011101000	0	1
28	1101111110	0	0001110100	0	0
29	0110111111	1	0000111010	0	1
30	1010011111	1	0000011101	1	0
31	1100001111	1	1011000101	1	0
32	1111000111	1	1110101001	1	0
.	.	.	.	.	.
.	.	.	.	.	.
1022	1001001101	1	1101011011	1	0
1023	1101100110	0	1101100110	0	0

3.4.1 Implementation of Gold code using 10-bit linear feedback shift register

Gold sequences aid in the generation of additional sequences from a pair of m-sequences, providing a much broader range of sequences to a large number of users. Only one sequence of sufficient length emerged from the m-sequences. Combining two of these sequences with the two m sequences themselves yields sequences. Modulo-2 is used to produce gold codes that show strong cross-correlation behaviour [105]. Gold codes are generated by modulo-2 by adding the desired pair of sequences' specific relative phases (maximally length sequences). In terms of operation. Fig.3.5 depicts a schematic representation of a Gold code generator with n =10. Linear feedback shift registers are used in conjunction with a shift register for each of the two m-sequences (LFSR). The diagram above depicts a Gold code sequence processing using a shift register with the length 10-bit, capable of generating output stages ($2^n - 1$). The shifting of the input data sequence with feedback taps (2, 3, 6, 8, 10) and feedback taps (3, 10) is listed in Table 3.5 and Table 3.6, and the corresponding output is in Table 3.7. The generated polynomial for LFSR-1 and LFSR-2 are given as

$$F_1 = 1 + p^2 + p^3 + p^6 + p^8 + p^9 + p^{10} \tag{3.6}$$

$$F_2 = 1 + p^3 + p^{10} \tag{3.7}$$

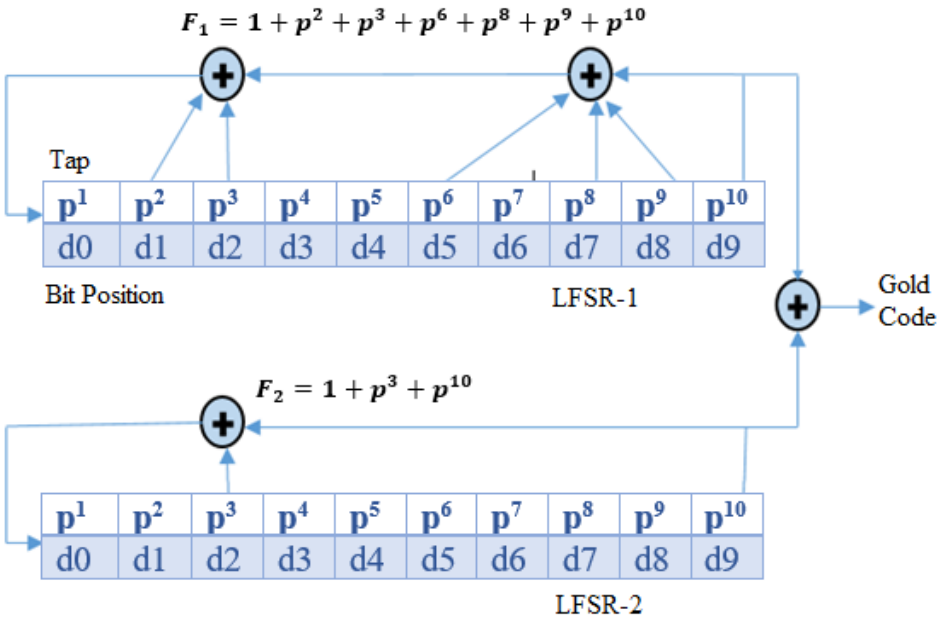


Fig.3.5 Gold code generator

Table 3.5 LFSR maximum-length sequence with (2, 3, 6, 8, 9, 10) feedback taps

	State change in the shift register										Output Symbol
	d0	d1	d2	d3	d4	d5	d6	d7	d8	d9	
Input Sequence	**1**	**1**	**0**	**1**	**1**	**0**	**0**	**1**	**1**	**0**	
Feedback											
1	1	1	1	0	1	1	0	0	1	1	1
1	1	1	1	1	0	1	1	0	0	1	1
0	0	1	1	1	1	0	1	1	0	0	0
1	1	0	1	1	1	1	0	1	1	0	0
0	0	1	0	1	1	1	1	0	1	1	1
0	0	0	1	0	1	1	1	1	0	1	1
0	0	0	0	1	0	1	1	1	1	0	0
1	1	0	0	0	1	0	1	1	1	1	1
1	1	1	0	0	0	1	0	1	1	1	1
1	1	1	1	0	0	0	1	0	1	1	1
0	0	1	1	1	0	0	0	1	0	1	1
.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.
1023	1	1	0	1	1	0	0	1	1	0	0

Coded sequence: 11001101111….0

Table 3.6 LFSR maximum-length sequence with (3, 10) feedback taps

	State change in the shift register										Output Symbol
	d0	d1	d2	d3	d4	d5	d6	d7	d8	d9	
Input Sequence	**1**	**1**	**0**	**1**	**1**	**0**	**0**	**1**	**1**	**0**	
Feedback											
0	0	1	1	0	1	1	0	0	1	1	**1**
0	0	0	1	1	0	1	1	0	0	1	**1**
0	0	0	0	1	1	0	1	1	0	0	**0**
0	0	0	0	0	1	1	0	1	1	0	**0**
0	0	0	0	0	0	1	1	0	1	1	**1**
1	1	0	0	0	0	0	1	1	0	1	**1**
1	1	1	0	0	0	0	0	1	1	0	**0**
0	0	1	1	0	0	0	0	0	1	1	**1**
0	0	0	1	1	0	0	0	0	0	1	**1**
0	0	0	0	1	1	0	0	0	0	0	**0**
0	0	0	0	0	1	1	0	0	0	0	**0**
.	.	.	.	.	.	.	.	.	.	.	**.**
.	.	.	.	.	.	.	.	.	.	.	**.**
1023	1	1	0	1	1	0	0	1	1	0	**0**

Coded Sequence: 1100110110….0. Table 3.7 lists the Gold code output for the input data sequence 1101100110 using LFSR 1 with feedback taps (2, 3, 6, 8, 9, 10) and LFSR 2 with feedback taps (3, 10).

Table 3.7 Gold sequence output

Stages	LFSR-1 Feedback taps (2, 3, 6, 8, 9, 10)	Output	LFSR-2 Feedback taps (3, 10)	Output	Gold Code output
1	1101100110	1	1101100110	1	0
2		1		1	0
3		0		0	0
4		0		0	0
5		1		1	0
6		1		1	0
7		0		0	0
8		1		1	0
9		1		1	0
10		1		0	1
.		.		.	.
.		.		.	.
1023		0		0	0

3.5 LFSR Implementation

The two types of LFSR implementation are Galois implementation and Fibonacci implementation.

3.5.1 Implementation of Galois

Fig.3.6 depicts the data flow from the left side to the right side, whereas the feedback path is displayed from the right side to the left side. The polynomials acquire values from the left side to the right side, starting with the p^0 in MSB and p^{16} in LSB. In mathematics, this polynomial is known as a tap polynomial, and it identifies which taps from the shift register should always be brought back to the computer. The generator LFSR polynomial is $\boldsymbol{F_1(p) = 1 + p^5 + p^{15}}$**.** The Galois implementation is referred to as modular type, M-type, or in-line LFSR implementation because the XOR gate is in the path of the shift register.

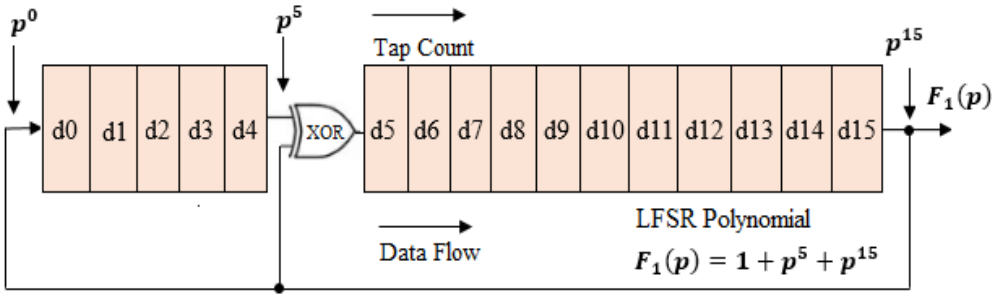


Fig.3.6 Galois Implementation

3.5.2 Fibonacci Implementation

The data flow in Fibonacci implementation occurs from the right side to the left side as depicted in Fig.3.7. In the Fibonacci implementation the generated polynomial decrements from the left side to the right side, p^0 is the final term of the LFSR polynomial. p^0 is the MSB p^{15} is in LSB, p^0 is known as the reverse tap polynomial, and the associated feedback taps are labeled sequentially from the right to the left side, indicating the direction of the shift registers. The generator LFSR polynomial is $\boldsymbol{F_2(p) = p^{15} + p^5 + 1}$. Since the XOR gate is located in the feedback route, the Fibonacci implementation is also referred to as a simple type (S-type) or out-of-line LFSR implementation.

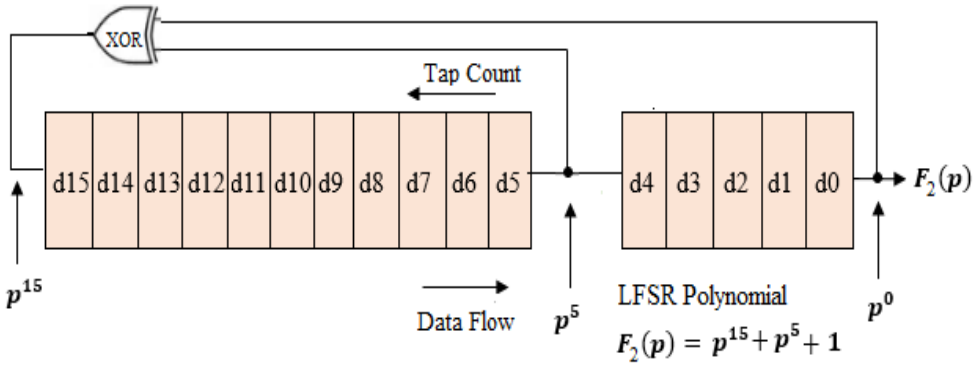


Fig.3.7 Fibonacci Implementation

3.6 LFSR and Gold Codes for Communication Technologies

In the present telecommunication systems, the transmission of data at high speed, voice quality, and expansion of broadband services are the most challenging areas for improvement. Thus, the growth of technology from the first generation to the third, third to fourth, and now the fifth generation has

experienced a dynamic change in data transmission rate [41]. Another important aspect of any wireless communication system [42] is its reliable data transmission means data transmission and reception should be of the same quality. During transmission, the data is subjected to noise or distortion when transmitted via the channel. The noise corrupts the data [43] due to which failure in data transmission occurs. Therefore, error detection and error correction [104] play an important role in data communication. For this purpose, different channel coding techniques are used. Some of these coding methods are LDPC, convolution, polar, and turbo codes [107].

The performance of these codes can be analyzed by the expression of its bit error rate, complexity, functionality, and hardware device utilization. In the late 1940s, a researcher named Shannon gave a concept describing the theoretical limit [124] of maximum capacity for data transmission in any communication channel. This channel capacity limit has now become the goal for any communication system. To better understand the need for this theoretical limit, the modern mobile industry can be considered. As the number of users is emerging enormously day by day the need and the complexity of the application are also increasing. For example, earlier mobile phones were only used for making calls or text messages but now we can access multiple features like sending pictures, and videos, and using the internet. This technological advancement demands a higher data rate.

As the number of consumers is emergent to access the existing channel different multiplexing techniques are used such as TDMA, FDMA, and CDMA. However, these methods do not increase the rate of data transmission. To increase the data rate one way is to increase the bandwidth of the passage but this reduces the number of available channels thus, this method does not solve the problem. Another way is by improving the coding scheme so that the existing channel can be utilized to its full strength. This allows more channels to accommodate a particular range of bandwidth. The proposed work stated turbo codes based on an iterative decoding scheme. The encoder for turbo codes uses two convolution encoders connected in equivalent and detached by an interleaver. Moreover, the receiver consists of the serial concatenation of

convolution decoders. These decoders use the data from the previous decoder for the error-controlling process.

The whole iteration process can be repeatedly run for one received codeword until further improvement. For the iterative decoding mechanism, two algorithms are present: (i) SOVA and (ii) MAP algorithm [57]. Earlier ASICs were considered the standard structure for implementing wireless transmitters and receivers. For multiple wireless standards, devices require separate ASICs for each wireless standard which makes the circuit more complex, costly, and power-consuming. The wireless standards have limited use of fixed ASIC and demand a new flexible architecture. The flexible architectures share hardware for common wireless standard features. Therefore, constraints like reduced chip area, complexity, and low power consumption can be optimized while using flexible architecture.

Various reconfigurable architectures, such as FPGA [48] are now available with user-programmable logic blocks that may be altered based on the requirements or applications. These architectures are attractive solutions to get the desired functionality with relatively high flexibility. The error correction method [59] is incorporated into a wireless communication system for reliable data transmission in channels subjective to noise. The encoding is performed by the transmitter based on a particular encoding scheme. This encoding adds some redundant bits/ data in the data sequence and at the receiver end, these encoded bits are separated for reliable detection of the data sequence. In any wireless channel whenever data bits are transmitted they are subjective to the noise that causes the error. Forward error correction codes can be used to detect and rectify this problem. This work is accomplished by encoders such as convolution or block codes, which add redundant bits to the data sequence. If an error occurs during transmission, the decoder detects it and corrects it at the receiver.

3.7 Gold Code Sequence Generation Using LFSR

The LFSR is essentially a shift register that moves data from one register to the next with each clock signal [51]. Linear feedback is formed by XORing some bits of the shift register to drive the input values of the LFSR. The selected bits

in the preceding states influence the input bits and are referred to as taps. The feedback loop is formed by XORing a series of such taps that are dependent on the feedback polynomial. A lot of work is carried out in the direction to make the reconfigurable LFSR which can be future utilized for gold code, different communication systems, and switching applications. Re-configurability and speed are the essential properties to make the system suitable for coding and communication. Re-configurability refers to re-programmability that provides the platform to the reprogrammable generators with significant capabilities to control LFSR operations at the same speed [63]. The extraordinary advancements in optical components and devices in current years have resulted in major efforts in implementing optical signal processing methods. LFSR played an important role in classical electronics, such as circuits that generate encryption and decryption, PRBS, and a bit error rate measuring system [53], [54]. Optical LFSRs, which have the advantage of speed, are predicted to provide benefits in similar types of applications. Optical fiber shift registers [65] with the integration of XOR gate [61], [67] and SOA, having low transferring energies and advanced switching speed(> 40Gb/s) proved to achieve this goal. The enormous length of optical shift registers limits the usefulness of optical LFSRs because of the complexity of programming while generating the regulated PRBSs.

In the scheme of grouping numerous concurrent transmissions for a multichannel network using the same physical link, the bandwidth of optical fiber can be utilized properly. In a network setting, this goal can be met by employing multiple access techniques, like CDMA and FDMA probably in conjunction with logical communication techniques. CDMA [58] has been used to address multiple users in a rational optical network, while [59] proved the practical viability of a spreading dispreading [50] electro-optic modulator. Gaussian approximation was proposed [41] for the random variable in the deficiency of noise to solve the interferer effect and the performance of a CDMA system with gold sequences is evaluated. The flaws in both assumptions have been thoroughly examined in recent papers for optical and wireless communication. In the disparity to the particulars of electro-optical conversion, it is an essentially random behavior that can be experienced in the presence of

shot noise. The non-existence of noise provides error-free transmission in the case of a small population of interferers. The feedback parameters are acquired by a matrix transformation from the control LFSR output. The disparity between the controller and controlled FSR in the design provides the period benefits over the suggested approach of varying feedback shift resisters [52]. The ability to use arbitrary feedback functions for the controlled FSR and an arbitrary controller additionally increases our design's degree of freedom. Variable feedback has been implemented earlier. Jansen and Boekee have also suggested this. Their switch-controlled feedback sequentially moves a single linear tap. Concerning the Roggeman scheme, there is no change in speed between the control and the reconfigured FSR is confined to an LFSR. As a result, complexity is highly promising for a simple method.

Decimation has been used to create variable characteristic polynomials in a variety of ways, the simplest of which is the stop-and-go generator, in which the output of one LFSR controls the clock of another. For example, when the output of a controlling LFSR is ‘1’, an LFSR may step through one cycle of its clock, but not when the output is ‘0’. The study examined several clock controlled LFSR designs "including cascades of LFSRs driven by the same clock source. Due to the necessity that each LFSR has several gates, this requirement may be a challenge for optical logic implementation due to gate count constraints. Another type of clock-controlled sequence is the LFSR-based shrinking generator, but these generators are insecure when the feedback functions contain a small number of nonzero terms. However, optical memory is built by circulating bits rapidly through fiber and has few points of access for users. The feedback function will utilize the majority of bits in the register, i.e. the optical memory, at any one time. As a result, the feedback function will contain a high proportion of zeros, which is precisely the limitation one wants to avoid in the taps and the feedback function. The speed differential concept was discussed between two FSR [59], in which the outputs of two LFSR operating at different speeds are combined nonlinearly but the speed difference between the two sequences ‘m’ is not reconfigured. This fiber has previously been demonstrated to be insecure. The approach is different from the reconfigurable FSR in the absence of a sequence-controlled FSR.

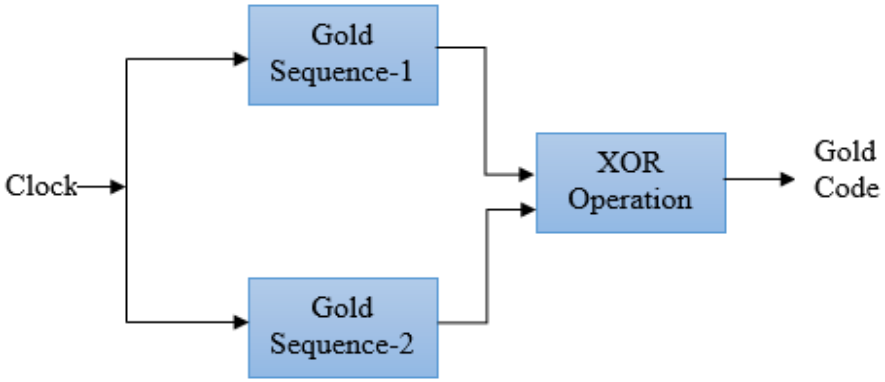


Fig.3.8. LFSR and gold code output

The gold sequence generator is using the two identical input sequences of LFSR output which are XORed. The two input sequences should have the same phase connection until all of the additions are made. A completely different gold sequence occurs from even a slight phase modification in one of the input sequences. This behavior is depicted in Fig.3.8. LFSRs are frequently used in stream ciphers to produce pseudorandom sequences which are further used in data security encryption and decryption. The novelty of the study is that it offers a platform for chip designers and researchers to replicate the capability of scalable LFSR and gold sequence generator on high-end FPGA, ensuring the integration of the same chip for the next-generation systems, such as 5G and 6G. The data size of the LFSR used in the simulation is 5 bits. Therefore, the data path size is 5-bit as one-bit storage applies to one flip-flop. The objective of the work is to perform the simulation for the same functionality.

3.8 LFSR Serial Algorithm

The serial implementation of LFSR [44, 45] is shown in Fig.3.9. It can be noted from the system's behavior that the serial architecture has two drawbacks: first, every clock causes the entire structure to become time-locked, and second, only one information bit is produced. It is possible to overcome these issues by utilizing a parallel design in which one or two cells are synchronized at the same time. LFSR circuit has many connections between its structure and the XOR tree because all the taps [26] are connected to the flip-flops for the circuit to function properly. Fig.3.10 depicts an example of parallel LFSR architecture.

Individual cells in the register remain unchanged in this configuration unless updated once in every 'N' clock cycle. Taps generally move in the direction [37] following the preceding cell with each clock cycle, but the bits contained within the memory elements remain unchanged.

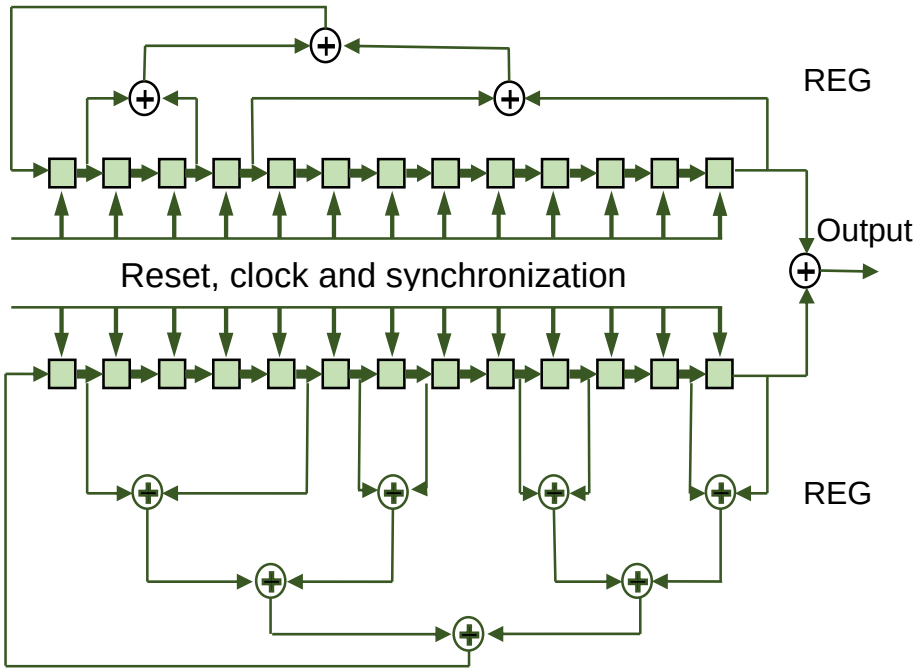


Fig.3.9 The basic principle of serial LFSR

As a result, the previously connected tap to cell number 6 has been moved to cell number 5, and the tap previously connected to cell number 5 has been moved to cell number 4, while the tap earlier connected to cell number 1 (initial cell) has been moved to the cell number 7. The tap initially connected to cell number 1 (initial cell) has been reconnected to cell number 7 (last cell). It is similar to the impact of bits being redistributed to the next higher cell in traditional architecture. The control unit [38] instructs the switches to update every cell in sequential order with the values which are obtained from the XOR tree on each clock cycle. The array memory elements only transition once as a result, and the remaining FFs stay static and use no power. While this architecture consumes more hardware than traditional architecture, it switches less circuitry per clock cycle, resulting in lower power dissipation. A large number of components are required in this architecture, as each XOR tree tap must first scan the entire array and it should be connected to each cell, totaling N(cells number) x M switches (taps number). Besides this, the total number of

components can be reduced to (N + M). Only one transistor is required to physically implement each switch.

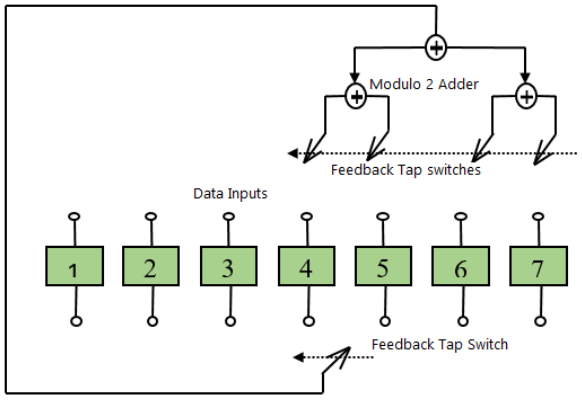


Fig.3.10 Basic Principle of the parallel LFSR

3.8 Method for Gold Codes

Gold codes [12] are created by XORing two maximum-length sequences having the same length. These sequences are added to the computer chip using synchronous clocking. The difference in the phase position of two generated m-sequences results in the generation of a fresh sequence. A Gold code is produced using specially chosen m-sequences, also known as preferred m-sequences. An LFSR has two (2n - 1) states [29], [20], where ‘n’ is the LFSR's register count. At the starting of each clock edge, the data of the shift registers is moved one position to the right [1]. Preset registers or register taps are used in conjunction with the register to provide feedback. These will use an XNOR or XOR gate to provide feedback to the leftmost register. When employing XOR feedback, it is forbidden to have a value of all 1s or all 0s because doing so would result in the counter becoming trapped in this state. LFSRs are used to convert a narrowband signal to a wideband signal, which generates pseudo-noise in the process. The feedback function of an LFSR is one of the device's two fundamental components, the other being the shift register. Shift registers are primarily used to relocate register contents into neighboring positions inside a register, or out of the register if the position is at the end of the register. The area will remain empty unless fresh content is provided to the register's other end.

In Gold code generators, LFSRs are the most fundamental functional block. When ‘N’ registers are used in the LFSR, it cycles through (2n-1) states, which

is a prime number. As shown in the diagram, each clock cycle causes the contents of the registers to be moved from one position to the right. The right most register is produced by XORing the left most register with feedback from predetermined registers or taps. When designing LFSRs, the number of taps in each feedback route, as well as the number of steps in the shift register, must be considered. Another variable is the shift register stage or the location of each tap. To function efficiently, a class of spreading codes in a multiple-access system must have the minimum possible value of mutual interference. The technique uses the gold subclass of PN sequences [31]. These are chosen in such a way that the cross-correlation scores between the codes in a specific set are uniform and bound across the entire set. The addition of the code sequences occurs one at a time using synchronous clocking. The length of the generated codes is equal to the length of the two m-sequences added together. Consider the following case: $g_1(X)$ and $g_2(X)$ are assumed to be the desired primitive polynomial pair of degree 'n', in which the respective shift registers produce a maximal length sequence of the period 2(n-1), and its cross-correlation function is less or equal to 1.

$$2^{\frac{n+1}{2}} + 1 \qquad \text{for n is odd} \qquad (3.8)$$

$$2^{(n+2)/2} + 1 \qquad \text{for n is even} \qquad (3.9)$$

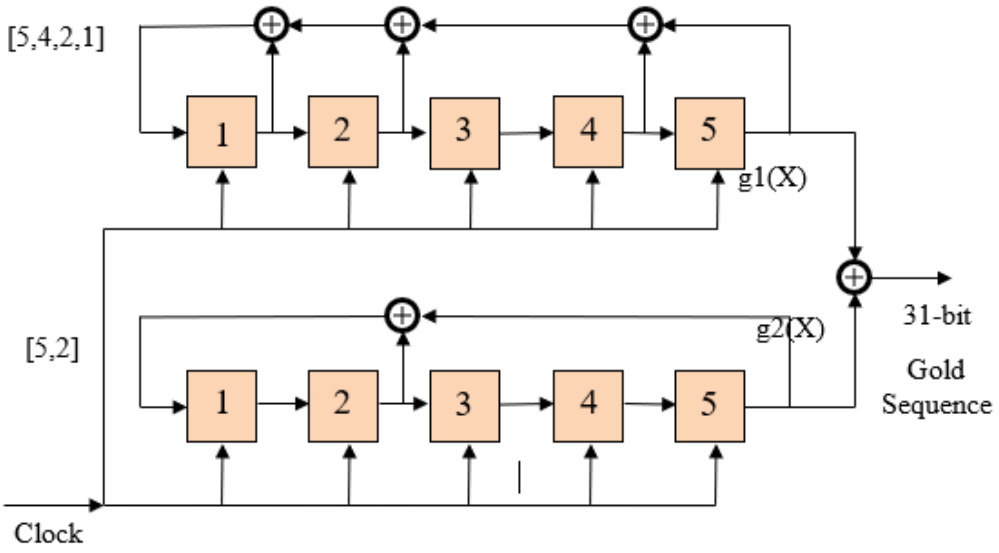


Fig.3.11 Gold code sequence generator with shift register of length m = 5

If the preceding condition is met, $g_2(X)$ will produce 2(n+1) distinct sequences, each having a duration of 2(n-1). The process of making Gold set is featured in the illustration in Fig.3.11 follows the Gold Code sequence generator with a shift register of length m = 5. LFSR is used to reduce the length of a sequence to its shortest possible length [21], [22]. The length of the shift register is m = 5. The feedback taps are chosen from the sets (5, 2) and (5, 4, 2, 1) as shown in Table 3.8 below. Table 3.8 illustrates the evolution of maximal length sequences, where the generator goes back to its initial state after 31 iterations, assuming the initial state is 10000.

Table 3.8 The maximum length sequence generation using a feedback-shift register with 5-bit input data 10000.

State	LFSR-1 Tap (5,2)	Output LFSR-1	LFSR-2 Tap (5,4,2,1)	Output LFSR-2	Gold Sequence
0	10000	0	10000	0	0
1	01000	0	01000	0	0
2	00100	0	00100	0	0
3	00010	0	00010	0	0
4	00001	1	00001	1	0
5	10100	0	11101	1	1
6	01010	0	10011	1	1
7	00101	1	10100	0	1
8	10110	0	01010	0	0
9	01011	1	00101	1	0
10	10001	1	11111	1	0
11	11100	0	10010	0	0
12	01110	0	01001	1	1
13	00111	1	11001	1	0
14	10111	1	10001	1	0

15	11111	1	10101	1	0
16	11011	1	10111	1	0
17	11001	1	10110	0	1
18	11000	0	01011	1	1
19	01100	0	11000	0	0
20	00110	0	01100	0	0
21	00011	1	00110	0	1
22	10101	1	00011	1	0
23	11110	0	11100	0	0
24	01111	1	01110	0	1
25	10011	1	00111	1	0
26	11101	1	11110	0	1
27	11010	0	01111	1	1
28	01101	1	11010	0	1
29	10010	0	01101	1	1
30	01001	1	11011	1	0
31	10000	0	10000	0	0

The design technology of the work is as follows. The chip functionality follows the steps of the chip design, and simulation modeling. The VHDL-based programming is used to design the LFSR-1 chip that accepts the 5-bit input stream with the tap sequence (5, 2). After that, the LFSR-2 chip is designed that also accepts the 5-bit input stream with the tap sequence (5, 4, 2, 1). The waveform simulation is carried out for both the LFSRs in which the test cases are verified. The real-time verification is done on the FPGA board to check the feasibility of the design. The methodology for the same is given in Fig.3.12.

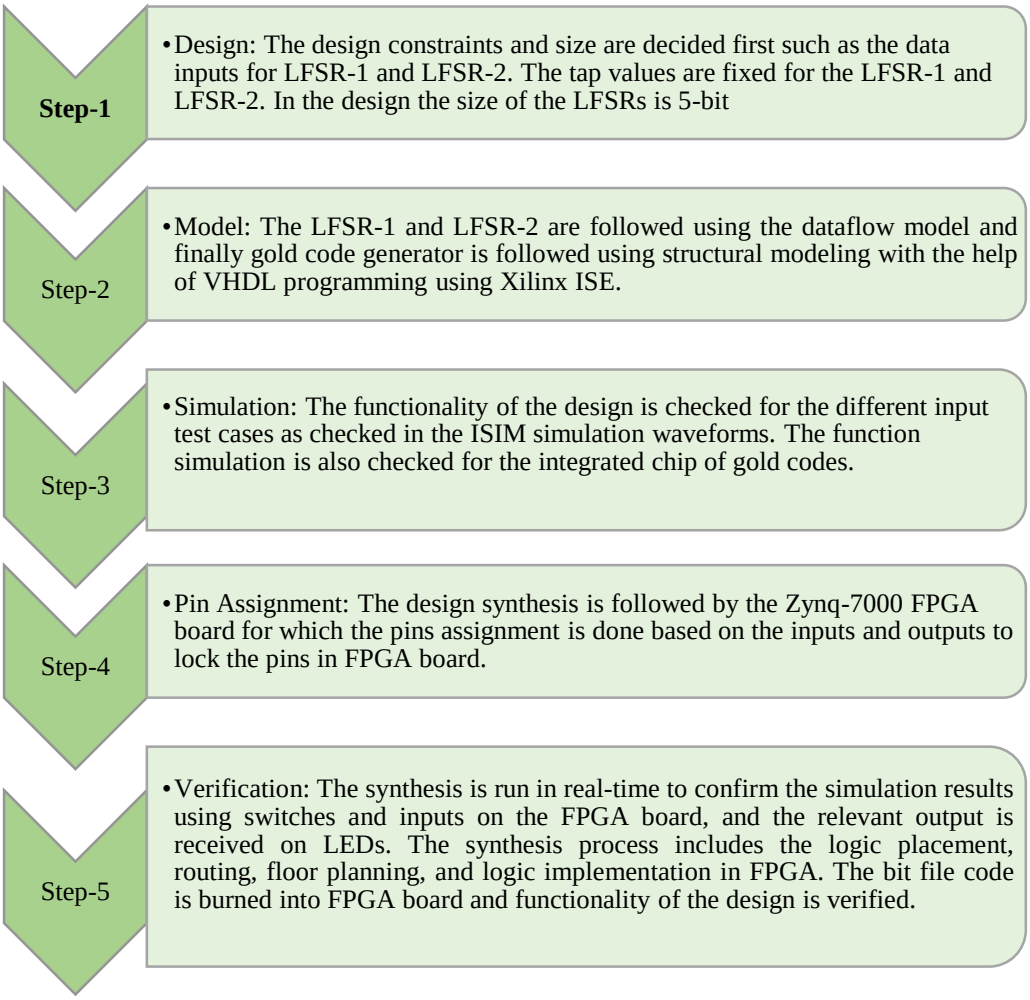


Fig.3.12. LFSR simulation methodology

CHAPTER 4

TURBO AND LDPC CODES

Chapter 4 explains the block diagram of an FEC communication system. The concept of Turbo and LDPC codes is introduced. It also covers the encoder and decoder concept and the entire research methodology for designing the structure of the Turbo and LDPC encoder-decoder.

4.1 Forward Error Correction Communication System

Fig.4.1 presents an FEC communication system block diagram that shows the data transmission from sender to receiver. The first step from the source is data encoding with the help of an encoder and performing modulation using a modulation method and then transmitting the data over the AWGN channel. During this transmission, noise can be added to data through the channel. This data along with noise is provided to the demodulator where the carrier signal is separated from the message signal and the data is decoded back to its original form using an error-correcting decoder at the receiver end. The two most significant blocks in the wireless communication system block diagram are the encoder and decoder.

Recent communication systems are facing difficulty to satisfy the need and requirements of users. The next generation mobile systems are expected to be 5G Era with improved channel coding schemes. There are multiple channel coding techniques available such as turbo, polar codes, LDPC, non-systematic convolutional, and systematic codes [81, 82]. The codes can be analyzed based on their complexity, reliability, flexibility, and latency as per system requirements. In cellular systems, channel coding methods are used to ensure reliable transmission by offering minimum errors during transmission. Wireless systems have shown remarkable development in satisfying user demands for the

last few years. In the current scenario, the coding methods available for 5G mobile systems are LDPC, turbo codes, and polar codes.

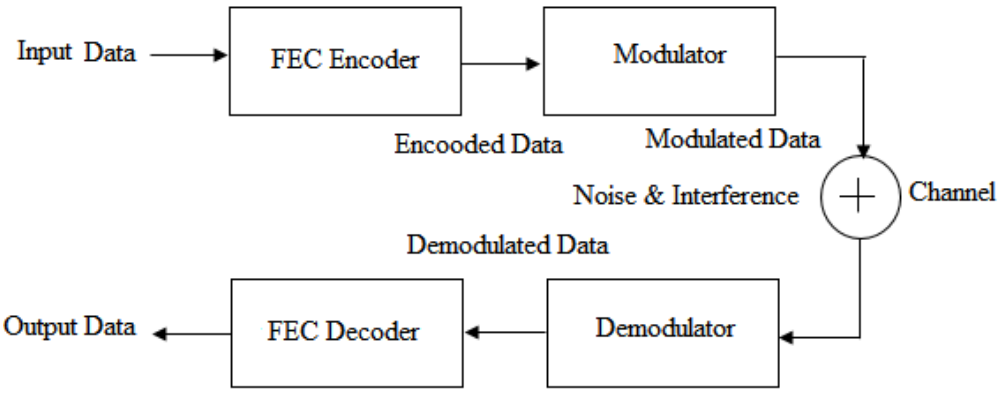


Fig.4.1 Block diagram of FEC communication

Turbo codes are high-performance FEC codes invented by C. Berrou [84, 85] in 1993. The performance of these codes is close to SHA. The theorem presents the channel capacity (C) in bits/s, to transmit information at a faster rate without error over channel bandwidth (B). The channel capacity, $C = B \log_2 (1+S/N)$, in which S/N denotes the SNR. The coding community was excited to provide promising solutions about channel capacity using an iterative decoding method based on simple constituent code. The coding block consists of the interlevel and the code length is long and random, which can emulate the performance of the system. The convolution turbo code is formed by parallel concatenated recursive convolution codes, separated by an interlever.

4.2 LDPC Codes

The LDPC codes were initially introduced in the 1960s by a scientist named Gallager [82, 84]. This discovery was unnoticed by many investigators for up to 20 years, but in 1981, a scholar by the name of Tanner gave a novel understanding of these codes from a graphical point of view. LDPC codes based on the Belief Propagation Algorithm (BPA) [86] have shown a channel's capacity performance close to Shannon's limit [87]. Consequently, LDPC codes proved powerful relative to other classes of linear block codes named turbo codes for error corrections where reliability is the concern. On the other hand,

turbo codes are very efficient in terms of their performance reaching Shannon's limit [60]. LDPC code [147] is a category of linear block code, which is comprised of a parity check matrix represented by 'H', which has less number of non-zero elements in every column and row. Such codes can be irregular as well as regular, and this entirely depends on the number of once available row-wise and column-wise. If the parity check matrix consists of an equal number of one's in every row and column, then it is called a regular LDPC code. It means column weight (w_c) is equal to row the weight (w_r). If the no. of 1's in each row and column of the matrix is unequal, then it is called an irregular LDPC code.

These codes are called low-density, as the number of 1's is lesser than the number of 0's. At present communication systems like DVB-S2 (802.11n), and mobile WiMax (802.16e), are widely utilized. LDPC codes can be represented in the form of (n, w_c, w_r), here 'n' stands for code length, 'w_c' stands for the weight of the column, and 'w_r' means the weight of the row. The parity checking matrix 'H' is symbolized with the help of Tanner graphs and algebraic construction. The columns number in the matrix represents the variable nodes (V_N) or bit nodes in the tanner graph, whereas the rows number in the matrix represents the check nodes (C_N). The connectivity between these nodes is shown by logic '1' in the matrix. The equ. (4.1) represents the 'H' matrix and is given as

$$H = \begin{bmatrix} C_{N1} \\ C_{N2} \\ C_{N3} \\ C_{N4} \\ C_{N5} \end{bmatrix} \begin{bmatrix} V_{N1} & V_{N2} & V_{N3} & V_4 & V_{N5} & V_{N6} & V_{N7} & V_{N8} & V_{N9} & V_{N10} \end{bmatrix} \quad (4.1)$$

$$H = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \end{bmatrix} \quad (4.2)$$

Fig.4.2 represents the tanner graph equivalent to the matrix 'H'. One method for LDPC encoding [122] is the pre-processing in which for a given 'H' matrix a generator matrix is developed that is used for encoding some random input (m) having size (1 × m). The other encoding method utilizes the parity check

matrix directly and thus, the method is less complex compared to the first method. The decoding part involves a bit-flipping procedure for hard decision channels and a message-passing procedure for soft decision channels. The message-passing procedure comprises the passing of messages in the backward and forward direction between checks nodes and variable nodes till no more iteration is required.

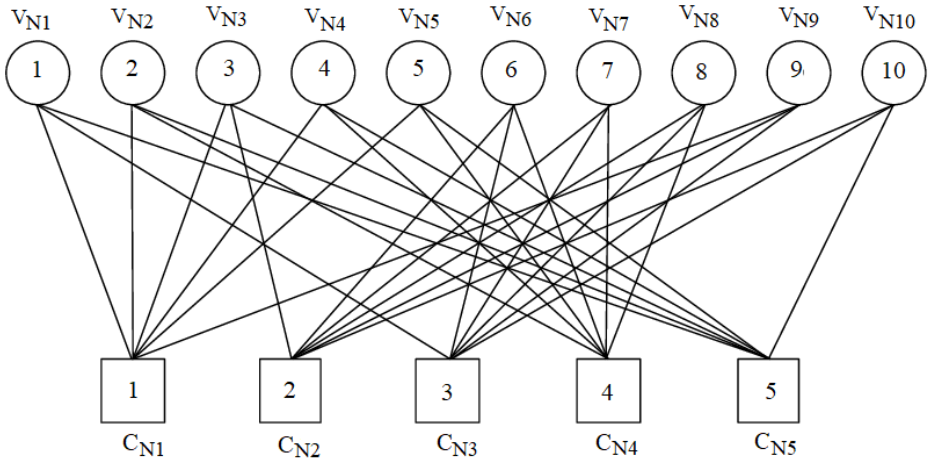


Fig.4.2 Tanner graph example using ‘H’ matrix

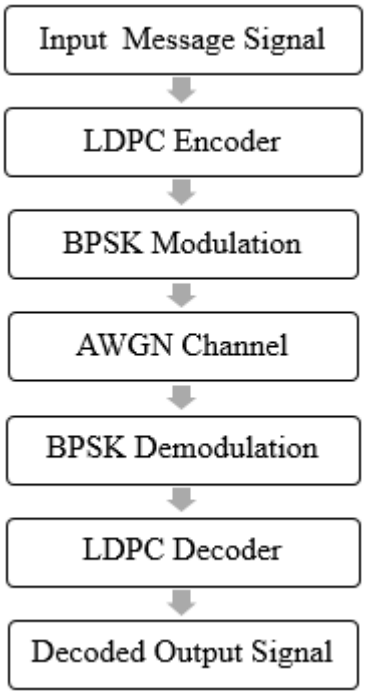


Fig.4.3 LDPC coding method

The LDPC coding process is shown with the help of Fig.4.3. The transmitter depicts the random data bits which are modulated using a BPSK modulator. Channel encoding is achieved using an LDPC encoder and the encoded data then passes across the channel. BPSK demodulator is used to perform the demodulation and LDPC decoder [128, 129] decodes the signal back into its original form. The length of the codeword plays a significant role in the whole encoding process. The encoding process mainly consists of two steps: creation of a code-word using sparse matrix and the formation of the sparse parity check. LDPC decoding involves iterative algorithms like the belief propagation algorithm [126]. A "Min- Sum" approach is adapted for implementing the hardware of the belief propagation algorithm. In this method, once the node checking occurs, the next step is updating the data node information. Once the decoding process is over a hard decision is done depending on the probability of the code word.

4.2.1 LDPC encoder and decoder

These codes are used as cyclic codes considering the exceptional case to abstract the parity check codes. The generation of 'H' is performed using the formation of block circulant code. The block circulant code is considered here because of more accuracy in error correction also, it provides an organized construction of the decoder. The 'H' is generated using a binary matrix in which each row is formed by means of a cyclic shift operation to the right in its previous row. The configuration of 'H' with a size (rT × nT) is achieved by performing the concatenation of a (r × n) light circulant with a size of (T × T). The Tanner graph corresponding to 'H' is referred to as a protograph. In the design, the photographs [115] to be used are AR4A and AR3A having a code rate of ½, and are shown using Fig.4.4 (a) and Fig.4.4 (b).

The circle in Fig.4.4 signifies the variable nodes and the squares in the figure denote the parity check nodes. The open circle denotes the penetrated symbols and the solid circle denotes the communication symbols. Here three level decoding process is considered: accumulate, re-occurrence by 4 or 3, and then accumulate. The Protograph denotes a circulant 'H' of (3 × 5) where the degree of consistent circulant is denoted through the number of parallel ends. These

protographs cannot be directly extended without low weight codewords, irrespective of the optimal selection of circulants. Therefore, protographs can only be extended twice, but with small variation, having a size of (4 × 4) and (8 × 8). The complete code can be constructed using these circulants. For the formation of the AR4A protograph, the check nodes are placed in the order of (C_{N1}, C_{N2}, C_{N3}) and variable nodes in the sequence (4, 2, 1, 5, 3). In matrix representation, a dot is used for non-zero entries. These nodes are denoted using solid lines initially having a (4 × 4) permutation and then elevating to (16 × 16) circulants. The resulting construction is a block circulant of (12 × 20). This idea is demonstrated using the protograph example of AR3A and AR4A. In this case, the reorganization of the columns and rows in AR4A is done.

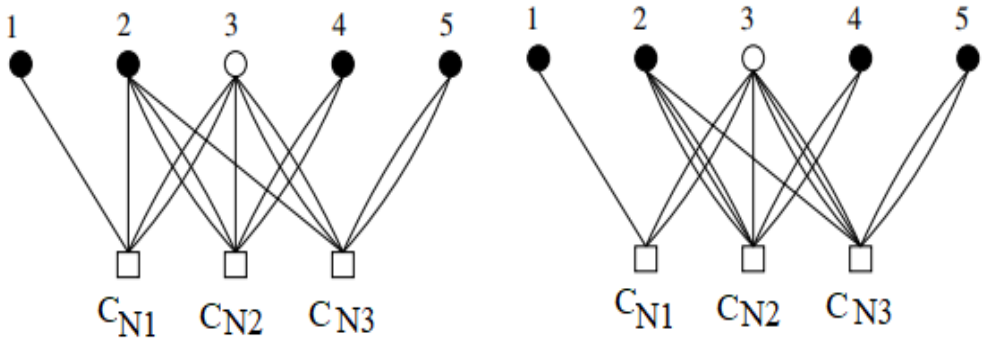


Fig.4.4 (a) Protograph (AR3A) (b) Protograph (AR4A)

Considering a sequence as (4, 2, 3, 1, 5), the base matrix 'H' for (C_{N2}, C_{N1}, C_{N3}) can be given as

$$H = \begin{bmatrix} 2 & 3 & 1 & 0 & 0 \\ 0 & 0 & 2 & 1 & 0 \\ 0 & 1 & 3 & 0 & 2 \end{bmatrix} \qquad (4.3)$$

4.2.2 Hardware Architecture

The design and chip implementation of the LDPC encoder is based on a block circulant matrix that can be created by the row-wise cyclic shift performed to the contents of shift registers. A matrix is a group of (n-k) encoder bits having a constraint length of 'T' that uses a recursive convolution approach. Fig.4.5 shows the feeding method where 'k' input bits are sequentially given to the

encoder and then with every shift, the data of the registers is updated sequentially.

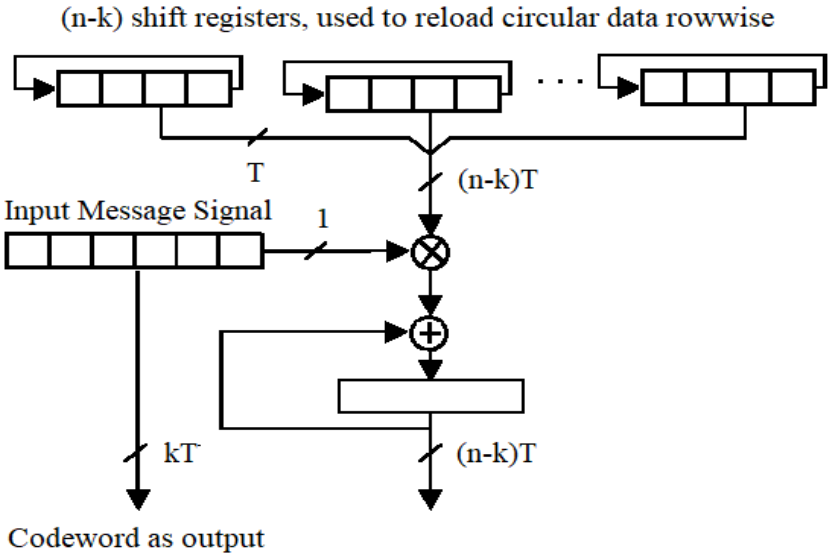


Fig.4.5 Row-wise circular data processing with (n-k) soft operations

Once the updated data is stored in registers, switches are changed, and data is read out from the registers along with the parity bits. The circular data is processed row-wise with (n-k) soft operations. Fig.4.6 presents the circular and shift operation to configure .the encoder design.

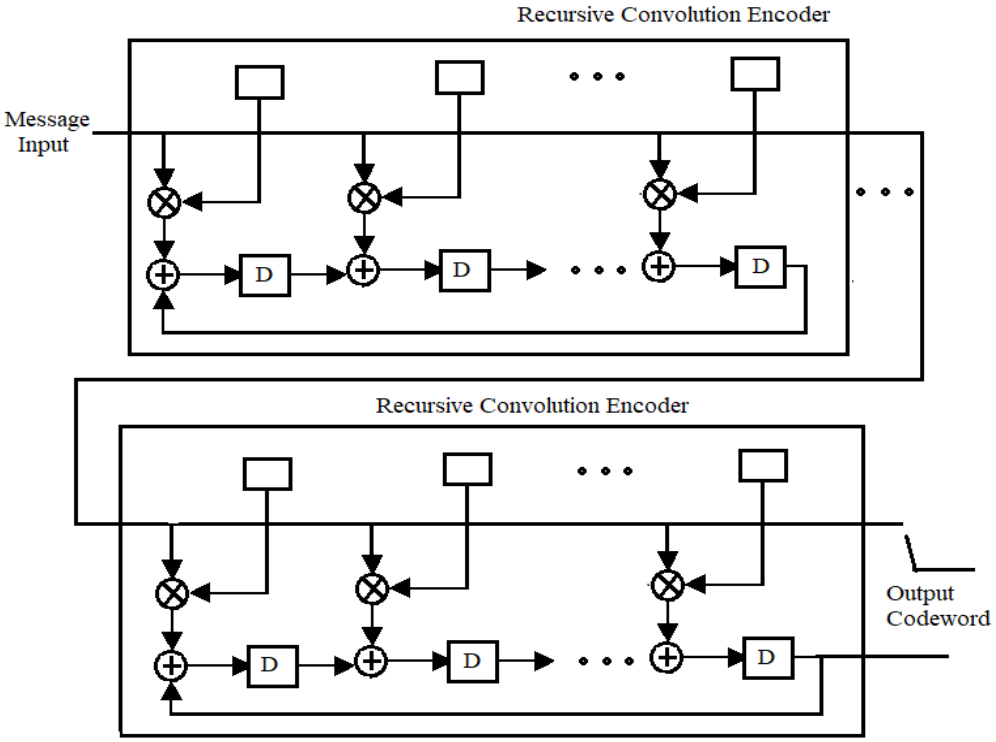


Fig.4.6 Circular and shift register operation for encoder

The LDPC encoder's output is the multiplication operation performed on the input message matrix and generator matrix. For a larger message sequence, this operation becomes very complex. LDPC encoding process uses the following steps:

Step-1: Compute the contents of parity check matrix 'H'

$$H = [-P^T : I_{n-k}] \qquad (4.4)$$

Step-2: Compute the contents of the generator matrix (G)

$$G = [I_K : P] \qquad (4.5)$$

Step-3: Determine the contents of the transmitting codeword (C_w)

$$C_w = m.G \qquad (4.6)$$

Circulant data encoding is utilized to select a parity check matrix (H). It also indicates the sparse matrix with a code rate value of 1/2. The size of the 'H' matrix and 'G' matrix is (2048 x 4096) and (2048 x 4096) respectively. Since the design and execution of such a large data size in real-time is a challenging task. Therefore, a (32 x 64) matrix is used for the implementation of a large-scale matrix.

4.3 Turbo Codes

Turbo codes were invented back in 1993 as a new type of error correction code. These codes are formed using two or more recursive systematic convolutional encoders [110] coupled in equivalent and detached by an interleaver. These codes perform better concerning BER than convolution codes because of their concatenation structure and iterative decoding method. Turbo codes are used in existing wireless protocols like LTE, UMTS, and HSDPA. A binary turbo encoder with two recursive convolution encoders is shown in Fig 4.7. Two parity bits are generated for each message bit given to the encoder. The production of the turbo encoder's output is the interleaved version of the input data. An interleaver is employed because consecutively transmitted bits can be influenced by faulty bits. The generation of the turbo encoder is comprised of the parity bits, which are produced by a recursive convolution encoder 1 (RSE-

1) and a recursive convolution encoder 2 (RSE-2). A parity frame and a systematic frame are generated by two convolution encoders. The feedforward polynomial ($1 + m + m^3$) is generated by RSE-1 and feedback generator polynomial ($1 + m^2 + m^3$) is generated by RSE-2. The value of the code rate for each encoder is R = 1/3, and the length of the sequence is 3N bits. A state diagram with (n = 8) is considered for the turbo encoder as depicted in Fig.4.8. Initially, the state of the turbo encoder is represented by (S_0 = '0'). S(n) ϵ {0, 1… 7} are the subsequent states for S(n) ϵ {0, 1} message bit. The states are S_0(000), S_1(001), S_2(010), S_3(011), S_4(100), S_5(101), S_6(110), and S_7(111) corresponding to binary message inputs (d_i = 0, di = 1).

The interleaver is the core block in the turbo encoding and decoding process. Due to the interleaving process burst errors in consecutive bits are spread over the transmitted sequence. Thus, BER performance is improved as the bits are reordered in some pseudo-random manner. Another process is puncturing, which increases the code rate by removing the coded parity bits of the sequence thus making the channel utilization efficient. The interleaver in the turbo encoder structure generates a long data block by reshuffling the data and the interleaver in the decoder structure corrects the errors by bypassing the data to the first decoder. After this process, the interleaver further reshuffles the first decoded data and once shuffling is done passes the data to the second decoder for remaining error corrections. The same way the process is repeated.

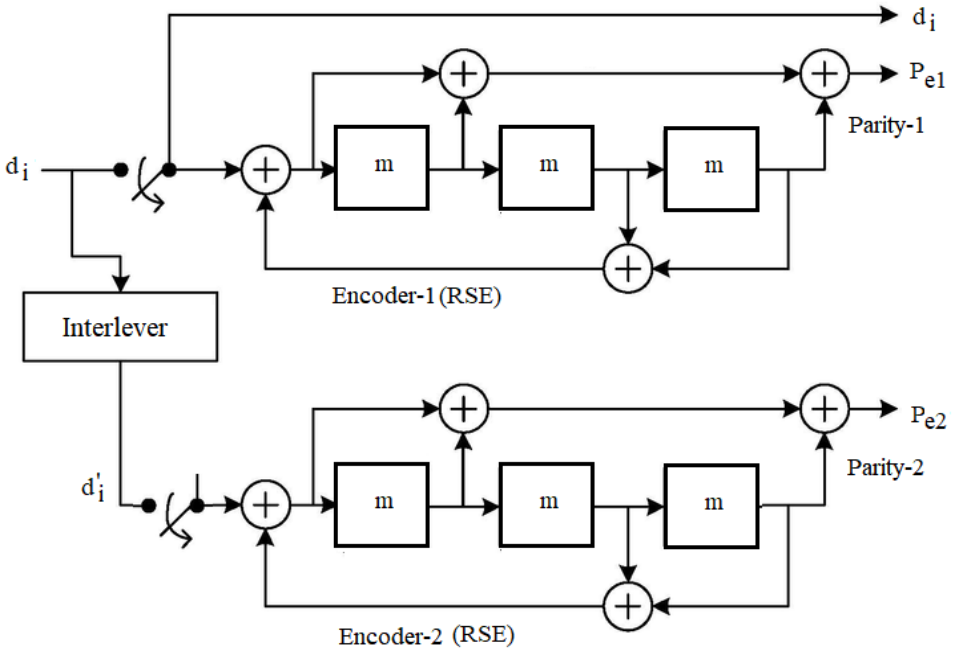


Fig.4.7. Turbo encoder [73]

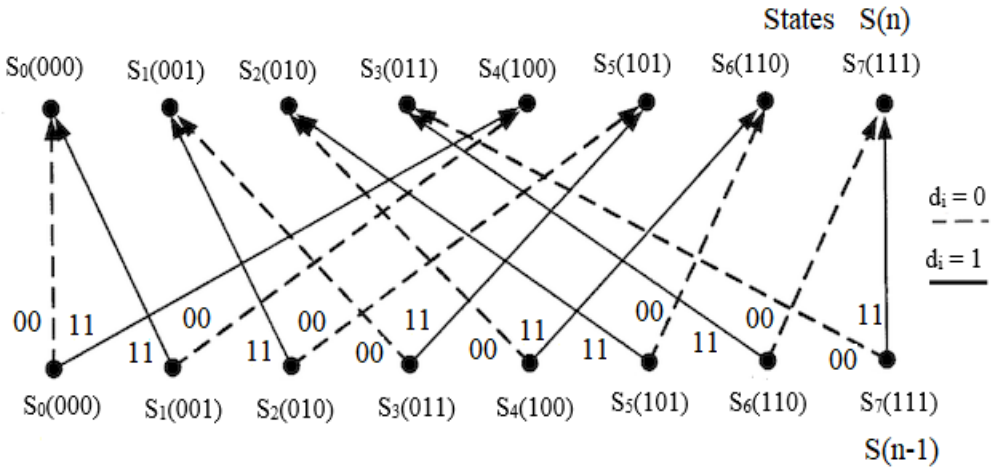


Fig.4.8. Trellis diagram configuration [73].

4.3.1 Algorithm

The decoder structure is a parallel concatenated arrangement of convolution codes that takes the independent set of parity bits. After a systematic bit sequence is formed this process is known as the iterative decoding method [125]. The decoding mechanism and the configuration of the turbo decoder are shown with the help of Fig.4.9 and Fig.4.10. The information or the message bits are processed between SISO-1 to SISO-2. The extrinsic information generated by SISO-1 is de-interleaved and then fed as an input to the next SISO-2. This building block calculates the forward matrix value (α), backward matrix value (β), and the output SISO matrix [126].

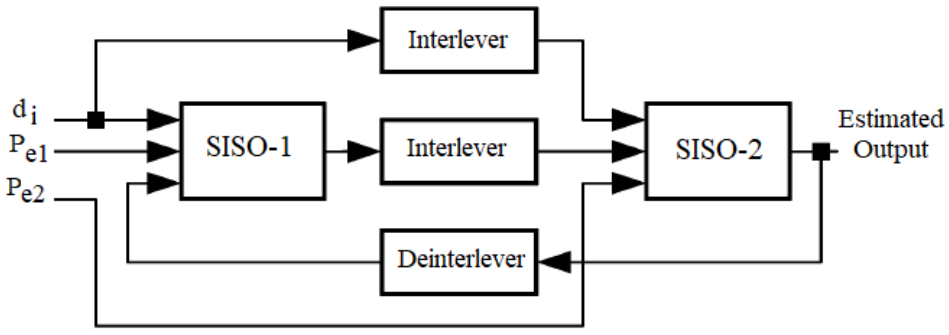


Fig. 4.9. Turbo decoding [126]

To calculate α and β values the BCJR algorithm is applied. The SISO decoders perform the turbo decoding process and are referred to as LLR [26]. For the

encoded sequence, d_i = [d_1, d_2…d_N], and generated codeword against each sequence d_i = [d_1, d_2…d_N], the LLR is articulated as

$$L(d_i) = \ln \frac{P(d_i = 1)}{P(d_i = 0)} \qquad (4.7)$$

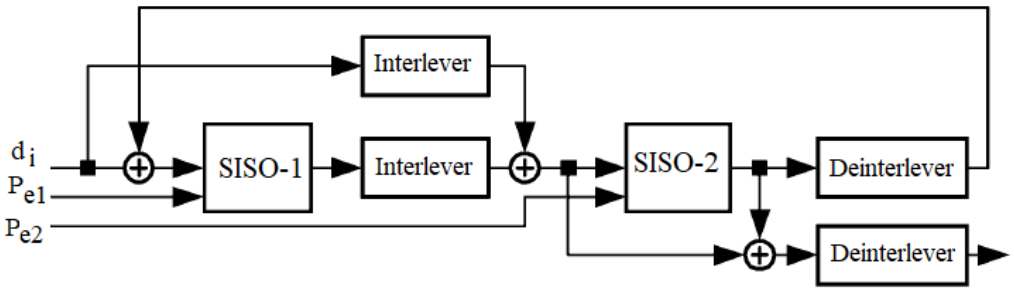


Fig.4.10 Iterative turbo decoding [127]

For the decoded sequence h = [h_1, h_2…h_N],

$$L(d_i|h) = \ln \frac{P(d_i = 1|h)}{P(d_i = 0|h)} \qquad (4.8)$$

[1] offered the MAP algorithm in 1974, applied to breed the probabilities of individual bits and descend the extrinsic information. The LLR of the k[th] symbol is stated as

$$L(d_k|h) = ln \frac{\sum_{d_k=1} \alpha'_{k-1}(s_{k-1}) \gamma'_k(s_{k-1}, s_k) \beta'_k(s_k)}{\sum_{d_k=0} \alpha'_{k-1}(s_{k-1}) \gamma'_k(s_{k-1}, s_k) \beta'_k(s_k)} \qquad (4.9)$$

In this equ. (4.9)

α'_k = Forward state matric

β'_k = Backward state matric

γ'_k = Branch matric

s_k = Trellis diagram states at a time ‘k’

The MAP algorithm [27] comprises addition processes and multiplication processes. For a large classification, the logarithm and estimate are applied, and the eq. (4.10) for max-log-MAP is specified as

$$\ln\left(\sum e^{g_i}\right) \approx maximum\ (g_i) \qquad (4.10)$$

Then the algorithm is streamlined to the expression

$$L(d_k|h) = max_{d_k=1}(\alpha_{k-1}\,\gamma_k(s_{k-1}, s_k)\,\beta_k\,(s_k)) -$$

$$max_{d_k=0}(\alpha_{k-1}\,\gamma_k(s_{k-1}, s_k)\,\beta_k\,(s_k)) \quad (4.11)$$

The expression has addition operations and association operations. The state matrix is similarly condensed and written as

$$\alpha_k\,(s_k) = \ln(\alpha'_k\,(s_k)) = maximum\,\left(\alpha_{k-1}(s_{k-1}) + \gamma_{k-1}(s_{k-1})\right) \quad (4.12)$$

$$\beta_k(s_k) = \ln(\beta'_k\,(s_k)) = maximum\,\left(\beta_{k-1}(s_{k-1}) + \gamma_{k-1}(s_{k-1})\right) \quad (4.13)$$

The branch matrix is expressed as

$$\gamma_k(s_{k-1}, s_k) = \begin{Bmatrix} \gamma_{k00} = 0.5\,(L_a(d_k) + L(d_{in}) + L(P_{k1}) \\ \gamma_{k01} = 0.5\,(L_a(d_k) + L(d_{in}) - L(P_{k1}) \\ \gamma_{k10} = 0.5\,(-L_a(d_k) - L(d_{in}) + L(P_{k1}) \\ \gamma_{k11} = 0.5\,(-L_a(d_k) - L(d_{in}) - L(P_{k1}) \end{Bmatrix} \quad (4.14)$$

Here,

L_a = Priori LLR

d_{in} = Methodical bit to the realization of extrinsic information

P_{k1} = Bit for parity check

Subsequently, after processing the calculations of the LLR, the resultant extrinsic information is presented as

$$L_e(d_k) = \mu.\left(L(d_k|h) - L(d_{in}) - L_a(d_k)\right) \quad (4.15)$$

Where μ denotes the scaling factor, and for accessing the extrinsic information in the MAP procedure to recompense for the losses, μ < 1.

4.3.2 Implementation Strategy

To meet the decoding requirements in the turbo coding scheme, interleaver and de-interleaver blocks are placed in parallel. The most challenging factor in the interleaving process is the contention of memory. Fig.4.11 shows the interleaving and parallel processing of the turbo decoder. To extract extrinsic information in terms of forwarding and backward variables, successive interleaving is used. Depending on the block size of data-parallel execution is used. An AGU and a CU are used to map the interleaver and de-interleaved with random access memory (RAM).

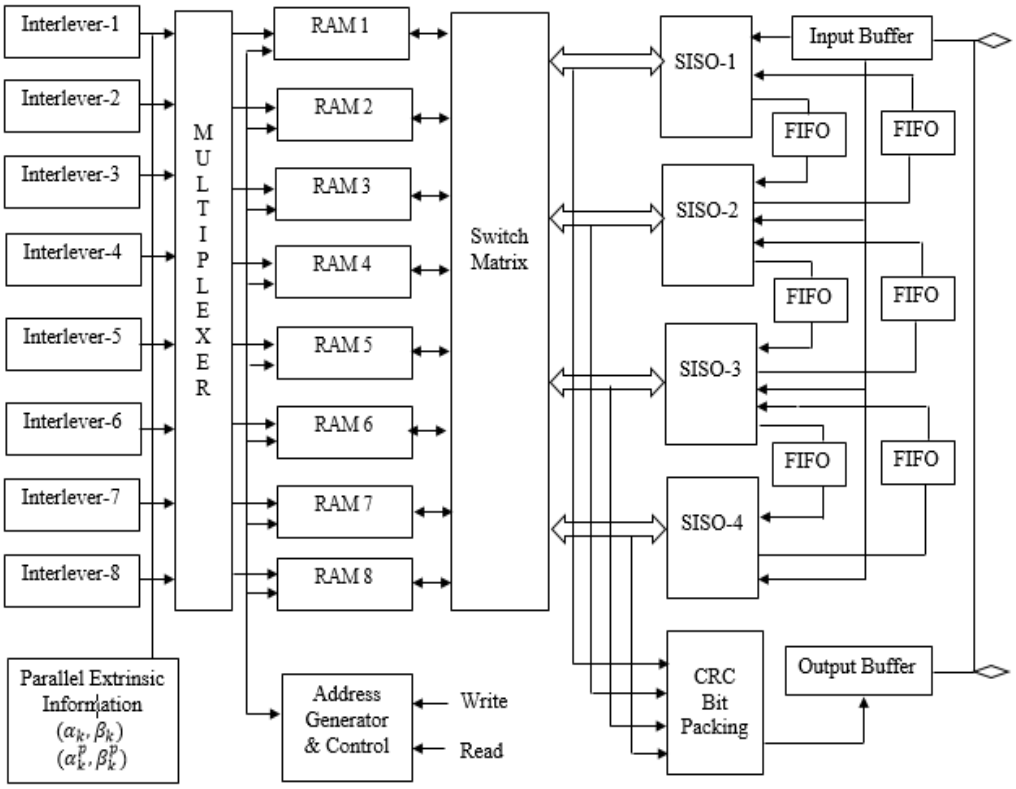


Fig. 4.11 The execution and parallel realization of the interleaving concept in the turbo hardware decoder

The real-time log-likelihood ratio and the parallel extrinsic information are selected with the help of multiplexers to perform the interleaving process. To perform the sequential read and write operations control signals are linked with RAM. To improve latency and reduce processing time during each iteration the AGU is used. For the selection of SISO and FIFO modules switch matrices are used. Depending upon the priority, each FIFO module is attached to its current SISO module for input-output buffer synchronization. Since the LLR, branch matrices, forward matrices, and backward matrices parameters are sequential. The throughput of the turbo decoder can be maximized using input and output buffers [129]. The design methodology of the work is shown in Fig.4.12. VHDL-based programming is used to design the turbo encoder and decoder. Waveform simulation is carried out for both the integrated modules in which the test cases are verified. The real-time verification is performed using a Virtex-5 FPGA board to check the feasibility of the design.

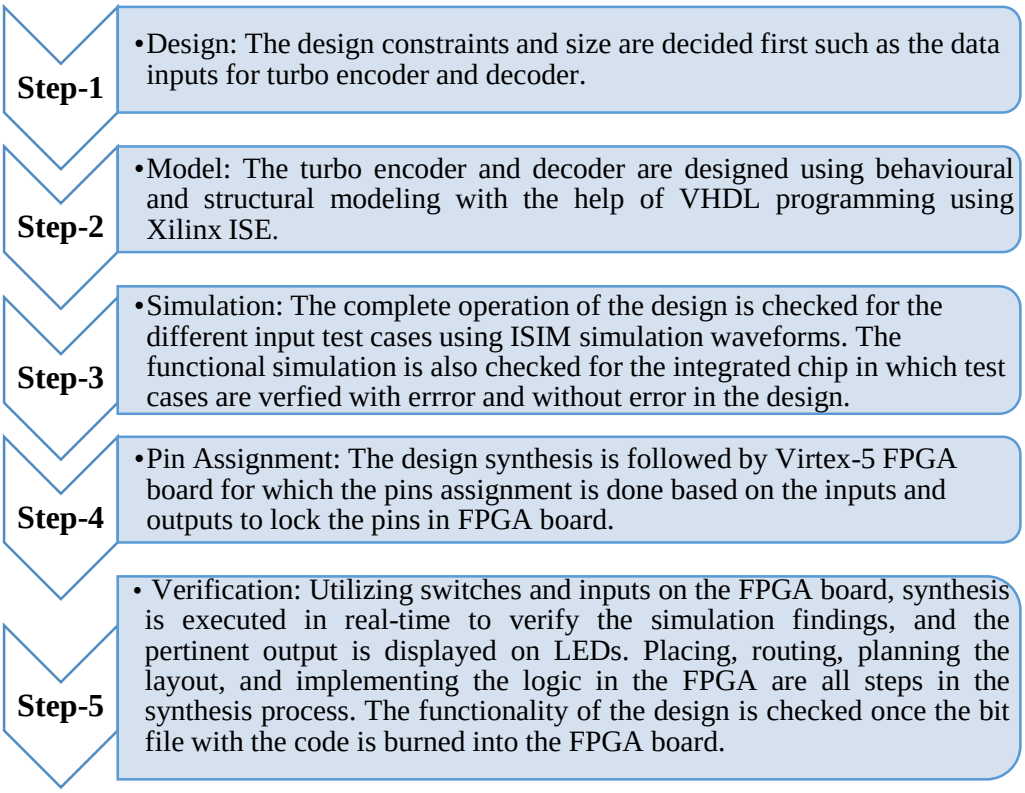


Fig. 4.12. Simulation and synthesis methodology

CHAPTER 5

POLAR CODES FOR 5G COMMUNICATION SYSTEM

5.1 Channel Coding Conditions for 5G-URLLC

Erdal Arikan introduced the concept of polar codes, a kind of linear block code [141], the first capacity-attaining channel code with a clear verification. These codes comes within the category of linear block codes. One of the newest challengers in the race to 5G is polar codes, and they have the distinction of being the first to break new ground in the field of channel coding thanks to their unique approach to the construction of codes in comparison to more standard approaches. In addition, polar code's simple encoding mechanism and very straightforward SC decoding make it an appealing choice for a wide choice of 5G applications. The idea of channel polarization is what allows polar codes to be created. During this procedure, ‘N’ duplicates of an ordinary channel ‘W’ are converted into extreme channels in which the symmetric capacities, depending on the value of ‘N’, either reach ‘1’ or reach ‘0’ as the number of copies increases.

Polar codes can only exist in this form. These ‘N’ polarized channels that are generated have these challenges (i) highly noisy and unreliable channels that broadcast random noise and (ii) ideal and reliable channels that can transmit data without noise. Next, the polar encoder will transmit ‘K’ information bits through the reliable channels, whereas the residual N–K bits will be turned into frozen bits (either 0 or 1) and are transmitted through un-reliable channels. In this ongoing competition to achieve 5G, polar codes have newly appeared as one of the best promising new contenders. The basic polar decoder is called a SC decoder, and it has the capability of achieving Shannon's capacity while having a relatively low level of complexity. Several enhanced SC decoding algorithms have been proposed [103] to enhance the finite length efficiency of a polar code.

This was done in both of these studies. Research is now being conducted on SCL decoders for their potential application in 5G studies that make use of L simultaneous decoding routes. This is being done to obtain performance that is superior to that of MAP decoders. There is a possibility that the CA-Polar along with SCL could potentially outperform the most advanced codes that are now available. PC–CA is suggested for use in low-latency application areas and is, as a result, taken into consideration for URLLC [98].

The following segment will provide a concise explanation of the primary requirements of the NR-URLLC before mapping these needs onto various channel coding methods. 5G-NR [142] with URLLC use case has stringent constraints on two characteristics that conflict with one another: (i) An extremely minimal amount of lag, and (ii) A very high degree of dependability [143]. To reduce the latency, shorter information must be used, which ultimately leads to a lower coding gain and a decline in reliability. Increasing dependability necessitates the addition of parity or redundant bits to the information at times, and re-transmission, all of which will lead to an increase in latency. Therefore, use caution when making your choice regarding the channel coding strategy to use for this case. Following the adoption of a standard for the eMBB scenario, the 3GPP began functioning toward the 5G-URLLC use case. The following are the significant performance indicator requirements that are well-defined by 3GPP in line with IMT-2020 for 5G-URLLC (i) Reliability- It is defined as subjective to the desired BLER with a value of 10^{-5} within a 1 ms period (with or without) HARQ. In addition, to this error floor, free BLER capability with a greater coding gain is sought for extremely high dependability of an order of 99.9 % and even higher. (ii) Delay- Total end-to-end latency for one packet is 1 millisecond, with a transmission time of 0.5 milliseconds for the uplink and 0.5 milliseconds for the downlink. The approaches to channel coding are changed to meet the necessities of URLLC for short latency and reliable transmission. To discover effective channel coding methods that meet the requirements of NR-URLLC for high reliability and low latency, the current state of art codes need to be re-considered. Fig.5.1 depicts NR-URLLC scenario requirements for channel coding. For the 5G-URLLC scenario, the conclusion

on the ideal channel coding method is still an open question that must be explored.

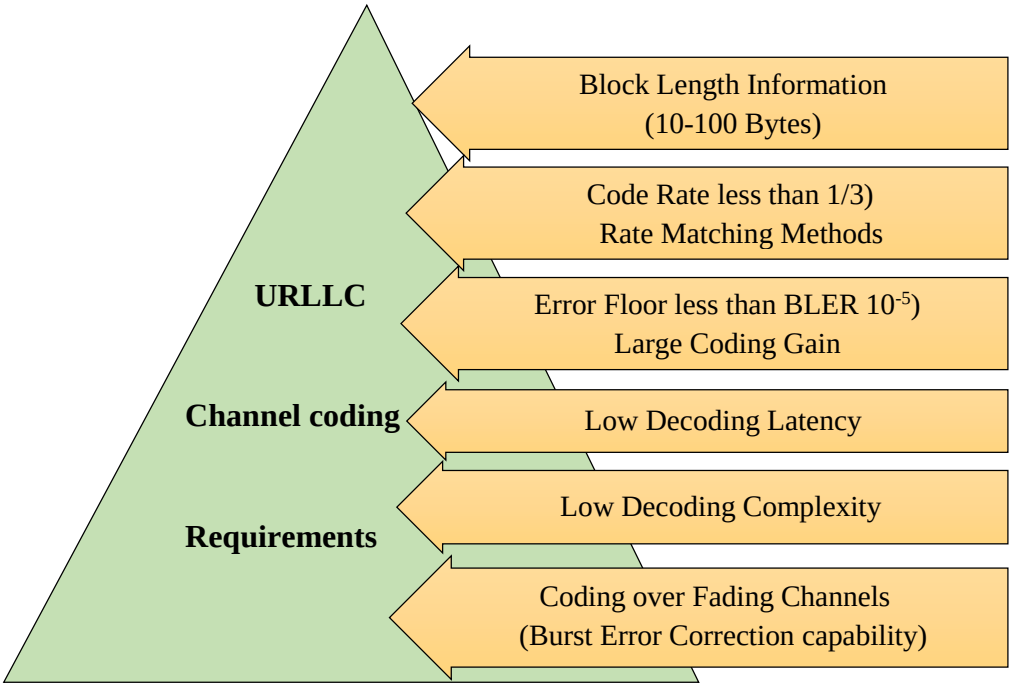


Fig.5.1 NR-URLLC scenario requirements for channel coding.

5.2 Polar Code for 5G-URLLC

Polar code comes under the category of linear block error correction codes. These are also known as the capacity achieving correction codes. Erdal Arkan at Bilkent University comes up with polar codes in 2009 [144]. Polar codes are easy to both encode and decode. The main purpose of this work is to make a polar code decoder that works well and has as little hardware complexity as possible. The most important problem in any communication structure is the effect of noise when the signal passes across a channel. It is impractical to make a channel perfect, so it's impossible to get rid of all noise. Most of the time, the AWGN noise model is used in communication systems. Erdal Arkan thought of a new idea called "polar codes." Polar codes are used which are based on the idea of channel polarization [147]. The whole channel is divided into channels that are noisy and channels that aren't noisy. Data bits are sent only through those channels which have space, so noise has a very small effect. The hardest part is finding the channels with the most people who can use them at once. Polar codes are very useful because these make mistakes as rare as possible. A

generalized digital communication block diagram is presented in Fig.5.2. It has a sender, a receiver, and a channel. Encrypting the data is needed to send it over a communication channel. This is done using encoder. At the receiver, a signal that was sent along with the noise is received, and the decoder puts the signal back together.

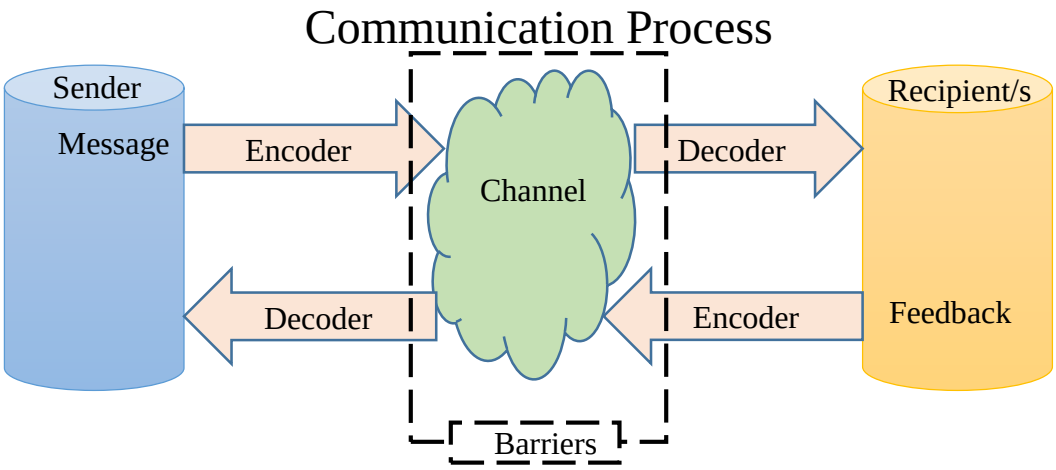


Fig.5.2 Block diagram of a digital communication system

5.2.1 Channel Polarization

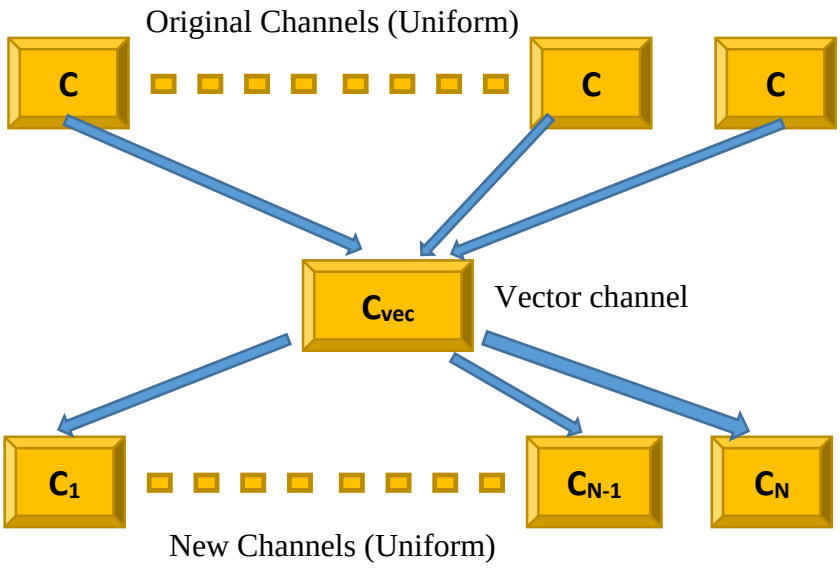


Fig.5.3 Concept of the channel polarization

Arkan [138] came up with the idea of channel polarization. Original channels are divided into noisy and perfect channels. Data can be sent through the perfect

channels, and thus channels are said to be polarized. An ordinary channel 'C' is turned into a perfect channel and an imperfect channel. There are three main steps in the process: making the code, encoding it, and decoding. Here, a polarization transform is used to improve the chances of the right choice about bits sent over a channel with no memory. Fig.5.3 represents the concept of channel polarization. Assuming X_m to be a binary discrete memoryless channel with an output denoted by Z_d and an input denoted by Y_d. The transition probability of the channel is stated as $X_m(z\text{-}y)$, where 'y' belongs to Y_d and 'z' belongs to Z_d. Two parameters measure the quality of a channel, one such parameter is the symmetric channel capacity, represented as $J_c(X_m)$, and the other is the Bhattacharyya parameter, represented as $A_c(X_m)$ as shown below:

$$J_c(X_m) = \sum_{z=Z_d}\sum_{y=y_d.} \frac{1}{2} X(x/y)\log \frac{X(\frac{Z_d}{y_d.})}{\frac{1}{2}X(Z_d/0)+X(Z_d/1)} [2] \quad (5.1)$$

$$A_c(X_m) = \sum_{z=Z_d} \sqrt{X(Z_d/0)X(Z_d/1)} \ [2] \quad (5.2)$$

The symmetric capacity defines the maximum rate at which a predictable communication can occur through 'X_m', while the Bhattacharyya parameter [145] limits the likelihood of incorrect detection when the maximum likelihood is used to analyze a received symbol 'Z_d.'. When data is transmitted over a binary discrete memoryless channel, both 'X_m' and the Bhattacharyya parameter have their values ranging between {0, 1}, and these two inequalities are demonstrated with the help of the following equation:

$$J_c(X_m) \geq \log \frac{2}{1+A_c(X_m)} \quad (5.3)$$

$$J_c(X_m) \leq \sqrt{1+(X)^2} \quad (5.4)$$

The above observations signify that $J_c(X_m) \approx 1$ if $A_c(X_m) \approx 0$ indicates an error-free transmission means the channel here is a perfect channel [147] and $J_c(X_m) \approx 0$ if $A_c(X_m) \approx 1$ indicates that no information may be transmitted as the channel is completely unreliable. When two bits (t_0, t_1) are transmitted from N_2 to M_2 with two distinct instances of XOR. N_2 returned two values (s_0,s_1) and has the mutual information values as follows:

$$J(Z_0, Z_1 : T_0) = J(X) = J(Z_0, Z_1 : 1) \qquad (5.5)$$

The mutual information value between the symbols and the received data, while converting (t_0, t_1) to (x_0, x_1) such that $x_1 = t_1$ and $x_0 = t_0 \oplus t_1$ is:

$$J(Z_0, Z_1 : T0) \leq J(X) \geq J(Z_0, Z_1 : T1) \qquad (5.6)$$

In other words, chance of getting the correct value of t_0 reduces while the chance of getting the correct value of t_1 increases. Polar transformation is used to change the multiple instances of 'X'. As the number of transformed channels, 'C_N' goes up, the chance of choosing a correct symbol 'ti' gets closer to high (1.0) or low (0.5). If the number of reliable bits get closer to the capacity of the channel, the code rate 'R' gets closer to the 'C_R'.

5.2.2 Polar Encoder

Polar codes are formed using a concatenation of numerous fundamental polarization [150] kernels. This results in cascading of values which quickens the process of polarizing synthetic channels while reducing the amount of complexity involved in encoding and decoding. This concatenation results in the production of a channel transformation matrix given as $G_N = G_2^{\otimes n}$. This matrix is represented by an 'n' folds Kronecker product of G_2 and its value may be iteratively determined as $G_N = \begin{bmatrix} \frac{G_N}{2} & 0 \\ \frac{G_N}{2} & \frac{G_N}{2} \end{bmatrix}$. For small values of 'n', the polarization of the synthetic channels is incomplete, which results in partially noisy intermediary channels. This build-up generates channels that are either completely noise-less or purely noisy, depending on the value of n. However, for larger values of n, these channels are completely noiseless or noisy. The equation $N = 2^n$ shows that polar codes have code lengths which is the power of two. The purpose of this design process for the (N, K) polar code is to determine the 'K' best synthetic channels, which are also called the most reliable channels, and then send the bits of information over those channels. By estimating how reliable each synthetic channel is, it is possible to put them in order of how reliable they are and give the 'K' bits of information to the best reliable synthetic channels [148]. The frozen set denoted by $F = I^C$ is defined by the remaining N-

K indices, and the channels that are linked with it are "frozen," meaning that they do not transmit any data.

The standard equation for a 2-bit polar transform is represented as:

$$G_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \text{ kernel} \qquad (5.7)$$

code word can be generated as $[m_1\ m_2]\ G_2 = [m_1+m_2\ m_2]$

The binary tree representation for a 2-bit polar transform is shown in the Fig.5.4.

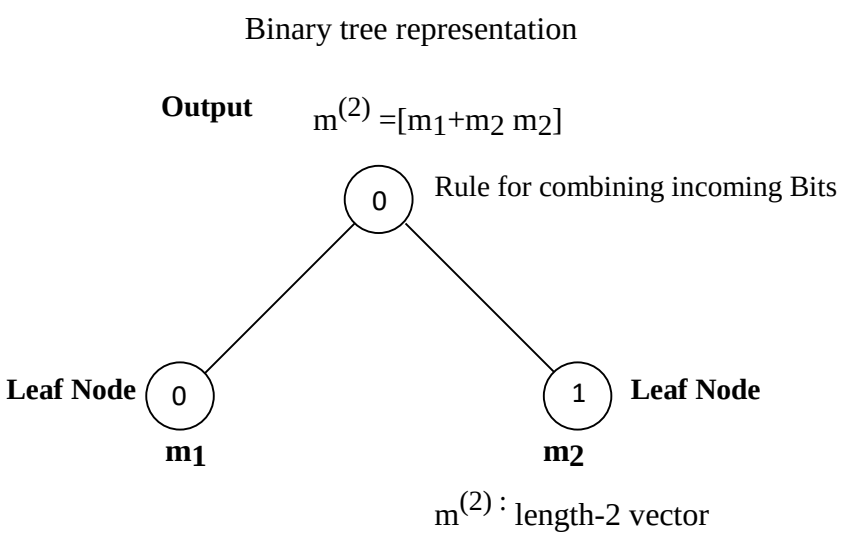


Fig.5.4 Binary tree representation for a 2-bit polar transformation

A 4-bit polar transform can be represented as:

$G_4 = G_2 \otimes G_2$, here $\otimes$ represents the kronecker product or tensor product

$$G_4 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \qquad (5.8)$$

$$G_4 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \qquad (5.9)$$

$$\begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$$

The generated 4-bit vector polar transform is given by the equation.:

$$[m_1\ m_2\ m_3\ m_4]\ G_4 = [m_1+m_2+m_3+m_4 \quad m_2+m_4 \quad m_3+m_4\ m_4] \qquad (5.10)$$

The binary tree representation for a 4-bit polar transform is shown in the Fig.5.5

Binary tree representation

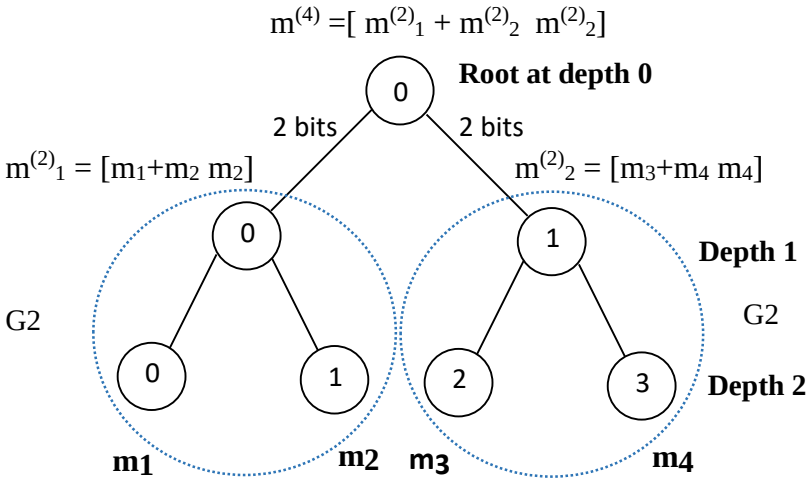


Fig.5.5 Binary tree representation for a 4-bit polar transformation

Similarly, the 8-bit polar transform can be represented using G_8 generator matrix which can be produced using G_2 and G_4.

$$G_8 = G_2 \otimes G_2 \otimes G_2$$

$$= \begin{bmatrix}1 & 0\\ 1 & 1\end{bmatrix} \otimes \begin{bmatrix}1 & 0\\ 1 & 1\end{bmatrix} \otimes \begin{bmatrix}1 & 0\\ 1 & 1\end{bmatrix}$$

or $G4 \otimes G_2$

$$= \begin{bmatrix}1 & 0\\ 1 & 1\end{bmatrix} \begin{bmatrix}0 & 0\\ 0 & 0\end{bmatrix} \otimes \begin{bmatrix}1 & 0\\ 1 & 1\end{bmatrix} \qquad (5.11)$$

$$\begin{bmatrix}1 & 0\\ 1 & 1\end{bmatrix} \begin{bmatrix}1 & 0\\ 1 & 1\end{bmatrix}$$

The generated 8-bit vector polar transform is given by the equ. (5.12):

$[m_1\ m_2\ m_3\ m_4\ m_5\ m_6\ m_7\ m_8]\ G_8 = [m_1+m_2+m_3+m_4+m_5+m_6+m_7+m_8\ m_2+m_4+m_6+m_8\ m_3+m_4+m_7+m_8\ m_4+m_8\ m_5+m_6+m_7+m_8\ m_6+m_8\ m_7+m_8\ m_8]$ (5.12)

The binary tree representation for an 8-bit polar transformation is depicted in Fig.5.6.

Binary tree representation

$m^{(8)} = [\, m^{(4)}_1 + m^{(4)}_2 \;\; m^{(4)}_2]$

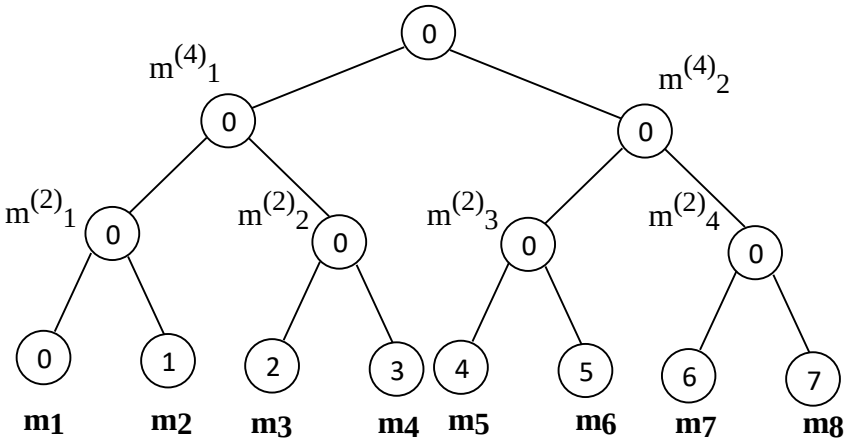


Fig.5.6 Binary tree representation of an 8-bit polar transformation

The generalized formula for Polar transformation is given by the equ. (5.13):

$$G_2^n = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \otimes n \qquad (5.13)$$

where $N=2^n$, G_N represents N x N matrix, and the kronecker product of 2 x 2 kernel.

5.2.3 Channel Representation of Polar Codes

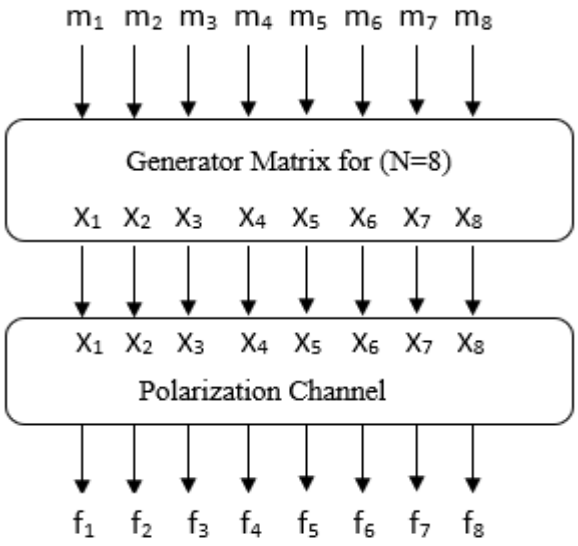


Fig.5.7 Polar transform with channel representation

Fig.5.7 depicts the channel represenation of an 8-bit polar transformation. Input message bits are denoted using m_1,m_2,m_3,m_4,m_5,m_6,m_7,m_8, G_N denotes the generator matrix here N=8. The encoded code words are represented by X_1, X_2, X_3, X_4, X_5, X_6, X_7, and X_8. the received codewords after transmission across the channel is represented by f_1, f_2, f_3, f_4, f_5, f_6, f_7, and f_8.

Fig.5.8 presents the channel realization for an 8-bit polar transform. Channel is divided into multiple bit channels and the entire received vector is represented by f^N.

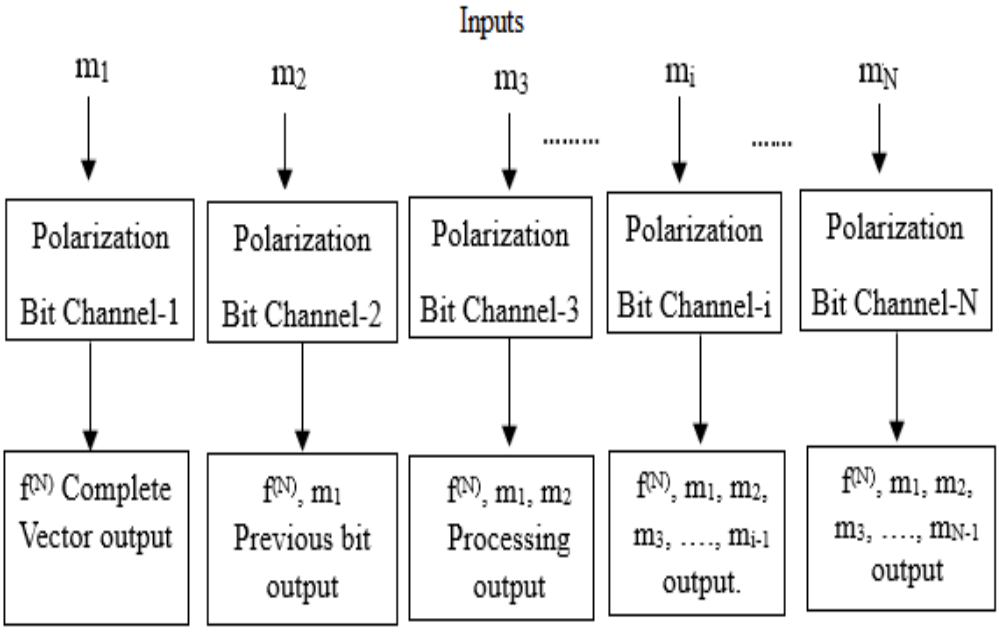


Fig.5.8 Channel realization for an 8-bit polar transform

The bit channels shown in Fig.5.8 are polarized and can be ordered based on the quality of the channels i.e. from good to bad. Entire channel is split into multiple bit channels. The first output for m_1 is the entire vector f_n.

For the second bit channel output is the entire vector along with the previous bit. For the third bit channel output is the entire vector along with two previous bits.

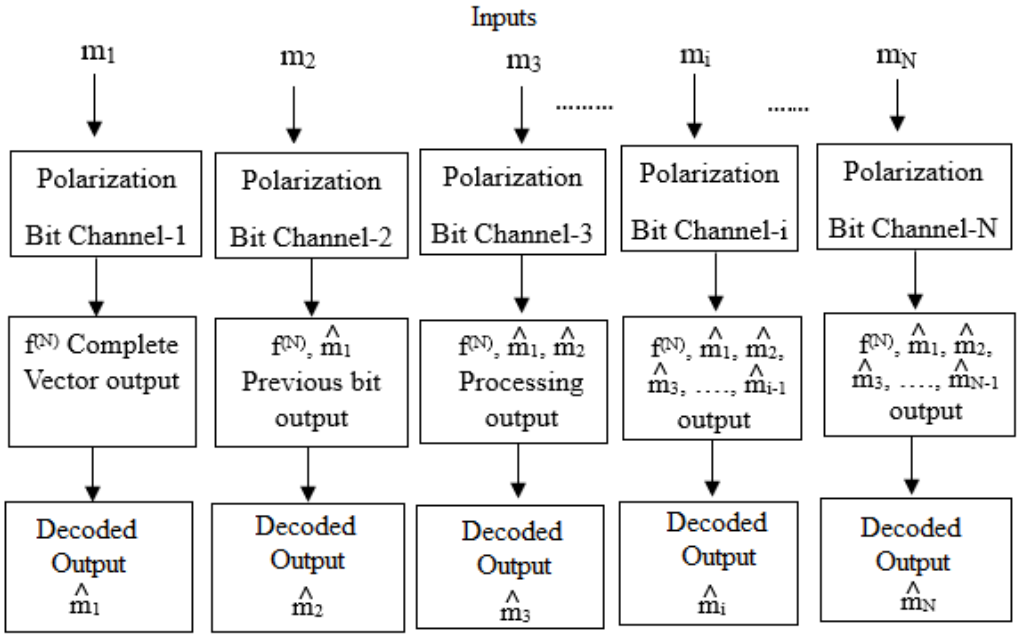


Fig.5.9 SC decoding for bit estimation

Fig.5.9 shows the bit estimation using SC decoding. If bit 'i' is frozen, $m_i = 0$, If bit 'i' is a message bit, bit estimation is found using a SISO decoder. With the help of the received vector decoding is performed and first bit $\hat{m}$i is estimated. If the bit is frozen no special decoding mechanism is used and $\hat{m}_1$ is assumed to be '0'. If the bit is the message bit $\hat{m}_1$ can be found using SISO decoder. Once $\hat{m}_1$ is estimated with the help of entire vector m_1 and $\hat{m}_2$ can be estimated. Once $\hat{m}_2$ is estimated with the help of entire vector, $\hat{m}_1$, $\hat{m}_2$, and $\hat{m}_3$ can be estimated. Similarly, bits can be estimated for rest of the bit channels.

5.2.4 Reliability Sequence

The best and worst channels are decided using the concept of reliability sequence. Reliability sequence decides the ordering of the bit channels from worst to the best. The reliability sequence for N = 8 is denoted by S= [1 2 3 5 4 6 7 8]. Here the frozen positions or bits are the first N-K positions and the rest positions are the message positions. Therefore, the frozen bit positions are 1, 2, 3, and 5. The message bit positions are 4, 6, 7, and 8 respectively. The binary tree representation for a (8,4) polar code with frozen and message bits is depicted in Fig.5.10.

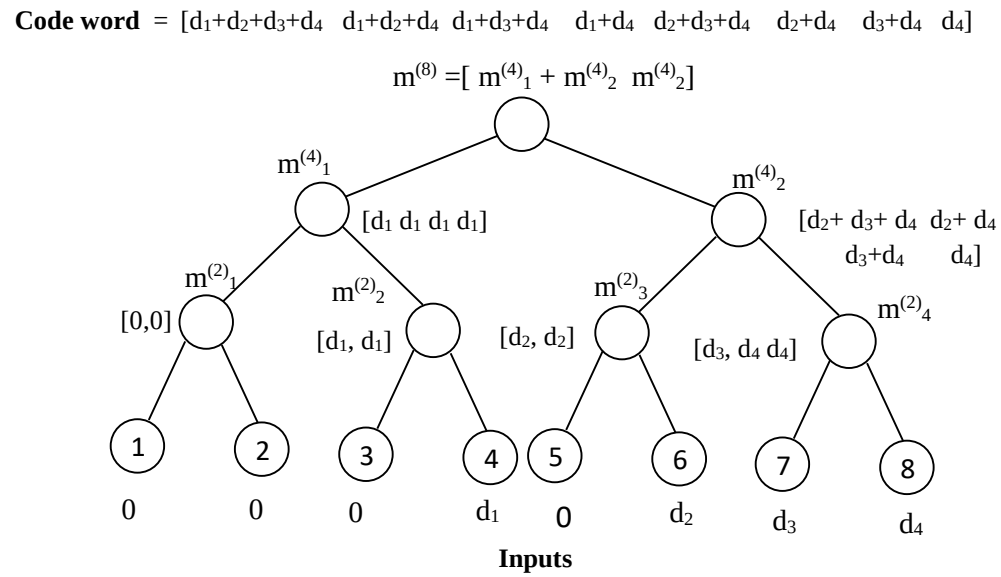


Fig.5.10 (8,4) polar code binary tree with frozen and message bits

The reliability sequence for N=16 is denoted by S= [1 2 3 5 9 4 6 10 7 11 13 8 12 14 15 16]. Here the frozen positions or bits are the first N-K positions and the rest positions are the message positions. Therefore, the frozen bit positions are 1, 2, 3, 5, 9, and 4. The message bit positions are 6, 10, 7, 11, 13, 8, 12, 14, 15, and 16 respectively. The binary tree representation for a (16,10) polar code with frozen and message bits is depicted in Fig.5.11.

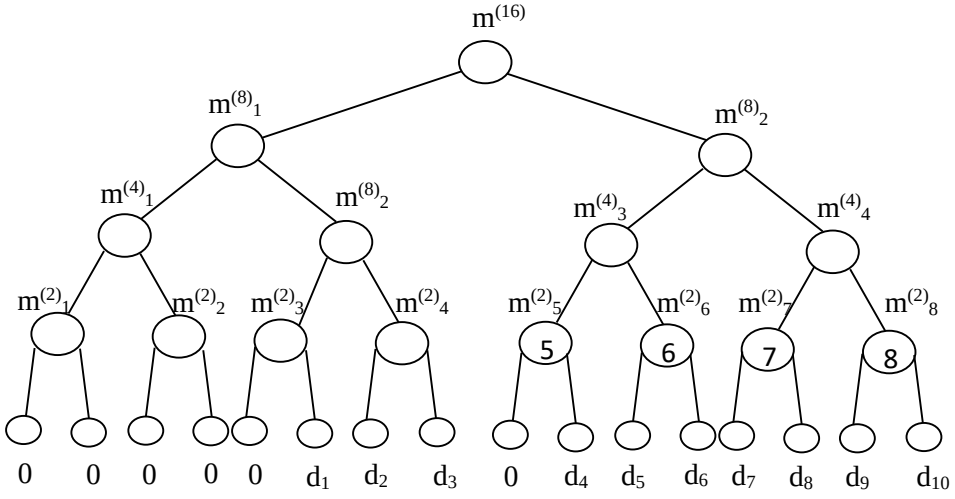


Fig.5.11 (16,10) polar code binary tree with frozen and message bits

The reliability sequence for N=32 is denoted as S= [1 2 3 5 9 17 4 6 10 7 18 11 19 13 21 25 8 12 20 14 15 22 27 26 23 29 16 24 28 30 31 32]. Here the frozen

positions or bits are the first N-K positions and the rest positions are the message positions. Therefore, the frozen bit positions are 1, 2, 3, 5, 9, 17, 4, 6, 10, 7, 18, and 11. The message bit positions are 19, 13, 21, 25, 8, 12, 20, 14, 15, 22, 27, 26, 23, 29, 16, 24, 28, 30, 31, and 32 respectively.

5.2.5 Encoding Method

The channel transformation matrix G_N is denoted as $G_2^{\otimes n}$, and the information set 'I' states an (N, K) polar code. The generator matrix [150] for the code is represented by a submatrix of G_N, which is made up of rows, and the indices for those rows are kept in 'I'. But because G_N's structure is recursive, it is possible to make the encoding process less complicated by adding an extra input vector 'm' with a length of 'N'bits. The input vector, denoted by M = [M(0), M(1), M(2),…… M(N-1)], is made by setting $M_i = 0$ if, I Ɛ F and then putting the information in the rest of the vector's entries. After this, the code word X= [X = $[X_o, X_1, X_2, X_3,\ldots\ldots X_{N-1}]$ can be calculated as x = M. G_N.

Since the matrix multiplication can be done at the same time as other G_2 matrix multiplications, the complexity of the encoding can be cut down to $O(\log_2(N))$. Because G_N has a recursive structure, decoding can be completed in $\log_2(N)$ stages, each of which is made up of N/2 fundamental polarization kernels that are similar to one another. The encoder for the polar code generates a code word using an equation:

$$X = m.G_N \qquad 5.14$$

Where 'm' is a vector of length 'N' bits and can be formed using following steps:

(i) Find the N-K least reliable (worst) channels using the reliability sequence.

(ii) Set m_i for those N-K channels to '0' which are referred to as frozen positions.

(iii) Set the remaining 'k' bits of mi to the message bits.

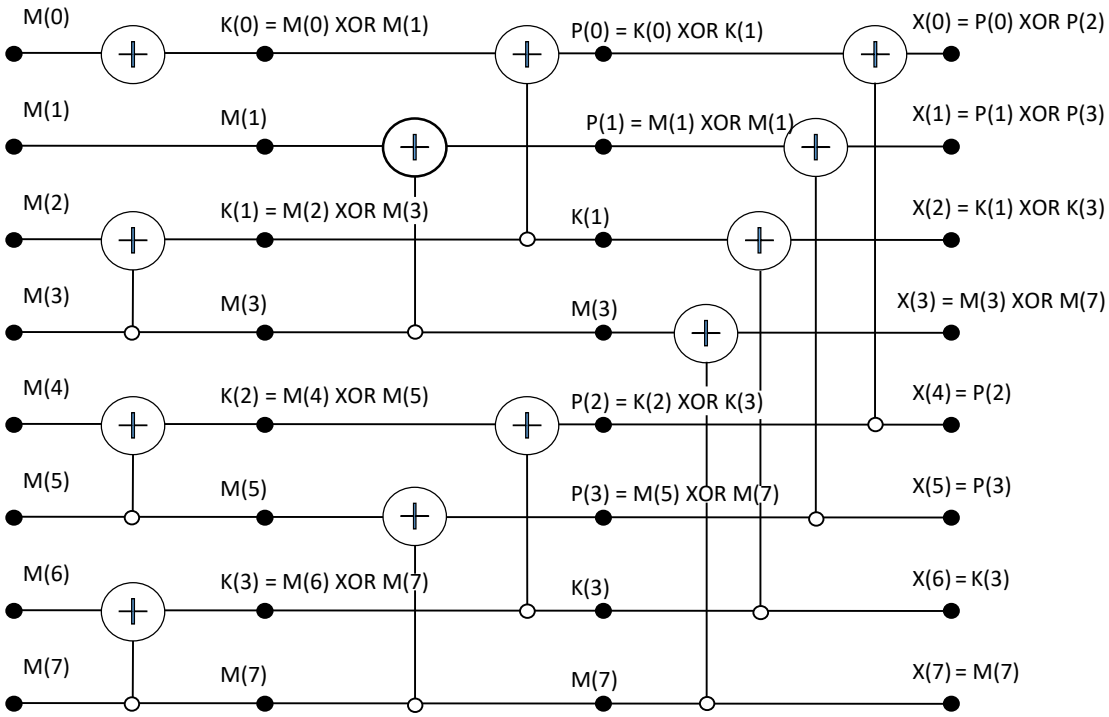


Fig.5.12 Encoder of an (8,4) polar code

Fig.5.12 depicts the structure of a polar encoder for N=8 length. With the help of message bits k=4 and the input vector M= [M(0) to M(7)]. The intermediate states are represented as K(0) to K(3), P(0) to P(3). The code word is generated as X = [X(0) to X(7)]. While designing the 8-bit polar encoder following representation is considered as follows:

M(0) = polar_message_data(0)

M(1) = polar_ message_ data(1)

M(2) = polar_ message_ data(2)

M(3) = polar_ message_ data(3)

M(4) = polar_ message_ data(4)

M(5) = polar_ message_ data(5)

M(6) = polar_message_data(6)

M(7) = polar_ message_ data(7)

X(0) = polar_encoded_data(0)

X(1) = polar_encoded_data(1)

X(2) = polar_encoded_ data(2)

X(3) = polar_encoded_data(3)

X(4) = polar_encoded_data(4)

X(5) = polar_encoded_ data(5)

X(6) = polar_encoded_data(6)

X(7) = polar_encoded_data(7)

The encoding takes place as follows:

K(0) <= M(0) XOR M(1);

K(1) <= M(2) XOR M(3);

K(2) <= M(4) XOR M(5);

K(3) <= M(6) XOR M(7);

P(0) <= K(0) XOR K(1);

P(1) <= M(1) XOR M(3);

P(2) <= K(2) XOR K(3);

P(3) <= M(5) XOR M(7);

X(0) <= P(0) XOR P(2);

X(1) <= P(1) XOR P(3);

X(2) <= K(1) XOR K(3);

X(3) <= M(3) XOR M(7);

X(4) <= P(2);

X(5) <= P(3);

X(6) <= K(3);

X(7) <= M(7);

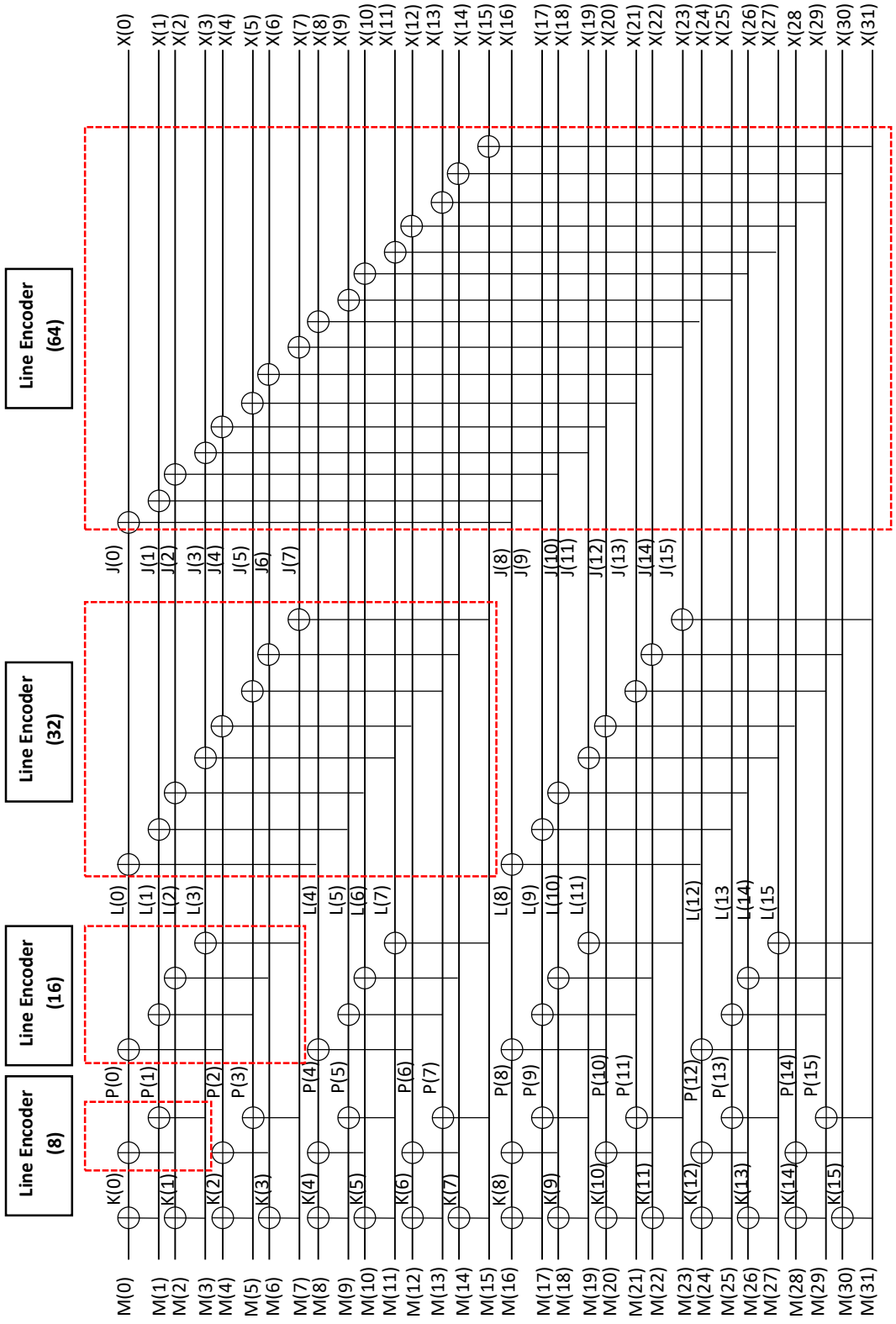


Fig.5.13 Encoder of an (32,20) polar code

Fig.5.13 depicts the structure of a polar encoder for N=32 length. With the help of message bits k = 20 and the input vector M= [M(0) to M(31)]. The intermediate states are represented as [K(0) to K(15), P(0) to P(3), J(0) to J(23), L(0) to L(15)]. The code word is generated as X = [X(0) to X(31)]. While designing the 8-bit polar encoder-decoder following representation is considered as follows:

M(0) = polar_message_data(0)
M(1) = polar_ message_ data(1)
M(2) = polar_ message_ data(2)
M(3) = polar_ message_ data(3)
M(4) = polar_ message_ data(4)
M(5) = polar_ message_ data(5)
M(6) = polar_message_data(6)
M(7) = polar_ message_ data(7)
M(8) = polar_ message_ data(8)
⋮
M(31) = polar_ message_ data(31)

X(0) = polar_encoded_data(0)
X(1) = polar_encoded_data(1)
X(2) = polar_encoded_ data(2)
X(3) = polar_encoded_data(3)
X(4) = polar_encoded_data(4)
X(5) = polar_encoded_ data(5)
X(6) = polar_encoded_data(6)
X(7) = polar_encoded_data(7)
X(8) = polar_encoded_data(8)
⋮
X(31) = polar_encoded_data(31)

5.2.6 The Successive cancellation list decoding

[150] gave a proposal for the native polar code decoding technique, which was referred to as SC. It is possible to think of it as a depth-first binary tree having a left branch as its priority. The leaf nodes of the tree represent 'N' bits that need to be calculated, and the root node contains soft information about the code bits that have been received. The complexity of the decoding method can be given as O. (N $\log_2$(N)). Fig.5.7 depicts the decoding structure using a tree for an (8,4) polar code. The leaf nodes of the decoding tree are shown in black, indicating information bits, while the nodes that are shown in white indicate frozen bits.

The list decoding mechanism is explained using Fig.5.14. It produces a list of all possible code words instead of producing single code word. Out of multiple code words a valid code word is estimated using a CRC. This procedure increases the complexity but improves performance. The message that passes CRC is the valid code word.

If none of the message passes CRC the one with the lowest path metric is considered to be valid code word. Bits at the receiver are considered using two terms path metric and decision metric. Decision metric is assigned whenever any decision is made to the bit encountered at any step. Path metric is the total of all the decision metric associated with any particular bit. For every bit both the decisions of '1' and '0' is considered and a decision metric is assigned to every incoming bit. If the decision is made as per the belief than the decision metric value is zero and if the decision is against the belief than a penalty is added which is the belief of the bit. SC decoding using path metric and decision metric is explained using equ. (5.15) and equ. (5.16). Here $P_{(}m_{i)}$ denotes the belief associated with the first encountered node , Dm_i is the decision metric assigned to the node and $\widehat{mi}$ is the bit estimation based on the threshold value comparison. Length of the CRC bits is denoted by 'L', List decoding produces 'H' code words. A decision is made on the first bit. If the bit is frozen the decision is assumed to be '0', if not decision is made based on the belief of the bit. It is assumed that decision can be made that the bit can i.e. either '0' or '1'. Instead of making one decision both the paths are considered.

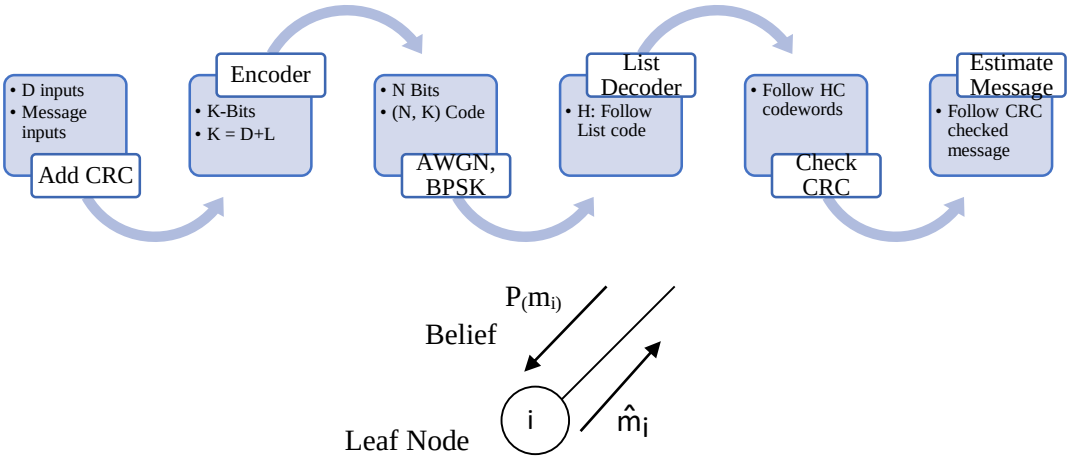


Fig.5.14 SC list decoding mechanism

If $P(m_i) \geq 0$, $\widehat{mi} = 0$ then $Dm_i = 0$, $\widehat{mi} = 1$ then $Dm_i = |P(mi)|$ (5.15)

If $P(m_i) < 0$, $\widehat{mi} = 1$ then $Dm_i = 0$, $\widehat{mi} = 0$ then $Dm_i = |P(mi)|$ (5.16)

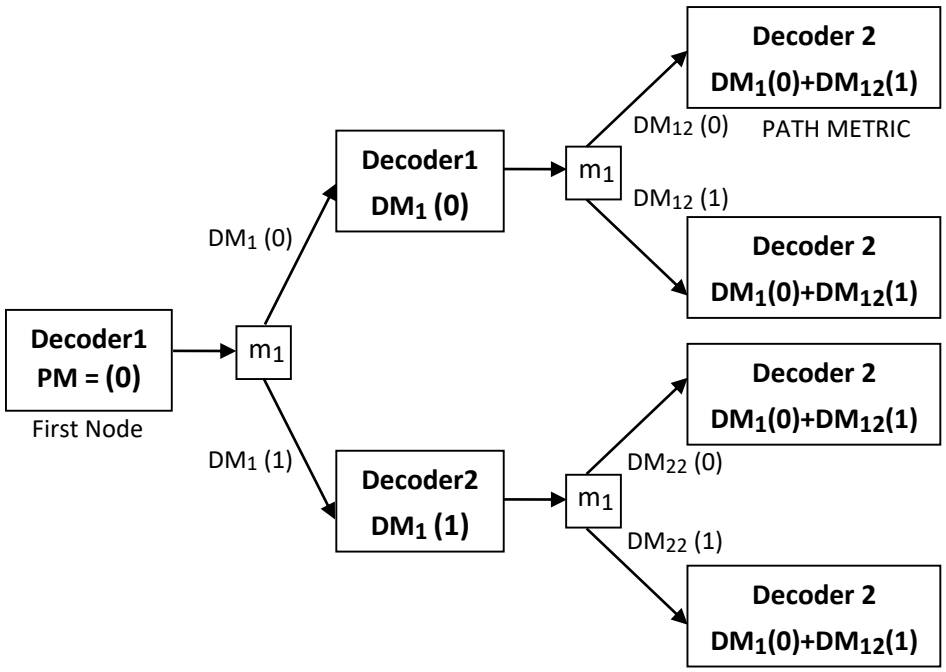


Fig.5.15 SCL decoding for N=4

Fig.5.15 depicts the SCL decoding mechanism for N=4. At the initial point when first received bit is encountered the path metric = 0, if the first encountered bit is frozen the decoding continues to be in one path and the bit is assumed to be '0'. If 'm_1' is not frozen two decisions are made either in paths '0' or '1'. If it is assumed as '0' some penalty associated with '0' i.e. $DM_1(0)$ is added. If it

is assumed as '1' some penalty associated with '0' i.e. $DM_1(1)$ is added. Initially, there is only one decoder but now in the next step it becomes two decoders. Every decoder requires a memory and states for storing belief values and 'm_i' values. In the next step decoding reaches 'm_2' and again two paths are considered either a decision of '0' or '1' is made. A decision metric associated with '0' and '1' is again added. Therefore, now the path metric becomes $DM_1(0)$ + $DM_{12}(1)$. Similarly, the decision metric associated with '0' and '1' is again added in next steps and it goes on. $DM_{12}(0)$ is the decision metric on 1st decoder for the second bit when a decision '0' is made. $DM_{12}(1)$ is the decision metric on 1st decoder for the second bit when a decision '1' is made. $DM_{22}(0)$ is the decision metric on 2nd decoder for the second bit when a decision '0' is made. $DM_{22}(1)$ is the decision Metric on 2nd decoder for the second bit when a decision '1' is made.

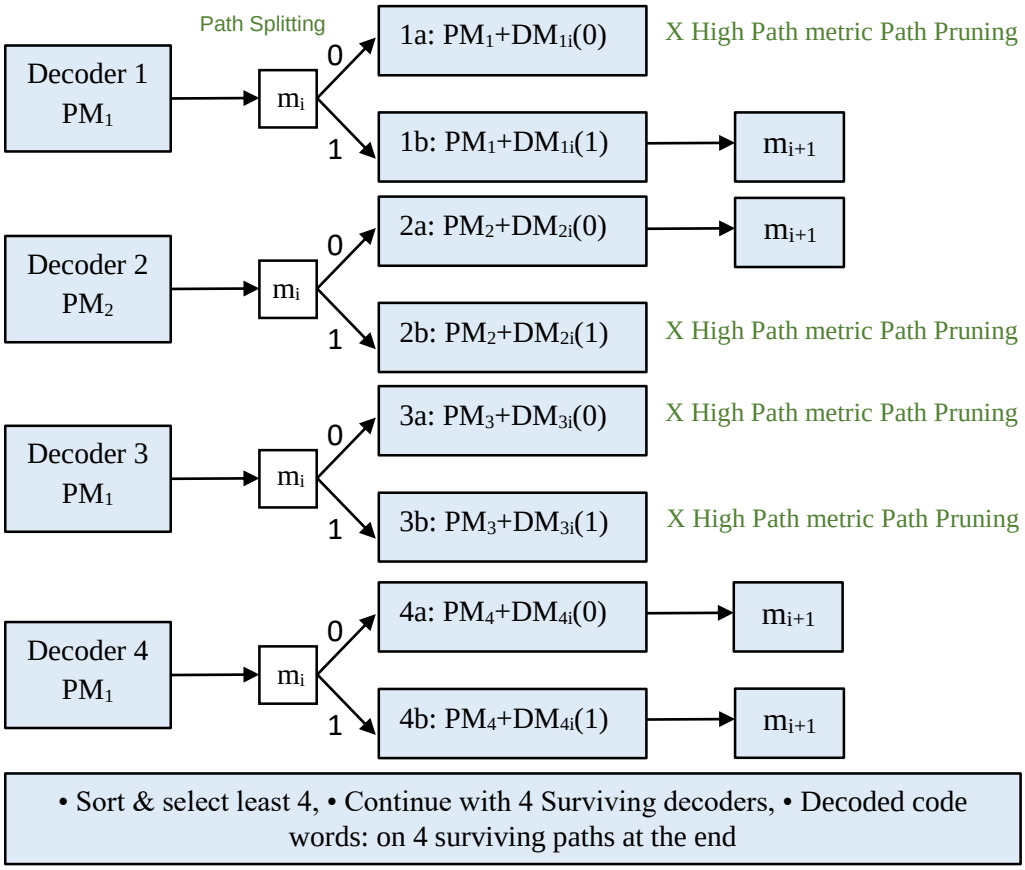


Fig.5.16 SCL decoding with path splitting and path pruning

The generalized structure showing the decoding mechanism of a size N= 4 is depicted using Fig.5.16. Before the decision on the node 'm_i' is made, there are four decoders and four path metrics. Again at node 'm_i' decision could be either '0' or '1'. If the bit is frozen it is assumed to be '0' and the decoder number remains four. Henceforth, the path metrics are updated depending on the sign values of beliefs. The four bits again generate eight paths assuming both the paths for each m_i. Out of eight paths only four paths are selected and path pruning is done. The paths with the highest value of path metrics is cancelled or pruned and the path with lowest path metrics are selected.

Thus, four surviving paths are considered. The decoded code words are the decisions made on the surviving paths. Finally, the decisions i.e. mi values are added and the CRC is performed on those m_i values.

Fig.5.17 presents the basic building block for N=2 SC decoder where f_1 and f_2 represents the beliefs for X_1 and X_2. Beliefs for m_1 and m_2 is denoted by $P(m_1)$ and $P(m_2)$ respectively.

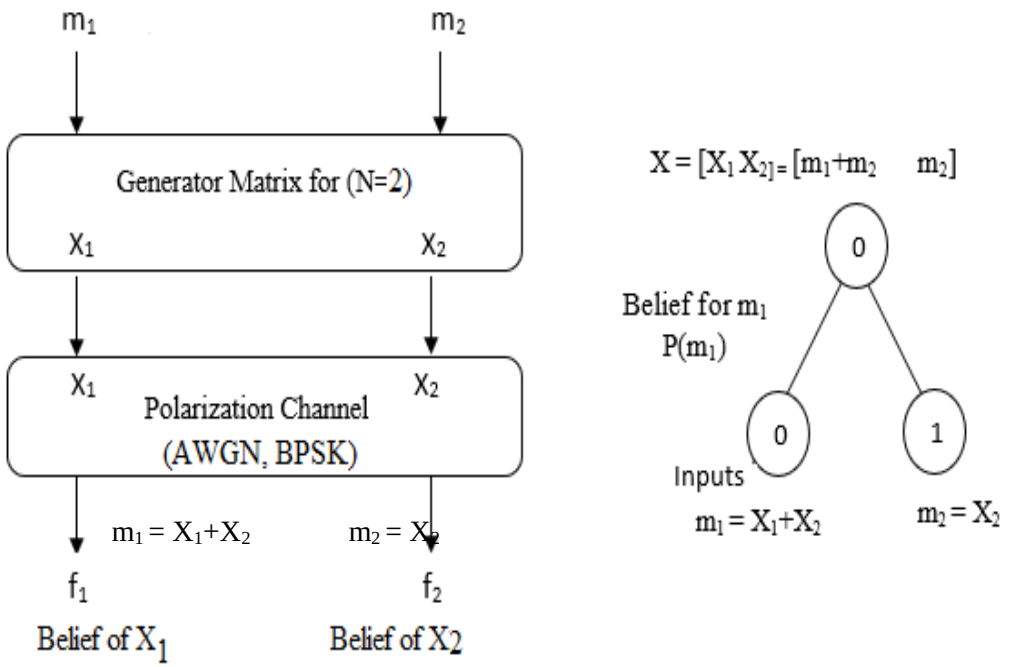


Fig.5.17 Basic building block of N=2, SC decoder

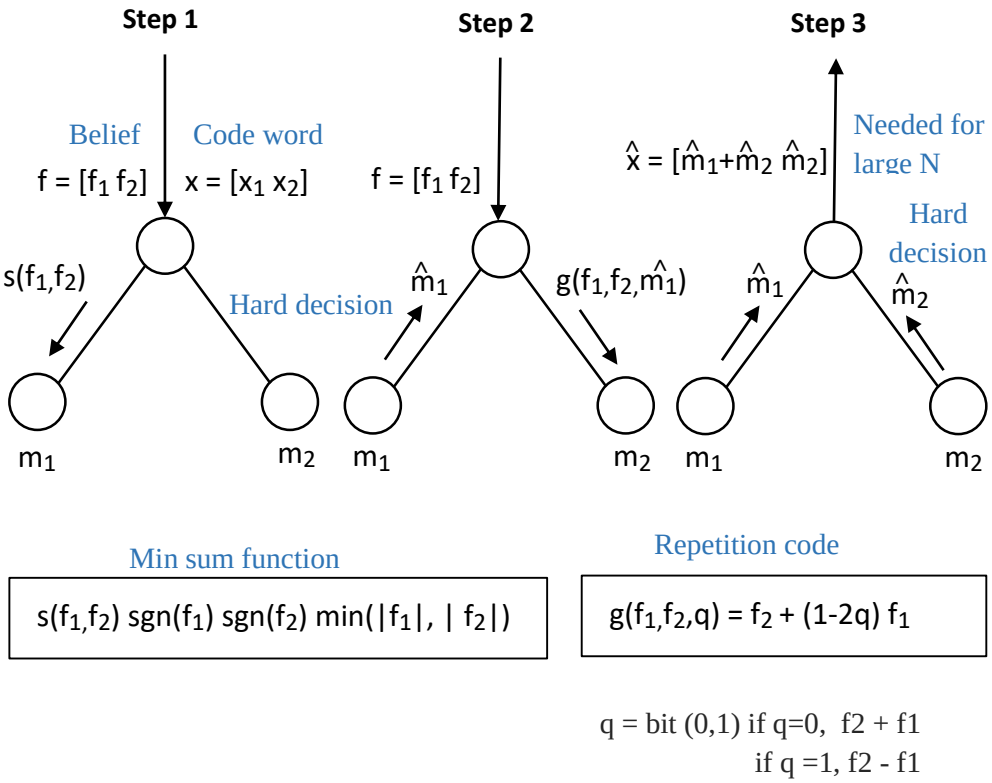


Fig.5.18 Message passing mechanism in SC decoder

Fig.5.18 shows the basic building block of SCL decoder with depth 1 binary tree showing root and two nodes. G_2 is the kernel, m_1 and m_2 are the inputs, the generated vector or code word is X_1 and X_2. X_1 and X_2 is transmitted over BPSK channel with noise, f_1 and f_2 are the received code words where f_1 corresponds to X_1 and f_2 corresponds to X_2. The aim is to estimate soft information for u_1 and u_2. For this single parity check code and repetition code is utilized. The equation for min sum algorithm and repetition code is given as follows:

$$P(m_1)= s(f_1,f_2) = \text{sgn}(f_1)\ \text{sgn}(f_2)\ \min(|f_1|, |f_2|) \quad (5.17)$$

$$\hat{m}_i = 0, \text{ if } P(m_1) \geq 0, \hat{m}_1, \text{ if } P(m_1) < 0 \quad (5.18)$$

$$\text{If } \hat{m}_1 = 0,\ P(m_2) = f_2+f_1\ (X= m_2\ m_2) \quad (5.19)$$

$$\text{If } \hat{m}_1 = 1,\ P(m_2) = f_2-f_1\ (X= \bar{m}_2\ m_2) \quad (5.20)$$

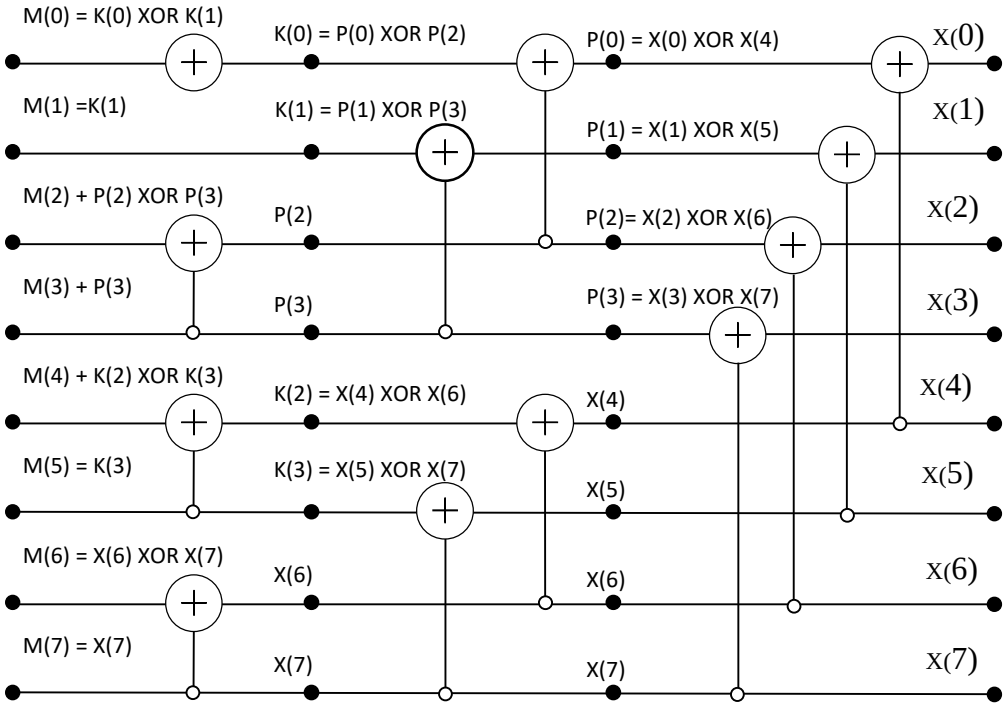


Fig.5.19 Decoder of an (8,4) polar code

Fig. 5.19 depicts the structure of a polar decoder with N=8 length that can be used to decode the the input vector M= [M(0) to M(7)] using the generated code word X = [X(0) to X(7)]. The intermediate states are represented as K(0) to K(3), P(0) to P(3). The decoding procedure takes place as follows: While designing the 8-bit polar decoder following representation is considered as follows:

M(0) = polar_message_data(0)
M(1) = polar_ message_ data(1)
M(2) = polar_ message_ data(2)
M(3) = polar_ message_ data(3)
M(4) = polar_ message_ data(4)
M(5) = polar_ message_ data(5)
M(6) = polar_message_data(6)
M(7) = polar_ message_ data(7)

X(0) = polar_decoded_data(0)
X(1) = polar_ decoded _data(1)
X(2) = polar_ decoded _ data(2)

X(3) = polar_ decoded _data(3)

X(4) = polar_ decoded _data(4)

X(5) = polar_ decoded _ data(5)

X(6) = polar_ decoded _data(6)

X(7) = polar_ decoded _data(7)

The decoding takes place as follows:

P(0) <= X(0) XOR X(4);

P(1) <= X(1) XOR X(5);

P(2) <= X(2) XOR X(6);

P(3) <= X(3) XOR X(7);

K(0) <= P(0) XOR P(2);

K(1) <= P(1) XOR P(3);

K(2) <= X(4) XOR X(6);

K(3) <= X(5) XOR X(7);

M(0) <= K(0) XOR K(1);

M(1) <= K(1);

M(2) <= P(2) XOR P(3);

M(3) <= P(3);

M(4) <= K(2) XOR K(3);

M(5) <= K(3);

M(6) <= X(6) XOR X(7);

M(7) <= X(7);

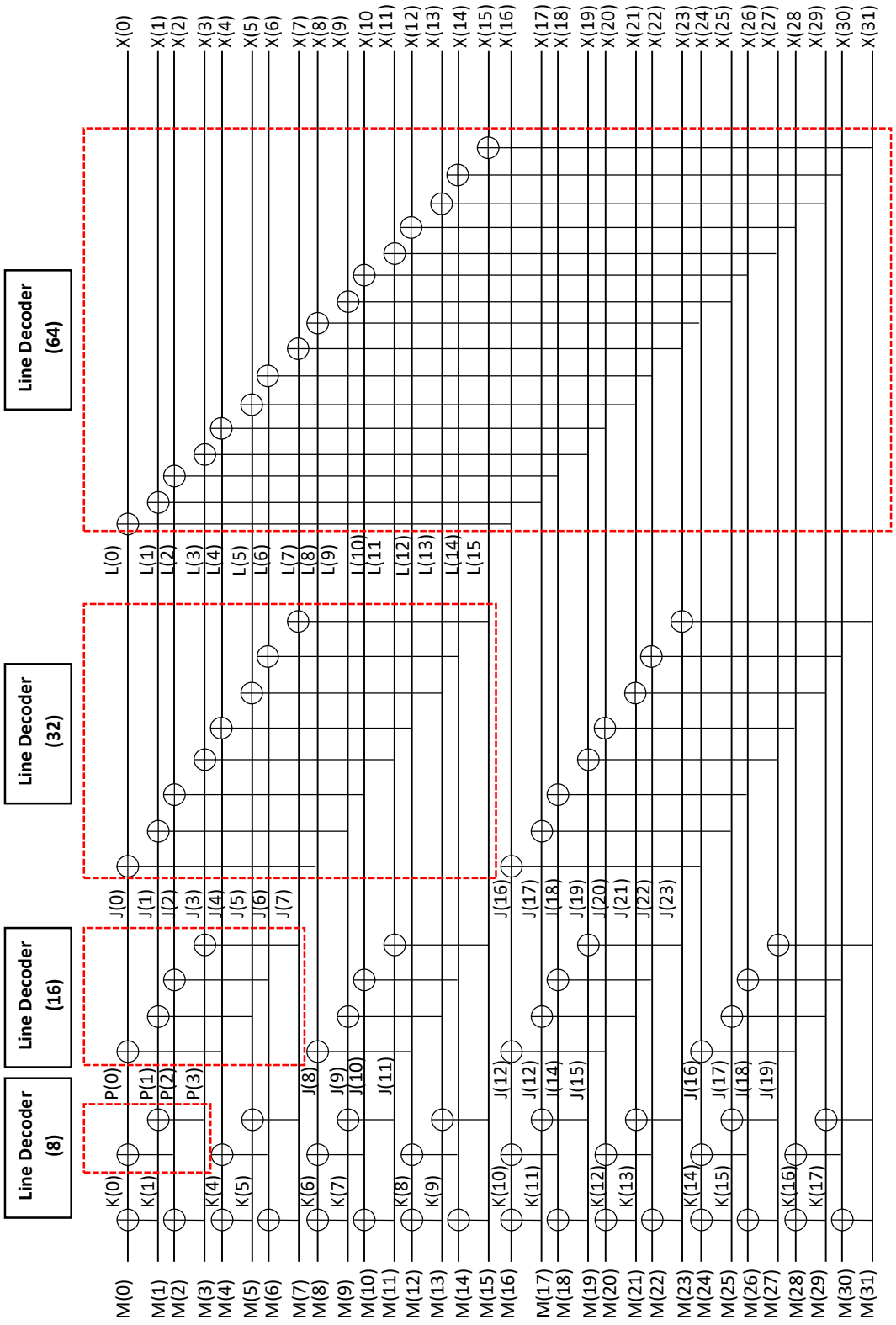


Fig.5.20 Decoder of an (32,20) polar code

Fig. 5.20 depicts the structure of a polar decoder for N=32 that can be used to decode the the input vector M= [M(0) to M(31)] using the generated code word X = [X(0) to X(31)]. The intermediate states are represented as [K(0), K(1), P(0) to P(17), J(0) to J(23), L(0) to L(15)]. While designing the 8-bit polar decoder following representation is considered as follows:

M(0) = polar_message_data(0)
M(1) = polar_ message_ data(1)
M(2) = polar_ message_ data(2)
M(3) = polar_ message_ data(3)
M(4) = polar_ message_ data(4)
M(5) = polar_ message_ data(5)
M(6) = polar_message_data(6)
M(7) = polar_ message_ data(7)
M(8) = polar_ message_ data(8)
⋮
M(31) = polar_ message_ data(31)

X(0) = polar_decoded_data(0)
X(1) = polar_ decoded _data(1)
X(2) = polar_ decoded_ data(2)
X(3) = polar_ decoded_data(3)
X(4) = polar_ decoded_data(4)
X(5) = polar_ decoded_ data(5)
X(6) = polar_ decoded_data(6)
X(7) = polar_ decoded_data(7)
X(8) = polar_ decoded_data(8)
⋮
X(31) = polar_ decoded_data(31)

CHAPTER 6

RESULTS & ANALYSIS

The chapter covers the RTL view of various functional modules, simulation waveforms, FPGA device utilization summary of the turbo, LDPC, and polar encoder & decoder as well as the device utilization summary of all the modules. The chapter also details the FPGA synthesis, experimental verification, and real-time signal processing in chip scope pro-analyzer. The performance of the chip is assessed based on different FPGA parameters and a comparative analysis is carried out between turbo, LDPC, and polar encoder-decoder to estimate the optimal solution for a 5G communication system.

6.1 Xilinx Simulation of LFSR and Gold Code Generator with 5-Bit Input Sequence

The simulation of LFSR-1, LFSR-2, and the gold code generator is done in Xilinx ISE 14.6. The RTL of the same module is depicted in Fig.6.1, which presents the input and output pins utilized in the design. The details for the utilized pins are listed in Table 6.1. In the design, the 5-bit input sequence is processed and controlled using the tap sequence. The tap bit positions are (5, 2) and (5, 4, 2, 1) for LFSR-1 and LFSR-2 respectively.

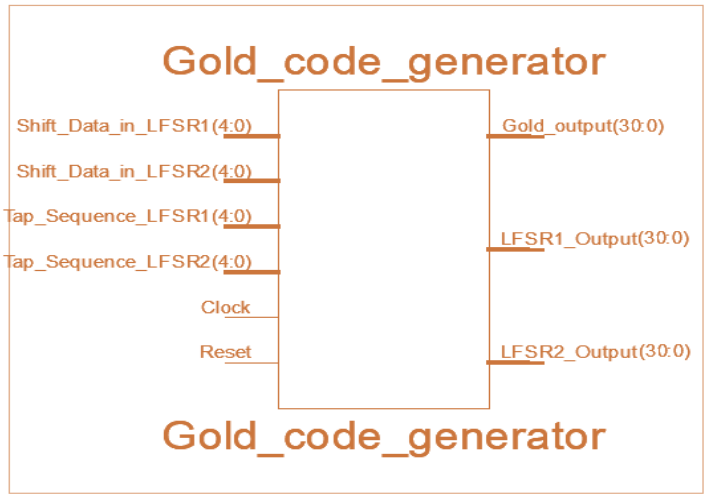


Fig.6.1 RTL of gold code generator with 5-bit input

Table 6.1 Pins details of the LFSR and gold code generator

Pins	**Function**
Shift_data_in_LFSR1(4:0)	It presents the 5-bit input to the LFSR-1 module given as the sequence-1 input
Shift_data_in_LFSR2(4:0)	It presents the 5-bit input to the LFSR-2 module given as the sequence-2 input
Tap_Sequence_LFSR1(4:0)	The taps are the position of the bits that affect the subsequent state. It presents the bit position for feedback, and the output bit is sent back into the leftmost bit after each tap is successively XORed with it. It is the 5-bit input given to the LFSR-1 module.
Tap_Sequence_LFSR2(4:0)	It presents the bit position for feedback for the LFSR-2 module, in which the output bit is sent back into the leftmost bit after each tap is successively XORed.
Clock (1-bit)	Click is the input given to LFSR-1 and LFSR-2 to work on the active edge of the clock signal. In the design, a 50% duty cycle is used.
Reset (1-bit)	Presents the input applied for the rest of the LFSR contents as zero.
LFSR_out1(30:0)	The output of the LFSR-1 module is derived as the 31-bit simulated output.
LFSR_out2(30:0)	The output of the LFSR-2 module is derived as the 31-bit simulated output.
Gold_out(30:0)	The output of the gold code sequence module, which is derived from the 31-bit XORed output

6.1.1 Xilinx ISIM Simulation Waveform of LFSR-1, LFSR-2 and Gold Code Generator for 5-Bit Input Sequence

The simulation of the LFSR-1 is shown in Fig.6.2, the simulation of LFSR-2 is shown in Fig.6.3, and the gold code simulation output waveform is shown with the help of Fig.6.4. The simulation waveform is taken from the Xilinx ISIM simulator [23], [25]. The simulation provides the hardware and timing reports for further analysis [24]. The test inputs are given below.

Test input for LFSR-1: Sequence input, Shift_data_in_LFSR1 (4:0) = "10000", Tap inputs [5, 2] as Tap Sequence_LFSR1 (4:0) = "01001", the output of LFSR-1(30:0) = 000 0100 1011 0011 1110 0011 0111 0101 (in binary) = 04B3E375 (hexadecimal).

Test input for LFSR-2: Sequence input, Shift_data_in_LFSR2 (4:0) = "10000", Tap inputs [5 4 2 1] as Tap_Sequence_LFSR2 (4:0) = "11011", the output of LFSR-1(30:0) = "000 0111 0011 0111 1101 0001 0010 1011" (in binary) = 04B3E375 (hexadecimal). The output of the gold sequence is Gold_out (30:0) = "000 0011 1000 0100 0011 0010 0101 1110" (in binary) = 0384325E (hexadecimal).

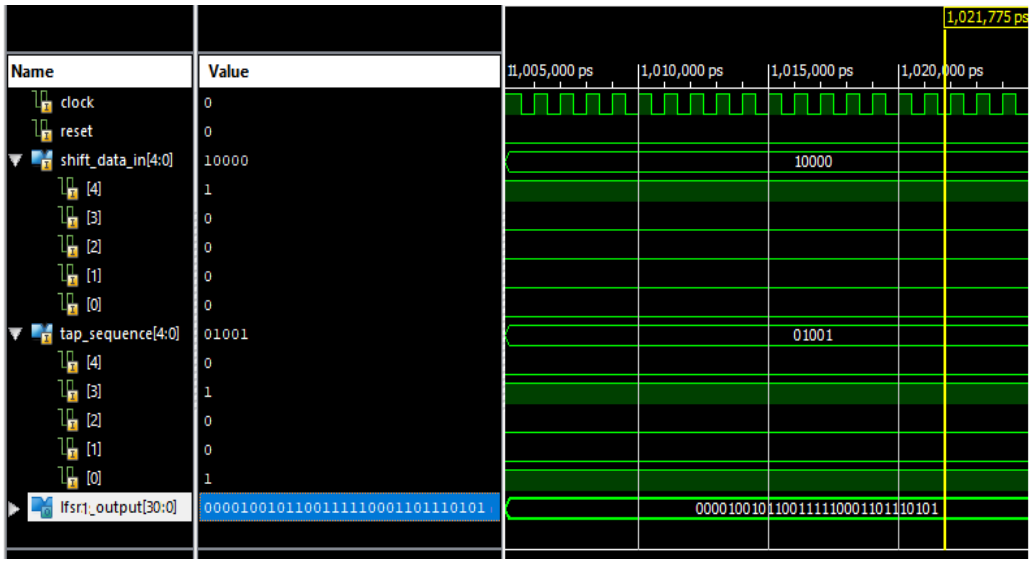


Fig.6.2 The output of LFSR-1 with feedback taps (5, 2)

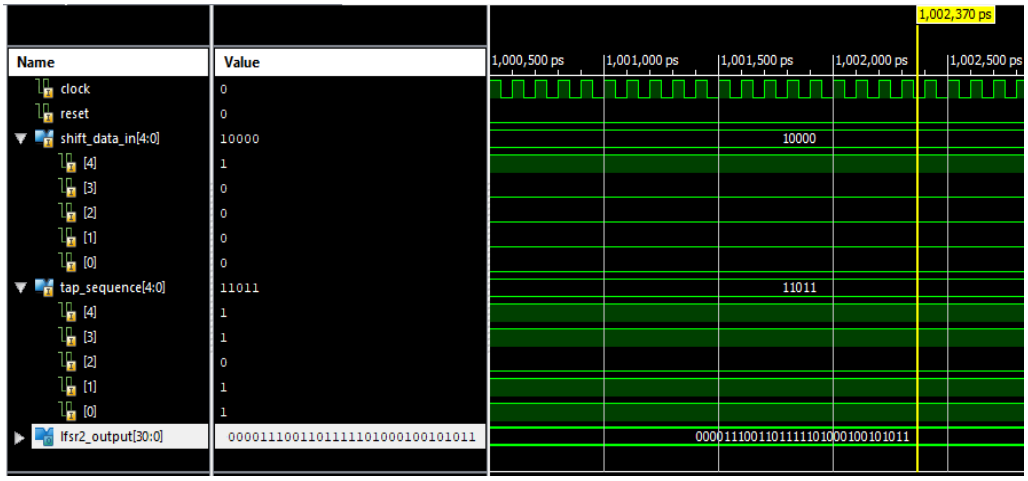


Fig.6.3 The output of LFSR-2 with feedback taps (5, 4, 2, 1)

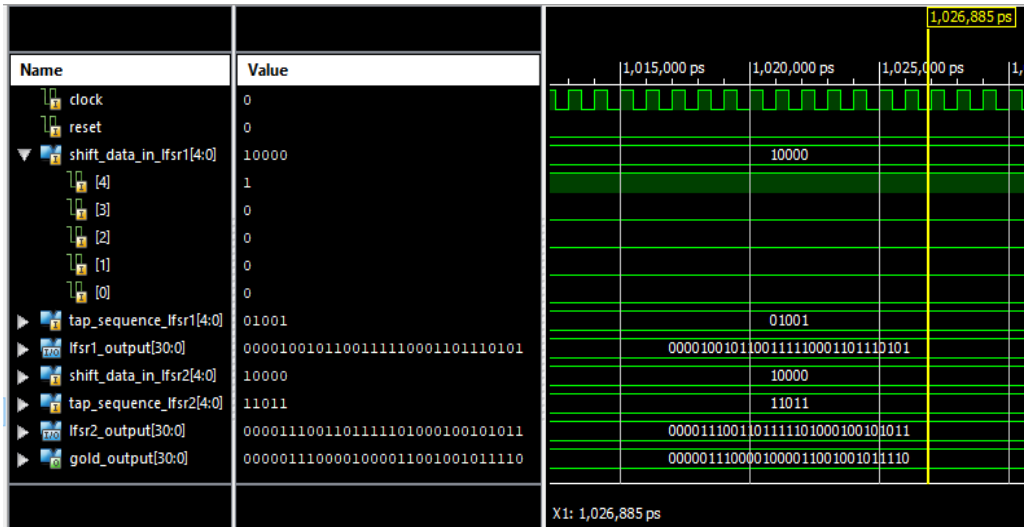


Fig.6.4 Gold sequence output

6.2 Xilinx Simulation of LFSR and Gold Code Generator with 10-bit Input Sequence

Xilinx ISE 14.7 is used to simulate the LFSR module-1, LFSR module-2, and the gold code sequence generator. The explanation of the input and output pins of the design is shown using Fig. 6.5, which presents the register transfer level (RTL) [62, 63] of the designed hardware chip. Table 6.2 contains information on the pins utilized in the design. The architecture uses the tap sequence to process and regulate the 10-bit input sequence. For LFSR-1 and LFSR-2, the tap bit locations are (3, 10) and (2, 3, 6, 8, 9, 10), respectively.

Table 6.2 Pins details of the LFSR and gold code generator

Pins	**Function**
Shift_data_in_LFSR1(9:0)	It provides the sequence-1 input of the LFSR-1 module with the 10-bit input.
Shift_data_in_LFSR2(9:0)	It provides the sequence-2 input of the LFSR-2 module with the 10-bit input.
Tap_Sequence_LFSR1(9:0)	The bit positions called taps are those that affect the next state. The output bit is transmitted back into the leftmost bit after each tap is successively XORed with it. It then displays the bit location for feedback. This is the 10-bit input for the LFSR-1 module.
Tap_Sequence_LFSR2(9:0)	The output bit is delivered back into the leftmost bit after each tap is successively XORed, presenting the bit position for feedback for the LFSR-2 module.
Clock (1-bit)	The clock is the input used by LFSR-1 and LFSR-2 to operate on the clock signal's active edge. The design employs a 50% duty cycle.
Reset (1-bit)	It presents the input used for the initial contents of the LFSR as zero during reset.
LFSR_out1 (1022:0)	The 1023-bit simulated output is the result of the LFSR-1 module output.
LFSR_out2 (1022:0)	The 1023-bit simulated output is the result of the LFSR-2 module output.
Gold_out (1022:0)	The gold sequence output module, which is obtained from the output of the 1023-bit XOR operation

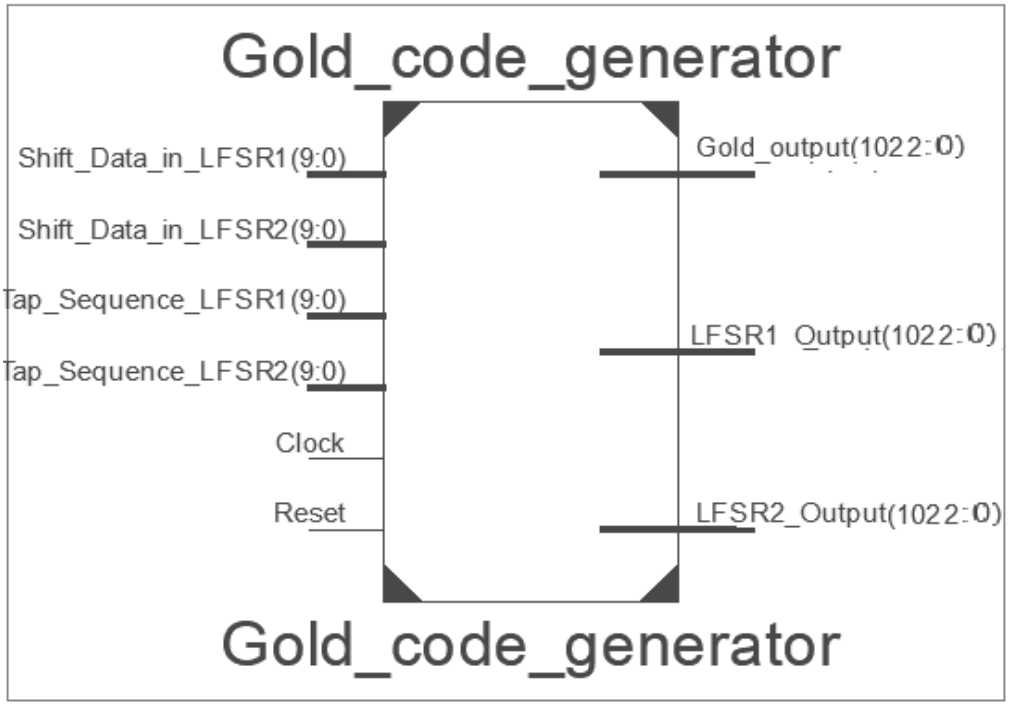


Fig.6.5 Gold code sequence generator RTL description with 10-bit LFSR inputs

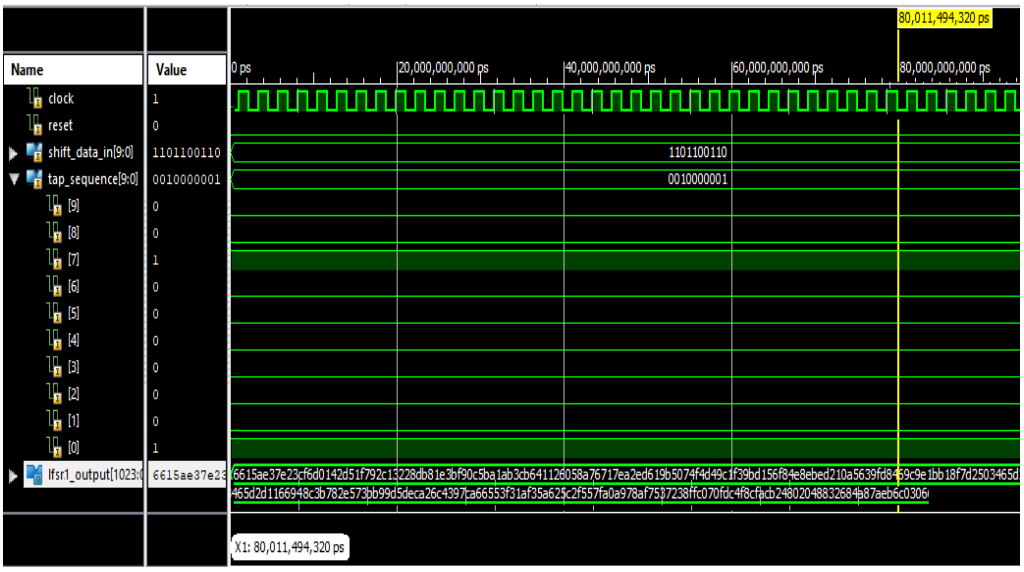


Fig.6.6 Output simulation waveform of LFSR-1 module with tap (3, 10)

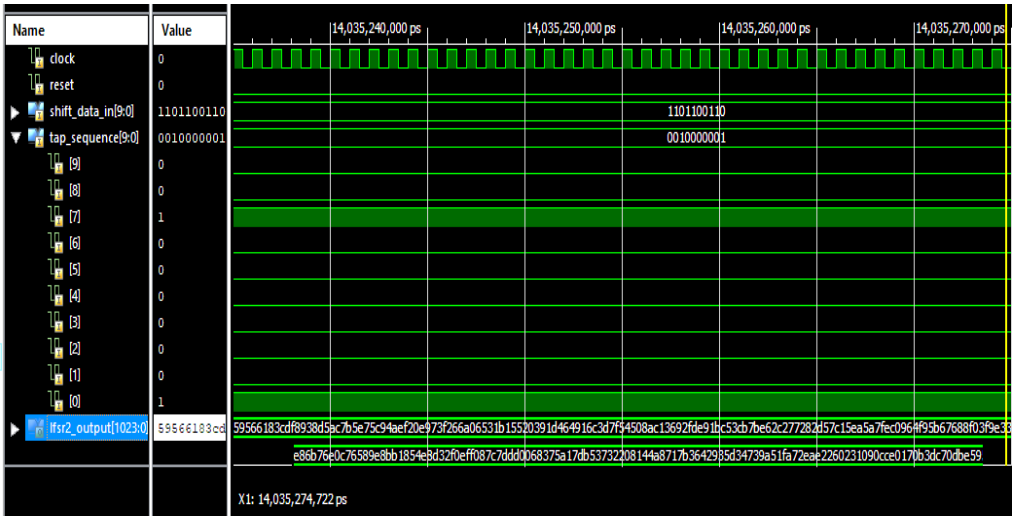


Fig.6.7 Output simulation waveform of LFSR-2 module with tap (2,3,6,8,9,10)

Fig.6.8 Simulation waveform of Gold sequence generated output

Fig.6.6 and Fig.6.7 depict the simulations of the LFSR module-1 and LFSR module-2, respectively. Fig.6.8 illustrates the output waveform of the gold code simulation. The simulation waveform is taken from the Xilinx ISIM simulator [34, 35]. The test inputs are given below. The software and hardware requirements used in the simulation: 64-bit Window-10, DDR4 memory 8 GB, 1. GHz speed, Intel Core i5 8th Gen to run the test cases in Xilinx ISE 14.7 software.

Test simulation input for LFSR-1: The input sequence is given as Shift_data_in_LFSR1 (9:0) = "1101100110", Tap inputs (3, 10) as Tap Sequence_LFSR1 (9:0) = "0010000001", The output presents a 1023-bit sequence for easy understanding in hexadecimal form. LFSR-1 Output data

stream with 1023 iterations is given as = [01100110 00010101 10101110 00110111 11100010 00111100 11110110 11010000 00010100 00101101 01010001 11110111 10010010 11000001 00110010 00101000 11011011 10000001 11100011 10111111 10010000 11000101 10111010 00011010 10110011 11001011 01100100 00010001 00100110 00000101 10001010 01110110 01110001 01111110 10100010 11101101 01100001 10011011 01010000 01110100 11110100 11010100 10011100 00011111 00111001 10111101 00010101 01101111 10000100 11101000 11101011 11101101 00100001 00001010 01010110 00111001 11111101 10000100 01101001 11001001 11100001 10111011 00011000 11110111 11010010 01010000 00110100 01100101 11010010 11010001 00010110 01101001 01001000 11000011 10110111 10000010 11100101 01110011 10111011 10011001 11010101 11011110 11001010 00100110 11000100 00111001 01111100 10100110 01100101 01010011 11110011 00011010 11110011 01011010 01100010 01011100 00101111 01010101 01111111 10100000 10101001 01111000 10101111 01110101 00110111 00100011 10001111 11111100 00000111 00001111 11011100 01001111 10001100 11111010 11001011 00100100 10000000 00100000 01001000 10000011 00100110 10000100 10101000 01111010 11101011 01101100 00000011 0000011] (in binary) = 6615AE37E23CF6D0142D51F792C13228DB81E3BF90C5BA1AB3CB6411 26058A76717EA2ED619B5074F4D49C1F39BD156F84E8EBED210A5639F D8469C9E1BB18F7D2503465D2D1166948C3B782E573BB99D5DECA26C 4397CA66553F31AF35A625C2F557FA0A978AF7537238FFC070FDC4F8C FACB24802048832684A87AEB6C0303 (in hexadecimal).

Test simulation input for LFSR-2: The input sequence is given as Shift_data_in_LFSR1 (9:0) = “1101100110”, Tap inputs (2,3,6,8,9,10) as Tap Sequence_LFSR1 (9:0) = “0110010111”, The output is presented in 1023-bit sequence for easy understanding in hexadecimal form. LFSR-2 Output data stream with 1023 iterations is given as = [01011001 01010110 01100001 10000011 11001101 11111000 10010011 10001101 01011010 11000101 10110101 11100111 01011100 10010100 10101110 11110010 00001110 10010111 00111111 00100110 01101010 00000110 01010011 00011011 00010101 01010010 00000011 10010001 11010100 01100100 10010001

01101100 00111101 01111111 01010100 01010000 10001010 11000001 00110110 10010010 11111101 11101001 00011011 11000101 00111100 01010011 11001011 01111011 11100110 00101100 0010011 01110010 10000010 11010101 01111100 00010101 11101010 01011010 01111111 11101100 00001001 01100100 11111001 01011011 01100111 01101000 10001111 00000011 11111001 11100011 00111110 10000110 10110111 01101110 00001100 01110110 01011000 10011110 10001011 10110001 10000101 01001110 10001101 00110010 11110000 11101111 11110000 10000111 11000111 11011101 11010000 00000110 10000011 01110101 10100001 01111101 10110101 00110111 00110010 00100000 10000001 01000100 10101000 01110001 01111011 00110110 01000010 10010011 01011110 00110100 01110011 10011010 01010001 11111010 01110010 11101010 11100010 00100110 00000010 00110001 00001001 00001100 11001110 00000001 01110000 10110011 11101100 01110000 11101011 11110000] (in binary) = 59566183CDF8938D5AC7B5E75C94AEF20E973F266A06531B15520391D464916C3D7F54508AC13692FDE91BC53CB7BE62C277282D57C15EA5A7FEC0964F95B67688F03F9E33E86B76E0C76589E8BB1854E8D32F0EFF087C7DDD0068375A17DB53732208144A8717B3642935D34739A51FA72EAE2260231090CCE0170B3DC70DBE (in hexadecimal).

Gold sequence output:

The output of the gold sequence generator, gold_output [1022:0] = “00111111 01000011 11001111 10110100 00101111 11000100 01100101 01011101 01001110 11101010 11100100 00010000 11001110 01010101 10011100 11011010 11010101 00010110 11011100 10011001 11111010 11000011 11101001 00000001 10100110 10011001 01100111 10000000 11110010 01100001 00011011 00011010 01001100 00000001 11110110 10111101 11101011 01011010 01100110 11100110 00001001 00111101 10000111 11011010 00000101 00001010 10101011 00001101 01000110 10011111 11000011 11000000 01110110 11001011 00001000 10011100 01011010 01111010 10101001 01011111 10101110 00101110 10101110 10000001 01011010 10100000 00001011 11111011 11100001 00111001 01111101

00011111 10101000 00000100 11010010 00001011 00001101 11001000 10100011 11001101 00111101 00001101 11100101 00101000 00111011 00110001 00000000 11011011 10111000 01010011 10011011 00101101 10101001 01001101 10111001 00001111 01011100 01110111 01110111 10110100 11 100011 11111111 10111000 11000110 01010011 00001010 10111010 00101111 01000000 00110110 01111001 01010000 00101011 11010100 01100101 00000110 11100000 00000011 01011000 00010011 11101010 01100100 10111111 01110001 11010110 10101011 00001110 10111000" (in binary)
=3F43CFB42FC4655D4EEAE410CE559CDAD516DC99FAC3E901A69967
80F2611B1A4C01F6BDEB5A66E6093D87DA050AAB0D469FC3C076CB0
89C5A7AA95FAE2EAE815AA00BFBE1397D1FA804D20B0DC8A3CD3D
0DE5283B3100DBB8539B2DA94DB90F5C7777B4E3FFB8C6530ABA2F4
03679502BD46506E0035813EA64BF71D6AB0EB8 (in hexadecimal).

The 10-bit test input for LFSR-1 is processed with the 10-bit tap sequence and produces the corresponding output-1 according to the defined case and the same is simulated. The 10-bit test input for LFSR-2 is processed with the 10-bit tap sequence and produces the corresponding output-2 according to the defined case and the same is simulated. Output-1 and output-2 are used to provide the claimed output of 1024-bit as listed in the binary Gold sequence output. The analysis of the chip design is done according to Virtex-5 FPGA hardware and timing summary reported by the software. The design provided 9.285 ns of delay and 310 MHz of frequency support.

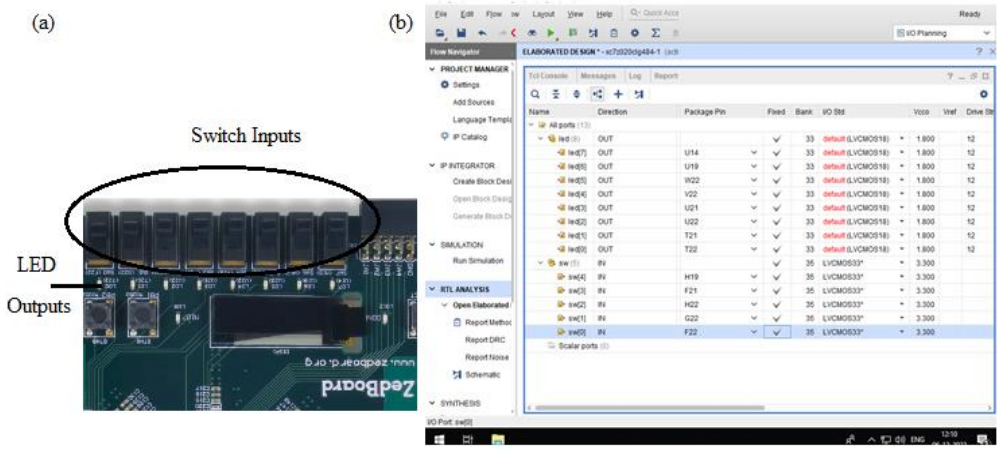


Fig.6.9 Pin assignment in FPGA and logic verification

Fig.6.10 Experimental verification on FPGA

The data of the simulation waveform is verified in a real-time environment for that the FPGA kit is configured with the pin assignment as depicted in Fig.6.9. The bit file is burned into FPGA as shown in Fig.6.10. The switches are applied inputs and “01001”, and “11011” for both cases respectively, and the corresponding output is verified on LED output byte by a byte which is “00000100 10110011 11100011 01110101” and “0000 0111 00110111 11010001 00101011” respectively. The verified gold output on the kit is “00000011 10000100 00110010 01011110”. The performance of the design is compared with the existing work. In [26], the reported delay is 6.205 ns and the frequency is 74.464 MHz on Xilinx Virtex-6 FPGA. Our design provides a maximum delay of 2.192 ns and 215 MHz clock frequency which means minimum delay and good frequency in comparison to existing work.

6.3 Xilinx Simulation of Turbo Encoder & Decoder

In Xilinx ISE 14.7, behavioral modeling is used to develop the turbo encoder and decoder. The encoder and decoder integrated chip RTL view are shown in Fig.6.11 Table 6.3 lists the specifics of RTL pins. Fig.6.12 depicts the waveform in Xilinx ISIM for the decoder for test-1 and test-2 in binary without reception error. Fig.6.13 depicts the waveform in Xilinx ISIM for the decoder for test-1 and test-2 in hexadecimal without reception error. Fig.6.14 depicts the

waveform in Xilinx ISIM for the decoder for test-3 and test-4 in binary with a reception error. Fig.6.15 depicts the waveform in Xilinx ISIM for the decoder for test-3 and test-4 in hexadecimal with a reception error. The major inputs and outputs of the decoder's main stimuli are displayed on the simulation test bench.

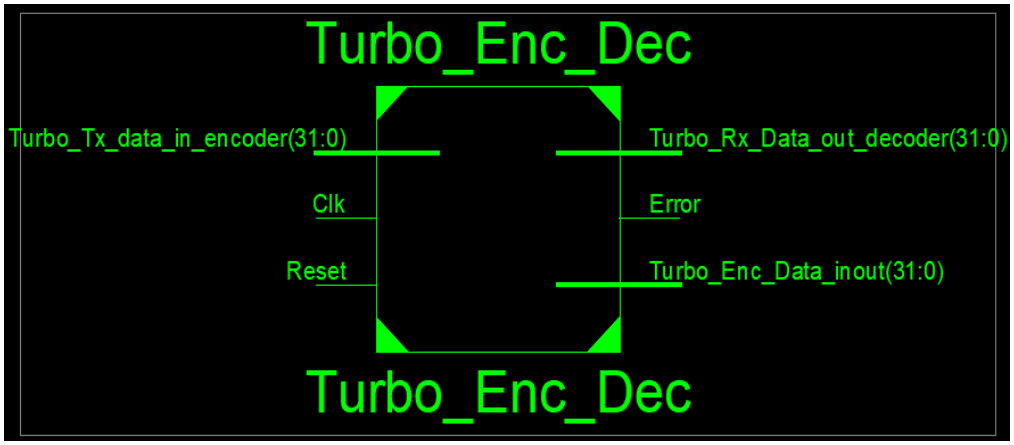


Fig.6.11 RTL view of the integrated Turbo encoder and decoder

Table 6.3 Pin description of Turbo encoder module and decoder module RTL chip

Pins	**Direction**	**Description**
Clk	Input	The clock signal has a 50% duty cycle and is an input clock signal that can provide a rising edge-based clock signal.
Reset	Input	To maintain the contents 0 for the encoder and decoder, reset is the chip design's default input.
Turbo_tx_data_in_encoder [31:0]	Input	It gives the turbo encoder provided at the transmitter section of the design the 32-bit input message.
Turbo_rx_data_out_decoder [31:0]	Output	The 32-bit turbo-encoded output can be treated as an AWGN channel's input.
Turbo_enc_data_inout [31:0]	Inout	It represents the 32-bit output of the AWGN communication channel and feedback to the turbo decoder.
Error	Output	It is the indicator output that a correct or incorrect message is received in the decoder section.

6.3.1 Xilinx ISIM Simulation Waveform of Turbo Encoder & Decoder

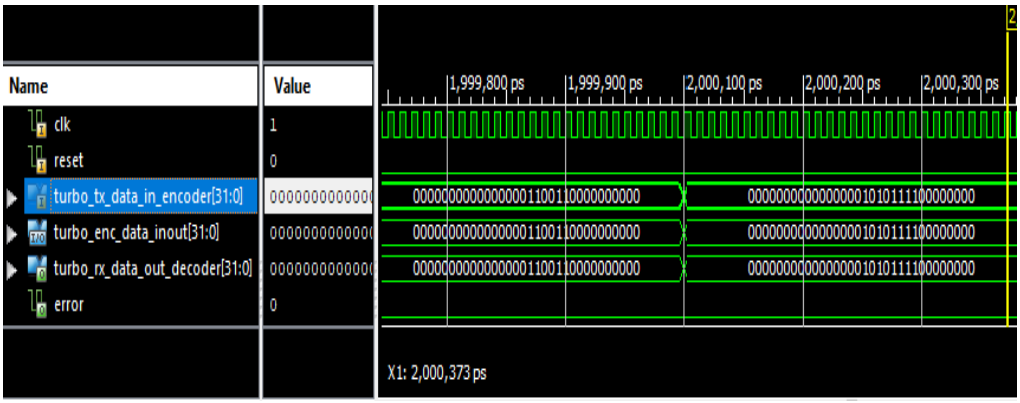


Fig.6.12. Simulation waveform in Xilinx ISIM for turbo encoder-decoder with test-1 and test-2 in binary form

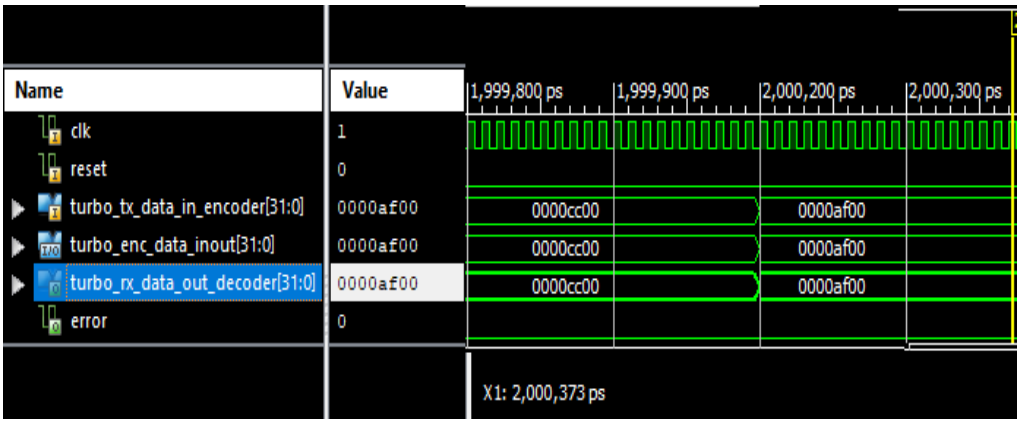


Fig.6.13 Simulation waveform in Xilinx ISIM for encoder- decoder with test-1 and test-2 in hexadecimal form

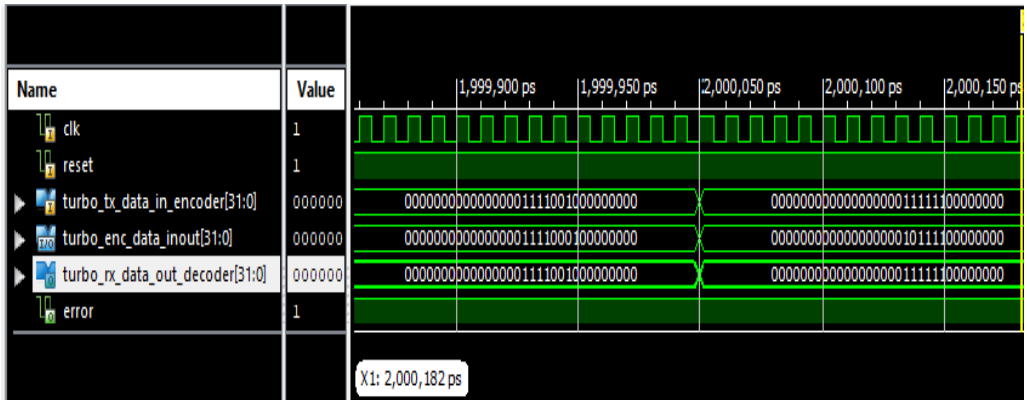


Fig.6.14 Simulation waveform in Xilinx ISIM for turbo encoder- decoder decoder with test-3 and test-4 in binary form

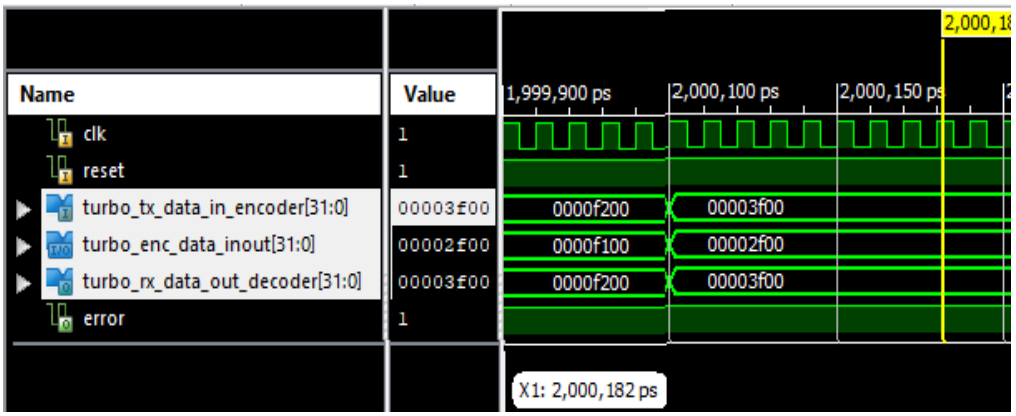


Fig.6.15. Waveform in Xilinx ISIM for decoder with test-3 and test-4 in hexadecimal

Test Case 1 for Turbo Encoder/ Decoder: Apply Clk = rising edge clock signal, reset = '0', turbo_tx_data_in_encoder [31:0] = "0000 0000 0000 0000 1100 1100 0000 0000" (binary) = 0000CC00 (hexadecimal), then turbo_enc_data_inout [31:0] = "0000 0000 0000 0000 1100 1100 0000 0000" (binary) = 0000CC00 (hexadecimal), and turbo_rx_data_out_decoder [31:0] = "0000 0000 0000 0000 1100 1100 0000 0000" (binary) = 0000CC00 (hexadecimal), with error = '0'

Test Case 2 for Turbo Encoder/ Decoder: Apply Clk = rising edge clock signal, reset = '0', turbo_tx_data_in_encoder [31:0] = "0000 0000 0000 0000 1010 1111 0000 0000" (binary) = 0000AF00 (hexadecimal), then turbo_enc_data_inout [31:0] = "0000 0000 0000 0000 1010 1111 0000 0000" (binary) = 0000AF00 (hexadecimal), and turbo_rx_data_out_decoder [31:0] = "0000 0000 0000 0000 1010 1111 0000 0000" (binary) = 0000AF00 (hexadecimal) with error = '0'

Test Case 3 for Turbo Encoder/ Decoder: Apply Clk = rising edge clock signal, reset = '0', turbo_tx_data_in_encoder [31:0] = "0000 0000 0000 0000 1111 0010 0000 0000" (binary) = 0000F200 (hexadecimal), then turbo_enc_data_inout [31:0] = "0000 0000 0000 0000 1111 0001 0000 0000" (binary) = 0000F100 (hexadecimal), and turbo_rx_data_out_decoder [31:0] = "0000 0000 0000 0000 1111 0010 0000 0000" (binary) = 0000F200 (hexadecimal) with error = '1'

Test Case 4 for Turbo Encoder/ Decoder: Apply Clk = rising edge clock signal, reset = '0', turbo_tx_data_in_encoder [31:0] = "0000 0000 0000 0000 0011 1111 0000 0000" (binary) = 00003F00 (hexadecimal), then

turbo_enc_data_inout [31:0] = “0000 0000 0000 0000 0010 1111 0000 0000” (binary) = 00002F00 (hexadecimal), and turbo_rx_data_out_decoder [31:0] = “0000 0000 0000 0000 0011 1111 0000 0000” (binary) = 00003F00 (hexadecimal) with error = ‘1’.

Table 6.4 summarizes the hardware parameters of the turbo encoder and decoder hardware with MAP algorithm in Xilinx ISE 14.7 software for various parameters targeting the Virtex-5 FPGA. Slices, flip-flops, LUTs, and IOBs are the hardware parameters [29]. Table 6.5 shows the results of timing simulations for the turbo encoder and decoder chip hardware for time-concerned parameters like maximum supporting frequency, maximum and minimum duration before and after the clock signal (ns), and minimum period (ns).

Table 6.4 Summary of FPGA hardware usage

Hardware	**Turbo Codes**		**Turbo Codes with Max Log MAP**	
	Encoder	**Decoder**	**Encoder**	**Decoder**
Slices	75	150	67	140
Flip Flops	87	103	77	80
LUTs	120	125	103	107
IOBs	64	96	64	96
GCLKs	1	1	1	1

Table 6.5 Parameters connected to timing information

Timing Parameter	**Turbo Codes**		**Turbo Codes with Max Log MAP**	
	Encoder	Decoder	Encoder	Decoder
Frequency (MHz)	314.0	320.0	332.00	355.0
Minimum period (ns)	1.710	1.420	1.6120	1.390
Time (minimum) before clock (ns)	2.109	2.917	1.964	2.851
Time (maximum) after clock (ns)	4.320	5.325	4.124	4.109
Combinational path delay (ns)	8.139	9.662	6.700	8.350
Speed Grade	-5	-5	-5	-5

Fig.6.16 presents the FPGA hardware utilization graph for the turbo encoder and decoder with a simple and integrated Max log-MAP algorithm chip. Fig.6.17 presents the time-related parameters of FPGA hardware. The combinational path delay and related time parameters are reduced by the integration of the Max log-MAP procedure in the turbo encoder and decoder hardware due to parallel execution.

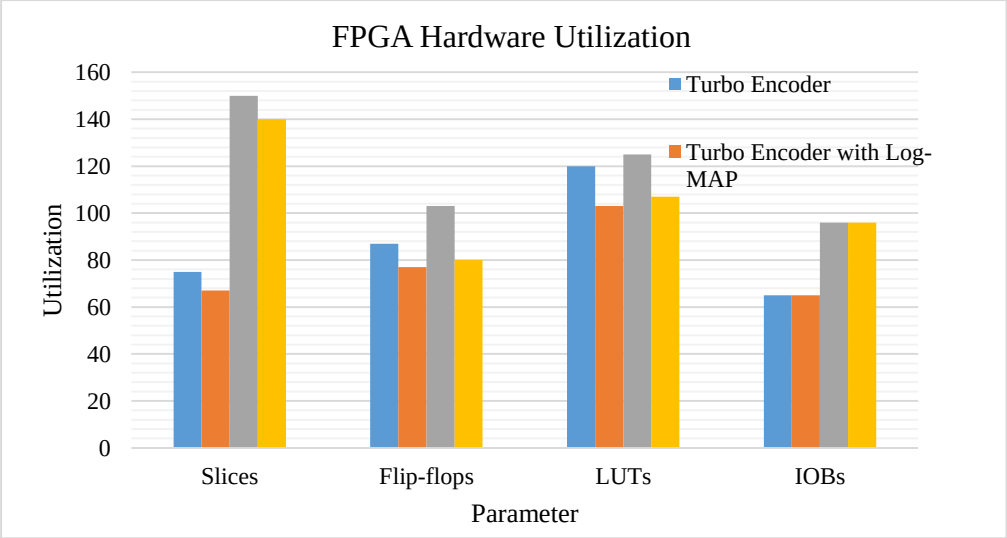


Fig.6.16. FPGA hardware usage

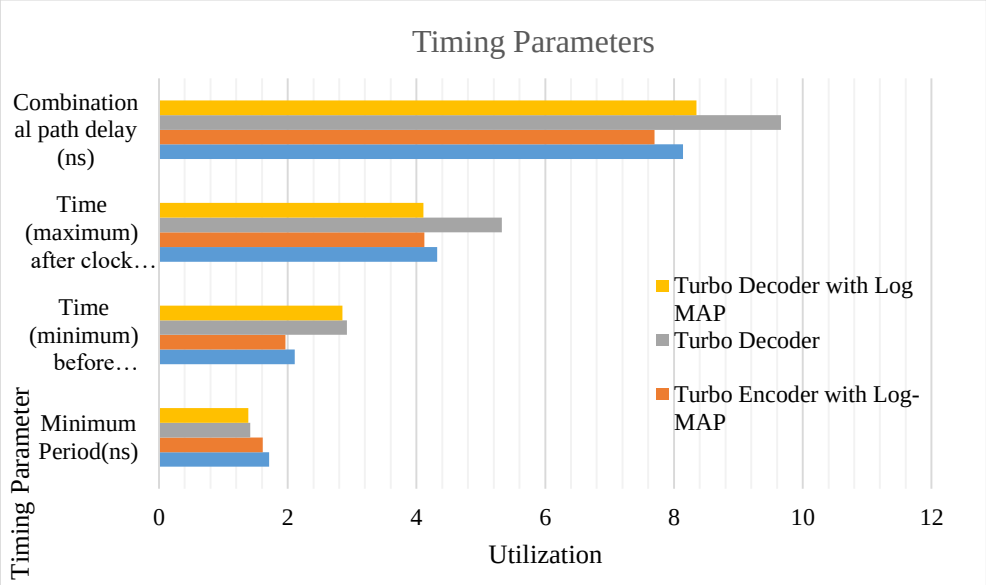


Fig.6.17. FPGA timing parameters

The major motivation for parallel implementation is to efficiently execute code, as it saves time and allows programmers to be executed in a shorter clock time. There is significantly less disruption for workers and equipment because multiple processes can run simultaneously. The encoder and decoder both utilize fewer hardware resources in comparison to the turbo encoder and decoder due to the parallel execution of the block. The designed hardware chip reduces the hardware utilization in the FPGA synthesis.

6.4 Xilinx Simulation of LDPC Encoder & Decoder

The chip design is done using Xilinx ISE 14.7 software. Fig.6.18 presents the RTL view of the turbo encoder and decoder. Fig.6.19 presents the RTL view of the LDPC encoder and decoder. Table 6.6 lists the description of the pins used in the design of the encoder and decoder chip.

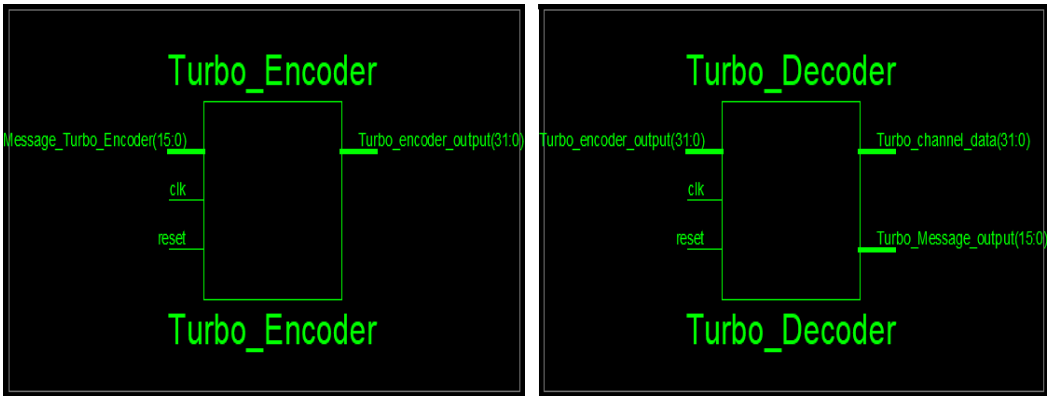


Fig.6.18 RTL view of Turbo encoder and decoder

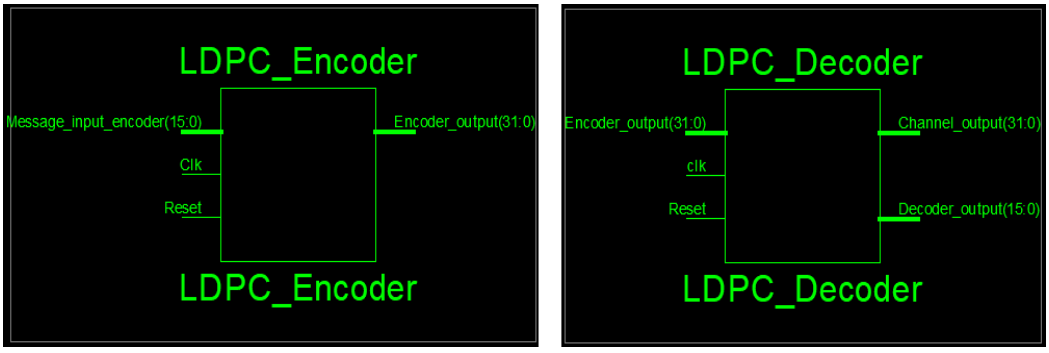


Fig.6.19 RTL view of LDPC encoder and decoder

Table 6.6 Pin description of LDPC encoder and decoder designed RTL chip

Pins	**Direction**	**Description**
Clk	Input	The clock signal is an input clock signal applicable to offer the rising edge-based clock signal, with the 50 % duty cycle.
Reset	Input	Reset is the default value of input to the chip design and it is used to keep the contents zero for the encoder and decoder.
Message_Turbo_Encoder <15:0>	Input	It presents a 16-bit input message to the turbo encoder, given at the transmitter section in the design
Turbo_Encoder_output <31:0>	Inout	It is the turbo-encoded output of 32-bit and can be processed as input to the AWGN channel.
Turbo_Channel_Data <31:0>	Inout	It denotes the output of the AWGN channel and input to the turbo decoder.
Turbo_Message_Output <15:0>	Output	It is the 16-bit output of the turbo decoder as the decoded signal or message at the receiver end.
Message_Input_Encoder <15:0>	Input	It presents the 16-bit input message to the LDPC encoder, given in the transmitter section in the design
Encoder_output<31:0>	Inout	It is an LDPC-encoded output of 32-bit and can be processed as input to the AWGN channel.
Channel_Output <31:0>	Inout	It denotes the output of the AWGN Channel and input to the LDPC decoder.
Decoder_Output <15:0>	Output	It is the 16-bit output of the LDPC decoder as the decoded signal or message at the receiver end.

Table 6.6 gives a detailed description of the LDPC encoder and decoder-designed RTL chip. The chip design of the LDPC code comprises the matrix multiplication of the input message in matrix form and the generator matrix denoted by 'G'. If the size of the input message is considered to be 16-bit, and the encoder output is 32-bit data, then the generator matrix is of size (16 x 32). The generator matrix requires 32 registers to store the contents. The matrix multiplication is done using AND logic and OR logic between the input message (1 x 16) and the generator matrix. The VHDL programming is used and the corresponding test bench is used to perform the functional simulation. The output of the turbo and LDPC encoder passes through the AWGN channel. The behavior simulation of the turbo and LDPC encoder and decoder chip is done in the Xilinx ISIM simulator. Fig.6.20 presents Xilinx ISIM simulation for turbo encoder and decoder. Fig.6.21 presents Xilinx ISIM simulation for the LDPC encoder and decoder. The simulation test case stimuli are given.

6.4.1 Xilinx ISIM Simulation Waveform of LDPC Encoder & Decoder

Test case-1 (Turbo Encoder/ Decoder): Apply clk = rising edge clock signal, reset = '0', message_turbo_encoder [15:0] = "0000000011001100", then turbo encoder output is turbo_encoder_output ="00000000000000001100110000000000", turbo_channel_data <31:0> = "00000000000000001100110000000000" and decoded output is turbo_message_output <15:0> = "0000000011001100".

Test case -2 (Turbo Encoder/ Decoder): Apply clk = rising edge clock signal, reset = '0', message_turbo_encoder [15:0] = "0000000011111100", then encoder output is turbo_encoder_output ="00000000000000001111110000000000", turbo_channel_data <31:0> = "00000000000000001111110000000000" and decoded output is turbo_message_output <15:0> = "0000000011111100".

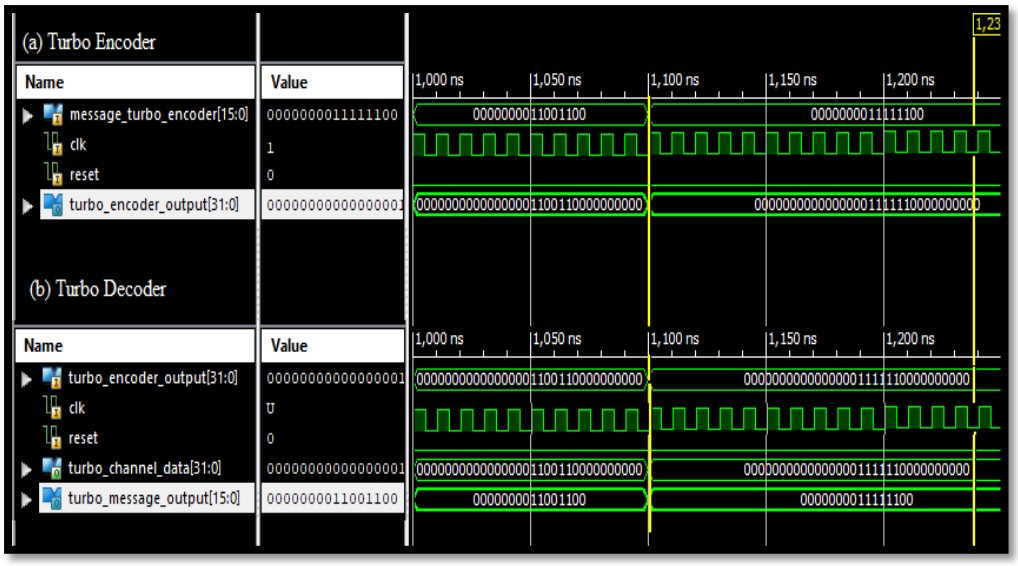


Fig.6.20 Xilinx ISIM simulation for Turbo encoder and decoder

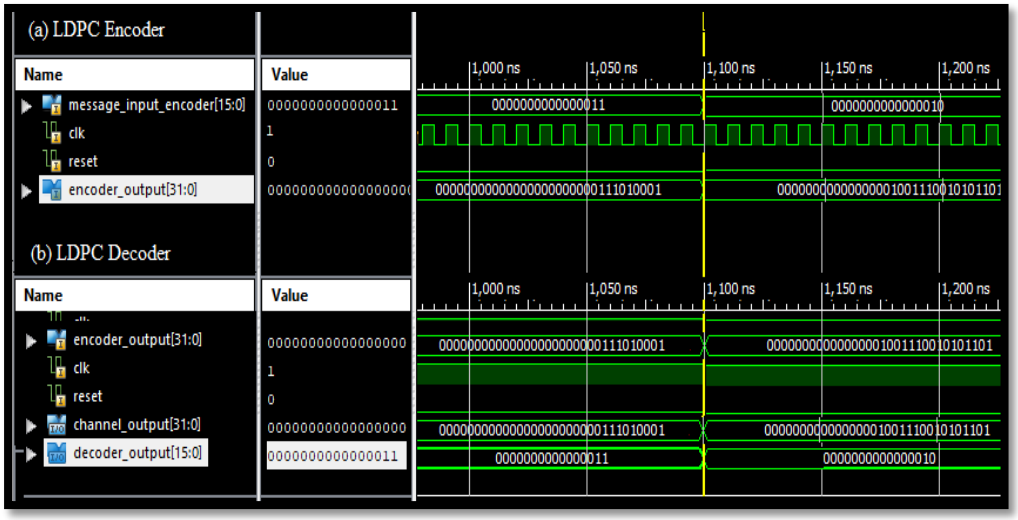


Fig.6.21 Xilinx ISIM simulation for LDPC encoder and decoder

Test case-3 (LDPC Encoder/ Decoder): Apply Clk = rising edge clock signal, reset = '0', message_input_encoder [15:0] = "0000000000000011" then the encoder output is encoded_output = "00000000000000000000000111010001", channel_output <31:0> = "00000000000000000000000111010001" and decoded output of LDPC decoder is decoder_output <15:0> = "0000000000000011".

Test case -4 (LDPC Encoder/ Decoder): Apply Clk = rising edge clock signal, reset = '0', message_input_encoder [15:0] = "0000000000000010" then the encoder output is encoded_output =

“00000000000000001001110010101101”, channel_output <31:0> = “00000000000000001001110010101101” and decoded output of LDPC decoder is decoder_output <15:0> = “0000000000000010”.

Table 6.7 presents the hardware parameter summary of turbo and LDPC encoder and decoder hardware for different parameters targeting Virtex-5 FPGA using Xilinx ISE 14.7 software. The hardware design parameters are flip-flops, LUTs, IOBs, and slices. Table 6.8 presents the timing results for turbo and LDPC encoder and decoder like the minimum-maximum time before and after the clock signal (ns), minimum period (ns), and maximum frequency.

Table 6.7 FPGA hardware utilization summary

Hardware	**Turbo Codes**		**LDPC Codes**	
	Encoder	**Decoder**	Encoder	Decoder
Slices	60/12480	145/12480	32/12480	110/12480
Flip Flops	72/12480	88/12480	45/12480	52/12480
LUTs	105/330	110/330	58/330	70/330
IOBs	50/172	72/172	50/172	72/172
GCLKs	1/32	1/32	1/32	1/32

Table 6.8 Timing information-related parameters

Timing Parameters	**Turbo Codes**		**LDPC Codes**	
	Encoder	**Decoder**	**Encoder**	**Decoder**
Frequency (MHz)	310.0	315.0	375.00	390.0
Minimum period(ns)	1.615	1.365	1.053	1.275
Time (minimum) before clock (ns)	1.980	2.825	1.750	2.113
Time (maximum) after clock (ns)	4.174	4.250	2.710	2.914
Combinational path delay (ns)	6.769	8.640	6.513	6.302
Speed Grade	-5	-5	-5	-5

6.5 Xilinx Simulation of Polar Encoder & Decoder

A polar encoder and decoder are implemented using VHDL. Fig.6.22 depicts the RTL view of an 8-bit polar encoder with an input sequence denoted by M

(7:0) which is an 8-bit data sequence. The output of the encoder is represented by X(7:0) which is an 8-bit generated output sequence. Fig.6.23 depicts the detailed RTL view of an 8-bit polar encoder. Table. 6.9 gives the pin description of an 8-bit polar encoder RTL chip.

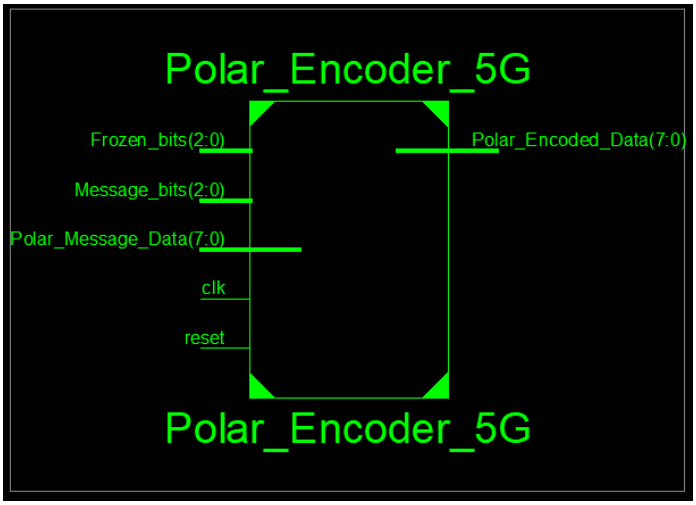


Fig.6.22 RTL view of 8-bit Polar encoder

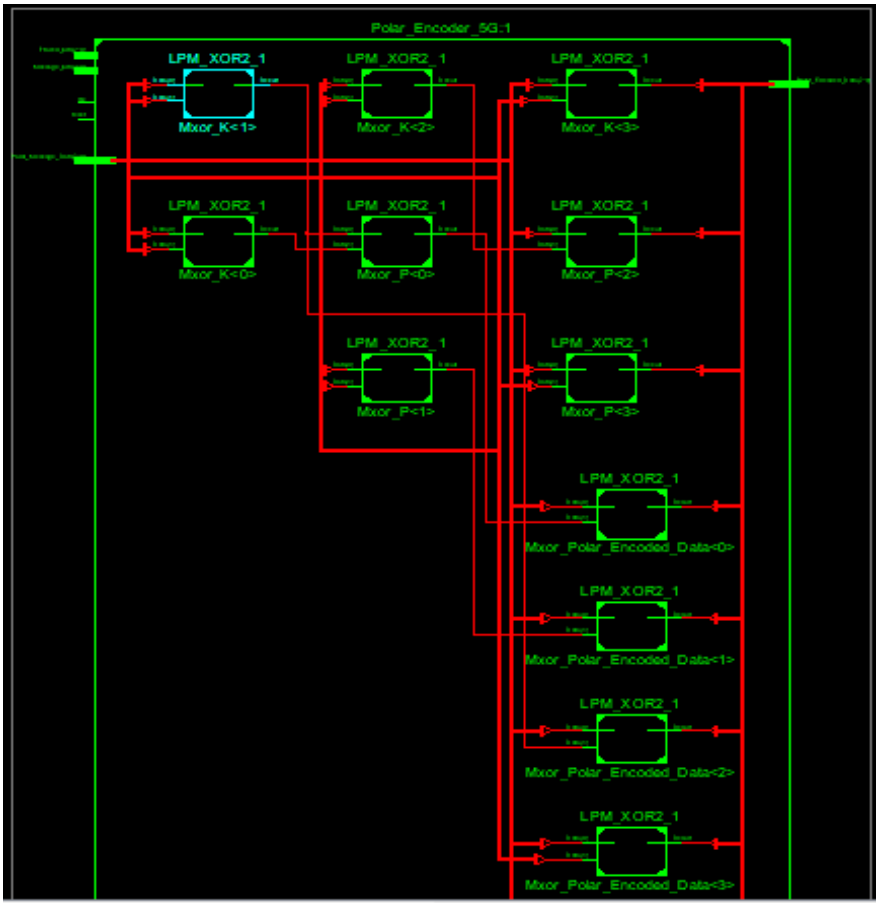


Fig.6.23 Detailed RTL view of 8-bit Polar encoder

Table 6.9 Pin description of an 8-bit Polar encoder RTL chip

Pins	**Direction**	**Description**
Clk	Input	The clock signal is an input clock signal applicable to offer the rising edge-based clock signal, with the 50 % duty cycle.
Reset	Input	Is the default value of input to the chip design and it is used to keep the contents zero for the encoder and decoder.
M <7:0>	Input	It presents an 8-bit input message to the polar encoder, given at the transmitter section in the design
Polar_Encoded_Data<7:0>	Inout	It is the polar-encoded output of 8-bit and can be processed as input to the AWGN channel.
Frozen_bits <2:0>	Input	It is a 3-bit data sequence representing the bits to be '0' in the transmitted data sequence.
Message_bits <2:0>	Input	It is a 3-bit data sequence representing the bits to be '1' in the transmitted data sequence.

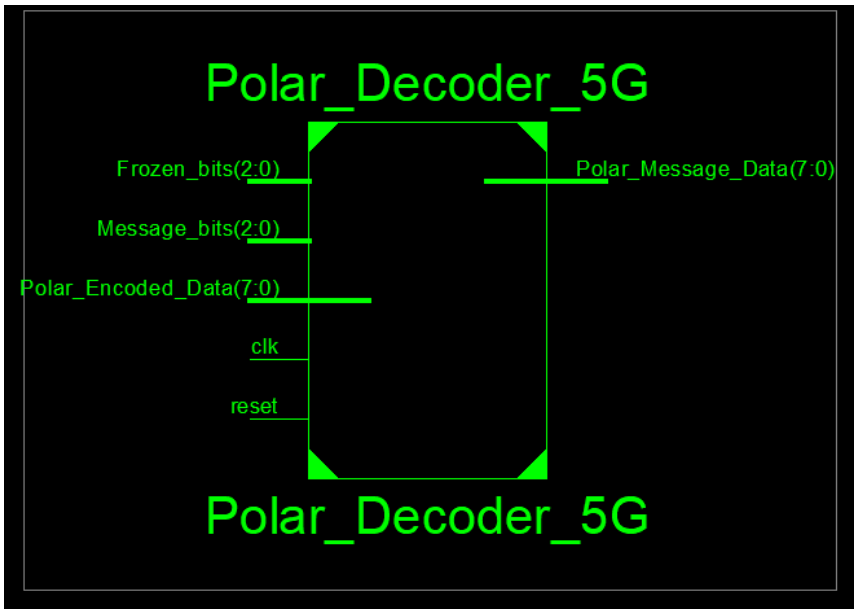


Fig.6.24 RTL view of an 8-bit Polar decoder

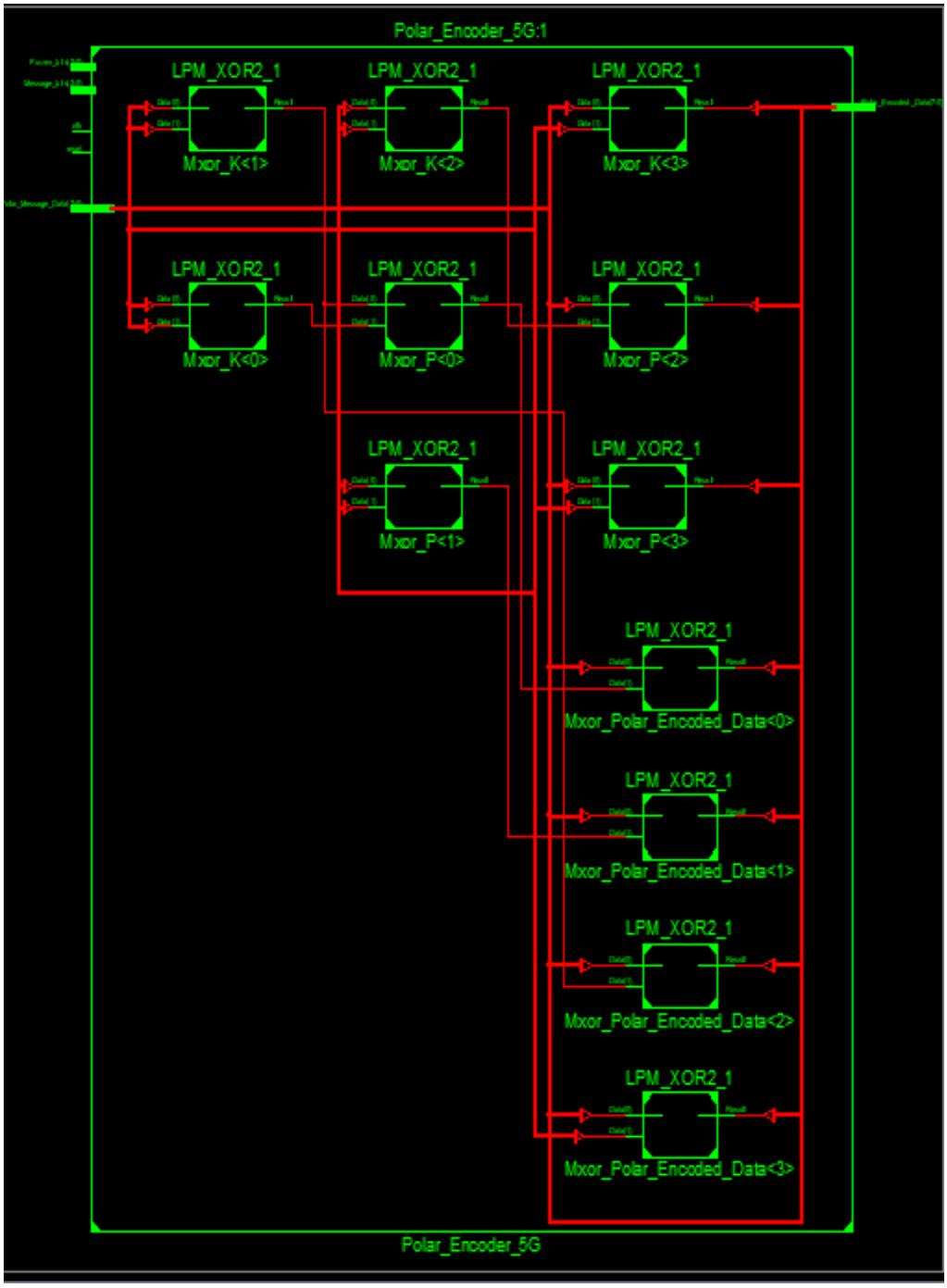


Fig.6.25 Detailed RTL view of an 8-bit Polar decoder

Similarly, the input to the decoder is represented by X(7:0) which is an 8-bit sequence and the output of the decoder is an 8-bit sequence represented by M (7:0) as shown in Fig.6.24. Fig.6.25 depicts the detailed RTL view of an 8-bit polar decoder. Table. 6.10 gives the pin description of an 8-bit polar decoder RTL chip.

Table 6.10 Pin description of an 8-bit Polar decoder RTL chip

Pins	**Direction**	**Description**
Clk	Input	The clock signal is an input clock signal applicable to offer the rising edge-based clock signal, with the 50 % duty cycle.
Reset	Input	Reset is the default input to the chip design and it is used to keep the contents zero for the encoder and decoder.
M <7:0>	Output	It is the received polar-decoded output of 8-bit from the polar decoder.
Polar_Encoded_Data<7:0>	Input	It presents an 8-bit input message to the polar decoder, given at the receiver section in the design.
Frozen_bits <2:0>	Input	It is a 3-bit data sequence representing the bits to be '0' in the transmitted data sequence.
Message_bits <2:0>	Input	It is a 3-bit data sequence representing the bits to be '1' in the transmitted data sequence.

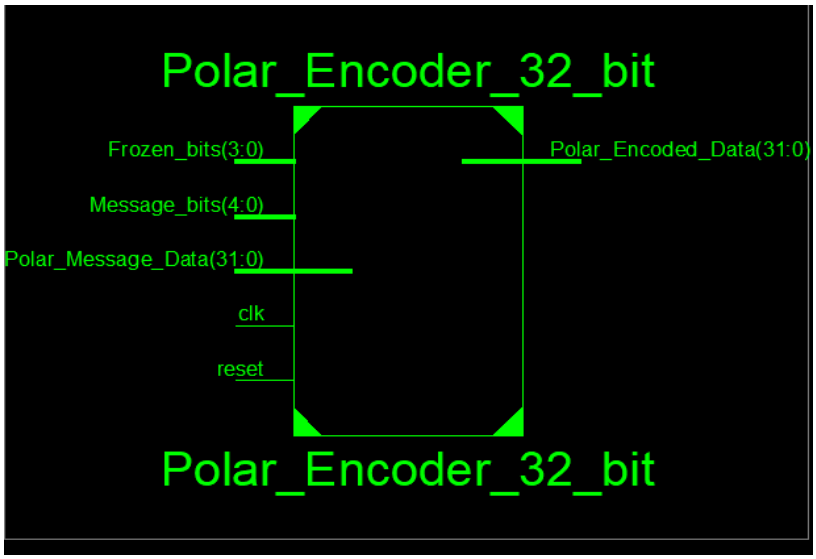


Fig.6.26 RTL view of a 32-bit Polar encoder

Fig.6.26 depicts the RTL view of a 32-bit polar encoder with an input sequence denoted by M (31:0) which is a 32-bit data sequence. The output of the encoder is represented by X(31:0) which is a 32-bit generated output sequence. Fig.6.27 depicts the detailed RTL view of a 32-bit polar encoder. Table. 6.11 gives the pin description of a 32-bit polar encoder RTL chip.

Table 6.11 Pin description of a 32-bit Polar encoder RTL chip

Pins	**Direction**	**Description**
Clk	Input	lock signal is an input clock signal applicable ed clock signal, with the 50 % duty cycle.
Reset	Input	Reset is the default input to the chip design and it is used to keep the contents zero for the encoder and decoder.
M <31:0>	Input	It presents a 32-bit input message to the polar encoder, given at the transmitter section in the design
Polar_Encoded _Data<31:0>	Inout	It is the polar-encoded output of 32-bit and can be processed as input to the AWGN channel.
Frozen_bits <3:0>	Input	It is a 4-bit data sequence representing the bits to be '0' in the transmitted data sequence.
Message_bits <4:0>	Input	It is a 5-bit data sequence representing the bits to be '1' in the transmitted data sequence.

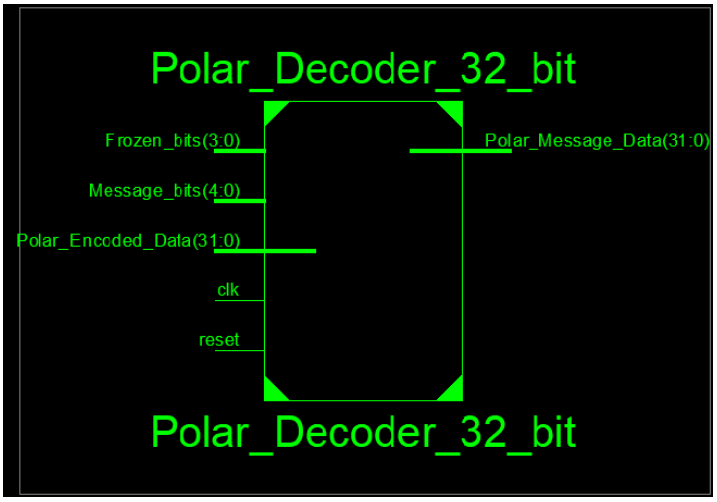


Fig.6.27 RTL view of a 32-bit Polar decoder

Similarly, the input to the decoder is represented by X(31:0) which is a 32-bit sequence and the output of the decoder is a 32-bit sequence represented by M (31:0) as shown in Fig.6.24. Fig.6.25 depicts the detailed RTL view of an 8-bit polar decoder. Table. 6.12 gives the pin description of a 32-bit polar decoder RTL chip.

Table 6.12 Pin description of a 32-bit Polar decoder RTL chip

Pins	**Direction**	**Description**
Clk	Input	The clock signal is an input clock signal applicable to offer the rising edge-based clock signal, with the 50 % duty cycle.
Reset	Input	Reset is the default value of input to the chip design and it is used to keep the contents zero for the encoder and decoder.
M <31:0>	Output	It is the received polar-decoded output of 32-bit from the polar decoder.
Polar_Encoded_Data<31:0>	Input	It presents a 32-bit input message to the polar decoder, given at the receiver section in the design.
Frozen_bits <3:0>	Input	It is a 4-bit data sequence representing the bits to be ‘0’ in the transmitted data sequence.
Message_bits <4:0>	Input	It is a 5-bit data sequence representing the bits to be ‘1’ in the transmitted data sequence.

6.5.1 Xilinx ISIM Simulation Waveform of an 8-bit Polar Encoder & Decoder

Fig.6.28 represents the simulation waveform for an 8-bit Polar encoder in binary form and Fig.6.29 represents the simulation waveform for a 8-bit Polar encoder in decimal form.

Test case (Polar Encoder): Apply Clk = rising edge clock signal, reset = ‘0’, M [7:0] = “00010111”, then polar encoder output is X[7:0] =“00010110”.

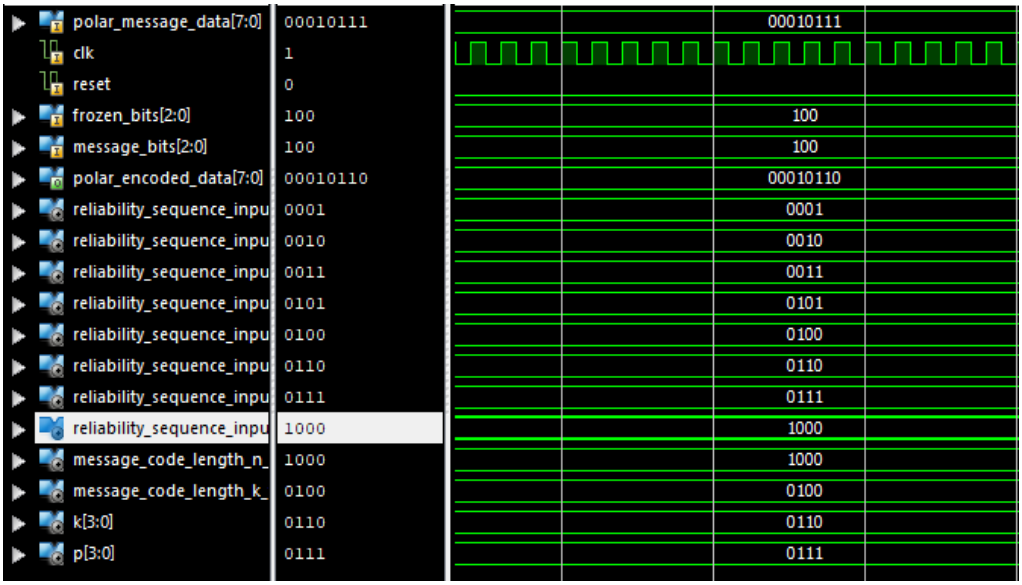


Fig.6.28 Simulation waveform for an 8-bit Polar encoder in binary form

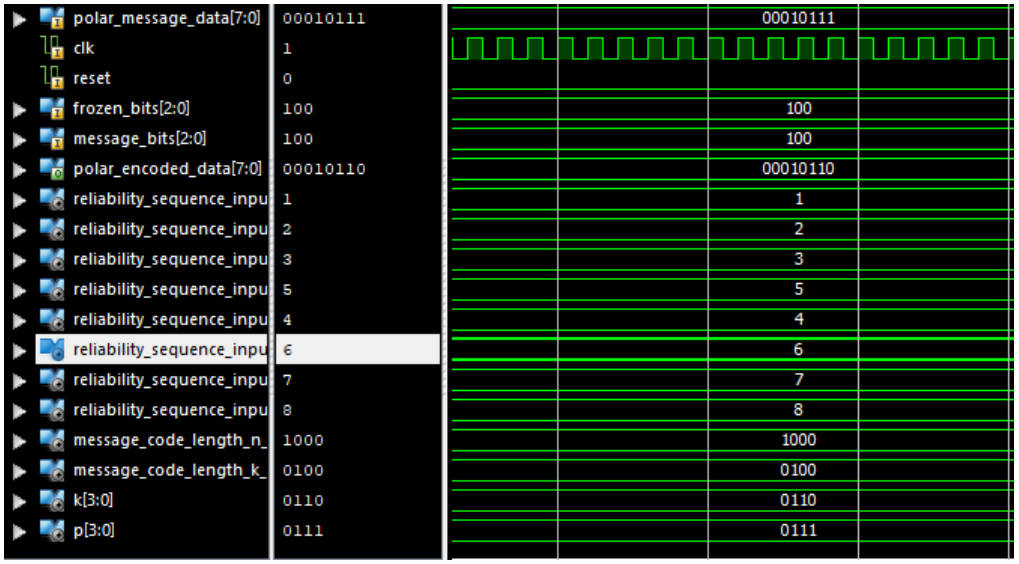


Fig.6.29 Simulation waveform for an 8-bit Polar encoder in decimal form

Fig.6.30 represents the simulation waveform for an 8-bit Polar decoder in binary form and Fig.6.31 represents the simulation waveform for a 8-bit Polar decoder in decimal form.

Test case (Polar Decoder): Apply clk = rising edge clock signal, reset = '0', X[7:0] = "00010110", then polar decoder output is M [7:0] ="00010111".

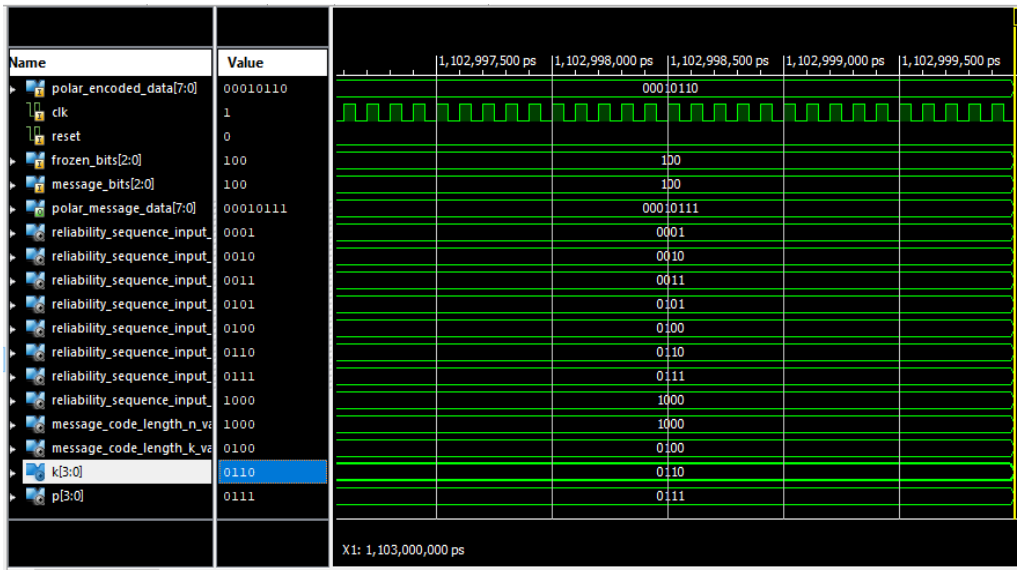


Fig.6.30 Simulation waveform for an 8-bit Polar decoder in binary form

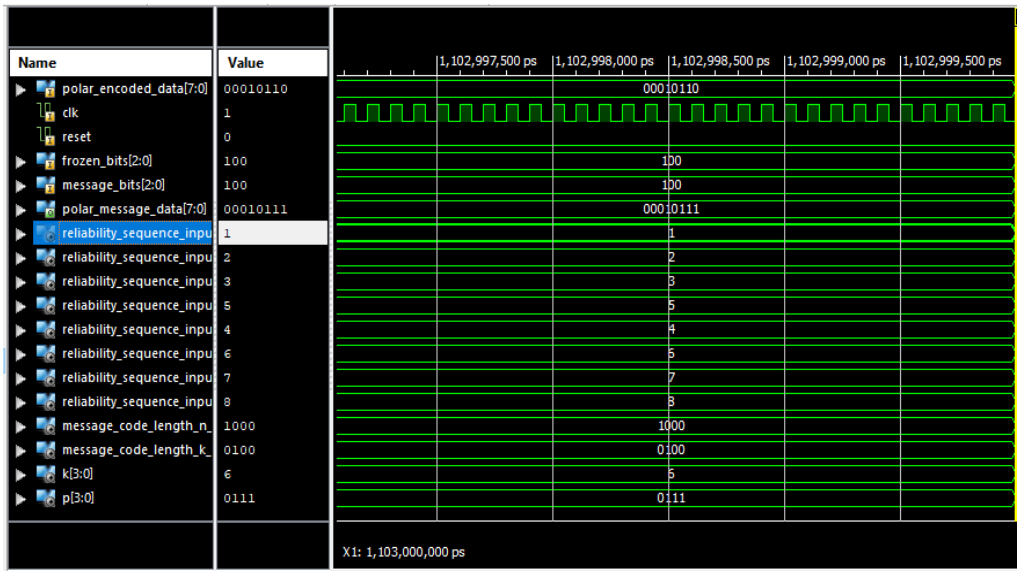


Fig.6.31 Simulation waveform for an 8-bit Polar decoder in decimal form

6.5.2 Xilinx ISIM Simulation Waveform of a 32-bit Polar Encoder & Decoder

Fig.6.32 represents the simulation waveform for a 32-bit Polar encoder in binary form and Fig.6.33 represents the simulation waveform for a 32-bit Polar encoder in decimal form.

Test case (Polar Encoder): Apply clk = rising edge clock signal, reset = '0', M [31:0] = "00000001000111110011111111111111", then polar encoder output is X[31:0] =" 00000000111101111100110100100001".

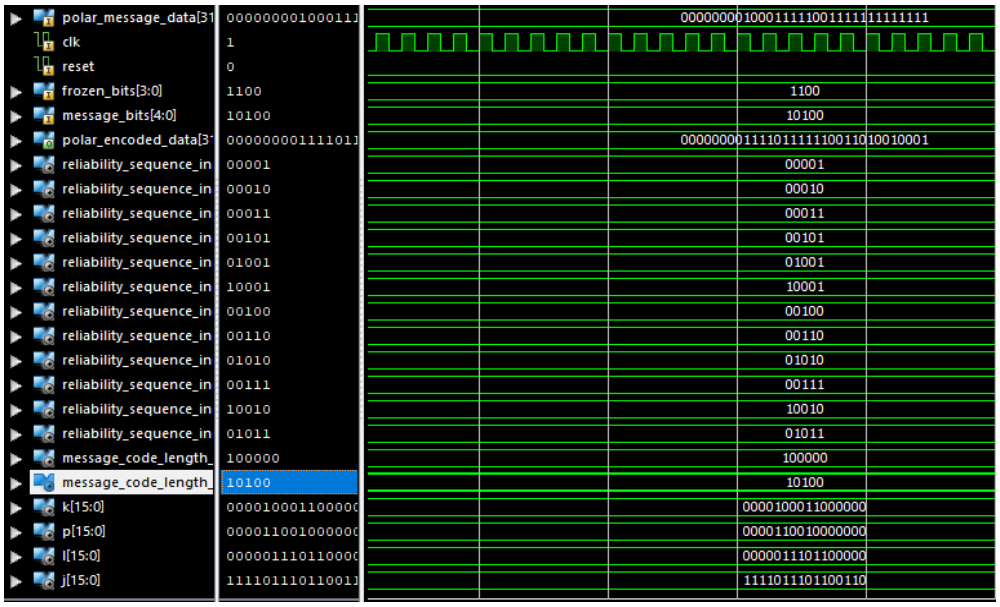


Fig.6.32 Simulation waveform for a 32-bit Polar encoder in binary form

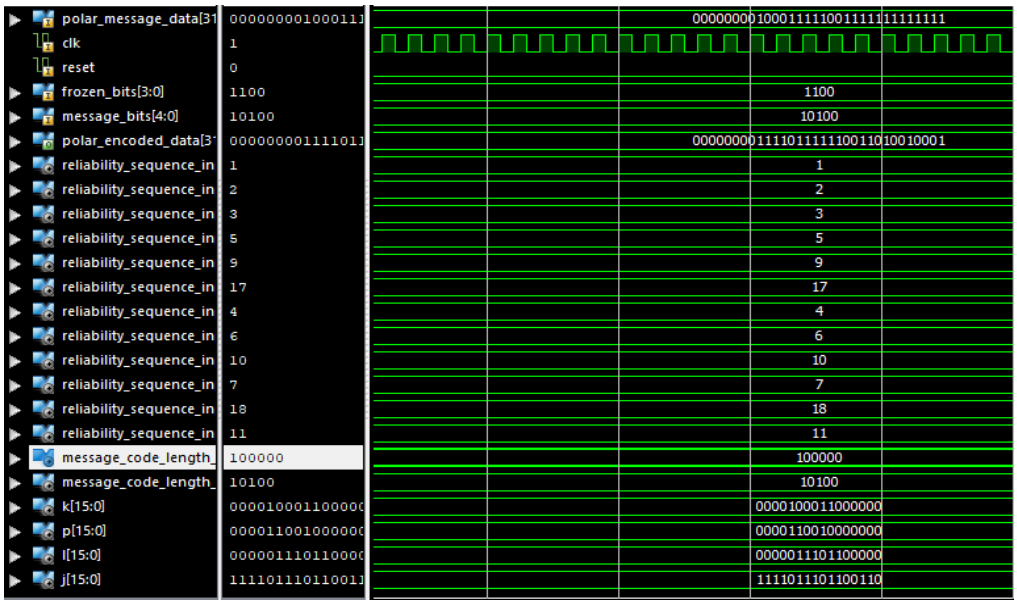


Fig.6.33 Simulation waveform for a 32-bit Polar encoder in decimal form

Fig.6.34 represents the simulation waveform for a 32-bit Polar decoder in binary form and Fig.6.35 represents the simulation waveform for a 32-bit Polar decoder in decimal form.

Test case (Polar Decoder): Apply Clk = rising edge clock signal, reset = '0', X[31:0] = "00000000111101111110011010010001 ", then polar decoder output is M [31:0] =" 00000001000111110011111111111111".

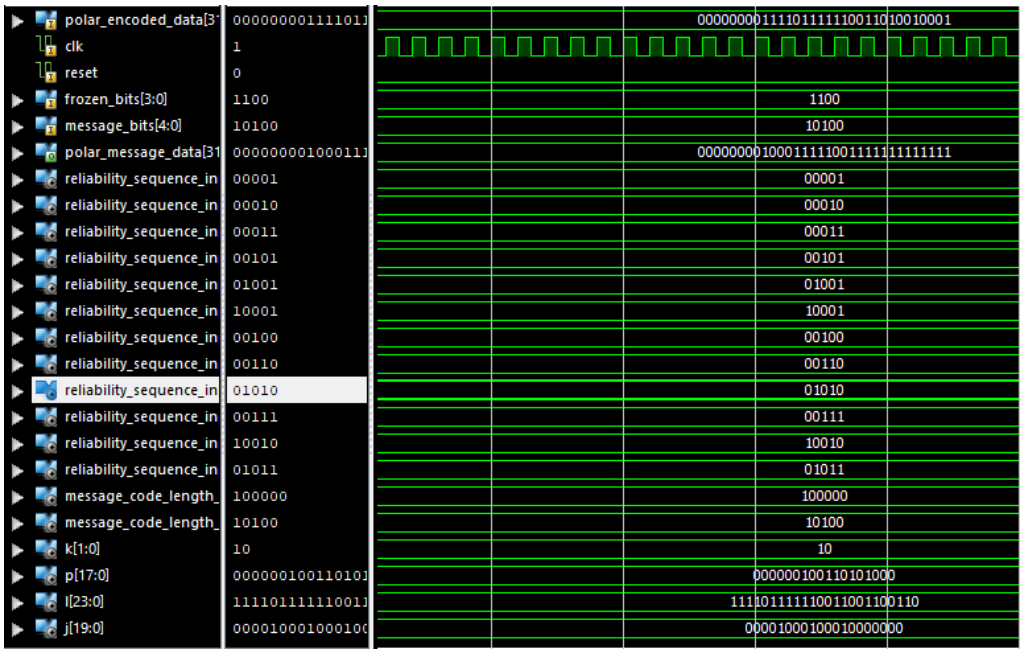


Fig.6.34 Simulation waveform for a 32-bit Polar decoder in binary form

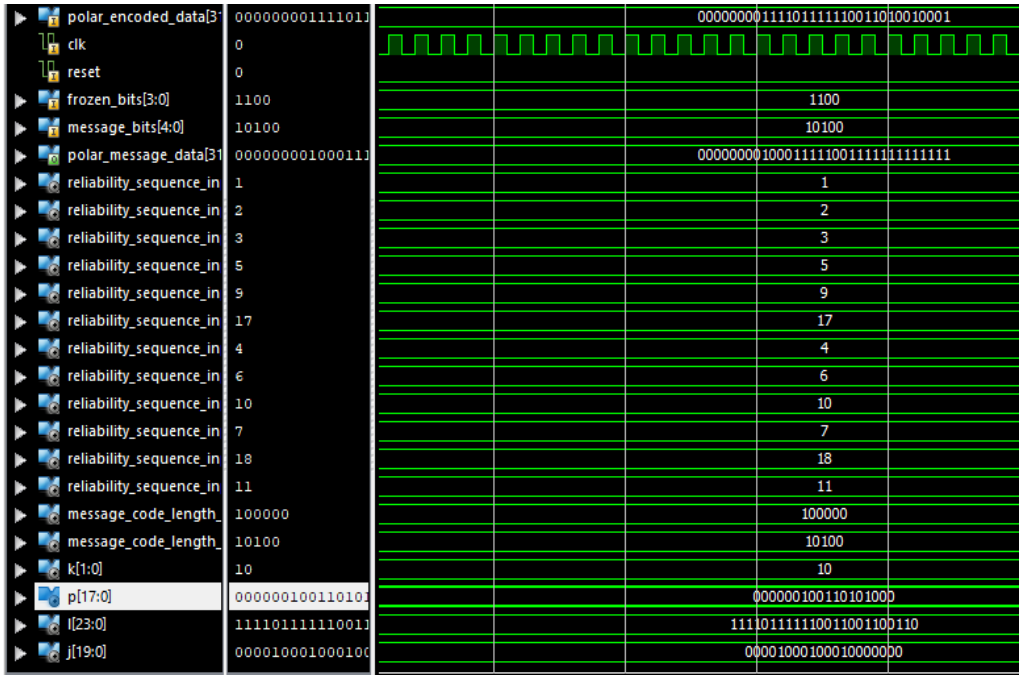


Fig.6.35 Simulation waveform for a 32-bit Polar decoder in decimal form

6.6 Analysis

Based on hardware device usage and timing reports, hardware chip design is analyzed. Directly from the Xilinx software comes a hardware design report that contains information on the FPGA device's hardware and design characteristics like the number of flip-flops, the logic-gate value, and the number of slices and LUTs that were utilized. The degree of optimization needed for a design is entirely up to the designer.

6.6.1 Comparative analysis of Xilinx hardware summary using Turbo, LDPC, and Polar encoder-decoder

Fig.6.36 illustrates a comparison of the hardware usage characteristics for Turbo, LDPC, and polar encoder and decoder on Virtex-5 FPGA with a 16-bit input data sequence. Device utilization for a 16-bit input data in terms of the number of flip flops, number of slices, LUTs, IOBs, and GCLKS for turbo encoder-decoder, LDPC encoder-decoder, and polar encoder-decoder is presented. Table. 6.13 shows the hardware utilization of a 16-bit turbo, LDPC, and polar encoder-decoder

Table. 6.13 Hardware utilization of a 16-bit Turbo, LDPC, and Polar encoder-decoder

Hardware Utilization	**Turbo Codes**		**LDPC Codes**		**Polar codes**	
	Encoder	Decoder	Encoder	Decoder	Encoder	Decoder
Slices	60/12480	145/12480	32/12480	110/12480	29/12480	103/12480
Flip Flops	72/12480	88/12480	45/12480	52/12480	37/12480	45/12480
LUTs	105/330	110/330	58/330	70/330	48/330	65/330
IOBs	50/172	72/172	50/172	72/172	50/172	72/172
GCLKs	1/32	1/32	1/32	1/32	1/32	1/32

Similarly, Fig.6.37 displays a comparison of the hardware usage characteristics for Turbo, LDPC, and polar encoder and decoder on Virtex-5 FPGA with 1024-bit input data sequence on Virtex-5 FPGA. Device utilization for a 1024-bit input data in terms of the number of slices, several flip flops, LUTs, IOBs, and GCLKS for turbo encoder-decoder, LDPC encoder-decoder, and polar encoder-decoder are presented. Table. 6.14 lists the hardware utilization of a 1024-bit Turbo, LDPC, and Polar Encoder-Decoder

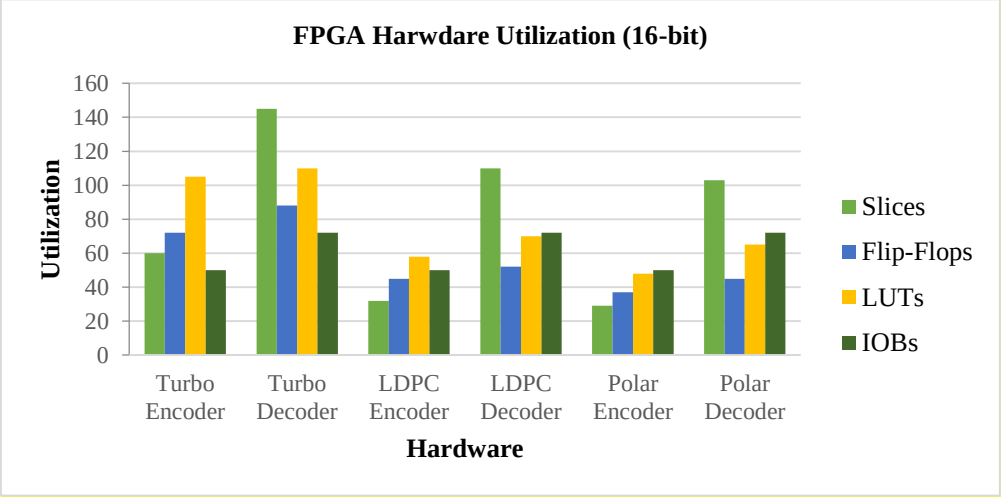


Fig.6.36 FPGA hardware resources utilization of turbo, LDPC, and polar encoder and decoder using a 16-bit input data sequence

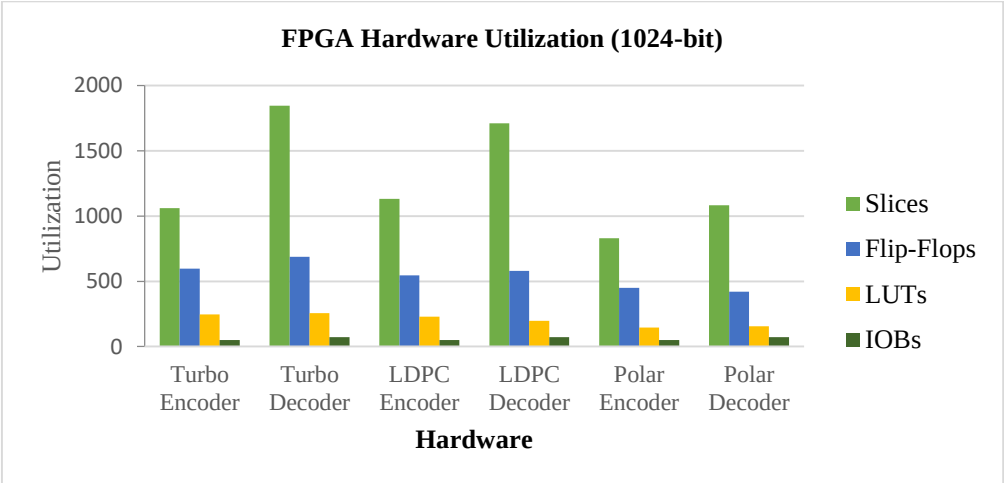


Fig.6.37 FPGA hardware resources utilization of turbo, LDPC, and polar encoder and decoder using a 1024-bit input data sequence.

Table. 6.14 Hardware Utilization of a 1024-bit Turbo, LDPC, and Polar encoder-decoder

Hardware	Turbo codes		LDPC codes		Polar codes	
	Encoder	Decoder	Encoder	Decoder	Encoder	Decoder
Slices	1060/12480	1845/12480	1132/12480	1710/12480	829/12480	1083/12480
Flip Flops	597/12480	688/12480	545/12480	580/12480	450/12480	420/12480
LUTs	245/330	256/330	228/330	198/330	145/330	155/330
IOBs	50/172	72/172	50/172	72/172	50/172	72/172
GCLKs	1/32	1/32	1/32	1/32	1/32	1/32

6.6.2 Comparative analysis of Xilinx timing summary using Turbo, LDPC, and Polar encoder-decoder

Fig.6.38 shows the comparison of timing parameters in terms of frequency, minimum period, time (min-max) before the clock, and combination path delay of the turbo, LDPC, and polar encoder and decoder using a 16-bit input data sequence on Virtex-5 FPGA. Table 6.15 lists the timing parameters for a 16-bit Turbo, LDPC, and polar encoder-decoder on Virtex-5 FPGA.

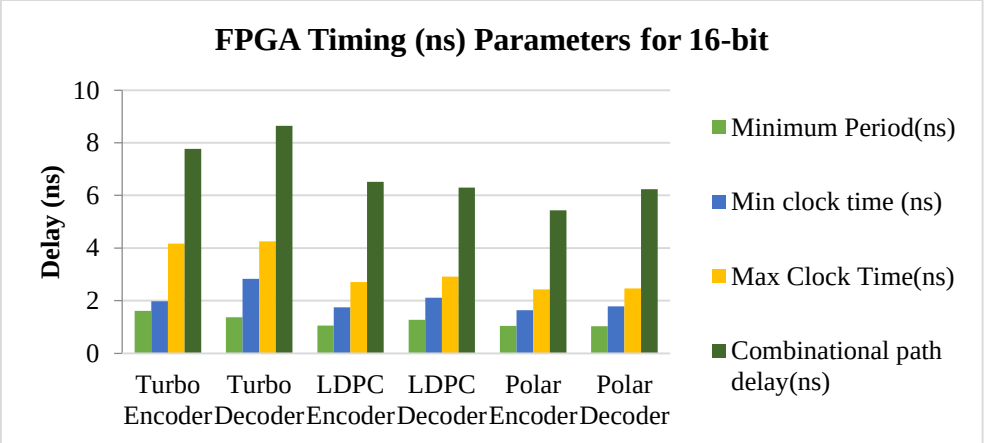


Fig.6.38 Timing parameter summary of the turbo, LDPC, and polar encoder and decoder using a 16-bit input data sequence.

Table 6.15 Timing parameters for a 16-bit Turbo, LDPC, and Polar encoder-decoder.

Timing Parameters	**Turbo Codes**		**LDPC Codes**		**Polar Codes**	
	Encoder	Decoder	Encoder	Decoder	Encoder	Decoder
Frequency (MHz)	310.0	315.0	375.00	390.0	382.00	410.0
Minimum period (ns)	1.615	1.365	1.053	1.275	1.042	1.034
Time (minimum) before clock (ns)	1.980	2.825	1.750	2.113	1.640	1.785
Time (maximum) after clock (ns)	4.174	4.250	2.710	2.914	2.434	2.465
Combinational path delay (ns)	6.769	8.640	6.513	6.302	5.432	6.231

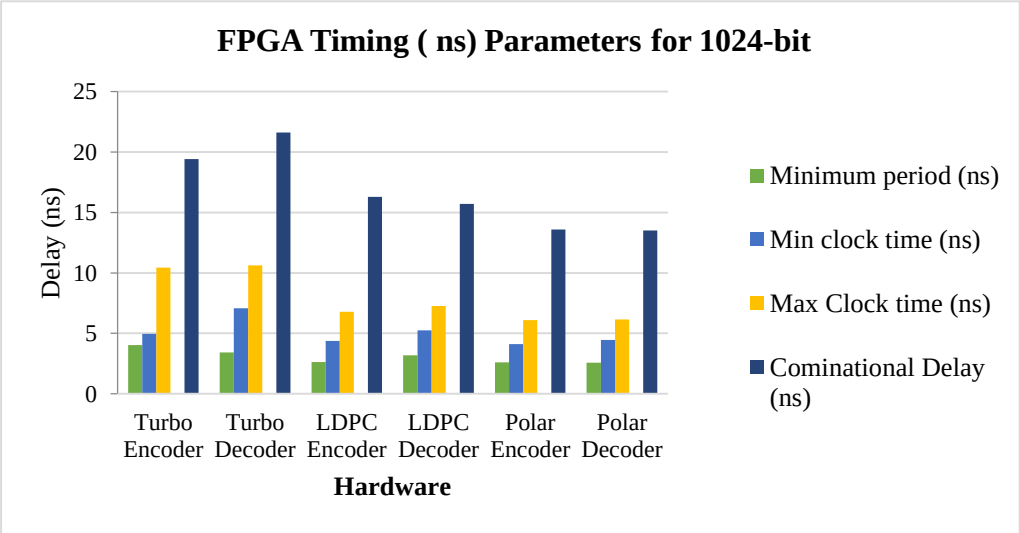


Fig.6.39 Timing Parameter summary of the turbo, LDPC, and polar encoder and decoder using a 1024-bit input data sequence.

Fig.6.39 illustrates the comparison of timing parameters in terms of frequency, minimum period, time (min-max) before clock, and combination path delay of the turbo, LDPC, and polar encoder and decoder using a 1024-bit input data sequence on Virtex-5 FPGA Table 6.16 lists the timing parameters for a 1024-bit Turbo, LDPC, and polar encoder-decoder on Virtex-5 FPGA.

Table 6.16 Timing parameter for a 1024-bit Turbo, LDPC, and Polar Encoder-Decoder.

Timing Parameter	Turbo codes		LDPC codes		Polar codes	
	Encoder	Decoder	Encoder	Decoder	Encoder	Decoder
Frequency (MHz)	775	788	938	971	955	1020
Minimum period (ns)	4.0375	3.4125	2.6325	3.17475	2.605	2.57466
Time(minimum) before clock (ns)	4.95	6.0625	4.375	5.26137	4.1	4.44465
Time (maximum) after clock (ns)	10.435	10.625	6.775	6.25586	6.085	6.13785
Combinational path delay (ns)	19.4225	21.6	16.2825	15.69198	13.58	13.51519

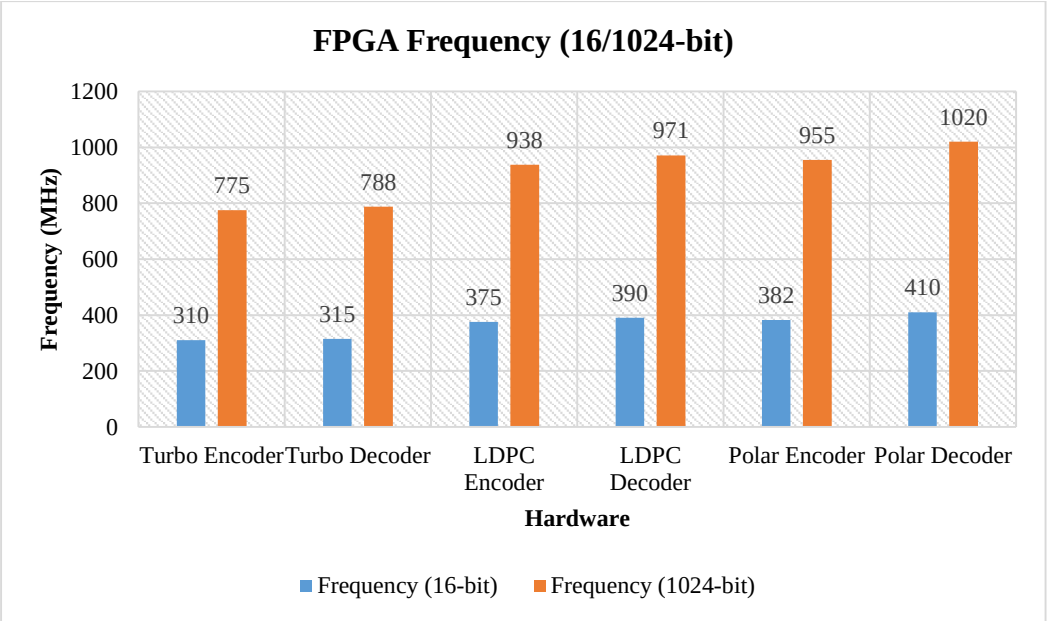


Fig.6.40 Frequency support for the synthesized coding

Fig. 6.41 FPGA logic verification

Virtex 5, XC5VX-110t-1ff1136 device with the simulated program in Xilinx simulator is used for the verification of the logic. The 32-bit simulated out is verified byte by byte on the LEDs on the Virtex-5 FPGA. The encoder output = "0000 0000 1000 1111 1001 1111 1111 1111" = ENCODER: 008F9FFFF (in hexadecimal), and the decoder output = "0000 0000 1111 0111 1110 0110 1001 0001" which is in hexadecimal DECODER: 00F7E691 (hexadecimal) output shown on LCD

CHAPTER 7
CONCLUSION & FUTURE WORK

7.1 Conclusion

The hardware chip design and performance analysis of LDPC, turbo, and polar codes for AWGN channel is successfully done in Xilinx ISE 14.7 software. The functional simulation and data communication with the designed hardware is verified using Xilinx ISIM simulation on target Virtex- 5 FPGA for device XC5Vlx20t-2-ff32.

It is observed that the device utilization in terms of number of slices, number of flip flops, LUTs, IOBs, and GCLKS with a 16-bit input data for turbo encoder is 60 out of 12480, 72 out of 12480, 105 out of 330, 50 out of 172, 1 out of 32, for turbo decoder, is 145 out of 12480, 88 out of 12480, 110 out of 330, 72 out of 172, 1 out of 32, the device utilization for LDPC encoder is 32 out of 12480, 45 out of 12480, 58 out of 330, 50 out of 172, 1 out of 32, for LDPC decoder is 110 out of 12480, 52 out of 12480, 70 out of 330, 72 out of 172,1 out of 32 and the device utilization for the polar encoder is 29 out of 12480, 37 out of 12480, 48 out of 330, 50 out of 172, 1 out of 32, for the polar decoder is 103 out of 12480, 45 out of 12480, 65 out of 330, 72 out of 172, 1 out of 32 respectively.

For 1024-bit input data the hardware device utilization in terms of the number of slices, the number of flip flops, LUTs, IOBs, and GCLKS for turbo encoder is 1060 out of 12480, 597 out of 12480, 245 out of 330, 50 out of 172, 1 out of 32, for turbo decoder is 1845 out of 12480, 688 out of 12480, 256 out of 330, 72 out of 172, 1 out of 32, the device utilization for LDPC encoder is 1132 out of 12480, 545 out of 12480, 228 out of 330, 50 out of 172, 1 out of 32, for LDPC decoder is 1710 out of 12480, 580 out of 12480, 198 out of 330, 72 out of 172, 1 out 32, and the device utilization for the polar encoder is 829 out of 12480, 450 out of 12480, 145 out of 330, 50 out of 172, 1 out of 32 and for and polar decoder is 1083 out of 12480, 420 out of 12480, 155 out of 330, 72 out

of 172, 1 out of 32 respectively. The timing parameters in terms of the minimum period, time (minimum before clock), and time (maximum after clock) with a 16-bit input data for turbo encoder are 1.615 ns, 1.980 ns, 4.174 ns, for turbo decoder is 1.365 ns, 2.825 ns, 4.250 ns, for LDPC encoder is 1.053 ns, 1.750 ns, 2.710 ns, for LDPC decoder is 1.275 ns, 2.113 ns, 2.914 ns, for the polar encoder is 1.042 ns, 1.640 ns, 2.434 ns, and for the polar decoder is 1.034 ns, 1.785 ns, 2.465 ns respectively. For 1024-bit input data, the timing parameters in terms of the minimum period, time (minimum before clock), and time (maximum after clock) for the turbo encoder is 4.0375 ns, 4.95 ns, 10.435 ns, for turbo decoder is 3.4125 ns, 7.0625 ns, 10.625 ns, for LDPC encoder is 2.6325 ns, 4.375 ns, 6.775 ns, for LDPC decoder is 3.17475 ns, 5.26137 ns, 7.25586 ns, for the polar encoder is 2.605 ns, 4.1 ns, 6.085 ns, and for the polar decoder is 2.57466 ns, 4.44465 ns, 6.13785 ns respectively.

For 16-bit input data, the percentage hardware device utilization for a turbo encoder in terms of the number of slices is 0.48 %, the number of flip-flops is 0.57 %, LUTs is 31.8 %, for turbo decoder the number of slices is 1.16 %, number of flip flops is 0.70%, LUTs is 33.3 %. The hardware device utilization for the LDPC encoder in terms of the number of slices is 0.25 %, the number of flip-flops is 0.36 %, LUTs is 17.5 %, for the LDPC decoder the number of slices is 0.88 %, the number of flip-flops is 0.41%, LUTs is 21.2 %. The hardware device utilization for the polar encoder in terms of the number of slices is 0.23 %, the number of flip-flops is 0.29 %, LUTs is 14.5 %, for the polar decoder the number of slices is 0.82 %, the number of flip-flops is 0.36 %, LUTs is 19.64 % respectively.

Similarly, for 1024-bit input data, the hardware device utilization for the turbo encoder in terms of the number of slices is 8.4 %, the number of flip-flops is 4.78 %, LUTs is 74.2 %, for turbo decoder the number of slices is 14.7 %, number of flip-flops is 5.51 %, LUTs is 77.5 %. The hardware device utilization for the LDPC encoder in terms of the number of slices is 9.07 %, the number of flip-flops is 4.36 %, LUTs is 69.09 %, for the LDPC decoder the number of slices is 13.7 %, the number of flip-flops is 4.64 %, LUTs is 60 %. The hardware device utilization for the polar encoder in terms of the number of slices is 6.64

%, the number of flip-flops is 3.605 %, LUTs is 43.9 %, for the polar decoder the number of slices is 8.67 %, the number of flip-flops is 3.36 %, LUTs is 46.9 % respectively.

Results indicate that polar encoder & decoder is taking lesser hardware resources in comparison to turbo and LDPC codes on FPGA. For 16-bit input data, the turbo encoder and decoder support 310.0 MHz and 315.0 MHz frequencies respectively. The LDPC encoder and decoder support 375.0 MHz and 390.0 MHz frequencies. In the same way polar encoder and decoder support 382 MHz and 410 MHz frequencies respectively. For 1024-bit input data, the turbo encoder and decoder support 775 MHz and 788 MHz frequencies respectively. The LDPC encoder and decoder support 938 MHz and 971 MHz frequencies. In the same way polar encoder and decoder support 955 MHz and 1020 MHz frequencies respectively. Polar codes provide fast switching in comparison to turbo and LDPC codes in FPGA hardware. Apart from this, polar codes provide optimal timing-related parameter results in comparison to turbo and LDPC codes. For 16-bit input data, the turbo encoder and decoder are having combination path delay of 7.767 ns and 8.64 ns for encoder and decoder hardware, the LDPC encoder and decoder are having combination path delay of 6.513 ns and 6.302 ns, and the polar encoder and decoder having combination path delay of 5.432 ns and 6.231 ns respectively. For 1024-bit input data, the turbo encoder and decoder are having combination path delay of 19.4225 ns and 21.6 ns, the LDPC encoder and decoder are having combination path delay of 16.2825 ns and 15.69168 ns, and the polar encoder and decoder having combination path delay of 13.58 ns and 13.51519 ns respectively.

The other important block in coding is LFSR. The LFSR shift register shifts the signal from one bit to the subsequent MSB and the outputs are connected in an exclusive-OR fashion to form a feedback loop. The XOR operation is used to generate a linear feedback shift register by combining the outputs of two or more flip-flops and then feeding those outputs back to the flip-flop's inputs. The behavior is used in different communication system coding, error correction, and detection methods. In this work, we have simulated the same behavior for the gold code sequence generator. The VHDL simulation of the LFSR modules

is successfully done in the Xilinx ISE 14.7. The RTL and waveform simulation verify the functionality of the chip design. The behavior of the LFSR-1 and LFSR-2 is verified for the test input given in the simulation and test benches created for the testing of the designed chip functionality of the chip. The behavior model-based simulation is carried out for both LFSR and further applied for the gold sequence generator to provide 31-bit output. The design is scalable and reprogrammable which can be extended based on the need of the communication system. The final design reports are verified in the simulation environment and the reported delay is 2.192 ns and 215 MHz clock frequency. The reported frequency is significant for the high-speed communication system with the optimal delay reported. In the future, we are planning to implement large-scale input LFSRs with tap sequences that will produce a larger gold code sequence generator. The logic synthesis can be carried out on high-end FPGA so that it can support fast-switching communication systems with optimum hardware utilization in chip design.

The hardware chip design of the turbo encoder and decoder is successfully done in Xilinx ISE 14.7. The max log MAP algorithm is integrated with the chip. The comparative performance of the chip is studied based on hardware and timing parameters on Virtex-5 FPGA synthesis. The ISIM simulation waveform shows the successful decoding of the data using the turbo decoder. The turbo encoder with max log-MAP utilizes less number of slices (67), flip-flops (71), and LUTs (103) in comparison to the turbo encoder. In the same way, the turbo decoder with max log-MAP utilizes less number of slices (140), flip-flops (80), and LUTs (107) in comparison to the turbo decoder. The frequency of the turbo encoder and decoder is 314 MHz and 320 MHz respectively.

The frequency of the turbo encoder and decoder with the max log-MAP algorithm is 332 MHz and 350 MHz respectively. The higher frequency support indicates that the turbo encoder and decoder with the max log-MAP algorithm provide a faster response in comparison to the turbo encoder and decoder. The combination path delay of the turbo encoder and decoder with the max log-MAP algorithm is 7.700 ns and 8.350 ns respectively, which is less in comparison to the combinational path delay of the turbo encoder and decoder. The designed

encoder and decoder hardware chip with the max log-MAP algorithm provides less hardware utilization, higher frequency support, and less delay in comparison to the turbo encoder and decoder.

7.2 Future Work

High spectral efficiency is a crucial need of 5G systems since it lowers the cost per bit significantly. Channel coding and modulation are very important for achieving high spectral efficiency. Mobile communications are projected to undergo significant changes because of 5G NR. 5G facilities are divided into three categories: mMTC, URLLC, and eMBB are predictable to be embraced by a variety of vertical businesses with varying needs. Beamforming, small cells, full-duplex, millimeter waves, and massive MIMO communication are among the technologies that make up 5G. Many protocols such as mMTC, D2D, and URLLC, have already replaced 4G LTE turbo codes with LDPC codes for ultra-reliable low delay reliable communications and low-density parity-check. Because of their integral benefits of rapid encoding-decoding operations and outstanding BER performance, LDPC as well as polar codes are gaining a lot of attention, making them strong contenders for the 5G Channel. The research work offers a widespread overview of the turbo, LDPC, and polar codes, as well as their benefits and shortcomings, to aid in the development of upcoming wireless networks.

The compressive study overviews the 5G networks, standardization, requirements, multiple carrier scheme, OFDM, NOMA, and Sub-band filtering based on 5G modulation techniques, and different channel coding schemes. Polar codes have been finally accepted in 5G eMBB control channel standards, after 10 years of its invention. This standardization has prompted more industrial and academic research in the field of polar coding and has given the flexibility of the code and design. It is analyzed that polar codes provide an optimum solution in terms of hardware complexity, timing response parameters, and high-frequency support. Polar codes are optimal, highly efficient, and reliable solutions in 4G and 5G wireless communication, and digital broadcasting as their performance is very good in the machine to machine communication and FPGA hardware

REFERENCES

1. A. Sahin, I. Guvenc and H. Arslan, "A Survey on Multicarrier Communications: Prototype Filters, Lattice Structures, and Implementation Aspects", *IEEE Communications Surveys & Tutorials*, vol. 16, no. 3, pp. 1312-1338, Third Quarter 2014, doi: 10.1109/SURV.2013.121213.00263.
2. Osseiran, "Scenarios for 5G mobile and wireless communications: the vision of the METIS project", *IEEE Communications Magazine*, vol. 52, no. 5, pp. 26-35, May 2014, doi: 10.1109/MCOM.2014.6815890.
3. A. C. Cirik, Y. Rong and Y. Hua, "Achievable Rates of Full-Duplex MIMO Radios in Fast Fading Channels with Imperfect Channel Estimation", *IEEE Transactions on Signal Processing*, vol. 62, no. 15, pp. 3874-3886, Aug.1, 2014, doi: 10.1109/TSP.2014.2330806.
4. B. Farhang-Boroujeny and H. Moradi, "OFDM Inspired Waveforms for 5G", *IEEE Communications Surveys & Tutorials*, vol. 18, no. 4, pp. 2474-2492, doi: 10.1109/COMST.2016.2565566.
5. B. F. Boroujeny, "Filter Bank Multicarrier (FBMC): An integrated solution to spectrum sensing and data transmission in cognitive radio networks", *Presented at the PHYDYAS Workshop, Perugia, Italy, 2009, doi:* 10.13140/2.1.4808.0160.
6. H. Chen, "Ultra-Reliable Low Latency Cellular Networks: Use Cases, Challenges and Approaches", *IEEE Communications Magazine*, vol. 56, no. 12, pp. 119-125, December 2018, doi: 10.1109/MCOM.2018.1701178.
7. A. Kumar, A. Kumar, A. Devrari and S. Singh, "Design and FPGA implementation of 32-point FFT processor", Proceeding of International Conference on Intelligent Communication, *Control and Devices, Springer, Singapore*, pp. 285-292, 2017, doi: https://doi.org/10.1007/978-981-10-1708-7_33.

8. D. Feng, L. Lu, Y. Yuan-Wu, G. Y. Li, S. Li and G. Feng, "Device-to-device communications in cellular networks", *IEEE Communications Magazine*, vol. 52, no. 4, pp. 49-55, April 2014, doi: 10.1109/MCOM.2014.6807946.
9. A. Kumar, A. Kumar, A. Devrari, "Hardware chip performance analysis of different FFT architecture", *International Journal of Electronics*, 108(7), pp.1124-1140, 2021, doi: https://doi.org/10.1080/00207217.2020.1819441.
10. Ericsson, "5G radio access", *White paper*, 2015.
11. F. Boccardi, R. W. Heath, A. Lozano, T. L. Marzetta and P. Popovski, "Five disruptive technology directions for 5G", *IEEE Communications Magazine*, vol. 52, no. 2, pp. 74-80, February 2014, doi: 10.1109/MCOM.2014.6736746.
12. F. Dai and J. Wu, "Efficient broadcasting in ad hoc wireless networks using directional antennas", *IEEE Transactions on Parallel and Distributed Systems*, vol. 17, no. 4, pp. 335-347, April 2006, doi: 10.1109/TPDS.2006.46.
13. N. Baliyan, M. Verma, and A. Kumar, "Channel Capacity in MIMO OFDM System", *Proceeding of International Conference on Intelligent Communication, Control and Devices*, pp. 1113-1120, 2017, doi: https://doi.org/10.1007/978-981-10-1708-7_132.
14. G. L. Stuber, J. R. Barry, S. W. McLaughlin, Ye Li, M. A. Ingram and T. G. Pratt, "Broadband MIMO-OFDM wireless communications", in *Proceedings of the IEEE*, vol. 92, no. 2, pp. 271-294, Feb. 2004, doi: 10.1109/JPROC.2003.821912.
15. GSMA Intelligence, "Understanding 5G: Perspectives on future technological advancement in mobile", *White paper*, 2014.
16. A. Kumar, A. Kumar, and G.S. Tomar, "Hardware Chip Performance of CORDIC Based OFDM Transceiver for Wireless Communication", *Computer Systems Science and Engineering*, 40(2), pp. 645-659, 2022, doi: https://doi.org/10.32604/csse.2022.019449.

17. G. Wunder, Martin Kasparick, and Thorsten wild, "5GNOW: Intermediate frame structure and transceiver concepts", *2014 IEEE Globecom Workshops (GC Wkshps)*, Austin, TX, USA, 2014, pp. 565-570, doi: 10.1109/GLOCOMW.2014.7063492.
18. H. Nikopour and H. Baligh, "Sparse code multiple access", *2013 IEEE 24th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, London, UK, 2013, pp. 332-336, doi: 10.1109/PIMRC.2013.6666156.
19. H. Holma, A. Toskala, and J. Reunanen, "LTE Small Cell Optimization: 3GPP Evolution to Release 13", *Hoboken, NJ, USA: Wiley*, 2015.
20. Huawei, "5G a technology vision", *White paper*, 2013.
21. V.M. Mishra, and A. Kumar, "Zigbee internode communication and FPGA synthesis using mesh, star and cluster tree topological chip", *Wireless Personal Communications*, 119(2), pp. 1321-1339, 2021, doi: https://doi.org/10.1007/s11277-021-08282-w.
22. J. Cui, Z. Ding and P. Fan, "A Novel Power Allocation Scheme Under Outage Constraints in NOMA Systems", *IEEE Signal Processing Letters*, vol. 23, no. 9, pp. 1226-1230, Sept. 2016, doi: 10.1109/LSP.2016.2591561.
23. A. Kumar, P. Kuchhal, and S. Singhal, "Design and FPGA synthesis of three stage telecommunication switching in HDL environment". *Procedia Computer Science*, 48, pp. 454-460, 2015, doi: https://doi.org/10.1016/j.procs.2015.04.119.
24. J. B. Kim and I. H. Lee, "Non-Orthogonal Multiple Access in Coordinated Direct and Relay Transmission", *IEEE Communications Letters*, vol. 19, no. 11, pp. 2037-2040, Nov. 2015, doi: 10.1109/LCOMM.2015.2474856.
25. J. Choi, "Non-Orthogonal Multiple Access in Downlink Coordinated Two-Point Systems" , *IEEE Communications Letters*, vol. 18, no. 2, pp. 313-316, February 2014, doi: 10.1109/LCOMM.2013.123113.132450.

26. A. Devrari, A. Kumar, A. Kumar, and S. Singh, “Design and FPGA Implementation of DSSS for Near-Far Effect in MANET”, *Proceeding of International Conference on Intelligent Communication, Control and Devices*, , pp. 425-434, 2017, https://doi.org/10.1007/978-981-10-1708-7_48.
27. K. Higuchi and Y. Kishiyama, "Non-Orthogonal Access with Random Beamforming and Intra-Beam SIC for Cellular MIMO Downlink", *2013 IEEE 78th Vehicular Technology Conference (VTC Fall)*, Las Vegas, NV, USA, 2013, pp. 1-5, doi: 10.1109/VTCFall.2013.6692307.
28. L. Dai, B. Wang, Y. Yuan, S. Han, I. Chih-lin and Z. Wang, "Non-orthogonal multiple access for 5G: solutions, challenges, opportunities, and future research trends”, *IEEE Communications Magazine*, vol. 53, no. 9, pp. 74-81, September 2015, doi: 10.1109/MCOM.2015.7263349.
29. L Chhaya, P. Sharma, G. Bhagwatikar, and A. Kumar, “Wireless sensor network based smart grid communications: Cyber attacks, intrusion detection system and topology control”, *Electronics*, 6(1), 5, 2017, doi: ; https://doi.org/10.3390/electronics6010005.
30. M. Zeng, A. Yadav, O. A. Dobre, G. I. Tsiropoulos and H. V. Poor, "Capacity Comparison Between MIMO-NOMA and MIMO-OMA With Multiple Users in a Cluster”, *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 10, pp. 2413-2424, Oct. 2017, doi: 10.1109/JSAC.2017.2725879.
31. N. Otao, Y. Kishiyama and K. Higuchi, "Performance of non-orthogonal access with SIC in cellular downlink using proportional fair-based resource allocation”, *2012 International Symposium on Wireless Communication Systems (ISWCS)*, Paris, France, 2012, pp. 476-480, doi: 10.1109/ISWCS.2012.6328413.
32. N. Zabetian, M. Baghani and A. Mohammadi, "Rate optimization in NOMA cognitive radio networks”, *2016 8th International Symposium on Telecommunications (IST)*, Tehran, Iran, 2016, pp. 62-65, doi: 10.1109/ISTEL.2016.7881783.

33. Nokia Networks, “Looking ahead to 5G: Building a virtual zero latency gigabit experience”, *White paper,* 2014.
34. P. Adhikari, “Understanding millimeter wave wireless communication”, Loea Corp., *White paper,* 2008.
35. Q. Sun, S. Han, C. -L. I and Z. Pan, "On the Ergodic Capacity of MIMO NOMA Systems”, *IEEE Wireless Communications Letters*, vol. 4, no. 4, pp. 405-408, Aug. 2015, doi: 10.1109/LWC.2015.2426709.
36. Service Requirements for Machine-Type Communications (MTC) V. 13.1.0, *3GPP Standard TS 22.368,* 2017.
37. S. M. R. Islam, N. Avazov, O. A. Dobre and K. -s. Kwak, "Power-Domain Non-Orthogonal Multiple Access (NOMA) in 5G Systems: Potentials and Challenges”, *IEEE Communications Surveys & Tutorials*, vol. 19, no. 2, pp. 721-742, Secondquarter 2017, doi: 10.1109/COMST.2016.2621116.
38. *Sophia Antipolis* , “Study on scenarios and requirements for next generation access technologies”, *3GPP, France,* Tech. Rep. 38.913, Jun. 2017.
39. Y. Tao, L. Liu, S. Liu and Z. Zhang, "A survey: Several technologies of non-orthogonal transmission for 5G”, in *China Communications*, vol. 12, no. 10, pp. 1-15, Oct. 2015, doi: 10.1109/CC.2015.7315054.
40. W. Shin, M. Vaezi, B. Lee, D. J. Love, J. Lee and H. V. Poor, "Non-Orthogonal Multiple Access in Multi-Cell Networks: Theory, Performance, and Practical Challenges”, *IEEE Communications Magazine*, vol. 55, no. 10, pp. 176-183, Oct. 2017, doi: 10.1109/MCOM.2017.1601065.
41. X. Zhang, M. Jia, L. Chen, J. Ma and J. Qiu, "Filtered-OFDM - Enabler for Flexible Waveform in the 5th Generation Cellular Networks”, *2015 IEEE Global Communications Conference (GLOBECOM),* San Diego, CA, USA, 2015, pp. 1-6, doi: 10.1109/GLOCOM.2015.7417854.

42. Y. Liu, M. Elkashlan, Z. Ding and G. K. Karagiannidis, "Fairness of User Clustering in MIMO Non-Orthogonal Multiple Access Systems", *IEEE Communications Letters*, vol. 20, no. 7, pp. 1465-1468, July 2016, doi: 10.1109/LCOMM.2016.2559459.
43. Z. Qin, Y. Liu, Z. Ding, Y. Gao and M. Elkashlan, "Physical layer security for 5G non-orthogonal multiple access in large-scale networks", *2016 IEEE International Conference on Communications (ICC)*, Kuala Lumpur, Malaysia, 2016, pp. 1-6, doi: 10.1109/ICC.2016.7510755.
44. Z. Ding, F. Adachi and H. V. Poor, "The Application of MIMO to Non-Orthogonal Multiple Access", *IEEE Transactions on Wireless Communications*, vol. 15, no. 1, pp. 537-552, Jan. 2016, doi: 10.1109/TWC.2015.2475746.
45. S. Soliman, M. A. Jaela, A. M. Abotaleb, Y. Hassan, M. A. Abdelghany, A. T. Abdel-Hamid, N.S. Khaled, and H. Mostafa, "FPGA implementation of dynamically reconfigurable IoT security module using algorithm hopping", *Integration,* 68, pp.108-121, 2019, doi:10.1016/j.vlsi.2019.06.004.
46. Z. Z. Aminuddin, I. H. b. Hamzah, A. A Samat, M. Idris, A. F. A. Rahim, and Z. H. C. Soh "An FPGA application of home security code using Verilog" *International Journal of Reconfigurable and Embedded Systems,* vol. 11, no. 3, pp. 205-214, 2021, doi:10.11591/ijres.v11.i3.pp205-214.
47. A. S. Babu, and B. Anand "Modified dynamic current mode logic-based LFSR for low power applications", *Microprocessors and Microsystems,* vol.72, pp. 102945, 2020, doi: https://doi.org/10.1016/j.micpro.2019.102945.
48. R. Mishra, P. Kuchhal, and A. Kumar "Effect of Height of the Substrate and Width of the Patch on the Performance Characteristics of Microstrip Antenna", *International Journal of Electrical and Computer Engineering,* vol.5, no. 6, pp. 1441-1445, 2015, doi: 10.11591/ijece.v5i6.pp1441-1445.

49. K. H. Kaittan, and S. J. Mohammed “Implementing and designing a secure information system based on the DSSS gold sequence using MATLAB”, *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 3, pp.1455-1463, 2021, doi: 10.11591/eei.v10i3.2446.
50. V. M. Mishra, and A. Kumar “Zigbee internode communication and FPGA synthesis using mesh, star and cluster tree topological chip”, *Wireless Personal Communications*, vol. 119, no. 2, pp.1321-1339, 2021, doi:10.1007/s11277-021-08282-w.
51. C. Bintjas *et al*., "20 Gb/s all-optical XOR with UNI gate", *IEEE Photonics Technology Letters*, vol. 12, no. 7, pp. 834-836, July 2000, doi: 10.1109/68.853516.
52. M. A. Esmail and H. Fathallah, "Physical Layer Monitoring Techniques for TDM-Passive Optical Networks: A Survey", *IEEE Communications Surveys & Tutorials*, vol. 15, no. 2, pp. 943-958, Second Quarter 2013, doi: 10.1109/SURV.2012.060912.00057.
53. A. S. Rawat, A. Rana, A. Kumar, and A. Bagwari, “Application of multilayer artificial neural network in the diagnosis system: a systematic review”, *IAES International Journal of Artificial Intelligence*, vol. 7, no.3, pp.138, 2018, doi: 10.11591/ijai.v7.i3.pp138-142.
54. A. M. Gutierrez, A. Brimont, J. Herrera, M. Aamer, D. J. Thomson, F. Y. Gardes, and P. Sanchis, “Analytical model for calculating the nonlinear distortion in silicon-based electro-optic Mach Zehnder modulators”, *Journal of Lightwave Technology*, vol. 31, no.23, 2013, pp. 3603-3613, doi: 10.1109/JLT.2013.2286838.
55. K. E. Stubkjaer, "Semiconductor optical amplifier-based all-optical gates for high-speed optical processing”, *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 6, no. 6, pp. 1428-1435, Nov.-Dec. 2000, doi: 10.1109/2944.902198.
56. R. Gold, "Optimal binary sequences for spread spectrum multiplexing (Corresp.)”, *IEEE Transactions on Information*

Theory, vol. 13, no. 4, pp. 619-621, October 1967, doi: 10.1109/TIT.1967.1054048.

57. J. Espinosa García, G. Cotrina, A. Peinado, and A. Ortiz, "Security and efficiency of linear feedback shift registers in GF (2^n) using n-Bit grouped operations" *Mathematics,* vol. 10, no. 6, pp. 996, 2022, doi: 10.3390/math10060996.
58. M. Jayasanthi, and R. Kalaivani, "Low-power DSSS transmitter and its VLSI implementation", *Annals of Telecommunications*, vol. 76, no. 7, pp. 537-543, 2021, doi: 10.1007/s12243-021-00837-z.
59. G. Giustolisi, R. Mita, G. Palumbo and G. Scotti, "A Novel Clock Gating Approach for the Design of Low-Power Linear Feedback Shift Registers", *IEEE Access*, vol. 10, pp. 99702-99708, 2022, doi: 10.1109/ACCESS.2022.3207151.
60. M. Saber, and M. M. Eid, "Low Power Pseudo-Random Number Generator based on lemniscate chaotic map", *International Journal of Electrical and Computer Engineering,* 11(1), pp. 863, 2021, doi: 10.11591/ijece.v11i1.pp863-871.
61. S. P Kumar and Y. D. S Raju, "Realization of Speed Optimized VLSI Architecture of Multistage Linear Feedback Shift Register Counters Decoding Logic", *International Journal of Early Childhood*, vol. 14, no.2, 2022, pp.2080-2090, doi: 10.9756/INT-JECSE/V14I2.116.
62. S. Hou, Y. Guo and S. Li, "A Lightweight LFSR-Based Strong Physical Unclonable Function Design on FPGA", *IEEE Access*, vol. 7, pp. 64778-64787, 2019, doi: 10.1109/ACCESS.2019.2917259.
63. S. Karunamurthi, and V. K. Natarajan "VLSI Implementation of Reversible Logic Gates Cryptography with LFSR Key", *Microprocessors and microsystems*, vol. 69, pp.68-78, 2019, doi:10.1016/j.micpro.2019.05.015.
64. S. Sood, A.Pal Singh and A. Kumar, "VHDL design of OFDM transreceiver chip using variable FFT", Journal of Selected Areas in Microelectronics (JSAM), *Singaporean Journal of Scientific Research* (SJSR) Vol, 5, 2013, pp. 47-58.

65. N. Gupta, A. Jain, K. S. Vaisla, A. Kumar, and R. Kumar, “Performance analysis of DSDV and OLSR wireless sensor network routing protocols using FPGA hardware and machine learning”, *Multimedia Tools and Applications*, vol, 80, no. 14, 2021, pp. 22301-22319, doi: 10.1007/s11042-021-10820-4.
66. R. K. Dutt, R. Ganesh, and P. Premchand, “Neural net implementation of steam properties on FPGA”, *International Journal of Reconfigurable and Embedded Systems*, vol. 10, no. 3, 2021, pp. 186, doi: 10.11591/ijres.v10.i3.pp186-194.
67. V. Raj, S. Janakiraman, S. Rajagopalan, and R. Amirtharajan, “Security analysis of reversible logic cryptography design with LFSR key on 32-bit microcontroller”, *Microprocessors and Microsystems,* 84, 2021, pp.104265, doi: 10.1016/j.micpro.2021.104265.
68. S. Saravanan, M. Lavanya, R. V. Sai, and R. Kumar, “Design and analysis of linear feedback shift register based on various tap connections”, *Procedia engineering*, vol. 38, 2012, pp. 640-646, doi: 10.1016/j.proeng.2012.06.079.
69. L. Bahl, J. Cocke, F. Jelinek and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate (Corresp.)", *IEEE Transactions on Information Theory*, vol. 20, no. 2, pp. 284-287, March 1974, doi: 10.1109/TIT.1974.1055186.
70. C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: turbo-codes", *IEEE Transactions on Communications*, vol. 44, no. 10, pp. 1261-1271, Oct. 1996, doi: 10.1109/26.539767.
71. K. H. Kaittan, and S. J. Mohammed “Implementing and designing a secure information system based on the DSSS gold sequence using MATLAB” *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 3, 2021, pp.1455-1463, doi: 10.11591/eei.v10i3.2446.
72. M. J. Adamu, L. Qiang, R. S. Zakariyya, C. O. Nyatega, H. B. Kawuwa, and A. Younis, “An Efficient Turbo Decoding and Frequency Domain Turbo Equalization for LTE Based Narrowband

Internet of Things (NB-IoT) Systems", *Sensors*, vol. 21, no. 16, Aug. 2021, pp. 5351, doi: 10.3390/s21165351.

73. C. E. Shannon, "A mathematical theory of communication" *Bell System Technology, Journal*, vol. 27, 1948, pp. 379-423.
74. L. Hang, G. Zhen, Z. Ming, and W. Jing, "Partial iterative decode of turbo codes for on-board processing satellite platform", *China Communications*, vol. 12, no. 11, Nov. 2015, pp. 1-8, doi: 10.1109/CC.2015.7366233.
75. L. Fanari, E. Iradier, I. Bilbao, R. Cabrera, J. Montalban, and P. Angueira, "Comparison between Different Channel Coding Techniques for IEEE 802.11be within Factory Automation Scenarios", *Sensors*, vol. 21, no. 21, Oct. 2021, pp. 7209, doi: 10.3390/s21217209.
76. P. Salija, B. Yamuna, T. R. Padmanabhan, and D. Mishra, "Generic Direct Approach for Decoding Turbo Codes Using Probability Density Based Reliability Model", *Journal of Communications Technology and Electronics*, vol. 66, no. 2, Feb. 2021, pp. 175-183, doi: 10.1134/S1064226921020133.
77. L. F. Gonzalez-Perez, L. C. Yllescas-Calderon and R. Parra-Michel, "Parallel and configurable turbo decoder implementation for 3GPP-LTE", 2013 *International Conference on Reconfigurable Computing and FPGAs (Reconfig)*, 2013, pp. 1-6, doi: 10.1109/Reconfig.2013.6732316.
78. J. P. Woodard and L. Hanzo, "Comparative study of turbo decoding techniques: an overview", *IEEE Transactions on Vehicular Technology*, vol. 49, no. 6, pp. 2208-2233, Nov. 2000, doi: 10.1109/25.901892.
79. R. Dobkin, M. Peleg and R. Ginosar, "Parallel VLSI architecture for MAP turbo decoder", *The 13th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, Lisbon, Portugal, 2002, pp. 384-388 vol.1, doi: 10.1109/PIMRC.2002.1046727.

80. J. M Gilbert, C. Robbins, and W. Sheikh, “FPGA implementation of error control codes in VHDL: An undergraduate research project”, *Computer Applications in Engineering Education*, vol. 27, no. 5, 2019, pp.1073-1086, doi: 10.1002/cae.22137.
81. Z. R. M Hajiyat, A. Sali, M. Mokhtar, and F. Hashim, “Channel coding scheme for 5G mobile communication system for short-length message transmission”, *Wireless Personal Communications*, vol. 106, no. 2, 2019, pp. 377-400, doi: 10.1007/s11277-019-06167-7.
82. S. Kalyani, V. Raj and K. Giridhar, "Narrowband Interference Mitigation in Turbo-Coded OFDM Systems”, *2007 IEEE International Conference on Communications*, Glasgow, UK, 2007, pp. 1059-1064, doi: 10.1109/ICC.2007.180.
83. J. D Kene, and K. D Kulat, “Soft Output Decoding Algorithm for Turbo Codes Implementation in Mobile Wi-Max Environment”, *Procedia Technology*, vol.6, ,2012, pp. 666-673, doi: 10.1016/j.protcy.2012.10.080.
84. J. Li, X.Wang, J. He, C. Su, and L. Shan, “Turbo Decoder Design based on an LUT-Normalized Log-MAP Algorithm”, *Entropy*, vol.21, no.8, 2019, pp. 814, doi: 10.3390/e21080814.
85. H. Luo, Y. Zhang, W. Li, , L. K. Huang, and J. Cosmas, "Low Latency Parallel Turbo Decoding Implementation for Future Terrestrial Broadcasting Systems”, *IEEE Transactions on Broadcasting*, vol. 64, no. 1, March 2018, pp. 96-104, doi: 10.1109/TBC.2017.2704425.
86. V. Akshaya, K. N. Sreehari and A. Chalil, "VLSI Implementation of Turbo Coder for LTE using Verilog HDL”, *Fourth International Conference on Computing Methodologies and Communication (ICCMC)*, pp. 275-279, 2020, doi: 10.1109/ICCMC48092.2020.ICCMC-00051.
87. P. Salmela, H. Sorokin, and J. Takala, “A programmable Max-Log-MAP turbo decoder implementation”, *VLSI Design*, vol. 2008, pp. 1-17, 2008, doi:10.1155/2008/319095.

88. Z. Yan, G W. He, S. Wang, and Z. Mao, “High-performance parallel turbo decoder with configurable interleaving network for LTE application”, *Integration,* vol. 52, pp.77-90, 2016, doi: 10.1016/j.vlsi.2015.05.003.
89. G. Masera, G. Piccinini, M. R. Roch and M. Zamboni, "VLSI architectures for turbo codes”, *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 7, no. 3, pp. 369-379, Sept. 1999, doi: 10.1109/92.784098.
90. M. J. Adamu, L. Qiang, R. S. Zakariyya, C. O. Nyatega, H. B. Kawuwa, and A. Younis, “An Efficient Turbo Decoding and Frequency Domain Turbo Equalization for LTE Based Narrowband Internet of Things (NB-IoT) Systems”, *Sensors,* vol. 21, no. 16, pp. 5351, Aug. 2021, doi: 10.3390/s21165351.
91. A. A. Hamad,, M. T. Gatte, and L. A. Abdul-Rahaim, “Efficient systematic turbo polar decoding based on optimized scaling factor and early termination mechanism”, *International Journal of Electrical & Computer Engineering,* vol. 13, no.1, pp. 623-637, Feb 2023, doi: 10.11591/ijece.v13i1.pp629-637.
92. A. Kumar, P. Sharma, M. K. Gupta, and R. Kumar, “Machine learning-based resource utilization and pre-estimation for network on chip (NoC) communication”, *Wireless Personal Communications*, vol.102, no. 3, pp. 2211-2231, 2018, doi: 10.1007/s11277-018-5376-3.
93. A. Kumar, G. Verma, and M. K. Gupta, “FM receiver design using programmable PLL”, *Wireless Personal Communications,* vol. 97, no. 1, pp. 773-787, 2017, doi: 10.1007/s11277-017-4536-1.
94. F. Liu, M.M. Karbassian, and H. Ghafouri-Shiraz, “Novel family of prime codes for synchronous optical CDMA”, *Optical and quantum electronics,* 39(1), pp. 79-90, 2007,doi: 10.1007/s11082-007-9066-8.
95. A. Tsuneda and T. Yoshida, "Performance evaluation of asynchronous DS/CDMA communications using unipolar codes", *2011 20th European Conference on Circuit Theory and*

Design (ECCTD), Linköping, Sweden, 2011, pp. 648-651, doi: 10.1109/ECCTD.2011.6043626.

96. J. M. Velazquez-Gutierrez and C. Vargas-Rosales, "Sequence Sets in Wireless Communication Systems: A Survey", *IEEE Communications Surveys & Tutorials*, vol. 19, no. 2, pp. 1225-1248, Second quarter 2017, doi: 10.1109/COMST.2016.2639739.
97. R. Gold, "Optimal binary sequences for spread spectrum multiplexing (Corresp.)", *IEEE Transactions on Information Theory*, vol. 13, no. 4, pp. 619-621, October 1967, doi: 10.1109/TIT.1967.1054048.
98. M. George, M. Hamid, and A. Miller, "Gold code generators in Virtex devices", *Xilinx Application Note xapp217*, v1, 2001.
99. J. Alkasassbeh,M A. Al-Qaisi, and A. Khalifeh, "BER Performance Using Linear Phase Orthogonal Binary Codes for Multi-users Mobile Communication", *Wireless Personal Communications,* 119(1), pp. 259-274, 2021, doi: https://doi.org/10.1007/s11277-021-08206-8.
100. O.B. Wojuola, and S.H. Mneney, "Multiple-access interference of Gold codes in a DS-CDMA system", *SAIEE Africa Research Journal*, pp. 4-10, 2017, doi: 10.23919/SAIEE.2015.8531483.
101. S. Sajic, N. Maletic, B.M. Todorovic, and M. Sunjevaric, "Random binary sequences in telecommunications", *Journal of Electrical Engineering*, 64(4), pp. 230, 2013, doi: https://doi.org/10.2478/jee-2013-0033.
102. R. Sarojini, and & C. Rambabu, "Design and implementation of DSSS-CDMA transmitter and receiver for reconfigurable links using FPGA", *International Journal of Recent Technology and Engineering (IJRTE).* ISSN, pp. 2277-3878, 2012.
103. A. Kumar, P. Vishnoi, and S. SL, "Smart grid security with cryptographic chip integration", *EAI Endorsed Transactions on Energy Web,* 6(23), 2019, doi: 10.4108/eai.13-7-2018.157037.
104. M. Yadav, K. Singh, A.S. Pandey, A. Kumar, and R. Kumar, "Smart Communication and Security by Key Distribution in Multicast

Environment", *Wireless Communications and Mobile Computing,* 2022, doi: https://doi.org/10.1155/2022/1011407.

105. E. Kaltofen, and G. Yuhasz, "On the matrix Berlekamp-Massey algorithm", *ACM Transactions on Algorithms (TALG),* 9(4), pp. 1-24, 2013, doi: https://doi.org/10.1145/2500122.
106. M.Y. Liu, T.L. Wang, Y.C. Wang, and C.H. Chiang, "Optical CDMA system using duobinary signaling scheme and LDPC code", *Optical and Quantum Electronics,* 48(1), pp.1-14, 2016, doi: https://doi.org/10.1007/s11082-015-0288-x.
107. V. Sharma, and S. Banerjee, "Quantum communication using code division multiple access network", *Optical and Quantum Electronics,* 52(8), pp. 1-22, 2020, doi: https://doi.org/10.1007/s11082-020-02494-3.
108. A. Kumar and Y. Zhu, "Extending Direct Sequence Spread-Spectrum for Secure Communication", *2021 International Symposium on Networks, Computers and Communications (ISNCC),* Dubai, United Arab Emirates, 2021, pp. 1-5, doi: 10.1109/ISNCC52172.2021.9615783.
109. S.B. Lokanatha, and S.B.B Prashanth, "Design and performance analysis of human body communication digital transceiver for wireless body area network applications", *International Journal of Electrical & Computer Engineering* , pp.2088-8708, 2022, doi: http://doi.org/10.11591/ijece.v12i3.pp2206-2213.
110. Y. -L. Lee, Y. -R. Li and L. -C. Wang, "Comparative Study on CDMA Code Design of PMCW Radar on Long-Range Multi-Objects Detection: (Extended Abstract)", *2021 30th Wireless and Optical Communications Conference (WOCC),* Taipei, Taiwan, 2021, pp. 209-210, doi: 10.1109/WOCC53213.2021.9603078.
111. K. Kashyap, S. Pathak, and N.S. Yadav, "Optimization of Spreading Code Using Modified Differential Evolution for Wireless Communication", *Wireless Personal Communications,* 122(2), pp. 1283-1304, 2022, https://doi.org/10.1007/s11277-021-08949-4.

112. A. Gezer, “Improving IEEE 802.15.4 performance with a switched Gold sequence chip formation. *Wireless Networks*, 26(6), pp. 4579-4593, 2020, https://doi.org/10.1007/s11277-021-08949-4.
113. K.H. Rahouma and A.A. Elsayed, “Applying Software Defined Network Concepts for Securing the Client Data Signals Over the Optical Transport Network of Egypt”, *Machine Learning and Big Data Analytics Paradigms: Analysis, Applications and Challenges*, pp. 583-601, 2021, doi: https://doi.org/10.1007/978-3-030-59338-4_28.
114. G.K. Maity, A.K. Mandal, N.B. Manik, and J.D. White, “All-optical TOAD based new binary sequence generator”, *Optical and Quantum Electronics,* 48(6), pp. 1-15, 2016, doi: https://doi.org/10.1007/s11082-016-0604-0.
115. V. Shivakumar, C. Senthilpari and Z. Yusoff, "A Low-Power and Area-Efficient Design of a Weighted Pseudorandom Test-Pattern Generator for a Test-Per-Scan Built-in Self-Test Architecture”, *IEEE Access*, vol. 9, pp. 29366-29379, 2021, doi: 10.1109/ACCESS.2021.3059171.
116. P. S. Dilip, G. R. Somanathan and R. Bhakthavatchalu, "Reseeding LFSR for Test Pattern Generation”, *2019 International Conference on Communication and Signal Processing (ICCSP), Chennai, India,* pp. 0921-0925, 2019, doi: 10.1109/ICCSP.2019.8698025.
117. P. Zode, P. Zode and R. Deshmukh, "FPGA Based Novel True Random Number Generator using LFSR with Dynamic Seed”, *2019 IEEE 16th India Council International Conference (INDICON)*, Rajkot, India, pp. 1-3, 2019, doi: 10.1109/INDICON47234.2019.9029049.
118. Y. Ogasahara, Y. Hori, T. Katashita, T. Iizuka, H. Awano, M. Ikeda, and H. Koike. "Implementation of pseudo-linear feedback shift register- based physical unclonable functions on silicon and sufficient Challenge– Response pair acquisition using Built-In Self-Test before shipping." *Integration* 71, pp. 144-153, 2020, doi: https://doi.org/10.1016/j.vlsi.2019.12.002.

119. M. Hossain, K.E. Zoiros, T. Chattopadhyay, and J.K. Rakshit, “Speed enhancement of all-optical pseudo random binary sequence (PRBS) generator using microring resonator”, *Optical and Quantum Electronics*, 53(122), pp. 1-31, 2021, doi: https://doi.org/10.1007/s11082-021-03329-5.
120. N. Gupta, K.S. Vaisla, A. Jain, A. Kumar and R. Kumar, “Performance Analysis of AODV Routing for Wireless Sensor Network in FPGA Hardware”, *Computer Systems Science and Engineering*, 39(2), pp. 1-12, 2021, doi: 10.32604/csse.2022.019911.
121. N. Gupta, A. Jain, K.S. Vaisla, A. Kumar, and R. Kumar, “Performance analysis of DSDV and OLSR wireless sensor network routing protocols using FPGA hardware and machine learning”, *Multimedia Tools and Applications*, 80(14), pp. 22301-22319, 2021, doi: https://doi.org/10.1007/s11042-021-10820-4.
122. A. Kumar, P. Sharma, M.K. Gupta, and R. Kumar, “Machine learning-based resource utilization and pre-estimation for network on chip (NoC) communication”, *Wireless Personal Communications*, 102(3), pp. 2211-2231, 2018, doi: https://doi.org/10.1007/s11277-018-5376-3.
123. 3GPP Technical Specifications 38.211, *Technical Specification Group Radio Access Network, NR* (Release 15), v15.0.0, 2017.
124. H. Ji, S. Park, J. Yeo, Y. Kim, J. Lee, and B. Shim, “Introduction to ultra reliable and low latency communications in 5G”, *arXiv*: 1704.05565v1, 2017.
125. M. Shirvanimoghaddam *et al.*, "Short Block-Length Codes for Ultra-Reliable Low Latency Communications”, *IEEE Communications Magazine*, vol. 57, no. 2, pp. 130-137, February 2019, doi: 10.1109/MCOM.2018.1800181.
126. 3GPP TSG RAN WG1, R1-1611692, “Channel coding schemes for URLLC scenario”, *Huawei, HiSilicon*, pp. 93-99, 2016, doi: 10.2991/ijndc.k.190702.005.

127. A. Sharma and M. Salim, “Polar code appropriateness for ultra-reliable and low-latency use cases of 5G systems”, *International Journal of Networked and Distributed Computing,* 7(3), pp. 93-99, 2019, doi: https://doi.org/10.2991/ijndc.k.190702.005.
128. V. Bioglio, C. Condo and I. Land, "Design of Polar Codes in 5G New Radio”, *IEEE Communications Surveys & Tutorials*, vol. 23, no. 1, pp. 29-40, Firstquarter 2021, doi: 10.1109/COMST.2020.2967127.
129. G.K. Prayogo, R. Putra, A.H. Prasetyo, and M. Suryanegara, “Evaluation of LDPC code and polar code coding scheme in 5G technology massive machine type communication”, 2018 10th *International Conference on Information Technology and Electrical Engineering (ICITEE)*, pp. 170-174, 2018, doi: 10.1109/ICITEED.2018.8534937.
130. F. Ercan, C. Condo, S. A. Hashemi and W. J. Gross, "On error-correction performance and implementation of polar code list decoders for 5G", *2017 55th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, Monticello, IL, USA, 2017, pp. 443-449, doi: 10.1109/ALLERTON.2017.8262771.
131. S. A. Hashemi, C. Condo, F. Ercan and W. J. Gross, "On the performance of polar codes for 5G eMBB control channel”, *2017 51st Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, USA, 2017, pp. 1764-1768, doi: 10.1109/ACSSC.2017.8335664.
132. P. Yuan, T. Prinz, G. Boecherer, O. Iscan, R. Boehnke and W. Xu, "Polar Code Construction for List Decoding”, *SCC 2019; 12th International ITG Conference on Systems, Communications and Coding*, Rostock, Germany, 2019, pp. 1-6, doi: 10.30420/454862022.
133. M. Qin, J. Guo, A. Bhatia, A. Guillén i Fàbregas and P. H. Siegel, "Polar Code Constructions Based on LLR Evolution”, *IEEE*

Communications Letters, vol. 21, no. 6, pp. 1221-1224, June 2017, doi: 10.1109/LCOMM.2017.2656126.

134. A. Sharma and M. Salim, "Polar Code: The Channel Code contender for 5G scenarios", *2017 International Conference on Computer, Communications and Electronics (Comptelix),* Jaipur, India, 2017, pp. 676-682, doi: 10.1109/COMPTELIX.2017.8004055.
135. C. Albayrak, C and K.Turk, "Reduced-complexity decoding of LT codes", *Wireless Personal Communications,* 94(3), pp. 969-975, 2017, doi: https://doi.org/10.1007/s11277-016-3660-7.
136. K. Arora, J. Singh, and Y.S. Randhawa, "A survey on channel coding techniques for 5G wireless networks" *Telecommunication Systems,* pp. 1-27, 2019, doi: https://doi.org/10.1007/s11235-019-00630-3.
137. J.H. Bae, A. Abotabl, H.P. Lin, K.B. Song, and J. Lee",An overview of channel coding for 5G NR cellular communications", *APSIPA Transactions on Signal and Information Processing,* 2019, doi: 10.1017/ATSIP.2019.10.
138. A. Devrari, A.Kumar, H. Chauhan, and A. Kumar, "Design and FPGA Implementation of LDPC Decoder Chip for Communication System using VHDL", *International Journal of Recent Technology and Engineering,* Vol 8(2), pp 200-206, 2019,doi: 10.21203/rs.3.rs-1288700/v1.
139. Y. Delomier, B. Le Gal, J. Crenne, and C. Jego, "Model-Based Design of Flexible and Efficient LDPC Decoders on FPGA Devices", *Journal of Signal Processing Systems,* pp.1-19, 2020, doi: https://doi.org/10.1007/s11265-020-01519-0.
140. P. Yuan, T. Prinz, G. Boecherer, O. Iscan, R. Boehnke and W. Xu, "Polar Code Construction for List Decoding", *SCC 2019; 12th International ITG Conference on Systems, Communications and Coding,* Rostock, Germany, 2019, pp. 1-6, doi: 10.30420/454862022.

141. A. Fazeli, H. Hassani, M. Mondelli and A. Vardy, "Binary Linear Codes with Optimal Scaling: Polar Codes with Large Kernels", *2018 IEEE Information Theory Workshop (ITW)*, Guangzhou, China, 2018, pp. 1-5, doi: 10.1109/ITW.2018.8613428.
142. Y. Yu, Z. Pan, N. Liu and X. You, "Belief Propagation Bit-Flip Decoder for Polar Codes", *IEEE Access*, vol. 7, pp. 10937-10946, 2019, doi: 10.1109/ACCESS.2019.2891951.
143. J. Mosheiff, N. Resch, N. Ron-Zewi, S. Silas and M. Wootters, "LDPC Codes Achieve List Decoding Capacity", *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, Durham, NC, USA, 2020, pp. 458-469, doi: 10.1109/FOCS46700.2020.00050.
144. P. Panteleev and G. Kalachev, "Quantum LDPC Codes With Almost Linear Minimum Distance", *IEEE Transactions on Information Theory*, vol. 68, no. 1, pp. 213-229, Jan. 2022, doi: 10.1109/TIT.2021.3119384.
145. F. Hamidi-Sepehr, A. Nimbalker and G. Ermolaev, "Analysis of 5G LDPC Codes Rate-Matching Design", *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*, Porto, Portugal, 2018, pp. 1-5, doi: 10.1109/VTCSpring.2018.8417496.
146. L. Fanari, E. Iradier, J. Montalban, P. Angueira, Ó. Seijo and I. Val, "PEG-LDPC Coding for Critical Communications in Factory Automation", *2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, Vienna, Austria, 2020, pp. 1135-1138, doi: 10.1109/ETFA46521.2020.9211899.
147. J.M. Gilbert, C. Robbins and W. Sheikh, "FPGA implementation of error control codes in VHDL": An undergraduate research project. *Computer Applications in Engineering Education*, *27*(5), pp. 1073-1086, 2019, doi: 10.1002/cae.22137.
148. Z.R. Hajiyat, A. Sali, A, M. Mokhtar, and F. Hashim, "Channel Coding Scheme for 5G Mobile Communication System for Short

Length Message Transmission", *Wireless Personal Communications*, 106(2), pp. 377-400, 2019, doi: https://doi.org/10.1007/s11277-019-06167-7.

149. L. Li and W. Zhang, "On the encoding complexity of systematic polar codes", *2015 28th IEEE International System-on-Chip Conference (SOCC)*, Beijing, China, 2015, pp. 415-420, doi: 10.1109/SOCC.2015.7406996.
150. K.D. Rao, "Channel coding techniques for wireless communications", *Springer India*, 2018, doi: https://doi.org/10.1007/978-981-15-0561-4.

Publications

[1] Aakanksha Devrari, Adesh Kumar, Piyush Kuchhal "Global Aspects and Overview of 5G Multimedia Communication", *Multimedia Tools and Applications, Springer*, Vol. 83, 2024, pp: 26439-26484, doi: https://doi.org/10.1007/s11042-023-16549-6, (**SCI Indexed, Impact Factor = 3.6**).

[2] Aakanksha Devrari, Adesh Kumar, Piyush Kuchhal, Zoltán Illés, Chaman Verma, "Sequential Logic Circuit Gold Codes for Electronics and Communication Technologies", *MethodsX, Elsevier*, Vol. 12, 2024, pp: 102602-102615, doi: https://doi.org/10.1016/j.mex.2024.102602, **(SCI Indexed, Impact Factor =1.9).**

[3] Aakanksha Devrari, Adesh Kumar "Chip Design and FPGA Device Analysis of Turbo Encoder and Decoder for Communication System", *International Journal of Reconfigurable and Embedded Systems* (IJRES), Vol. 12, 2023, pp. 174-185, doi: http://doi.org/10.11591/ijres.v12.i2.pp174-185, (**Scopus Indexed).**

[4] Aakanksha Devrari, Adesh Kumar "Reconfigurable Linear Feedback Shift register for Wireless Communication", *International Journal of Reconfigurable and Embedded Systems* (IJRES), Vol. 12, No. 2, 2023, pp. 195-204, doi: http://doi.org/10.11591/ijres.v12.i2.pp195-204, (**Scopus Indexed).**

Author's Curriculum Vitae

Aakanksha Devrari is a Research Scholar in UPES, Dehradun, Uttarakhand (Jan 2020 - Jan 2023) pursuing Ph. D (Full time) from Department of Electrical & Electronics, School of Engineering, UPES, Dehradun, Uttarakhand. She has completed her M.Tech. degree in VLSI Design from Faculty of Technology, Uttarakhand Technical University. in the year 2013 and completed B.Tech. in Electronics & Communication in the year 2011.

Contact Information

Permanent Address: 161/1/3A Kochar Colony, Jakhan, Rajpur Road, Dehradun, 248001

Email: aks.uit08@gmail.com

Research Interests:

5G communication system, VLSI & Wireless communication.

Technical Skills:

(i) **Programming**: VHDL, VERILOG, Embedded C.

(ii) **Tools/Platforms:** Modelsim, Xilinx ISE 14.7, Xilinx Vivado, Spartan 3E FPGA, Virtex 5 FPGA, Zedboard Zynq-7000 Development Board, TSPICE simulation software, Keil C51 Development tool, Proteus Design suite 8.0 for electronic design automation.

Education:

(i) Master of Technology (VLSI): Faculty of Technology, Uttarakhand Technical University, Dehradun, Uttarakhand, 2011-2013, (82.3%).

(ii) Bachelor of Technology (Electronics & Communication): Uttaranchal Institute of Technology, Uttarakhand Technical University, Dehradun, Uttarakhand, 2007-2011, (74.0%).

(iii) Intermediate: ST. Theresa's, Srinagar Garhwal , Uttarakhand, 2007, (72.0%)

(iv) High school: ST. Theresa's, Srinagar Garhwal , Uttarakhand, 2005, (70.0%)

Experience:

(i) Currently working as an Assistant professor in Women Institute of Technology, Campus Govt. Institute of VMSBTU, Dehradun, Uttarakhand.

(ii) 3 years of experience as a Doctoral research Fellow, R&D Department, UPES, Dehradun, Uttarakhand (From Jan 2020 to Jan 2023).

(iii) 3 years of experience as an Assistant professor in Women Institute of Technology, Campus Govt. Institute of VMSBTU, Dehradun, Uttarakhand (From Dec 2014 to Dec 2017).

ad

ORIGINALITY REPORT

8%	4%	7%	1%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.ijrte.org Internet Source	<1%
2	www.atlantis-press.com Internet Source	<1%
3	mafiadoc.com Internet Source	<1%
4	link.springer.com Internet Source	<1%
5	spiral.imperial.ac.uk Internet Source	<1%
6	Submitted to University of Tennessee Knoxville Student Paper	<1%
7	Submitted to University of Hertfordshire Student Paper	<1%
8	"Proceeding of International Conference on Intelligent Communication, Control and Devices", Springer Nature, 2017 Publication	<1%

9	Namit Gupta, Arpit Jain, Kunwar Singh Vaisla, Adesh Kumar, Rajeev Kumar. "Performance analysis of DSDV and OLSR wireless sensor network routing protocols using FPGA hardware and machine learning", Multimedia Tools and Applications, 2021 Publication	<1 %
10	Muhammad Baqer Mollah, Md. Rashidul Islam. "Comparative analysis of Gold Codes with PN codes using correlation property in CDMA technology", 2012 International Conference on Computer Communication and Informatics, 2012 Publication	<1 %
11	Valerio Bioglio, Carlo Condo, Ingmar Land. "Design of Polar Codes in 5G New Radio", IEEE Communications Surveys & Tutorials, 2020 Publication	<1 %
12	"Cryptographic Hardware and Embedded Systems – CHES 2014", Springer Science and Business Media LLC, 2014 Publication	<1 %
13	Haesik Kim. "Coding and modulation techniques for high spectral efficiency transmission in 5G and satcom", 2015 23rd European Signal Processing Conference (EUSIPCO), 2015 Publication	<1 %

14	www.researchgate.net Internet Source	<1%
15	"Optical and Wireless Technologies", Springer Science and Business Media LLC, 2020 Publication	<1%
16	Furkan Ercan, Carlo Condo, Seyyed Ali Hashemi, Warren J. Gross. "On error-correction performance and implementation of polar code list decoders for 5G", 2017 55th Annual Allerton Conference on Communication, Control, and Computing (Allerton), 2017 Publication	<1%
17	Aarti Sharma, Mohammad Salim. "Polar Code: The Channel Code contender for 5G scenarios", 2017 International Conference on Computer, Communications and Electronics (Comptelix), 2017 Publication	<1%
18	Mostafa Rashdan. "Differential Time Signaling Data-Link Architecture", Journal of Signal Processing Systems, 02/29/2012 Publication	<1%
19	Zhiwei Zhong, Xiaohu You, Chuan Zhang. "Auto-generation of pipelined hardware designs for polar encoder", 2018 China	<1%

Semiconductor Technology International Conference (CSTIC), 2018
Publication

20	Sahni, Kanika, Kiran Rawat, Sujata Pandey, and Ziauddin Ahmad. "Low power approach for implementation of 8B/10B encoder and 10B/8B decoder used for high speed communication", 2014 2nd International Conference on Emerging Technology Trends in Electronics Communication and Networking, 2014. Publication	<1 %
21	"Communications, Signal Processing, and Systems", Springer Science and Business Media LLC, 2019 Publication	<1 %
22	www.arxiv-vanity.com Internet Source	<1 %
23	www.ncbi.nlm.nih.gov Internet Source	<1 %
24	Ali Gezer. "Improving IEEE 802.15.4 performance with a switched Gold sequence chip formation", Wireless Networks, 2020 Publication	<1 %
25	Aakanksha Devrari, Adesh Kumar, Himanshu Chauhan, Amit Kumar. "Design and FPGA Implementation of LDPC Decoder Chip for	<1 %

Communication System using VHDL", International Journal of Recent Technology and Engineering (IJRTE), 2019
Publication

26	Adesh Kumar, Gaurav Verma, Mukul Kumar Gupta, Mohammad Salauddin, B. Khaleelu Rehman, Deepak Kumar. "3D Multilayer Mesh NoC Communication and FPGA Synthesis", Wireless Personal Communications, 2018 Publication	<1 %
27	ds.amu.edu.et Internet Source	<1 %
28	epdf.tips Internet Source	<1 %
29	www.mdpi.com Internet Source	<1 %
30	www.techscience.com Internet Source	<1 %
31	Tewelgn Kebede, Yihenew Wondie, Johannes Steinbrunn, Hailu Belay, Kevin Kornegay. "Multi-carrier Waveforms and Multiple Access Strategies in Wireless Networks: Performance, Applications and Challenges", IEEE Access, 2022 Publication	<1 %
32	dokumen.pub Internet Source	<1 %

33	patents.justia.com Internet Source	<1 %
34	Goutam Kumar Maity, Ashis Kumar Mandal, Nabin Baran Manik, Jonathon David White. "All-optical TOAD based new binary sequence generator", Optical and Quantum Electronics, 2016 Publication	<1 %
35	Valerio Bioglio, Carlo Condo, Ingmar Land. "Design of Polar Codes in 5G New Radio", IEEE Communications Surveys & Tutorials, 2021 Publication	<1 %
36	Handbook of Combinatorial Optimization, 2013. Publication	<1 %
37	www.datasheet.hk Internet Source	<1 %
38	www.koreascience.or.kr Internet Source	<1 %
39	Submitted to University of Petroleum and Energy Studies Student Paper	<1 %
40	techscience.com Internet Source	<1 %
41	ia803400.us.archive.org Internet Source	<1 %

42	A.Albert Raj, M. Pooranasankari, S.Sri Abinayaa. "Design of Successive cancellation List Decoding of Polar Codes", 2021 6th International Conference on Communication and Electronics Systems (ICCES), 2021 Publication	<1 %
43	M. Jayasanthi, R. Kalaivani. "Low-power DSSS transmitter and its VLSI implementation", Annals of Telecommunications, 2021 Publication	<1 %
44	"Data Engineering and Communication Technology", Springer Science and Business Media LLC, 2021 Publication	<1 %
45	"Inventive Communication and Computational Technologies", Springer Science and Business Media LLC, 2020 Publication	<1 %
46	Mangal Deep Gupta, R.K. Chauhan. "Secure image encryption scheme using 4D-Hyperchaotic systems based reconfigurable pseudo-random number generator and S-Box", Integration, 2021 Publication	<1 %
47	Shibao Li, Yunqiang Deng, Xun Gao, He Li, Lin Guo, Zhenwei Dong. "Generalized Segmented Bit-Flipping Scheme for Successive	<1 %

	Cancellation Decoding of Polar Codes With Cyclic Redundancy Check", IEEE Access, 2019 Publication	
48	Swapnil P. Badar, Kamalesh Khanchandani. "Successive Cancellation Polar Decoder Implementation using Processing Elements", 2022 IEEE Region 10 Symposium (TENSYMP), 2022 Publication	<1 %
49	gyan.iitg.ernet.in Internet Source	<1 %
50	A. Kavitha. "Design of Low Power TPG Using LP-LFSR", 2012 Third International Conference on Intelligent Systems Modelling and Simulation, 02/2012 Publication	<1 %
51	Submitted to CSU, Fullerton Student Paper	<1 %
52	Modern Cryptography Primer, 2013. Publication	<1 %
53	V Sobolev, E Shchepakina. "Cheap control problem for micro drone quadcopter", Journal of Physics: Conference Series, 2018 Publication	<1 %
54	qabasjournals.com Internet Source	<1 %

55	"Communications and Networking", Springer Science and Business Media LLC, 2021 Publication	<1 %
56	Submitted to Rochester Institute of Technology Student Paper	<1 %
57	Souhila Sadeg, Mohamed Gougache, Nabil Mansouri, Habiba Drias. "An encryption algorithm inspired from DNA", 2010 International Conference on Machine and Web Intelligence, 2010 Publication	<1 %
58	pages.hmc.edu Internet Source	<1 %
59	"Intelligent Communication, Control and Devices", Springer Science and Business Media LLC, 2018 Publication	<1 %
60	Haesik Kim. "Design and Optimization for 5G Wireless Communications", Wiley, 2020 Publication	<1 %
61	Lecture Notes in Computer Science, 2003. Publication	<1 %
62	Ompal, Vishnu Mohan Mishra, Adesh Kumar. "ZigBee Communication in WSN with Mesh Configured Routers and Hardware Chip", 2023 International Conference on Power,	<1 %

Instrumentation, Energy and Control (PIECON), 2023
Publication

63	Submitted to Middlesex University Student Paper	<1%
64	Yunlong Cai, Zhijin Qin, Fangyu Cui, Geoffrey Ye Li, Julie A. McCann. "Modulation and Multiple Access for 5G Networks", IEEE Communications Surveys & Tutorials, 2018 Publication	<1%
65	Zahraa Raad Mayoof Hajiyat, Aduwati Sali, Makhfudzah Mokhtar, Fazirulhisyam Hashim. "Channel Coding Scheme for 5G Mobile Communication System for Short Length Message Transmission", Wireless Personal Communications, 2019 Publication	<1%
66	Ompal, Vishnu Mohan Mishra, Adesh Kumar. "Zigbee Internode Communication and FPGA Synthesis Using Mesh, Star and Cluster Tree Topological Chip", Wireless Personal Communications, 2021 Publication	<1%
67	Pooja Pathak, Richa Bhatia. "Performance Analysis of Polar codes for Next Generation 5G Technology", 2022 3rd International Conference for Emerging Technology (INCET), 2022	<1%

	Publication	
68	Vaibbhav Taraate. "Digital Logic Design Using Verilog", Springer Science and Business Media LLC, 2022 Publication	<1 %
69	biologicas.ucm.es Internet Source	<1 %
70	moam.info Internet Source	<1 %
71	www.granthaalayahpublication.org Internet Source	<1 %
72	Adesh Kumar, Piyush Kuchhal, Sonal Singhal. "Five Stage Telecommunication Switching Design and Synthesis", Wireless Personal Communications, 2017 Publication	<1 %
73	C. -L. Lu, H. -C. Wang, J. -C. Juang, H. -R. Chuang. "10-Gb/s CMOS ultrahigh-speed gold-code generator using differential-switches feedback", 2007 European Microwave Integrated Circuit Conference, 2007 Publication	<1 %
74	Submitted to CSU, San Jose State University Student Paper	<1 %
75	Mamta Agiwal, Abhishek Roy, Navrati Saxena. "Next Generation 5G Wireless Networks: A	<1 %

Comprehensive Survey", IEEE Communications Surveys & Tutorials, 2016
Publication

76	hdl.handle.net Internet Source	<1 %
77	patents.google.com Internet Source	<1 %
78	repository.wima.ac.id Internet Source	<1 %
79	theses.hal.science Internet Source	<1 %
80	web.archive.org Internet Source	<1 %
81	www.waset.org Internet Source	<1 %
82	Akash Goel, Amit Kumar Goel, Adesh Kumar. "Performance analysis of multiple input single layer neural network hardware chip", Multimedia Tools and Applications, 2023 Publication	<1 %
83	Submitted to Coventry University Student Paper	<1 %
84	Gianluca Giustolisi, Rosario Mita, Gaetano Palumbo, Giuseppe Scotti. "A Novel Clock Gating Approach for the Design of Low-Power	<1 %

Linear Feedback Shift Registers", IEEE Access, 2022
Publication

85	escholarship.mcgill.ca Internet Source	<1 %
86	etheses.whiterose.ac.uk Internet Source	<1 %
87	ir.lib.vntu.edu.ua Internet Source	<1 %
88	www.scribd.com Internet Source	<1 %
89	Seyyed Ali Hashemi, Marco Mondelli, Arman Fazeli, Alexander Vardy, John Cioffi, Andrea Goldsmith. "Parallelism versus Latency in Simplified Successive-Cancellation Decoding of Polar Codes", IEEE Transactions on Wireless Communications, 2021 Publication	<1 %
90	cyberleninka.org Internet Source	<1 %
91	Andrew J. Maier, Bruce F. Cockburn. "Optimization of Low-Density Parity Check decoder performance for OpenCL designs synthesized to FPGAs", Journal of Parallel and Distributed Computing, 2017 Publication	<1 %

92	H. TANAKA. "Linearization Method and Linear Complexity", IEICE Transactions on Fundamentals of Electronics Communications and Computer Sciences, 01/01/2008 Publication	<1 %
93	Submitted to University of Bristol Student Paper	<1 %
94	A Suresh Babu, B Anand. "Modified dynamic current mode logic based LFSR for low power applications", Microprocessors and Microsystems, 2020 Publication	<1 %
95	Salager, Elodie, Robin S. Stein, Chris J. Pickard, BÃ©nÃ©dicte Elena, and Lyndon Emsley. "Powder NMR crystallography of thymol", Physical Chemistry Chemical Physics, 2009. Publication	<1 %
96	Zhiting Yan, Guanghui He, Weifeng He, Shuaijie Wang, Zhigang Mao. "High performance parallel turbo decoder with configurable interleaving network for LTE application", Integration, the VLSI Journal, 2016 Publication	<1 %

Exclude quotes On

Exclude matches < 14 words