Error and Erasure Decoding for a CDPD System

by

Chenbo Zheng

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

IN

ELECTRICAL ENGINEERING

APPROVED:

Brian Woerner
Dr. Brian D. Woerner, Chair

Dr. Jeffery H. Reed

Dr. F. Gail Gray

June, 1996

Blacksburg, Virginia

Keywords: CDPD, Wireless, Reed-Solomon Code, GMSK
Error and Erasure Decoding for a CDPD System

by

Chenbo Zheng

Committee Chairman: Dr. Brian D. Woerner

Electrical Engineering

(ABSTRACT)

Cellular digital packet data (CDPD) is a new service for wide-area data communication with wireless mobile users. CDPD system uses the existing infrastructure of the analog Advanced Mobile Phone Service (AMPS) cellular telephone network to transmit data with a channel hopping technique. The CDPD system employs Gaussian minimum shift keying (GMSK) as a modulation scheme and a Reed-Solomon code for error control to transmit high quality data in the mobile and wireless environment. Most current CDPD receivers use errors only decoding of the Reed-Solomon code, although an improved errors and erasures decoding technique would also be possible.

This thesis undertakes a performance evaluation of the CDPD system with an errors and erasures decoder for the Reed-Solomon coding. A thorough system simulation is conducted for both white Gaussian noise and flat Rayleigh fading channel environments. Results show that improved coding gains of $0.5 \sim 1\, \text{dB}$ are possible for the additive white Gaussian noise channel and improved coding gains of $1.9 \sim 2.7\, \text{dB}$ are possible for the fading channel.
Acknowledgments

I wish to express my sincere thanks to Dr. Brian D. Woerner, my academic advisor and committee chairman, for his continuous advice, guidance and encouragement. It is largely due to his helpful assistance and suggestions throughout this study, that I have been able to complete this thesis.

Special thanks go to Dr. Jeffery H. Reed who gave me considerable advice and support on this project, and also served as a member of my thesis committee. I would like to thank Dr. F. Gail Gray for all the kind help he gave me during my graduate studies at Virginia Polytechnic Institute and State University and for serving on my thesis committee.

I wish to thank Mr. Scott Elson, Mr. Jeff Laster and Mr. Mike Buehrer for their helpful discussion and assistance in the simulation of CDPD and GMSK. And also I wish to thank Mr. Allen Comer, Mr. Farooq Azam, Mr. Chang Yong and Mr. Sanjay Shenoy for their great help on MATLAB and UNIX system.

Finally, I would like to express my deepest gratitude to my parents for their patience and encouragement, to my wife for her love and support and to my son for giving me his love and confidence. Without them, it is would not have been possible for me to finish my Master's degree in the United States.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Chapter 1. Introduction</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1 Signal Transmission in Wireless Network</td>
<td>2</td>
</tr>
<tr>
<td>1.2 Overview of CDPD System</td>
<td>4</td>
</tr>
<tr>
<td>1.2.1 CDPD Implementation Technique</td>
<td>4</td>
</tr>
<tr>
<td>1.2.2 CDPD Network</td>
<td>5</td>
</tr>
<tr>
<td>1.2.3 CDPD Network Entities</td>
<td>8</td>
</tr>
<tr>
<td>1.3 Airlink Specification of the CDPD System</td>
<td>12</td>
</tr>
<tr>
<td>1.4 Research Objective</td>
<td>13</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chapter 2. CDPD Signal Format and Signaling Technique</th>
<th>15</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1 Data Packet Transformation in CDPD</td>
<td>15</td>
</tr>
<tr>
<td>2.2 Forward Channel Block Format</td>
<td>17</td>
</tr>
<tr>
<td>2.2.1 Pseudo-Random Number (PN) Coverage</td>
<td>18</td>
</tr>
<tr>
<td>2.2.2 Synchronization Word</td>
<td>20</td>
</tr>
<tr>
<td>2.2.3 Control Flag</td>
<td>21</td>
</tr>
<tr>
<td>2.3 Reverse Channel Format</td>
<td>23</td>
</tr>
<tr>
<td>2.3.1 Synchronization Flag</td>
<td>24</td>
</tr>
<tr>
<td>2.3.2 Continuity Indicator</td>
<td>25</td>
</tr>
<tr>
<td>2.4 Gaussian Minimum Shift Keying (GMSK)</td>
<td>26</td>
</tr>
<tr>
<td>2.4.1 Minimum Shift Keying</td>
<td>26</td>
</tr>
<tr>
<td>2.4.2 Gaussian Filtered MSK</td>
<td>27</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chapter 3. Reed Solomon Coding in CDPD System</th>
<th>34</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1 Reed Solomon Encoding</td>
<td>34</td>
</tr>
<tr>
<td>3.1.1 Coding Theory Background</td>
<td>34</td>
</tr>
<tr>
<td>3.1.2 Reed-Solomon Systematic Encoding</td>
<td>34</td>
</tr>
<tr>
<td>3.2 Reed-Solomon Decoding</td>
<td>41</td>
</tr>
<tr>
<td>3.2.1 Berlekamp-massey Decoding Algorithm</td>
<td>42</td>
</tr>
<tr>
<td>3.2.2 Error Correction of Reed-Solomon</td>
<td>47</td>
</tr>
<tr>
<td>3.3 Error and Erasure Correction Decoding</td>
<td>51</td>
</tr>
<tr>
<td>3.3.1 Soft Decision and Erasure Decoding</td>
<td>51</td>
</tr>
<tr>
<td>3.3.2 Erasure Decoding Algorithm</td>
<td>56</td>
</tr>
</tbody>
</table>
Chapter 4. System Modeling and Simulation................................. 61
  4.1 Transmitter Block.............................................................. 62
  4.2 Receiver Block...................................................................... 64
    4.2.1 GMSK Demodulation...................................................... 65
    4.2.2 Reed-Solomon Decoder................................................... 72
  4.3 Physical Channel Models...................................................... 74
    4.3.1 AWGN Channel.............................................................. 75
    4.3.2 Rayleigh Fading Channel............................................... 77
  4.4 Performance Analysis.......................................................... 81
    4.4.1 Block Error Rate......................................................... 81
    4.4.2 Performance of Reed-Solomon Code................................. 82
    4.4.3 System Simulation....................................................... 85

Chapter 5 Simulation Results..................................................... 90
  5.1 AWGN............................................................................. 91
    5.1.1 $BT=0.5$ Case............................................................... 91
    5.1.2 $BT=0.3$ Case.............................................................. 96
  5.2 Flat Rayleigh Fading........................................................... 98
    5.2.1 $v=100$ km/hr and $BT=0.5$ Case.................................... 99
    5.2.2 $v=50$ km/hr and $BT=0.5$ Case..................................... 105
    5.2.3 $v=8$ km/hr and $BT=0.5$ Case....................................... 105
    5.2.4 Rayleigh Fading with $BT=0.3$...................................... 108

Chapter 6. Conclusions.............................................................. 113
  6.1 Summary.......................................................................... 113
  6.2 Future............................................................................. 115
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 1.1</td>
<td>Packet Data Format</td>
<td>3</td>
</tr>
<tr>
<td>Figure 1.2</td>
<td>CDPD Channel-Hopping</td>
<td>6</td>
</tr>
<tr>
<td>Figure 1.3</td>
<td>A, E and I Interfaces</td>
<td>7</td>
</tr>
<tr>
<td>Figure 1.4</td>
<td>CDPD Network Entities</td>
<td>8</td>
</tr>
<tr>
<td>Figure 2.1</td>
<td>CDPD Airlink Setup</td>
<td>15</td>
</tr>
<tr>
<td>Figure 2.2</td>
<td>Data Packet Transformation</td>
<td>16</td>
</tr>
<tr>
<td>Figure 2.3</td>
<td>Forward Channel Format Processing</td>
<td>18</td>
</tr>
<tr>
<td>Figure 2.4</td>
<td>Forward Frame and Block Structure</td>
<td>18</td>
</tr>
<tr>
<td>Figure 2.5</td>
<td>PN Sequence Generator</td>
<td>19</td>
</tr>
<tr>
<td>Figure 2.6</td>
<td>Control Symbol in Forward Channel</td>
<td>21</td>
</tr>
<tr>
<td>Figure 2.7</td>
<td>Reverse Channel Transmission Structure</td>
<td>24</td>
</tr>
<tr>
<td>Figure 2.8</td>
<td>In-Phase and Quadrature of a Rectangle Pulse Train</td>
<td>28</td>
</tr>
<tr>
<td>Figure 2.9</td>
<td>MSK Modulation Signal with $f_c = 15$ Hz</td>
<td>29</td>
</tr>
<tr>
<td>Figure 2.10</td>
<td>GMSK Implementation</td>
<td>30</td>
</tr>
<tr>
<td>Figure 2.11</td>
<td>Rectangle Pulse Response of Gaussian Filter</td>
<td>32</td>
</tr>
<tr>
<td>Figure 2.12</td>
<td>Filtered Rectangle Pulse Train</td>
<td>32</td>
</tr>
<tr>
<td>Figure 2.13</td>
<td>Frequency Response of the Rectangle Pulse Train</td>
<td>33</td>
</tr>
<tr>
<td>Figure 3.1</td>
<td>Reed-Solomon Systematic Encoder</td>
<td>39</td>
</tr>
<tr>
<td>Figure 3.2</td>
<td>Block Structure of (63,47) Reed-Solomon</td>
<td>40</td>
</tr>
<tr>
<td>Figure 3.3</td>
<td>LFSR Implementation</td>
<td>46</td>
</tr>
<tr>
<td>Figure 3.4</td>
<td>Program Structure of Berlekamp-Massey Decoding Algorithm</td>
<td>48</td>
</tr>
<tr>
<td>Figure 3.5</td>
<td>Program Structure of Error Magnitude Calculation</td>
<td>50</td>
</tr>
<tr>
<td>Figure 3.6</td>
<td>Hard-Decision Mechanism</td>
<td>52</td>
</tr>
<tr>
<td>Figure 3.7</td>
<td>Erasure-Soft Decision Mechanism</td>
<td>53</td>
</tr>
<tr>
<td>Figure 3.8</td>
<td>Probability Distribution of Binary Decision</td>
<td>53</td>
</tr>
<tr>
<td>Figure 3.9</td>
<td>Uniform Discrete Symmetric with Erasures</td>
<td>54</td>
</tr>
<tr>
<td>Figure 3.10</td>
<td>Erasure Decoding Algorithm</td>
<td>60</td>
</tr>
<tr>
<td>Figure 4.1</td>
<td>Simulation Structure of CDPD Airlink</td>
<td>61</td>
</tr>
</tbody>
</table>
Figure 4.2 Format/Reed-Solomon Sub-Block................................................. 62
Figure 4.3 GMSK Modulation Sub-Block..................................................... 63
Figure 4.4 Receiver Block of CDPD Airlink................................................ 64
Figure 4.5 One-Bit Differential Detector.................................................... 66
Figure 4.6 Phase-State Diagram $BT=0.3$.................................................... 70
Figure 4.7 Phase-State Diagram $BT=0.5$.................................................... 70
Figure 4.8 Error Correction Only S-R Decoder.......................................... 73
Figure 4.9 Errors and Erasures S-R Decoder.............................................. 73
Figure 4.10 AWGN Channel Model.......................................................... 75
Figure 4.11 Rayleigh Fading Simulation.................................................... 78
Figure 4.12 Rayleigh Fading Channel Model............................................. 79
Figure 4.13 Performance of (63,47) R-S Code in CDPD............................... 83
Figure 4.14 Assumption of the Maximum of $P_c$........................................ 84
Figure 4.15 System Simulation without Error Control Code.......................... 87
Figure 4.16 System Simulation with R-S Code........................................... 88
Figure 4.17 System Simulation with Error and Erasure Decoder..................... 89

Figure 5.1 BLERs of No-Coding and Error-Correction Only Decoding
($BT=0.5$ and AWGN).............................................................................. 92
Figure 5.2 BLERs of No-Coding and Error-Erasure Decoding
($BT=0.5$ and AWGN).............................................................................. 93
Figure 5.3 BLERs of Three Cases ($BT=0.5$ and AWGN)............................... 94
Figure 5.4 BERs of No-Coding and Error-Correction Only Decoding
($BT=0.5$ and AWGN).............................................................................. 95
Figure 5.5 BERs of No-Coding and Error-Erasure Decoding
($BT=0.5$ and AWGN).............................................................................. 95
Figure 5.6 BERs of Three Cases ($BT=0.5$ and AWGN)............................... 96
Figure 5.7 BLERs of Three Cases ($BT=0.3$ and AWGN)............................... 97
Figure 5.8 BERs of Three Cases ($BT=0.3$ and AWGN)............................... 98
Figure 5.9 BLER of No-Coding and Error Correction only Coding
in Rayleigh Channels ($v=100\text{km/hr}$ and $BT=0.5$)............................. 100
Figure 5.10 BLER of No-Coding and Error-Erasure Decoding
in Rayleigh Channels ($v=100\text{km/hr}$ and $BT=0.5$)............................. 102
List of Tables

Table 2.1 PN Sequence Relationship to Block Structure........................................... 20
Table 2.2 Busy/Idle Flag......................................................................................... 22
Table 2.3 Decode Status......................................................................................... 22
Table 2.4 Binary Values of Dotting Sequence and Synchronization Word.............. 25
Table 2.5 Continuity Indicator Status..................................................................... 25
Table 3.1 The Number of Error/Erasure Corrections in CDPD................................. 56
Table 4.1 Signal $\theta_0$ and ISI Terms Vs. $BT$..................................................... 68
Table 4.2 Message Data and Phase Difference ($BT=0.3$)..................................... 69
Table 4.3 Message Data and Phase Difference ($BT=0.5$)..................................... 69
Table 4.4 Parameters for Different Vehicle Speed................................................ 80
Table 5.1 The Optimum Threshold for AWGN with $BT=0.5$................................. 93
Table 5.2 The Optimum Threshold for Rayleigh Fading ($v=100$km/hr & $BT=0.5$).. 101
Table 5.3 Simulation results for $BT=0.5$............................................................. 111
Table 5.4 Simulation results for $BT=0.3$............................................................. 112
Chapter 1 Introduction

Cellular digital packet data (CDPD) is a service for wide-area data communication with wireless mobile users. The original version of the CDPD System Specification (Release 1.0) was developed by IBM and released in 1993 by a consortium of the major U.S. cellular telephone carriers, including Ameritech Mobile Communications, Bell Atlantic Mobile Systems, Contel Cellular, GTE Mobile Communications, McCaw Cellular Communication, and Southwestern Bell Mobile Systems. The CDPD standard is now controlled by an open trade organization known as the CDPD Forum, which released a revision of the CDPD standard (Release 1.1) in 1995 [CDW96].

The CDPD system uses the existing infrastructure of the analog Advanced Mobile Phone Service (AMPS) cellular telephone network to transmit data, utilizing channels in the cellular system to provide a connectionless digital data packet service [GTE94]. The CDPD system operates at the physical layer and data link layer of the Open System Interconnection (OSI) model and provides service to either of two network layer protocols: the Internet Protocol (IP) and the International Standards Organization (ISO) Connectionless Network Protocol (CLNP) [Hil96]. Development and molding of the CDPD system is the subject of much current research, particularly since CDPD has been adopted as a standard for ITS communications. Greg Bump described one implementation of CDPD hardware in his MS thesis [Bum95]. Scott Elson developed a simulation model of the CDPD system in his thesis [Els96]. The purpose of this thesis is
to evaluate the benefits of an advanced decoding technique for use in a CDPD system. This chapter provides an introduction to the problem of interest.

1.1 Signal Transmission in Wireless Networks

The technique for signal transmission is one of the main features of a wireless network. There are two types of signal transmission methods in a wireless network: circuit-switching and packet-switching [Ber95]. Circuit-switching provides connection-oriented services and packet-switching (also called virtual switching) provides connectionless services. Circuit-switching is mostly used in the cellular system for voice transfers. Voice channels are dedicated for specific user’s calls, and network resources are dedicated to the voice traffic when a call is initialized. Circuit-switching establishes a dedicated connection (e.g. a radio channel between the base station and the mobile user, and a dedicated phone line between the Mobile Switching Center (MSC) and the Public Switched Telephone Network (PSTN)) for the entire duration of a call. Even when a mobile user hands off from one base station to another, there is always a dedicated radio channel and a fixed full duplex phone line between MSC and the PSTN to provide connection-oriented service to the user.

Although circuit switching can be used to transmit any type of data, it is best-suited for dedicated voice-only traffic or for larger batch data transfer (e.g., facsimile).
Circuit-switching for transferring short bursts of data is not efficient because of the delays incurred when initializing and discontinuing a call. For a burst data transmission, packet-switching is more suitable.

Connectionless service eliminates the dedicated resource elements in message transmission. Packet-switching is the most common technique used to implement connectionless services and allows a large number of data users to remain virtually connected to the same physical channel in the wireless network. In a packet-switching network, all users may access the network randomly. Set-up procedures for the dedicated connection are not needed when a particular user needs to send data. Packet-switching breaks each message into smaller units for transmission and recovery. Figure 1.1 shows the basic format of a packet transmission. Each packet consists of a header, user data and a trailer. The header contains a source address, a destination address and some control information. The user data contains the transmitted message which is generally protected by error control coding. The trailer contains a cyclic redundancy checksum which is used for error detection and error correction at the receiver.

| Header | Users Data | Trailer |

**Figure 1.1 Packet Data Format**

In contrast to circuit-switching, packet-switching provides much higher channel efficiency for burst data transmission. One of the most important advantages of packet-
switching data is that the channel is utilized only when sending or receiving bursts of information. Since the available bandwidth is limited in wireless communications, the excellent channel efficiency realized with packet-switching is a primary benefit in CDPD systems.

1.2 Overview of CDPD Systems

CDPD is a wireless digital packet service which utilizes excess bandwidth from the analog cellular system AMPS [GTE94]. CDPD is implemented as an overlay to existing cellular networks and allows data to be sent in packets during idle time on cellular voice channels. This overlay implementation makes use of the existing cellular infrastructure, simultaneously transmitting voice and data from the same device. Because the voice cellular radio system must maintain low blocking probabilities, almost 30% of the cellular channel capacity is unused, even during the busy hour [CDW96]. The CDPD system can capture this unused air time by hopping between idle voice channels, working independently of the voice base station and cellular telephone switch. The following overview of the CDPD system briefly discusses CDPD implementation techniques, the CDPD network interface and CDPD network entities.
12.1 CDPD Implementation Techniques

In CDPD technology, data shares cellular spectrum with standard AMPS voice channels. In order to integrate voice and data traffic on the cellular system, the CDPD system implements a technique called channel hopping [GTE94] [Jac95]. Channel-hopping allows the CDPD system to share the voice system’s expensive radio frequency (RF) infrastructure, including antennas, filters, low-noise amplifiers (LNA), channel banks and base stations as well as the radio frequency channel licenses.

In the CDPD system, the channel (called the channel stream) is a shared medium employing digital sense multiple access (DSMA) with collision detection (CD) [Hil96]. The channel-hopping technique is based on the fact that CDPD occupies voice channels purely on a secondary, noninterfering basis, and packet channels are dynamically assigned (hopped) to different cellular voice channels as they become vacant, so the CDPD radio channel is time-variant. Figure 1.2 shows an example of channel-hopping in the CDPD system. Channels A, B, C and D are four cellular radio channels. The CDPD data channel stream only accesses the idle cellular radio channels, but these channel ‘hops’ are completely transparent to the mobile data user. Therefore, to the user, only one data channel stream is used to complete the entire transmission.
In addition to channel hopping, the CDPD system adopts the packet-switching technique for data transmission. The data stream is divided into separate sections. Each section adds a header containing all control information such as the source and destination addresses. Packets are sent through the CDPD network individually and each may take a different path to the final destination. However, if any of the packets are received out of order, the receiving device reassembles the packets in the order of the original sequence. As mentioned before, the packet-switching data transmission technique provides excellent channel efficiency for the CDPD system.

1.2.2 CDPD Network Interfaces

Different systems of any network communicate with each other through an interface. Basically there are three distinct interfaces defined within the CDPD network
shown in Figure 1.3: Airlink (A) Interface, External (E) Interface and Inter-Service Provider (I) Interfaces [GTE94].

Figure 1.3 A, E and I Interfaces [GTE94]

The A-Interface is the only wireless portion of the CDPD network. This is where data communication between CDPD mobile end systems (M-ESs) and serving portions of the CDPD network occur. As with cellular phones, these communications are transmitted over a cellular radio channel pair; i.e. a duplex radio channel: a forward channel and a reverse channel. The A-Interface encrypts data sent over these channels for nonbroadcast/multicast type data transfers, in order to prevent casual eavesdropping. The E-Interface provides wired communications between the CDPD networks and external networks such as the Internet, OsInet or any other networks compatible with the CDPD network. The I-Interface allows different CDPD carriers to communicate and exchange information with each other. For example, CDPD subscribers in Chicago can...
communicate with CDPD subscribers in New York regardless of who is providing the local service. The I-Interface supports CDPD network services across all geographical areas where CDPD is available. These services include M-ES authentication, network management and remote activation. The I Interface is not visible outside the CDPD network. In this research project, the A-Interface only is examined.

1.2.3 CDPD Network Entities

Although the cellular infrastructure is extensive with nationwide coverage, it was designed specifically to transmit analog voice signals. In order to provide a national seamless digital data overlay of the cellular network, the CDPD network entities must be added. Figure 1.4 illustrates that the CDPD elements interface with each other.

![Figure 1.4 CDPD Network Entities](image)

Figure 1.4 CDPD Network Entities [GTE94]
There are two basic classes of CDPD network entities [Hil94]: end systems (ESs) and intermediate systems (IS). The ESs are the CDPD Network hosts and the ISs are the CDPD network routes.

ESs can communicate with each other via the CDPD network. An ES is like a telephone, except it is used to send and receive digital data. There are two basic types of ES in the CDPD network: mobile end system (M-ES) and fixed end system (F-ES).

M-ESs are mobile, portable wireless computing devices, for example Personal Digital Assistant, which can change their physical location from cell-to-cell as the user desires and can communicate with the network using a CDPD modem without losing the connectivity. The M-ES is similar to, but more intelligent than, a cellular phone because the decision to initiate a hand-off from one cell to which is under the control of the M-ES itself instead of the Mobile Telephone Switching Office (MTSO). The M-ES monitors the received signal strength indicator (RSSI) of the cellular channels and if the RSSI drops below a predetermined level, then the M-ES will transfer to a new channel or cell. The M-ESs communicate with the mobile data base station (MDBS) of the CDPD network through the A-Interface.

The F-ESs can be many different stationary computing devices, for example UNIX workstations or host computers. The F-ES can communicate with M-ES in the same way as it does with other F-ES, so its configuration and applications do not have to be modified for the CDPD network. Although F-ESs are located in external networks, the E-Interface provides the communications between the CDPD Network and the F-ES.
In order to modify the existing cellular infrastructure by incorporating the CDPD technology, some new internal entities need to be added. These entities include an MDBS and mobile data intermediate system (MD-IS). The MD-IS is a stationary network component and has responsibilities similar to those of the MTSO in the cellular voice system. Like the MTSO, which is responsible for tracking a portable cellular phone’s location and the routing of calls, the MD-IS is responsible for keeping track of the M-ES’s location and routing of data packets between the CDPD network and the M-ES. The MD-IS can be considered as the control center of the CDPD network. The functions of the MD-IS are to guarantee that an M-ES is valid to log on the CDPD network and store information such as the M-ESs last known location, traffic statistics and billing information. The MD-ISs are the only network-relay systems that have any knowledge of mobility and exchange M-ES location information with each other by operating a specific CDPD protocol: Mobile Network Location Protocol (MNLP). This exchange of location information realizes seamless mobility in the CDPD system when an M-ES goes anywhere in the cellular domain.

The MDBS is a stationary network component similar to the Base Station (BS) in the cellular voice system. As the BS is responsible for establishing voice communications between the portable phone and the cellular network, the MDBS is primarily responsible for relaying data between the M-ES and the MD-IS. Since the connection for the M-ES is wireless, the data transmission between the M-ES and the MDBS is done through the A-Interface. The function of the MDBS is radio frequency (RF) channel management. The
MDBS creates an airlink comprising a pair of RF channels for forward and reverse communications. The forward channel is a connectionless broadcast channel extending from the MDBS to the M-ESs. Multiple M-ESs share the reverse channel, which extends from the M-ESs to the MDBS. Therefore, two M-ESs on the same reverse channel cannot communicate with each other. Due to voice priority, the MDBS has responsibility for requiring and instructing the M-ES to hop to the new channel for continued communication. The MDBS also aids the M-ES in making a hand-off from one cell to another by assisting in the location of a new channel. The MDBS keeps track of all adjacent cell’s channels being used or potentially usable for the CDPD network and periodically broadcasts the list to the surrounding M-ESs so that the M-ESs can quickly make decisions when a hand-off is needed.

All of the serving areas of the network are interconnected by a number of the other CDPD entities, the ISs. The IS equipment and the physical interconnections associated with each IS create the CDPD network backbone. Each IS provides network-larger relay functions that relay data between MD-ISs and other ISs throughout the network. An important characteristic of an IS’s routing capabilities is that it can receive data packets on one interface and forward them over another.
1.3 Airlink Specification of the CDPD System

The CDPD system, as a wide area packet data network overlaid to the cellular radio network, has duplex channels for airlink connections. The CDPD system shares the 30 kHz simplex channel at an 800 MHz carrier frequency used by AMPS. Given a data rate of 19.2 kbps, the spectrum efficiency is about 0.64 bps/Hz. In the CDPD system, the forward channel transmits data from the PSTN side of the network to all different mobile users and the reverse channel connects these mobile users to the CDPD network. The CDPD system uses a particular protocol called DSMA/CD to allow different mobile users to access the CDPD network simultaneously. Both the forward and reverse RF channel use the Gaussian minimum-shift-keying (GMSK) modulation scheme characterized by a bandwidth time product $BT = 0.5$. A fixed-length block is used for CDPD transmission and each block code carries 63 symbols with a total length of 378 bits. A (63, 47) Reed-Solomon code is used as the forward error correction (FEC) code and it provides error correction for up to seven symbol errors [Jac94]. The block error rate (BLER) is defined as the main criteria to measure the performance of a CDPD system, much as the bit error rate (BER) is used in assessing voice cellular networks. In the CDPD standard, a BLER of 0.05 is chosen as the maximum acceptable block error rate for both the additive White Gaussian Noise (AWGN) situation and the fading environment. Table 1.1 shows the basic airlink specification for a CDPD system [CDS95] [Rap96][Paw94].
Table 1.1 Airlink Specification of CDPD

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value/Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gross Data Rate</td>
<td>19.2 kbps</td>
</tr>
<tr>
<td>Frequency Band</td>
<td>824-849/869-894 MHz</td>
</tr>
<tr>
<td>Channel Spacing</td>
<td>30 kHz</td>
</tr>
<tr>
<td>Spectrum Efficiency</td>
<td>0.64</td>
</tr>
<tr>
<td>Error Control</td>
<td>(63, 47) Reed-Solomon Code</td>
</tr>
<tr>
<td>Max. Error Correction</td>
<td>7 Symbols with 6 bits/symbol</td>
</tr>
<tr>
<td>Maximum BLER</td>
<td>0.05</td>
</tr>
<tr>
<td>Modulation</td>
<td>GMSK ($BT=0.5$)</td>
</tr>
<tr>
<td>Multiple Access</td>
<td>DSMA/CD</td>
</tr>
</tbody>
</table>

1.4 Research Objective

The objective of this project is to simulate the CDPD airlink which consists of Reed-Solomon coding, GMSK and different types of wireless channels. Special emphasis is placed on examining the Reed-Solomon decoding algorithm and soft decision decoding. Three different decoding cases are investigated in this project: 1) no Reed-Solomon coding; 2) Reed-Solomon error correction coding; 3) Reed-Solomon error and erasure
decoding. Both the additive white Gaussian noise (AWGN) and Rayleigh fading channel models are employed in examining the three channel coding cases.

The project report is divided into six chapters. Chapter 1 has provided an overview of the CDPD system and the basic airlink specification of CDPD. Chapter 2 introduces the CDPD signal format and GMSK modulation scheme. Chapter 3 describes the key feature of the Reed-Solomon error control coding. Chapter 4 discusses the CDPD simulation including the Reed-Solomon error and erasure decoding algorithm, the GMSK differential decoder and wireless channel models. Chapter 5 presents the simulation results. Chapter 6 concludes the research report and suggests directions for future research.
Chapter 2  CDPD Signal Format and Signaling Technique

The only wireless portion of the CDPD system is the communication link between M-ES and MDBS over the A Interface. Data packets can be sent between M-ES and MDBS in either the forward (MDBS to M-ES) or the reverse (M-ES to MDBS) direction. Figure 2.1 shows that the multiple M-ESs share the RF channel to communicate with the local MDBS. This chapter describes the data packet transformation, the forward and reverse data format, and the signaling techniques in the CDPD system.

![CDPD Airlink Setup](image)

**Figure 2.1 CDPD Airlink Setup**

2.1 Data Packet Transformation in CDPD

Before the message data can be sent over the airlink in the CDPD system, it needs to be processed using a data packet transformation. The data packets provided by the network layer are known as Network Protocol Data Units (NPDUs). Each NPU
contains the user data and a header and has a maximum length of 2048 bytes [CDS95]. Figure 2.2 shows a simplified view of the data packet transformation.

![Data Packet Transformation Diagram]

*Figure 2.2 Data Packet Transformation [Hil96]*

First, the header of the packet may be compressed if desired [CDS95]. Then the packet is divided into segments (up to 128 bytes of user data), and a header is added which includes an identifier indicating the type of the header compression applied to the packet, and a protocol number to indicate the type of network protocol used in transmitting the segment. Secondly, each segment is encapsulated by the addition of a data link header to form a frame (frame length up to 130 bytes of user data). The frame header includes a sequence number (SN) to maintain the order of frames in transit and an
identifier for multiplexing separate data link connections. Finally, the resulting frames are concatenated together, with a flag bit delimiting the individual frames, and the resulting bit sequence is divided into a series of fixed-sized blocks (47 symbols of user data, i.e. 282 bits). Each block is subjected to (63, 47) Reed-Solomon forward error correction (FEC) code which allows the errors in transmission to be corrected at the receiver. This block is the basic unit, transmitted as a sequence of bits over the 30 kHz AMPS channel employed in the CDPD system.

On the receiver side, these operations are reversed in order to obtain the original data packet. After the original packet is formed, it can be transmitted over the existing wire-link network (e.g. Internet).

2.2 Forward Channel Block Format

The three main data processes before the signal is modulated are: Reed-Solomon encoding, Pseudo-Random Number (PN) coverage and insertion with a synchronization word and control flags [GTE94]. This is shown in Figure 2.3. Since the Reed-Solomon error control code is one of the major topics of this research, it will be discussed in detail in Chapter 3. Forward channel transmissions consist of a continuous, contiguous series of blocks. Each forward channel block contains fixed-length Reed-Solomon code words.
which are interleaved with the forward synchronization word and control flags. Figure 2.4 shows the forward channel framing and block structure.

![Diagram of Forward Channel Format Processing](image)

**Figure 2.3 Forward Channel Format Processing**

![Diagram of Forward Channel Frame and Block Structure](image)

**Figure 2.4 Forward Channel Frame and Block Structure [CDS95]**

### 2.2.1 Pseudo-Random Number (PN) Coverage

In the CDPD system, Reed-Solomon code is used as FEC for both burst and random error correction of errors introduced in transmission. Each block is encoded using a (63, 47) systematic Reed-Solomon code. It is possible that long strings of binary 0’s and 1’s exist within the information field of the block [Els96]. Therefore some demodulation methods (e.g. Phase-Locked Loop) may not be appropriate. A PN sequence is involved in
the CDPD system to solve this problem. The PN sequence is exclusive-OREd (XOREd) with each 378-bit Reed-Solomon code word block to minimize the likelihood of long strings of binary 0's or 1's in the block during transmission. At the receiver side, each block is XOREd with the same PN sequence before Reed-Solomon decoding to remove the PN sequence from the block. The generator polynomial of the PN sequence is given by [CDS95]

\[ g(D) = D^9 + D^8 + D^5 + D^4 + 1 \]  \hspace{1cm} (2.1)

It can be implemented by a shift register shown in Figure 2.5 initiated with the sequence 1 1 1 0 0 0 1 0 1.

![Figure 2.5 PN Sequence Generator [CDS95]](image)

The 378-bit sequence can be mapped to a sequence of 63 six-bit symbols that comprise the Reed-Solomon block shown in Table 2.1. The symbols are transmitted in order from \( S_{62} \) to \( S_0 \) with the least significant bit of a given symbol transmitted first.
### Table 2.1 PN Sequence Relationship to Block Structure [CDS95]

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Hex</th>
<th>Binary</th>
</tr>
</thead>
<tbody>
<tr>
<td>62</td>
<td>05</td>
<td>000101</td>
</tr>
<tr>
<td>61</td>
<td>17</td>
<td>010111</td>
</tr>
<tr>
<td>60</td>
<td>32</td>
<td>110010</td>
</tr>
<tr>
<td>59</td>
<td>22</td>
<td>100010</td>
</tr>
<tr>
<td>58</td>
<td>1A</td>
<td>011010</td>
</tr>
<tr>
<td>57</td>
<td>25</td>
<td>100101</td>
</tr>
<tr>
<td>56</td>
<td>1F</td>
<td>011111</td>
</tr>
<tr>
<td>55</td>
<td>2C</td>
<td>101100</td>
</tr>
<tr>
<td>54</td>
<td>05</td>
<td>000101</td>
</tr>
<tr>
<td>53</td>
<td>3A</td>
<td>111010</td>
</tr>
<tr>
<td>52</td>
<td>1E</td>
<td>011110</td>
</tr>
<tr>
<td>51</td>
<td>05</td>
<td>000101</td>
</tr>
<tr>
<td>50</td>
<td>18</td>
<td>011000</td>
</tr>
<tr>
<td>49</td>
<td>26</td>
<td>100110</td>
</tr>
<tr>
<td>48</td>
<td>18</td>
<td>011000</td>
</tr>
<tr>
<td>47</td>
<td>1A</td>
<td>011010</td>
</tr>
<tr>
<td>46</td>
<td>08</td>
<td>001000</td>
</tr>
<tr>
<td>45</td>
<td>33</td>
<td>110011</td>
</tr>
<tr>
<td>44</td>
<td>0B</td>
<td>001011</td>
</tr>
<tr>
<td>43</td>
<td>07</td>
<td>000111</td>
</tr>
<tr>
<td>42</td>
<td>39</td>
<td>111001</td>
</tr>
<tr>
<td>41</td>
<td>19</td>
<td>011001</td>
</tr>
<tr>
<td>40</td>
<td>33</td>
<td>110011</td>
</tr>
<tr>
<td>39</td>
<td>15</td>
<td>010101</td>
</tr>
<tr>
<td>38</td>
<td>2F</td>
<td>101111</td>
</tr>
<tr>
<td>37</td>
<td>0D</td>
<td>001101</td>
</tr>
<tr>
<td>36</td>
<td>18</td>
<td>011000</td>
</tr>
<tr>
<td>35</td>
<td>29</td>
<td>101001</td>
</tr>
<tr>
<td>34</td>
<td>0C</td>
<td>001100</td>
</tr>
<tr>
<td>33</td>
<td>20</td>
<td>100000</td>
</tr>
<tr>
<td>32</td>
<td>08</td>
<td>001000</td>
</tr>
<tr>
<td>31</td>
<td>1E</td>
<td>011110</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Hex</th>
<th>Binary</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td>27</td>
<td>100111</td>
</tr>
<tr>
<td>29</td>
<td>20</td>
<td>100000</td>
</tr>
<tr>
<td>28</td>
<td>3B</td>
<td>111011</td>
</tr>
<tr>
<td>27</td>
<td>1A</td>
<td>011010</td>
</tr>
<tr>
<td>26</td>
<td>34</td>
<td>110100</td>
</tr>
<tr>
<td>25</td>
<td>23</td>
<td>100011</td>
</tr>
<tr>
<td>24</td>
<td>22</td>
<td>100010</td>
</tr>
<tr>
<td>23</td>
<td>04</td>
<td>000100</td>
</tr>
<tr>
<td>22</td>
<td>0D</td>
<td>001101</td>
</tr>
<tr>
<td>21</td>
<td>2B</td>
<td>101011</td>
</tr>
<tr>
<td>20</td>
<td>2D</td>
<td>101101</td>
</tr>
<tr>
<td>19</td>
<td>1F</td>
<td>011111</td>
</tr>
<tr>
<td>18</td>
<td>23</td>
<td>100011</td>
</tr>
<tr>
<td>17</td>
<td>11</td>
<td>010001</td>
</tr>
<tr>
<td>16</td>
<td>00</td>
<td>000000</td>
</tr>
<tr>
<td>15</td>
<td>1E</td>
<td>011110</td>
</tr>
<tr>
<td>14</td>
<td>28</td>
<td>101000</td>
</tr>
<tr>
<td>13</td>
<td>34</td>
<td>110100</td>
</tr>
<tr>
<td>12</td>
<td>01</td>
<td>000001</td>
</tr>
<tr>
<td>11</td>
<td>1A</td>
<td>011010</td>
</tr>
<tr>
<td>10</td>
<td>19</td>
<td>011001</td>
</tr>
<tr>
<td>9</td>
<td>0F</td>
<td>001111</td>
</tr>
<tr>
<td>8</td>
<td>05</td>
<td>000101</td>
</tr>
<tr>
<td>7</td>
<td>06</td>
<td>000110</td>
</tr>
<tr>
<td>6</td>
<td>0E</td>
<td>001110</td>
</tr>
<tr>
<td>5</td>
<td>2C</td>
<td>101100</td>
</tr>
<tr>
<td>4</td>
<td>1B</td>
<td>011011</td>
</tr>
<tr>
<td>3</td>
<td>12</td>
<td>010010</td>
</tr>
<tr>
<td>2</td>
<td>2A</td>
<td>101010</td>
</tr>
<tr>
<td>1</td>
<td>04</td>
<td>000100</td>
</tr>
<tr>
<td>0</td>
<td>02</td>
<td>000010</td>
</tr>
</tbody>
</table>

### 2.2.2 Synchronization Word

The forward channel synchronization word in the CDPD system provides timing reference for the reverse channel DSMA/CD microslot clock [CDS95]. In addition, this
synchronization word serves as the reference marker, within the forward channel bit stream, needed to discriminate the reverse channel status flags and FEC Reed-Solomon block boundaries. Each block is interleaved with the 35-bit Forward Synchronization Word (FSW). This 35-bit FSW is divided into seven, 5-bit groups as shown below

11101 00001 11000 00100 11001 01010 01111

Each 5-bit group is XORed with one 5-bit Busy/Idle flag. Therefore, the actual transmitted value of a 5-bit group is affected by the current value of the control flag. The resultant 5-bit group with one extra single bit Decode Status flag in the sixth bit becomes a 6-bit symbol. These seven control flag symbols are inserted ahead of the $\mathcal{S}_{(62-9i)}$, $i = 0 \sim 6$ in each forward channel block.

2.2.3 Control Flag

The control flags transmitted in the forward channel are comprised of two types of reverse channel status: Busy/Idle Status and Block Decode Status. The combination of these two control flags forms the control symbol shown in Figure 2.6

```
6 5 4 3 2 1

| Decode Status | Busy/Idle Status |
```

Figure 2.6 Control Symbol in Forward Channel
As mentioned in Section 2.2.2, the 5-bits of Busy/Idle Status are XORed with each 5-bit group of the forward channel Synchronization Word. The sixth bit of the control symbol is one of the Decode Status flag bits.

Table 2.2 shows the Busy/Idle Status flag in binary value with channel busy indicated by five 0’s and channel idle indicated by five 1’s. Since each Busy/Idle flag is XORed with seven 5-bit groups from the forward synchronization word, the actual transmitted value of a Busy/Idle flag is changed by the flag’s position in the block.

<table>
<thead>
<tr>
<th>Flag State</th>
<th>Binary Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Channel Busy</td>
<td>00000</td>
</tr>
<tr>
<td>Channel Idle</td>
<td>11111</td>
</tr>
</tbody>
</table>

The Decode Status flag in the forward channel shows whether the MDBS is able to decode the preceding block received on the reverse channel in the current burst. Table 2.3 shows the coding of the Decode Status flag.

<table>
<thead>
<tr>
<th>Flag State</th>
<th>Binary Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Decode Success</td>
<td>00000</td>
</tr>
<tr>
<td>Decode Failure</td>
<td>11111</td>
</tr>
</tbody>
</table>
The default state of the Decode Status flag is decode-failure (i.e. when there are no blocks to decode, the Decode Status flag will revert to a state of all ones). Once the Decode Status flag has been changed to ‘0’ from ‘1’ by the MDBS after a decode success, it will set all subsequent bits to ‘0’ until a decode failure or decode default status is to be signaled. Similarly, once the Decode Status has been set to ‘1’ by the MDBS after a decode failure, it will set all subsequent bits to ‘1’ until a successful decoding. Since the MDBS is unable to distinguish random and burst errors on the radio channel transmission from two or more M-ESs, the Decode Status flag is used to signal both types of errors.

2.3 Reverse Channel Format

The reverse channel format in the CDPD system is based on Reed-Solomon block coding with a covering PN-sequence, the Dotting Sequence preamble, the reverse channel Synchronization Word, and the Continuity Indicator. The reverse channel transmission block is a (63,47) Reed-Solomon code word with the PN sequence coverage as described in Section 2.2.1. The reverse channel transmission structure is shown in Figure 2.7.
Figure 2.7 Reverse Channel Transmission Structure [CD895]

2.3.1 Synchronization Flag

There are two types of synchronization flags in the reverse channel of the CDPD system: Dotting Sequence (DS) and Synchronization Word (SW). The Dotting Sequence is used by the MDBS for reverse channel burst detection and bit-timing recovery. This 38-bit Dotting Sequence is always set at the beginning of the reverse channel burst data. The Reverse Channel Synchronization Word (RSW) is used by the MDBS to maintain block synchronization with the M-ES and is set after the Dotting Sequence. The binary
values of the reverse channel Dotting Sequence and Synchronization Word are listed in Table 2.4

<table>
<thead>
<tr>
<th>Bit</th>
<th>1</th>
<th>8</th>
<th>16</th>
<th>24</th>
<th>32</th>
<th>38</th>
</tr>
</thead>
<tbody>
<tr>
<td>DS</td>
<td>1010</td>
<td>1010</td>
<td>1010</td>
<td>1010</td>
<td>1010</td>
<td>1010</td>
</tr>
<tr>
<td>RSW</td>
<td>1011</td>
<td>1011</td>
<td>0101</td>
<td>1001</td>
<td>1100</td>
<td>00 (22 bits)</td>
</tr>
</tbody>
</table>

2.3.2 Continuity Indicator

The Continuity Indicator in the reverse channel format is a 7-bit sequence which signals whether the reverse transmission burst is complete. These seven bits of the Continuity Indicator are interleaved with the Reed-Solomon block, one bit every nine symbols. Such that each bit of the Continuity Indicator is inserted in the front of the Reed-Solomon code symbols $S_{(62−9i)}$, $(i = 0 ~ 6)$ in each reverse channel block. The Continuity Indicator encoding is shown in Table 2.5.

<table>
<thead>
<tr>
<th>Flag State</th>
<th>Binary Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Final Block</td>
<td>00000</td>
</tr>
<tr>
<td>More Block</td>
<td>11111</td>
</tr>
</tbody>
</table>

Table 2.5 Continuity Indicator Status

Chapter 2 CDPD Signal Format and Signaling Technique
2.4 Gaussian Minimum Shift Keying (GMSK)

Because of its superior spectral properties and ease of implementation, Gaussian Minimum Shift Keying (GMSK) has been chosen as the modulation scheme of the airlink interface standard for the CDPD system. GMSK is a form of minimum shift keying (MSK), which is discussed, in general, below.

2.4.1 Minimum Shift Keying

MSK is a special type of continuous-phase frequency shift keying (CPFSK) with a modulation index of $h = 0.5$ which is necessary for the minimum shift keying (FSK) signal to be coherently orthogonal [Pro94]. The name MSK is based on the minimum frequency separation that allows orthogonal detection. There are many variations of MSK; however, all variations of MSK are essentially CFSK employing different techniques to achieve spectral efficiency.

The most common method of mathematically describing MSK is within the context of FSK as in the following equation [Cov93]

$$S(t) = A_e \cos\left[2\pi f_c t + D_f \int m(\lambda) d\lambda + \Phi(t_0)\right]$$

$$= A_e \cos\left[2\pi f_c t + \theta(t) + \Phi(t_0)\right]$$

(2.2)
where \( m(t) \) is a baseband digital signal (\( \pm 1 \)), \( \theta(t) = D_f \int m(\lambda) \, d\lambda \) is the phase of the modulated signal and \( \Phi(t_0) \) is the initial phase.

The digital modulation index is defined as

\[
h = \frac{2\Delta F}{R} \tag{2.3}
\]

where \( \Delta F = \frac{D_f}{2\pi} \) is the peak frequency deviation and \( R \) is the data bit rate of the message signal. For orthogonal signaling and phase continuity at the bit transition period, the peak frequency separation \( \Delta F = \frac{1}{4} R \) or \( h = 0.5 \) is required.

In general, any physical bandpass signal can be represented by the complex envelop

\[
S(t) = \text{Re} \left\{ g(t) \, e^{j\omega_c t} \right\} \tag{2.4}
\]

or the in-phase-quadrature representation

\[
S(t) = X_i(t) \cos(\omega_c t) - X_q(t) \sin(\omega_c t) \tag{2.5}
\]
where $R_e\{\cdot\}$ denotes the real part of $\{\cdot\}$, $g(t)$ is called the complex envelop of $S(t)$, $\omega_c = 2\pi f_c$ and $f_c$ is the associated carrier frequency (Hz), $X_i(t) = R_e\{g(t)\}$ is the in-phase modulation associated with $S(t)$ and $X_q(t) = I_m\{g(t)\}$ is the quadrature modulation associated with $S(t)$. Figure 2.8 shows the in-phase and quadrature parts of a rectangle pause train and Figure 2.9 shows the modulating signal and the MSK modulated signal with $f_c = 15$Hz.

![In-Phase of Message and Quadrature of Message](image)

**Figure 2.8 The in-Phase and Quadrature of a Rectangle Pause Train**

From equation (2.2), it can be seen that MSK has constant amplitude and linear phase during each bit period. In addition, MSK provides relatively narrow bandwidth and coherent detection capability [Mur81]. These properties make MSK particularly attractive.
for use in mobile radio communication systems. However, MSK does not meet specifications regarding out-of-band radiation for a signal-channel-per-carrier (SCPC) mobile radio application. Typically, the out-of-band power radiated to an adjacent channel should be suppressed 60–80 dB below that in the desired channel [Sim84]. To make the output power spectrum compact for MSK, premodulation filtering must be used.

![Graph showing MSK modulated signal with $f_c = 15$Hz.](image)

**Figure 2.9** MSK modulated signal with $f_c = 15$Hz.

### 2.4.2 Gaussian Filtered MSK

One common way to suppress or control the out-of-band spectrum of MSK is to apply a premodulation Gaussian lowpass filter to the no-return-zero (NRZ) data sequence. The resulting modulation scheme is known as GMSK. As shown in Figure 2.6, the
GMSK signal is generated by first passing an NRZ data sequence through a Gaussian premodulation LPF, and then the filter output is fed into a FM modulator with modulation index \( h = 0.5 \). This filtering process provides a narrow bandwidth to suppress the high frequency components and a low overshoot impulse response to protect against excessive instantaneous frequency deviation.

![Diagram of GMSK Implementation]

**Figure 2.10 GMSK Implementation**

The impulse response and the frequency response of the GMSK premodulation LPF are given by [Fag92]

\[
    h_G(t) = B \sqrt{\frac{2\pi}{\ln 2}} e^{-\frac{2(\pi Bt)^2}{\ln 2}} \quad (2.6)
\]

\[
    H_G(t) = e^{-(\alpha t)^2} \quad (2.7)
\]

where the parameter \( \alpha = \frac{\sqrt{\ln 2}}{B \sqrt{2}} \) and \( B \) is the 3-dB bandwidth of the Gaussian LPF.

Equation (2.6) and (2.7) show that the output power spectrum can be controlled by varying the bandwidth of the Gaussian LPF. As the Gaussian LPF is completely
defined by the 3-dB bandwidth \( B \) and the data bit duration \( T \), the bandwidth-time product \( BT \) provides a normalized quantity of filter throughput.

The rectangle pulse response of the Gaussian LPF is determined by [Yao94]

\[
g(t) = h(t) \ast \text{rect}(t/T)
\]

(2.8)

with \( \text{rect}(t/T) = \begin{cases} 1/T & \text{for } |t| < T/2 \\ 0 & \text{otherwise} \end{cases} \).

Equation (2.8) can be also represented by the error function [Bri92]

\[
g(t) = \frac{1}{2T} \left[ Q \left( 2\pi B \frac{t - T/2}{\sqrt{\ln 2}} \right) - Q \left( 2\pi B \frac{t + T/2}{\sqrt{\ln 2}} \right) \right]
\]

(2.9)

where the error function \( Q(t) = \int_{t}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\tau^2/2} d\tau \).

Figure 2.11 depicts the different time-domain rectangle pulse response of the Gaussian LPF. As \( BT \) increases, the higher frequency components of the shaped signal increase. When \( BT = \infty \), there is no shaping effect on the signal. Figure 2.12 shows that the rectangle pulse train with bit pattern 0 0 0 1 1 0 1 0 0 0 is filtered by Gaussian LPF at \( BT = 0.5 \). Figure 2.13 shows the magnitude frequency response for this pulse train. In the
CDPD system, the specifications require that the modulation index \( h = 0.5 \) with the
tolerance of \( h \) is \( \pm 5\% \) and \( BT = 0.5 \) with a data rate \( R = 19.2 \) kbps, so the 3-dB
bandwidth of the Gaussian LPF is 9.6 kHz.

Figure 2.11 Rectangle Pulse Response of Gaussian Filter

Figure 2.12 Filtered Rectangle Pulse Train
Figure 2.13 Frequency Response of the Rectangle Pulse Train

In this chapter we have provided a detailed overview of the CDPD airlink. In the next chapter, we focus on the important aspects of coding and decoding Reed-Solomon codes which is a key contribution of the research in this thesis. In the Chapter 4, we describe how the CDPD airlink is modeled in simulations.
Chapter 3  Reed-Solomon Code in CDPD

In communication systems, the physical channel attenuates the transmitted signal and introduces noise, corrupting the signal. Error control codes (also called channel codes) are used to format the transmitted information so as to increase its immunity to noise. This is accomplished by inserting controlled redundancy into the transmitted information stream (i.e. by transmitting extra check bits), allowing the receiver to detect and possibly correct errors. In this chapter, Reed-Solomon codes and their application in the CDPD system are discussed in detail.

3.1  Reed Solomon Encoding

3.1.1  Coding Theory Background

There are a number of ways to define Reed-Solomon codes. One commonly used definition is based on BCH codes [Lin83]: a Reed-Solomon code is a non-binary $q^n$-ary BCH code with length (in symbols) $q^n-1$, where each symbol is an element of the Galois field GF($q^n$). As the most useful subclass of BCH codes, Reed-Solomon codes achieve the largest possible code minimum-distance and error-correcting capability for any linear code[Wic95]. In addition, Reed-Solomon codes are especially effective in combating long strings of errors (burst errors). This is because the performance of the code word with a
given symbol error is the same whether the symbol error is due to one bit being in error or
$m$ bits (in one symbol) error. Based on these excellent properties, Reed-Solomon codes
serve as FEC in the CDPD system.

For the construction of a $t$-symbol-error correcting Reed-Solomon code, assume
that $\alpha$ is a primitive element in $\text{GF}(q^m)$, the required roots in narrow-sense should be

$$\alpha, \alpha^2, \alpha^3, \cdots \alpha^{2t} \quad (3.1)$$

In other words, the generator polynomial for a $t$-symbol-error correcting Reed-Solomon
code is [Lin83]

$$g(x) = (x + \alpha)(x + \alpha^2) \cdots (x + \alpha^{2t}) = \prod_{i=1}^{2t} (x + \alpha^i) \quad (3.2)$$

Reed-Solomon codes have many interesting properties that are not shared by other
BCH codes [Ree60] [Wic95]. One of the most important properties of Reed-Solomon
codes is that an $(N, K)$ Reed-Solomon code always has minimum distance ($d_{\text{min}}$) exactly
equal to $(N - K + 1)$, i.e.

$$d_{\text{min}} = N - K + 1 = 2t + 1 \quad (3.3)$$

where $N$ is the length of the Reed-Solomon code word in symbols, $K$ is the length of
message data in symbols, and $t$ is the maximum error correcting symbol capacity for
random errors. The maximum number of error corrections in Reed-Solomon codes depends only on the length of the parity symbols.

\[ N - K = 2t \quad \text{or} \quad t = \frac{N - K}{2} \quad (3.4) \]

A special case in Reed-Solomon coding is that for which \( N = q^m = 2^m \). Then each element \( \beta \) of \( GF(2^m) \) can be presented as an \( m \)-tuple of elements from \( GF(2) \), i.e. in polynomial notation:

\[ \beta = a_{m-1} \alpha^{m-1} + a_{m-2} \alpha^{m-2} + \cdots + a_1 \alpha + a_0 \quad (3.5) \]

or in vector notation

\[ \beta = (a_{m-1}, a_{m-2}, \cdots, a_1, a_0) \quad (3.6) \]

where \( a_i \in GF(2), \ 0 \leq i \leq m - 1 \).

CDPD creates a (63,47) Reed-Solomon code over \( GF(64) \) or \( GF(2^6) \) for data error correction [CDS95]. \( GF(64) \) is the Galois field with 63 elements and the additive identity element (0) based on the following primitive polynomial [Wic95]:

\[ P(x) = x^6 + x + 1 \quad (3.7) \]
Each Reed-Solomon block in the CDPD system is characterized by the following parameters:

- length of each symbol in bits: \( m = 6 \);
- length of user data in symbols: \( K = 47 \), or in bits \( k = m \cdot K = 282 \);
- length of R-S block in symbols: \( N = 63 \), or in bits \( n = m \cdot N = 378 \);
- length of parity check in symbols: \( N - K = 16 \);
- maximum number of error correction in symbols: \( t = \frac{N - K}{2} = 8 \).

Therefore, the (63, 47) Reed-Solomon block employed in CDPD is able to correct up to 8 symbol errors in one block (64 symbols). However, the CDPD specification recommends that the code correct up to 7 symbol errors such that the remaining parity symbol can be used to provide better error detection performance by using an automatic repeat request (ARQ) scheme for the retransmission of an unsuccessfully decoded block [CDS95].

### 3.1.2 Reed-Solomon Systematic Encoding

In the CDPD system, the systematic encoding algorithm for Reed-Solomon codes [CDS95] is employed. This technique greatly simplifies the problem of recovering the user data from a code word. In this approach the user data is embedded without modification in the last \( K \) coordinates of the resulting code word:

\[
C = \left[ b_0, b_1, \ldots, b_{N-K-1} : m_0, m_1, \ldots, m_{K-1} \right] \quad (3.8)
\]
After decoding, the last $K$ symbols are removed from the selected code word to yield the received user data.

A $K$-symbol message block $m = (m_0, m_1, \ldots, m_{k-1})$ can be associated with a polynomial:

$$m(x) = m_0 + m_1 x + \cdots + m_{K-1} x^{K-1} \quad (3.9)$$

In systematic form, the encoded vector should be:

$$C(x) = b_0 + b_1 x + \cdots + b_{N-K-1} x^{N-K-1} +$$
$$m_0 x^{-k} + m_1 x^{N-K+1} + \cdots + m_{K-1} x^{N-1} \quad (3.10)$$

$$= b(x) + x^{N-K} m(x)$$

By the Euclidean Division Algorithm [Gra95], $b(x)$ is the remainder of $x^{N-K} m(x)$ divided by the generator polynomial $g(x)$.

$$b(x) = -\text{REM} \left\{ \frac{x^{N-K} m(x)}{g(x)} \right\} \quad (3.11)$$

The systematic encoding procedure can be summarized as follows [Gra95]:

Step 1 - multiply the message polynomial $m(x)$ by $x^{N-K}$,

Step 2 - find the remainder of $\frac{x^{N-K} m(x)}{g(x)}$, 

Chapter 3 Reed-Solomon Code in CDPD

38
Step 3 - concatenate $b(x)$ and $m(x)$ to form $C(x) = b(x) + x^{N-K}m(x)$.

A Shift-Register (SR) is commonly used to implement Reed-Solomon systematic encoding because Reed-Solomon codes are cyclic codes. The SR method can be used for both software and hardware implementations.

![Figure 3.1 Reed-Solomon Systematic Encoder [Wic95]](image)

Figure 3.1 depicts the Reed-Solomon systematic encoder using an SR. Since Reed-Solomon codes are nonbinary BCH codes, the nonbinary addition, multiplication and storage in the field $\text{GF}(2^n)$ must be used [Lin83] [Wic95].

Referring to Figure 3.1, during the first step of the encoding operation the three switches are placed in position “1” and the $K$ message symbols are fed into the encoder in order of decreasing index. These $K$ message symbols are simultaneously sent to the transmitter and perform as the last $K$ coordinates of the systematic code word. After the
$k$th message symbol has been fed into the SR, the switches are moved to position “2”. At this point the $(N-K)$ SR cells contain the remainder generated by the Euclidean Division Algorithm. These $(N-K)$ symbols are then shifted out of the SR to the transmitter, where they comprise the remaining systematic code word coordinates.

The CDPD system uses systematic encoding for $(63,47)$ Reed-Solomon codes. For encoding purposes, the bits in the transmission block can be considered to form a sequence of 63 six-bit Reed-Solomon code symbols $\{ S_i \}$. Figure 3.2 shows the block structure of Reed-Solomon codes in symbols used in the CDPD system.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Bit</th>
<th>$b_6$</th>
<th>$b_5$</th>
<th>$b_4$</th>
<th>$b_3$</th>
<th>$b_2$</th>
<th>$b_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S_{62}$</td>
<td>Message Data Field</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\vdots$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$S_{16}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$S_{15}$</td>
<td>Parity Field</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\vdots$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$S_0$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure 3.2 Block Structure of (63, 47) Reed-Solomon Codes**

The symbols are transmitted in sequential order from $S_{62}$ to $S_0$ and the least significant bit of each code symbol should be transmitted first. The symbols $S_{62}$ to $S_{16}$ contain message data and the symbols $S_{15}$ to $S_0$ are used to perform parity checking.

For the CDPD system, the generator polynomial for a Reed-Solomon code is given by,
\[ g(x) = (x + \alpha)(x + \alpha^2) \cdots (x + \alpha^{16}) = \prod_{i=1}^{16} (x + \alpha^i) \quad (3.12) \]

where \( \alpha \in \text{GF}(64) \). By simplifying,

\[ g(x) = \alpha^6 + \alpha^{23} x + \alpha^{60} x^2 + \alpha^{19} x^3 + \alpha^{23} x^4 + \alpha^{28} x^5 + \alpha^{9} x^6 + \]
\[ + \alpha^{25} x^7 + \alpha^{24} x^8 + \alpha^{23} x^9 + \alpha^{62} x^{10} + \alpha^{59} x^{11} + \alpha^{19} x^{12} + \]
\[ + \alpha^{42} x^{13} + \alpha^{55} x^{14} + \alpha^{20} x^{15} + x^{16} \quad (3.13) \]

This generator polynomial \( g(x) \) used for (63,47) Reed-Solomon systematic encoding in this thesis is realized by a 16-buffer SR implementation. An example of systematic encoding for (63,47) Reed-Solomon code in CDPD is shown in CDPD System Specifications [CDS95].

### 3.2 Reed-Solomon Decoding

Compared with Reed-Solomon systematic encoding, Reed-Solomon decoding is much more complicated. As a nonbinary BCH code, Reed-Solomon decoding must determine both error locations and error magnitudes. There are several different algorithms for Reed-Solomon decoding: the Peterson-Gorenstirn-Zierler decoding algorithm [Pet60] [Gor61], the Berlekamp-Massey algorithm [Mas68] and Euclid’s
algorithm. Since the Berlekamp-Massey algorithm is much more efficient than the other two decoding algorithms, it is the technique chosen for simulating the CDPD system in this project.

3.2.1 Berlekamp-Massey Decoding Algorithm

The Berlekamp-Massey decoding algorithm [Lin83] [Wic95] mainly provides the error location information for Reed-Solomon codes. Assume that all codes under discussion are narrow-sense. For a $t$-symbol-error correction Reed-Solomon code, Equation (3.12) describes the generator polynomial, repeated here for convenience,

$$g(x) = \prod_{i=1}^{2t} (x + \alpha^i)$$  \hspace{1cm} (3.14)

So $g(x)$ has $2t$ consecutive powers of $\alpha$ as zeros.

$$g(\alpha) = g(\alpha^2) = \cdots = g(\alpha^{2t}) = 0$$  \hspace{1cm} (3.15)

A nonbinary vector $C = (C_0, C_1, \cdots, C_{N-1})$ is a Reed-Solomon code word if and only if its associated polynomial $C(x) = C_0 + C_1x + \cdots + C_{N-1}x^{N-1}$ has these same $2t$ consecutive powers of $\alpha$ as zeros in $GF(q^n)$. So a received polynomial $r(t)$ can be
expressed as the sum of a transmitted code polynomial \( C(x) \) and an error polynomial

\[
e(x) = e_0 + e_1 x + \cdots + e_{N-1} x^{N-1},
\]

\[
r(x) = C(x) + e(x)
\]

(3.16)

Then a series of syndromes can be obtained by evaluating the received polynomial at the \( 2t \) zeros:

\[
S_j = r(\alpha^j) = C(\alpha^j) + e(\alpha^j)
\]

\[
e(\alpha^j) = \sum_{k=0}^{N-1} e_k (\alpha^j)^k, \quad 1 \leq j \leq 2t
\]

(3.17)

where \( S_j \) is one of \( 2t \) syndromes. A correct code word is received if and only if all syndromes \( S_j \) \((1 \leq j \leq 2t)\) are zero.

Assume that the received word \( r \) has \( v \) errors in position \( i_1, i_2, \cdots, i_v \), then the error polynomial becomes

\[
e(x) = e_{i_1} x^{i_1} + e_{i_2} x^{i_2} + \cdots + e_{i_v} x^{i_v}
\]

(3.18)

and the syndrome components are

\[
S_j = e(\alpha^j) = \sum_{l=1}^{v} e_{i_l} (\alpha^j)^l, \quad 1 \leq j \leq 2t
\]

(3.19)
Since the values of $\alpha^i$ ($1 \leq l \leq v$) indicate the positions of errors in the received word, define $X_i = \alpha^i$ ($1 \leq l \leq v$) as the error locations. Now, the syndromes can be expressed in terms of these error locations [Gra95]:

$$S_j = \sum_{i=1}^{v} e_i (X_i)^j \quad \text{for} \quad 1 \leq j \leq 2t \quad (3.20)$$

Equation (3.20) can be expanded into a sequence of $2t$ algebraic syndrome equations in the $v$ unknown error locations

$$S_1 = e_{i_1} X_1 + e_{i_2} X_2 + \cdots + e_{i_v} X_v$$
$$S_2 = e_{i_1} X_1^2 + e_{i_2} X_2^2 + \cdots + e_{i_v} X_v^2$$
$$\vdots$$
$$S_{2t} = e_{i_1} X_1^{2t} + e_{i_2} X_2^{2t} + \cdots + e_{i_v} X_v^{2t} \quad (3.21)$$

Since Equation (3.21) forms a system of nonlinear algebraic equations in multiple variables, they are difficult to solve in a direct manner. However, the BCH syndrome equation can be translated into a series of linear equations that are much easier to solve.

Let $\Lambda(x)$ be the error locator polynomial consisting of all error locations $\{X_i\}$.

$$\Lambda(x) = \prod_{i=1}^{v} (1 - X_i x) = \Lambda_0 + \Lambda_1 x + \cdots + \Lambda_v x^v \quad (3.22)$$

For all $X_i$, $(X_i)^{-1}$ is a root of the error locator polynomial,
\[
\Lambda(X^{-1}) = \Lambda_0 + \Lambda_1 X^{-1} + \cdots + \Lambda_v X^{-v} = 0 \quad (3.23)
\]

Equation (3.23) is multiplied by \( e_i X^{-i} \), then

\[
e_i X^{-i} \Lambda(X^{-1}) = e_i (\Lambda_0 X^{-i} + \Lambda_1 X^{-i-j^{-1}} + \cdots + \Lambda_v X^{-i-j^{-v}}) = 0
\]

\[ (1 \leq i \leq v) \quad (3.24)\]

Summing Equation (3.24) from \( l = 1 \) to \( l = v \):

\[
\sum_{l=1}^{v} e_i (\Lambda_0 X^{-i} + \Lambda_1 X^{-i-j^{-1}} + \cdots + \Lambda_v X^{-i-j^{-v}}) = 0
\]

\[ (3.25)\]

Compared with Equation (3.21), Equation (3.25) can be represented in terms of syndromes and the coefficients of the error locations.

\[
\Lambda_0 S_j + \Lambda_1 S_{j-1} + \cdots + \Lambda_v S_{j-v} = 0
\]

\[ (3.26)\]

Equation (3.26) indicates that each syndrome \( S_j \) can be expressed in recursive form as a function of the coefficients of the error locator polynomial \( \Lambda(x) \) and the previous syndromes \( S_{j-1}, S_{j-2}, \cdots, S_{j-v} \).

\[
S_j = - (\Lambda_1 S_{j-1} + \Lambda_2 S_{j-2} + \cdots + \Lambda_v S_{j-v})
\]

\[ (3.27)\]
The Berlekamp-Massey algorithm for Reed-Solomon codes is based on a linear feedback shift register (LFSR) interpretation of Equation (3.27), as shown in Figure (3.2). The first $2t$ elements in the LFSR output sequence are the syndromes $S_1, S_2, \ldots, S_{2t}$ and the tap of this shift register provides the desired error locator polynomial $\Lambda(x)$.

![Figure 3.2 LFSR Implementation [Wic95]](image)

The algorithm defines $\Lambda^{(k)}(x) = \Lambda_k x^k + \Lambda_{k-1} x^{k-1} + \cdots + \Lambda_1 x + 1$ as the connection polynomial of length $k$ whose coefficients specify the taps of a length-$k$ LFSR. The algorithm begins with finding $\Lambda^{(1)}$ such that the first element in the output of the LFSR is the first syndrome $S_1$. The second output of this LFSR is compared to the second syndrome. If these outputs are not equal, then the discrepancy between the two is used to construct a modified connection polynomial. If there is no discrepancy, then the same connection polynomial is used to generate the next sequence element, which is compared
to the third syndrome. This process continues until a connection polynomial is obtained that specifies an LFSR capable of generating all \(2t\) elements of the syndrome sequence. It can be proved that for an error pattern of weight \(v \leq t\), the error locator polynomial is uniquely specified by the connection polynomial resulting from the Berlekamp-Massey algorithm [Mas68].

The Berlekamp-Massey algorithm has five basic parameters: the length of the shift register \(L\); the indexing variable \(k\); the connection polynomial \(\Lambda^{(k)}(x)\); the correction polynomial \(T(x)\); and the discrepancy \(\Lambda^{(k)}\). The error locations \(\{X_i\}\) can be determined by finding the roots of \(\Lambda(x) = \Lambda^{(2t)}(x)\). The algorithm simulation structure is shown in Figure 3.3.

### 3.2.2 Error Correction of Reed-Solomon

The Berlekamp-Massey algorithm provides the error locator polynomial for a received word, but there remains the problem of finding the error magnitudes. For this purpose, an equation (referred to as the *key equation*) that relates the known syndrome values to the error locator and error magnitude polynomials must be introduced.

\[
\Lambda(x)[1 + S(x)] = \Omega(x) \mod x^{2t+1}
\]  
(3.28)
Calculate $S_j$

Initialize: $k = 0$, $\Lambda^{(0)}(x) = 1$,
$L = 0$, $T(x) = x$

Set: $k = k + 1$,
Compute: $\Lambda^{(k)}(x) = S_k - \sum_{i=1}^{L} \Lambda_i^{(k-1)} S_{k-i}$

$\Lambda^{(k)} = 0$?

$\Lambda^{(k)}(x) = \Lambda^{(k-1)}(x) - \Lambda^{(k)}T(x)$

$2L \geq k$

Set: $L = k - L$
Computer: $T(x) = \Lambda^{(k-1)}(x) / \Lambda^{(k)}$

$T(x) = x \cdot T(x)$

$Y$

$k \leq 2t$?

$N$

Determine the roots of $\Lambda(x) = \Lambda^{(2t)}(x)$

Decoding failure
Determine the error magnitude

Stop
Continue

Figure 3.3 Program Structure of Berlekamp-Massey Decoding Algorithm
where \( S(x) = \sum_{j=1}^{2^r} S_j x^j \) is the syndrome polynomial, and \( \Omega(x) \mod x^{2r+1} \) is the key equation with terms up to \( x^{2r+1} \) because \( S(x) \) has non-zero coefficients only up to \( S_{2r} \).

In extracting the error magnitudes from Equation (3.28), a formal derivative that behaves like a derivative but does not have the corresponding interpretation must be used since the usual definition of a derivative cannot be applied to a finite field. The definition and properties of the formal derivative are presented in [Wic95].

The error magnitudes can be computed in terms of \( \Omega(x) \) and \( \Lambda'(x) \) as follows.

\[
e_k = \frac{-X_k \Omega(X_k^{-1})}{\Lambda'(X_k^{-1})} \tag{3.29}
\]

The computer simulation structure is shown in Figure 3.4.

The error polynomial consists of the error locations and error magnitudes

\[
e(x) = \sum_{k=1}^v e_k x^k \tag{3.20}
\]

where for the CDPD systems, \( 0 \leq i_k \leq 62 \).

The correct code word can be obtained by the sum of the received polynomial and the error polynomial.

\[
C(x) = r(x) + e(x) \tag{3.31}
\]
Syndrome Function:
\[ S(x) = \sum_{j=1}^{2n} S_j x_j \]

Error Locator Polynomial:
\[ \Lambda(x) = \prod_{i=1}^{n} (1 + X_i) \]

Key Equation:
\[ \Omega(x) \mod x^{2t+1} = \Lambda(x)[1 + S(x)] \]

Calculate:
\[-X_k \Omega(X_k^{-1}) \text{ and } \Lambda'(X_k^{-1})\]

Error Magnitude:
\[ e_i = \frac{-X_k \Omega(X_k^{-1})}{\Lambda'(X_k^{-1})} \]

Error Correction:
\[ C(x) = r(x) + e(x) \]

Stop

Figure 3.4 Program Structure of Error Magnitude Calculation
3.3 Error and Erasure Decoding

In the CDPD system, message data are formatted into a sequence of symbols by the Reed-Solomon encoder. Each of these symbols is represented by an element of GF($2^6$). These encoded symbols are modulated by GMSK and transmitted to receivers bit-by-bit. At the receiver, a detection circuit examines the received signal and decides which of the possible transmitted bits are most likely to have been sent before the received signal can be decoded. This section discusses two different decision receivers and the erasure decoding algorithm that could possibly be incorporated in the CDPD system.

3.31 Soft Decision and Erasure Decoding

There are two types of decision receivers: hard-decision receivers and soft-decision receivers. In a hard-decision receiver, the detection circuit has the same group of possible decision variables as the transmitter. Figure 3.5 illustrates the mechanism of the hard-decision receiver with binary decisions. In Figure 3.5 “0” is chosen as the threshold, “+1” and “-1” represent 1 and 0 in binary bits, respectively. For any received signal in the range of $(0,+1)$, the detection circuit makes a decision of binary 1, and for any received signal in the range of $(-1,0)$, the detection circuit makes a decision of binary 0. The CDPD standard (Release 1.1) adopts this hard-decision method as the receiver decision.
algorithm. The hard-decision algorithm simplifies the complexity of the receiver and is easy to implement.

![Decision range diagram]

**Figure 3.5 Hard-Decision Mechanism**

However, the noise and fading effects of the physical channels may corrupt the transmitted signal and the received signal may not make a clear choice as to which of the possible symbols has been transmitted. So hard-decision receivers destroy the message data that could improve the overall performance of the communication system.

Unlike the hard-decision receivers that force a decision which is likely to be incorrect, the soft-decision receiver employs an expanded selection of decision variables to minimize the probability of the error created by the detection circuit. The simplest form of soft-decision receiver is an erasure-decoding receiver that uses two thresholds as shown in Figure 3.6. For any received signal in the range between two threshold (\( Th1 \leq r \leq Th2 \)), the detection circuit declares an erasure to indicate the reception of a signal whose corresponding symbol value is in doubt. For any received signal in the other two ranges such as \(-1 \leq r \leq Th2\) and \( Th1 \leq r \leq 1\), the detection circuit makes a decision, in binary notation "0" and "1" respectively.
Figure 3.6 Erasure-Soft Decision Mechanism

Figure 3.7 shows the probability distribution of the two binary decision range and one erasure range. The overlapping region indicates that there exists some probability of error made by hard-decision.

Figure 3.7 Probability Distribution of Binary Decision

Now examine a $q$-ary Reed-Solomon code word as shown in Figure 3.8. The transmitter has $q$ symbol choices, but the receiver has $(q+1)$ detection choices. The $\{p_i\}$ are the corresponding probabilities of receiving the correct symbol, one of the error
symbols, or an erasure. All \( \{p_i\} \) values are related to the modulation schemes, transmitter power and the physical channels.

![Diagram of uniform discrete symmetric channel with erasures](image)

**Figure 3.8 Uniform Discrete Symmetric Channel with Erasures [Wic95]**

The erasure decoding receiver may improve the performance, in terms of the block error rate (BLER), for the CDPD system. Usually the error correction capabilities of block codes are measured by the minimum distance. If there is a received code word with a single erasure coordinate, then all pairs distinct code words over the unerased coordinates are separated by a Hamming distance of at least \( (d_{\text{min}} - 1) \). So, in general, for given \( f \) symbol erased coordinates, the code word will have an effective minimum distance of \( (d_{\text{min}} - f) \) over the unerased coordinates. This will give the number of symbol error corrections of a received code word.

\[
t = \frac{d_{\text{min}} - f - 1}{2}
\]  
(3.32)
In other words, the erasure decoding receiver can correct \( t \) errors and \( f \) erasures as long as \( t \) and \( f \) satisfy the following equation:

\[
2t + f + 1 \leq d_{\text{min}} \quad (3.33)
\]

Equation (3.33) reveals that the soft-decision receiver with erasure decoder can correct twice as many erasures as errors.

As mentioned in Section 3.1.1, an \((N,K)\) Reed-Solomon code always has a minimum distance exactly equal to \((N-K+1)\). Consider the error and erasure corrections as well as the error detection for (ARQ) in the CDPD system [CDS95], the minimum distance must have

\[
d_{\text{min}} = N - K + 1 \geq 2t + d + f + 1 \quad (3.34)
\]

For (63,47) Reed-Solomon codes, \( d_{\text{min}} = 17 \) and \( d=1 \), then Equation (3.34) can be rewritten,

\[
2t + f \leq 15 \quad (3.35)
\]

Table 3.1 gives the relationship of different error and erasure corrections in symbols. The maximum 23.8% of erasure corrections is provided by CDPD.
Table 3.1 The Number of Error/Erasure Corrections in CDPD

<table>
<thead>
<tr>
<th>$d_{\text{min}}$</th>
<th>$d$</th>
<th>$t$</th>
<th>$f$</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>17</td>
<td>1</td>
<td>0</td>
<td>15</td>
<td>23.8%</td>
</tr>
<tr>
<td>17</td>
<td>1</td>
<td>1</td>
<td>13</td>
<td>20.63%</td>
</tr>
<tr>
<td>17</td>
<td>1</td>
<td>2</td>
<td>11</td>
<td>17.46%</td>
</tr>
<tr>
<td>17</td>
<td>1</td>
<td>3</td>
<td>9</td>
<td>14.29%</td>
</tr>
<tr>
<td>17</td>
<td>1</td>
<td>4</td>
<td>7</td>
<td>11.11%</td>
</tr>
<tr>
<td>17</td>
<td>1</td>
<td>5</td>
<td>5</td>
<td>7.94%</td>
</tr>
<tr>
<td>17</td>
<td>1</td>
<td>6</td>
<td>3</td>
<td>4.76%</td>
</tr>
<tr>
<td>17</td>
<td>1</td>
<td>7</td>
<td>1</td>
<td>1.58%</td>
</tr>
</tbody>
</table>

3.3.2 Erasure Decoding Algorithm [Wic95]

The erasure decoding algorithm discussed in this section is based on the Berlekamp-Massey algorithm presented in section 3.2.1. Suppose a received code word has $v$ errors and $f$ erasures. The error locators $X_1 = \alpha^{i_1}, X_2 = \alpha^{i_2}, \ldots, X_n = \alpha^{i_n}$ indicate that these $v$ errors occurred in symbol coordinates $i_1, i_2, \ldots, i_v$. The erasure locators $Y_1 = \alpha^{j_1}, Y_2 = \alpha^{j_2}, \ldots, Y_n = \alpha^{j_n}$ indicate that the $f$ erasures occurred in symbol coordinates $j_1, j_2, \ldots, j_f$. It should be mentioned that the $f$ erasure locators have been determined by the detection circuit in the soft-decision receiver. The erasure decoding algorithm consists of three tasks: (1) determine the error locators; (2) calculate magnitude of errors associated with the error locators $\{e_i\}$; (3) find the magnitude of erasures associated with the erasure locators $\{f_j\}$.
Since the syndrome of a received code word is the only function which provides both error and erasure information, the erasure decoding algorithm starts with the syndrome calculation. For this purpose, each coordinate of the received code word, where an erasure has been indicated, is assigned the value zero, then 2t syndromes and the syndrome polynomial can be computed by

\[ S_l = r(\alpha^l) = \sum_{k=1}^{n} e_k X_k^l + \sum_{k=1}^{f} f_k Y_k^l \quad l = 1, 2, \ldots, 2t \quad (3.35) \]

\[ S(x) = \sum_{l=1}^{2t} S_l x^l \quad (3.36) \]

The erasure locator polynomial is given by the erasure locators from the detection circuit:

\[ \Gamma(x) = \prod_{l=1}^{f} (1 - Y_l x) \quad (3.37) \]

The combination of Equations (3.36) and (3.37) provides a modified syndrome expressed by,

\[ \Psi = \left[ \Gamma(x)[1 + S(x)] - 1 \right] \mod x^{2r+1} \quad (3.38) \]
The coefficients of this new modified syndrome $\Psi_1, \Psi_2, \ldots, \Psi_r$ are used to find the connection polynomial $\Lambda(x)$.

The procedure for calculating $\Lambda(x)$ is the same as the procedure shown in Figure 3.3. This error locator polynomial provides the coordinates of all errors in the received code word.

A single error-erasure locator polynomial can be obtained by

$$\Phi(x) = \Lambda(x) \Gamma(x) \quad (3.39)$$

Finally the magnitude of errors and erasures are calculated by

$$e_i = \frac{-X_k \Omega(X_k^{-1})}{\Phi'(X_k^{-1})} \quad \text{and} \quad f_i = \frac{-Y_k \Omega(Y_k^{-1})}{\Phi'(Y_k^{-1})} \quad (3.40)$$

where $\Omega(x) = \Lambda(x)(1 + \Psi) \mod x^{2t+1}$ is the error-erasure key function.

The error-erasure polynomial $E(x) = e(x) + f(x)$ is subtracted from the received polynomial to obtain the desire code polynomial

$$C(x) = r(x) - E(x) \quad (3.41)$$

The simulation structure of the erasure decoding algorithm is presented in Figure 3.9.

In this chapter we have described the operation of the Reed-Solomon decoder in the CDPD system. Since an entire packet must be received correctly to be of use to the
system, the Reed-Solomon coding is key to maintaining an acceptable BLER of less than five percent. Although, the standard CDPD receivers implements hard decision decoding, it is well known that errors and erasures decoding will provide improved performance. The remainder of this thesis is devoted to determining how much improvement will be possible in the wireless environments.
Assign 0 to erasure coordinates: \( f_i \leftarrow 0 \)

Calculate Syndromes
\[ S_l = r(\alpha^l) \quad 1 \leq l \leq 2t \]

Erasure Locator Polynomial
\[ \Gamma(x) = \prod_{i=1}^{f} (1 - Y_i x) \]

Modified Syndrome
\[ \Psi(x) = \Gamma(x)[1 + S(x)] - 1 \]

Find Error Locator Polynomial
\[ \Lambda(x) \]

Key Equation
\[ \Lambda(x)[1 + \Psi(x)] \equiv \Omega(x) \mod x^{2r+1} \]

Magnitudes of Errors and Erasures
\[ e_i = -X_k \Omega(X_k^{-1}) \quad \Phi(X_k^{-1}) \]
\[ f_i = -Y_k \Omega(Y_k^{-1}) \quad \Phi(Y_k^{-1}) \]

Corrected Code
\[ C(x) = r(x) + e(x) + f(x) \]

Figure 3.9 Erasure Decoding Algorithm
Chapter 4 System Modeling and Simulation

The CDPD airlink simulation in this research project was performed in MATLAB. The simulation structure for the CDPD airlink consists of four main blocks: transmitter, receiver, channel models and performance analysis. The MATLAB code developed for this simulation is organized into those main blocks. This chapter will discuss the system modeling and the simulation implementation in detail.

![Simulation Structure of CDPD Airlink](image)

Figure 4.1 Simulation Structure of CDPD Airlink
4.1 Transmitter Block

There are two sub-blocks in the transmitter block: data format/Reed-Solomon encoder and GMSK modulator. Figure 4.2 shows the CDPD data format and Reed-Solomon encoder sub-block.

![Diagram of Transmitter Block]

**Figure 4.2 Format/Reed-Solomon Sub-block**

Random binary data is generated by the MATLAB command `random(•)` with uniform distribution. These data are fed into the Reed-Solomon encoder in forms of block with 47 symbols, each symbol having 6 bits. A (63,47) Reed-Solomon encoder is based on the generator polynomial of \( g(x) \) shown in Equation (3.13) and implemented by Reed-Solomon systematic encoding algorithm. The shift-register implementation shown in Figure 3.1 provides a basic MATLAB code structure, but the operations of the 6-bit symbols such as addition, multiplication, division and shift-register are all in GF(64). This
systematic encoder creates a 63-symbol Reed-Solomon code word block with the first 16 symbols of parity check and the last 47 symbols of message data.

The PN sequence encoder is a 6-buffer shift-register which is specified and initialized by the CDPD specifications for minimizing the likelihood of long strings of binary 0's and 1's in a Reed-Solomon block [CDS95]. Each block of the Reed-Solomon encoder is XORed with the same pseudo-random number sequence provided by the PN sequence encoder.

Since both the Reed-Solomon encoder and the PN sequence encoder manipulate the data symbol by symbol, the symbol sequence must be converted into the bit sequence before the Reed-Solomon block can be interleaved with system control flags. The control flag insertions for the forward channel and reverse channel are slightly different. The output of the flag block on the forward channel gives a 420 bit sequence and the one on the reverse channel outputs a 385-bit sequence [CDS95]. For simulation, each bit of the Reed-Solomon code block is represented by the rate of 5 samples per bit before the bit stream is modulated and transmitted by the GMSK modulator.

![Figure 4.3 GMSK Modulator Sub-block](image)

Chapter 4 System Modeling and Simulation
The simulation of the GMSK modulation scheme in MATLAB is shown in Figure 4.3. The Reed-Solomon bit stream is filtered by a Gaussian low pass filter (GLPF), which is performed using the MATLAB command \texttt{filter()} with GLPF impulse response of \( h(t) \). The bandwidth-time product \( BT=0.5 \) is set in \( h(t) \) for CDPD specifications. This GLPF shaped signal is modulated by MSK modulator in baseband. The MATLAB command \texttt{cumsum()} performs integration for continuous phase calculation and the outputs of MSK modulator are complex envelope.

4.2 Receiver Block

The receiver block of the CDPD airlink in Figure 4.1 includes two sub-blocks: GMSK demodulation and Reed-Solomon decoder. The formatting is completed in the transmitter block such as PN sequence coverage, control flags insertion and sequence conversion are removed in the corresponding stages as shown in figure 4.4.

![Diagram of Receiver Block of CDPD Airlink](image)

**Figure 4.4 Receiver Block of CDPD Airlink**
4.2.1 GMSK Demodulation

For the detection of GMSK, there are many different detection methods such as a coherent detector [Mur81], a frequency discriminator [Cou93] and a differential detection [Yon88] [Sim84] [Smi94] which can be employed. In this thesis, a one-bit differential detection is used for the detection of GMSK. As the block diagram of a GMSK transmitter is shown in Figure 2.10, the input to the GLPF is a NRZ sequence and the output of the GPLF can be represented by [Pro94]:

\[
m'(t) = \sum_{i=-\infty}^{\infty} b_i g(t - iT) \tag{4.1}
\]

where \(b_i\) is the binary message signal, \(g(t)\) is the response of the GLPF to a unit amplitude rectangular pulse with duration \(T\) (shown in Equation (2.8) and (2.9)).

Substituting Equation (4.1) into Equation (2.2), the output of GMSK transmitter can be given by:

\[
S(t) = A_0 \cos(\omega_c t + \theta(t) + \phi(t_0)) \tag{4.2}
\]

and the phase of the modulated signal is

\[
\theta(t) = D_f \cdot \int_{-\infty}^{t} m'(\lambda) d\lambda
\]

\[
= D_f \cdot \sum_{i=-\infty}^{\infty} b_i \int_{-\infty}^{t} g(\tau - iT) d\tau
\tag{4.3}
\]
The phase difference in the one-bit duration $T$ is:

$$\Delta \theta_i = \theta(iT) - \theta(iT - T) = D_f \cdot \sum_{i=\infty}^{\infty} b_i \int_{\tau-iT}^{T} g(\tau - iT) d\tau$$  \hspace{1cm} (4.4)$$

In the case of MSK ($h=0.5$) and $b_i=\pm 1$, $(\Delta \phi_i)_{\text{max}} = \frac{\pi}{2}$, and $D_f = \frac{\pi}{2T}$. The decision variable $\Delta \theta$ is the key of one-bit differential detection of GMSK.

![Diagram](image)

**Figure 4.5 One-bit differential detector [Yon88]**

The block diagram of the conventional one-bit differential detector is presented in Figure 4.5. The received signal which may be corrupted by the physical channel is filtered by per detection bandpass filter (BPF) and the output of the BPF can be represented by:

$$y(t) = r(t)\cos[(\omega_c t + \phi(t))] + n_i(t)\cos\omega_c t - n_q(t)\sin\omega_c t$$  \hspace{1cm} (4.5)$$
where $r(t)$ is the time-varying envelope of the signal, $\omega_c$ is the carrying frequency, $\phi(t)$ is the distorted signal phase, $n_i(t)$ and $n_q(t)$ are the in-phase and quadrature components of the narrowband noise, respectively.

The differential detection is implemented by lowpass filtering the product of $y(t)$ and the signal $y(t)$ delayed by one bit duration $T$ and phase shifted by $90^\circ$. If assume that $\omega_c T = 2k\pi$, and $k$ is an integer, the output of LPF $d_1(t)$ can be given by:

$$d_1(t) = \gamma(t)\gamma(t-T) \cdot \sin(D_f \sum_{i=f}^{T} \int g(\tau - iT) d\tau) + n_1(t)$$  \hspace{1cm} (4.6)

where $n_1(t)$ represents all the noise terms.

At any instant time $kT$, $d_1(t)$

$$d_1(kT) = R(kT) \cdot \sin(\sum_{i=-\infty}^{\infty} b_i \theta_{k-i}) + n_1(kT)$$

$$= R(kT) \cdot \sin(\Delta \theta_k) + n_1(kT)$$  \hspace{1cm} (4.7)

where $R(kT) = \gamma(kT) \cdot \gamma(kT-T)$,

$$\theta_{k-i} = D_f \cdot \int_{kT-T}^{kT} g(\tau - iT) d\tau$$

$$\Delta \theta_k = \sum_{i=-\infty}^{\infty} b_i \theta_{k-i}$$

Table 4.1 shows the values of $\theta_i$ for different $BT$. $\theta_0$ represents the signal and $\theta_{-3}, \theta_{-2}, \theta_{-1}, \theta_1, \theta_2, \theta_3$ represent the Internal Symbol Interference (ISI) terms. When
$BT = \infty$, $\theta_0 = 90^\circ$, only the desired signal appears, and there are no ISI terms. When $BT$ decreases, the ISI terms increase greatly and the signal may not be recognized.

<table>
<thead>
<tr>
<th>$BT$</th>
<th>$\theta_{-3}$</th>
<th>$\theta_{-2}$</th>
<th>$\theta_{-1}$</th>
<th>$\theta_0$</th>
<th>$\theta_{+1}$</th>
<th>$\theta_{+2}$</th>
<th>$\theta_{+3}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.15</td>
<td>0.3</td>
<td>4.55</td>
<td>21.85</td>
<td>36.6</td>
<td>21.85</td>
<td>4.55</td>
<td>0.3</td>
</tr>
<tr>
<td>0.2</td>
<td>-</td>
<td>1.7</td>
<td>20.6</td>
<td>45.4</td>
<td>20.6</td>
<td>1.7</td>
<td>-</td>
</tr>
<tr>
<td>0.3</td>
<td>-</td>
<td>0.2</td>
<td>15.9</td>
<td>57.8</td>
<td>15.9</td>
<td>0.2</td>
<td>-</td>
</tr>
<tr>
<td>0.5</td>
<td>-</td>
<td>-</td>
<td>10.3</td>
<td>69.4</td>
<td>10.3</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>1.0</td>
<td>-</td>
<td>-</td>
<td>5.9</td>
<td>78.2</td>
<td>5.9</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>$\infty$</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>90.0</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

In Table 4.1, it is assumed that $h = 0.5$, for any $BT$, $\sum \theta_i = 90^\circ$. Since $|\theta_i| < 1$ as $|i| > 3$, $\max(|i|) = 3$ in Table 4.1.

The phase difference between the received signal and its delayed/shifted version provides key information for the GMSK detection. Table 4.2 and Table 4.3 show the differential phase angles $\Delta \theta_k$ corresponding to all possible input data combinations at $BT = 0.3$ and $BT = 0.5$ respectively. The $\Delta \theta_k$ from Table 4.2 and Table 4.3 can be plotted on the phase-state diagrams with y-axis of $\sin(\Delta \theta_k)$ and x-axis of $\cos(\Delta \theta_k)$. 

Chapter 4 System Modeling and Simulation
### Table 4.2 Message Data and Phase Difference ($BT=0.3$)

<table>
<thead>
<tr>
<th>State</th>
<th>Bit Combination</th>
<th>$\Delta \theta_k$ (in degree)</th>
<th>$\Delta \theta_k / 90^0$ ( % )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1 1 1</td>
<td>89.6</td>
<td>99.6</td>
</tr>
<tr>
<td>2</td>
<td>1 1 -1</td>
<td>57.8</td>
<td>64.5</td>
</tr>
<tr>
<td>2</td>
<td>-1 1 1</td>
<td>57.8</td>
<td>64.5</td>
</tr>
<tr>
<td>3</td>
<td>-1 1 -1</td>
<td>26</td>
<td>29.0</td>
</tr>
<tr>
<td>4</td>
<td>1 -1 1</td>
<td>-26</td>
<td>29.0</td>
</tr>
<tr>
<td>5</td>
<td>1 -1 -1</td>
<td>-57.8</td>
<td>64.5</td>
</tr>
<tr>
<td>5</td>
<td>-1 -1 1</td>
<td>-57.8</td>
<td>64.5</td>
</tr>
<tr>
<td>6</td>
<td>-1 -1 -1</td>
<td>-89.6</td>
<td>99.6</td>
</tr>
</tbody>
</table>

### Table 4.3 Message Data and Phase Difference ($BT=0.5$)

<table>
<thead>
<tr>
<th>State</th>
<th>Bit Combination</th>
<th>$\Delta \theta_k$ (in degree)</th>
<th>$\Delta \theta_k / 90^0$ ( % )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1 1 1</td>
<td>90.0</td>
<td>100</td>
</tr>
<tr>
<td>2</td>
<td>1 1 -1</td>
<td>69.4</td>
<td>77.1</td>
</tr>
<tr>
<td>2</td>
<td>-1 1 1</td>
<td>69.4</td>
<td>77.1</td>
</tr>
<tr>
<td>3</td>
<td>-1 1 -1</td>
<td>48.8</td>
<td>54.2</td>
</tr>
<tr>
<td>4</td>
<td>1 -1 1</td>
<td>-48.8</td>
<td>54.2</td>
</tr>
<tr>
<td>5</td>
<td>1 -1 -1</td>
<td>-69.4</td>
<td>77.1</td>
</tr>
<tr>
<td>5</td>
<td>-1 -1 1</td>
<td>-69.4</td>
<td>77.1</td>
</tr>
<tr>
<td>6</td>
<td>-1 -1 -1</td>
<td>-90.0</td>
<td>100</td>
</tr>
</tbody>
</table>
Figure 4.6 Phase-State Diagram $BT=0.3$

Figure 4.7 Phase-State Diagram $BT=0.5$
The phase-state diagrams of \( BT=0.3 \) and \( BT=0.5 \) present the relationship between the message bit \( (b_k) \) and the phase difference \( (\Delta \theta_k) \). It can be seen that the phase-states are symmetric with respect to the x-axis, and the location of the phase difference in the phase-state diagram implies the corresponding state of the message bit, i.e. when \( \Delta \theta \geq 0, b_k = \pm 1 \); or \( \Delta \theta < 0, b_k = -1 \). Therefore, if the x-axis is chosen as the decision threshold, the decision algorithm can be given by:

\[
\hat{b}_k = \text{sign}[d_1(kT)]
\] (4.8)

where \( \text{sign}[x]=1 \) for \( x \geq 0 \) and \( \text{sign}[x]=-1 \) for \( x<0 \).

Table 4.2 and Table 4.3 also include the normalized phase difference for all different bit combinations. The phase state separation between the closest states with the opposite polarity can be used to investigate the effect of \( BT \) upon one-bit differential detection performance. The minimum differential phase angle is defined by:

\[
\Delta \theta_{\min} = 2(\theta_0 - \sum_{i \neq 0} \theta_i)
\] (4.9)

For the case of \( BT=0.3 \), \( \Delta \theta_{\min}=52^\circ \) and for the case of \( BT=0.5 \), \( \Delta \theta_{\min}=97.6^\circ \). This implies that \( BT=0.5 \) provides a better performance than \( BT=0.3 \) for GMSK one-bit differential detection.

The one-bit differential detection discussed above has x-axis as a threshold for decision and it is a hard-decision algorithm. The phase-state of the received signal which
is corrupted by the physical channel may cross the x-axis threshold and provides the opposite polarity for decision. The performance of this detection algorithm relies on the bandwidth-time product $BT$ which determines the ratio of the message signal and ISI terms ($SIR$), and the physical channel which determines the ratio of the message signal and noise ($SNR$).

### 4.2.2 Reed-Solomon Decoder

There are two types of Reed-Solomon decoders employed in this project: the error-correction only Reed-Solomon decoder and the errors and erasures correction Reed-Solomon decoder. Both of these decoders are implemented by the Berlekamp-Massey algorithm [Wie95].

Figure 4.8 shows the error-correction only Reed-Solomon decoder based on the detailed discussion of Chapter 3. The decision variable sequence from GMSK detector is first converted from the bit stream to the symbol sequence before the decoding process. The syndrome calculator gives the 16 syndromes for the (63,47) Reed-Solomon code block and any one of these syndromes not equal to zero implies that there is at least one symbol error in this 63-symbol code block. Otherwise, this code block has no symbol errors and can be delivered to the next stage to process. The error locator decision variables
from GMSK detector calculator determines the error locations and the number of errors. By using key equation and final field derivative, the error magnitude calculator gives the error magnitudes for all corresponding error locations. The error correction is implemented by summing the received signal polynomial and the error polynomial.

Figure 4.8 Error Correction only R-S Decoder

Figure 4.9 Error and Erasures R-S Decoder
The errors and erasures decoder for the Reed-Solomon code in CDPD is shown in Figure 4.9 above. Errors and erasures decoder is similar to the error-correction only decoder. The main difference is that errors and erasures decoder marks erasures coordinates to the corresponding symbol in which at least one bit has the magnitude in the erasure range. The errors and erasures decoder can determine the magnitudes for both errors and erasures in the Reed-Solomon code. The final correction is implemented by summing received signals, error and erasure polynomials.

With the MATLAB implementation, the simulation of Reed-Solomon decoding is very time consuming. Since the encoding and decoding processes of the (63,47) Reed-Solomon code in CDPD are deterministic, the simulation of the error-correction only and error-erasure decoding can be simplified rather than directly using Berlekamp-Massey decoding algorithm. Section 4.4 will discuss the entire system simulation in more detail.

4.3 Physical Channel Models

Two physical channel models are considered in this project: additive white Gaussian noise (AWGN) channel and Rayleigh fading channel.
4.3.1 AWGN Channel

The system simulation with AWGN channel is shown in Figure 4.10. AWGN channel is simulated by the transmitted signal adding the white Gaussian noise and the resulting signal fed into the CDPD receiver as the received signal.

![Figure 4.10 AWGN Channel Model](image)

In this thesis, the white Gaussian noise is generated by a random generator function in MATLAB with zero mean and variance $\sigma^2$. For baseband simulation, the complex noise is composed of in-phase and quadrature parts, as shown below:

$$ n(t) = n_x(t) + jn_y(t) \quad (4.10) $$

The energy of the noise signal is expressed by the variance of noise:

$$ \sigma^2 = \frac{N_0}{2} \cdot B_{eq} = \frac{N_0}{2} \cdot \frac{1}{T_s} \quad (4.11) $$
where $\sigma^2$ is the variance of noise, $N_0$ is the noise constant in watts per Hz, $B_{eq}$ is the equivalent bandwidth, $T_s$ is the sampling duration.

Assume that the energy per bit

$$E_b = A^2 T$$  \hspace{1cm} (4.12)

where $A$ is the amplitude of signal, and $T$ is the bit duration. The parameter $\frac{E_b}{N_0}$ can be given by:

$$\frac{E_b}{N_0} = \frac{A^2 T}{2\sigma^2 T_s}$$  \hspace{1cm} (4.13)

Here we assume that $A=1$, and $T = mT_s$. Then:

$$\frac{E_b}{N_0} = \frac{m}{2\sigma^2}$$  \hspace{1cm} (4.14)

Consider that the in-phase and quadrature parts of noise have the same variance, i.e. $\sigma_1^2 = \sigma_x^2 = \sigma_y^2$, then

$$\sigma_1^2 = \frac{m}{4 \frac{E_b}{N_0}}$$  \hspace{1cm} (4.15)
4.3.2 Rayleigh Fading Channel

In mobile radio environment, the Rayleigh distribution is commonly used to describe the statistical time-varying nature of the received envelope of a flat fading signal. The probability density function (pdf) of the Rayleigh distribution is given by [Rap96]:

\[
p(r) = \begin{cases} 
\frac{r}{\sigma^2} e^{-\frac{r^2}{2\sigma^2}} & (0 \leq r \leq \infty) \\
0 & (r < 0)
\end{cases}
\]  

(4.16)

where \( \sigma \) is the rms value of the received signal and \( \sigma^2 \) is the time-average of the received signal.

Clarker developed a flat fading model [Cla68] and Ganes developed a spectrum analysis for Clarke’s model [Gan72]. The output spectrum of the Rayleigh flat fading can be expressed by

\[
S(f) = \frac{1.5}{\pi f_m \sqrt{1 - \left(\frac{f - f_c}{f_m}\right)^2}}
\]  

(4.17)

where \( f_m \) is the maximum Doppler shift frequency and \( f_c \) is the carrier frequency.
Figure 4.11 Rayleigh Fading Simulator [Rap96]

In this project, Smith’s simulation algorithm [Smi75] for Clark-Gans fading model is employed. Figure 4.11 shows the Smith’s algorithm frequency domain implementation of a Rayleigh fading simulator at baseband. Smith’s method uses a complex Gaussian random number generator to generate a baseband line spectrum with the maximum frequency component of the line spectrum $f_m$. This random valued line spectrum is then multiplied with a discrete frequency representation of $\sqrt{S_x(f)}$ having the same number of points as the noise source. The resulting frequency domain signals from the in-phase and quadrature arms are taken an IFFT and squared. The square root of the sum of the two arms is the time domain Rayleigh fading signal.
Figure 4.12 Rayleigh Fading Channel Model

A block diagram of the system simulation with a Rayleigh fading channel is shown in Figure 4.12. The transmitted signal from the GMSK modulator is multiplied by the fading signal from the Rayleigh fading simulator. The received signal is obtained by the resulting signal passed through the fading channel adding the noise signal. Assume the carrier frequency $f_c = 836\text{MHz}$ which is used for CDPD specification, then the wavelength

$$\lambda = \frac{c}{f_c} = \frac{3 \times 10^8}{836 \times 10^6} = 0.35885 \text{ (m)}$$ (4.18)

The maximum Doppler frequency is given by

$$f_m = \frac{v}{\lambda}$$ (4.19)

where $v$ is the speed of vehicles.

For the vehicle speeds of 100, 50, and 8 km/hr specified by CDPD system [CDS95], the Doppler frequency are 77.408, 38.704 and 6.193 Hz respectively. The
number of the points of the fading spectrum $\sqrt{S(f)}$ determines the accuracy of the system simulation in the Rayleigh fading channels. The number of the points of $\sqrt{S(f)}$ can be calculated by

$$N_s = \frac{N_{in}}{R_t} \cdot f_m + 1 = \frac{N_{in}}{mR_b} \cdot f_m + 1$$  \hspace{1cm} (4.20)$$

where $N_{in}$ is the number of IFFT points, $R_b$ is the data rate of CDPD ($R_b=19.2$ kbps), $R_s$ is the sampling rate ($R_s = mR_b$, $m=5$ in this thesis).

Usually $N_{in}$ is taken as the number of $2^n$. For higher accuracy of the system simulation, employing a larger $N_s$ is better. However, increasing $N_{in}$ will lead to increase computations and slow down simulation. Table 4.4 shows all those parameters for three different vehicle speeds.

**Table 4.4 Parameters for Different Vehicle Speed**

<table>
<thead>
<tr>
<th>$v$ (km/hr)</th>
<th>$f_m$ (Hz)</th>
<th>$R_s$ (bps)</th>
<th>$N_{in}$</th>
<th>$N_s$</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>77.408</td>
<td>96000</td>
<td>$2^{17}$</td>
<td>106</td>
</tr>
<tr>
<td>50</td>
<td>38.704</td>
<td>96000</td>
<td>$2^{18}$</td>
<td>106</td>
</tr>
<tr>
<td>8</td>
<td>6.193</td>
<td>96000</td>
<td>$2^{19}$</td>
<td>35</td>
</tr>
</tbody>
</table>
4.4 Performance Analysis

4.41 Block Error Rate

In the CDPD system, the receiver RF sensitivity, which is the minimum received signal strength in dBm at the receiver’s antenna input terminals, is refer to the received block error performance of the receiver with GMSK modulation. The block error rate is defined as [CDS95]:

\[
\text{BLER} = \frac{\text{Blocks Sent} - \text{Correctable Blocks Received}}{\text{Total Number of Block Sent}}
\] (4.21)

The airlink specification of the CDPD system requires the reverse channel format to be used for testing. A correctable block is a received Reed-Solomon block or code word that either contains no errors in the 378 bit block (reverse channel format) or that contains up to and including the maximum number of symbol errors that can be corrected by the (63,47) Reed-Solomon code word. As mentioned before, the CDPD specification recommends that the Reed-Solomon decoder be implemented to correct up to seven symbol errors with an ARQ for the error detection. The BLER of 5% for the receiver RF sensitivity measurement in both AWGN and Rayleigh fading channels is used. Therefore, performance evaluations in this thesis are based on this specification.
The bit error rate (BER) is examined simultaneously in this simulation. But BER is not directly proportional to the BLER. Because a Reed-Solomon code block is based on symbols, each symbol error can be caused by from 1 to 6 bit errors. So the performance metrics such as the coding gain between the no-coding system and the Reed-Solomon code system, and the coding difference between the error-correction only and the error-erasure correction are defined in terms of BLER.

4.4.2 Performance of Reed-Solomon Code

The most common metrics of performance for any error correction code are the error probability of transmission. Since Reed-Solomon codes act on symbols, the probability of channel-symbol error or the channel symbol error rate $P_{se}$ is an important parameter. This can be used to determine the probability of an uncorrectable error $P_u$.

The probability of an uncorrectable error in a code block is related to the probability of channel symbol errors $P_{se}$, the minimum distance $d_{\text{min}}$ and the code rate $\frac{K}{N}$.

The probability of uncorrectable error is bounded by the probability of occurrence of an error pattern of weight $d_{\text{min}}$ or greater [Wie95]:

$$P_u \leq \sum_{n_e=\frac{d_{\text{min}}-1}{2}}^{N} \binom{N}{n_e} P_{se}^{n_e} (1 - P_{se})^{N-n_e} = 1 - \sum_{n_e=0}^{\frac{d_{\text{min}}-1}{2}} \binom{N}{n_e} P_{se}^{n_e} (1 - P_{se})^{N-n_e} \quad (4.22)$$
where the symbol error \( P_{se} = 1 - (1 - P_b)^m \) and \( P_b \) is the bit error rate under the assumption of purely random errors.

In the CDPD system, the (63,47) Reed-Solomon code has a block length of \( N=63 \), the number of correctable symbol errors \( t = \frac{d_{\text{min}} - 1}{2} = 8 \). But \( t=7 \) symbols is adopted for ARQ. Figure 4.13 shows the performance of the (63,47) Reed-Solomon code with \( P_u \) versus \( P_{se} \). Figure 4.13 indicates that with a small improvement for the symbol error rate \( P_{se} \), a large improvement for \( P_u \) is possible and there is a steep slope when \( P_{se} \leq 10^{-2} \) which performs a threshold of \( P_{se} \) for the better performance of the code block.

![Figure 4.13 Performance of (63,47) R-S Code in CDPD [Els96]](image-url)
In the case of the error and erasure correction decoding, the probability of error correction can be given by the probabilities of symbol errors \( P_{se} \) and symbol erasure \( P_{sf} \):

\[
P_c = \sum_{n_e=0}^{t} \sum_{n_f=0}^{n_e-2} \binom{N-n_e}{n_f} \binom{N}{n_f} P_{se}^{n_e} P_{sf}^{n_f} (1 - P_{se} - P_{sf})^{N-n_e-n_f} \tag{4.23}
\]

and the probability of an uncorrectable error in the code block:

\[
P_u = 1 - P_c \tag{4.24}
\]

Equation (4.23) implies that the maximum value of \( P_c \) exists for the special values of \( P_{se} \) and \( P_{sf} \) as shown in Figure (4.14).

![Figure 4.14 Assumption of the Maximum of Pc](image)

Chapter 4 System Modeling and Simulation
In order to achieve the maximum value of \( P_c \) for each \( \frac{E_b}{N_o} \), the threshold of the soft decision must be adjusted. It seems possible to select the optimum threshold by Equation \((4.23)\) for \( P_{c_{\text{max}}} \), but the analytic formula for the optimum threshold would be very complicated. Instead of more detailed theoretical analysis, this project will develop the simulation of CDPD system for both error correction only and error-erasure correction decoders.

**4.4.3 System Simulation**

The system simulation implements three types of CDPD airlink: (1) the system without error control code (Figure 4.15), (2) the system with (63,47) Reed-Solomon error-correction only code (Figure 4.16), (3) the system with (63,47) Reed-Solomon error and erasure decoder (Figure 4.17). In each case, AWGN and Rayleigh fading channels are chosen as physical channel models.

In the simulation of no-coding CDPD system, any bit error of the 378-bit code block can cause one block error. The maximum number of block errors to be counted is set to fifteen. The simulation provides both BLER and BER versus \( \frac{E_b}{N_o} \) curves.

As mentioned in section 4.2, the encoding and decoding processes of the (63,47) Reed-Solomon code in CDPD are deterministic, therefore, the simulations of both error-correction only and error-erasure decoding can be implemented without using Reed-
Solomon systematic encoding and Berlekamp-Massey decoding algorithm. This will not results in any loss of system simulation and will be much faster. In Figure (4.16), each (63, 47) Reed-Solomon code block goes through GMSK modulation, channel models and GMSK one-bit differential detector. Since Reed-Solomon code can correct up to 7 symbol errors in one code block, any code block with less than or equal to seven symbol errors will be considered as a correctable block and no block error will be counted. Unlike above two system simulations, the system with errors and erasures correction adjusts the decision thresholds in a certain range for minimizing BLER. The soft decision produces three states of decision variables: two of binary states and one of erasure state. The program checks the number of symbol errors first: only more than seven symbol errors in one Reed-Solomon code block being considered as one block error. Then the program will checks the number of symbol erasure in one block according to Equation (3.35). Only no block error caused by either symbol errors or symbol erasures being considered as no block error in the code block. For each $\frac{E_b}{N_0}$, there exist a group of BLER and BER which correspond to the different thresholds. The minimum of BLER is provided by the optimum threshold.

In this chapter we have summarized the important aspects of our CDPD system simulation. In the next chapter, we will prevent simulation results which quantify the performance improvement which may be possible in CDPD receiver implementing errors and erasures decoding.
Figure 4.15 System Simulation without Error Control Code
Figure 4.16 System Simulation with R-S Code
Figure 4.17 System Simulation with Error and Erasure Decoder
Chapter 5 Simulation Results

In this chapter, the results obtained of MATLAB of CDPD with errors and erasures decoding is presented are presented. The simulations are grouped based on channel models, decoding algorithms and the bandwidth-time products associated with GMSK modulation. In addition to the bandwidth-time product $BT=0.5$ for a CDPD system, the bandwidth-time product, $BT=0.3$, is examined as this is characteristic of the widely-used GSM system.

The entire airlink simulation of CDPD system is based on a series of functional blocks: GMSK modulation, demodulation, $(63,47)$ Reed-Solomon systematic encoding, decoding and channel models. One-bit differential detection is employed for the GMSK demodulation and the Berlekamp-Masssey algorithm is adopted for the Reed-Solomon decoding. The soft-decision algorithm with two thresholds, as opposed to one fixed threshold, is implemented to make decisions for both message data and erasures. The errors and erasures decoder can correct any combination up to seven symbol errors or 15 symbol erasures in one Reed-Solomon code block, according to Equation (3.33): $2t + f + 1 \leq d_{\text{min}}$. The results performance of the CDPD system is evaluated in terms of both BLER and BER. In addition, AWGN and Rayleigh fading channels were both examined.
5.1 AWGN

In the following discussions regarding simulation results, a block error rate (BLER) of 5% was used as a performance benchmark according to the CDPD system specification [CDS95]. The threshold for soft-decision algorithm was varying from 0 to ±1 in a increment of 0.1.

5.1.1 $BT=0.5$ Case

Block error rate is a main consideration in implementing the CDPD system and a BLER of 5% is chosen as a performance metric for the CDPD standard in AWGN channels. Figure 5.1 depicts the BLER as a function of $E_b/N_o$ for the system of $BT=0.5$, without coding and with (63,47) Reed-Solomon code in AWGN. When applying the (63,47) Reed-Solomon code, a maximum of seven symbol errors can be corrected. The hard decision with unique threshold of '0' is adopted for one bit differential detection of GMSK demodulation. The $E_b/N_o$ for the case of no-coding and Reed-Solomon coding (BLER=5%) are required to be 7.7 dB and 12.5 dB respectively. The block coding gain of the system with $BT=0.5$ in AWGN is about 4.8 dB. This result is very close to the result given by Elson using SPW [Els96]. Figure 5.2 shows the BLER as a function of $E_b/N_o$ for the CDPD system with errors and erasures decoding. The $E_b/N_o$ in this case is required to be 7.15 dB. The coding gain difference between the (63,47) Reed-Solomon
code with 7 symbol error correction only (4.8 dB) and error-erasure correction (5.35 dB) in AWGN is about 0.55 dB. This implies that the Reed-Solomon code with errors and erasures decoder does improve BLER performance for CDPD systems compared with the hard-decision and error-correction only decoder, even though the improvement does not provide significant additional gain for AWGN channels. The BLER plots for each of the cases discussed above are combined in Figure 5.3 for the purpose of comparison.

Figure 5.1 BLERs of No-Coding and Error-Correction Only Decoding

(BT=0.5 and AWGN)
The threshold of the soft-decision and errors-erasures decoding algorithm was based on the minimum values of BLER while the threshold is adjusted from 0 to 1 in the increment of 0.1. The data in Table 5.1 shows that the optimum thresholds are close to 0.2 and fairly robust to different $E_b/N_0$.

**Table 5.1 The Optimum Threshold for AWGN with $BT=0.5$**

<table>
<thead>
<tr>
<th>$E_b/N_0$ (dB)</th>
<th>0~5</th>
<th>5</th>
<th>6</th>
<th>6.3</th>
<th>6.5</th>
<th>6.8</th>
<th>7</th>
<th>7.2</th>
<th>7.5</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Threshold</td>
<td>--</td>
<td>0.2</td>
<td>0.1</td>
<td>0.2</td>
<td>0.2</td>
<td>0.3</td>
<td>0.3</td>
<td>0.2</td>
<td>0.4</td>
<td></td>
</tr>
</tbody>
</table>

**Figure 5.2** BLERs of No-Coding and Error-Erasure Decoding

*(BT=0.5 and AWGN)*
Figure 5.3 BLERs of Three Cases (BT=0.5 and AWGN)

Having examined the performance of the coding algorithms with respect to BLER, which is the primary performance measure of interest for the packet oriented CDPD system, it is now useful to evaluate these same approaches in terms of BER. Figure 5.4 depicts the BER of the CDPD system with both no-coding and Reed-Solomon coding in AWGN. The coded BER under $E_b/N_o=7.7$ dB which is required given a BLER of 5% is $10^{-3}$ and the uncoded BER under the same $E_b/N_o$ is $10^{-2}$. Figure 5.5 shows that the system incorporating the errors and erasures decoder has a BER of $2 \times 10^{-4}$ under the same condition. The three BER plots are combined in Figure 5.6 where it is clearly seen that the errors and erasures decoder performs better, with respect to BER, than the hard-decision, error-correction only decoder.
Figure 5.4 BERs of No-Coding and Error-Correction Only Decoding

\((BT=0.5 \text{ and AWGN})\)

Figure 5.5 BERs of No-Coding and Error-Erasurere Decoding

\((BT=0.5 \text{ and AWGN})\)
5.1.2 $BT=0.3$ case

As stated earlier, the GMSK of $BT=0.3$ is also examined since GSK system, which is characterized by $BT=0.3$, is one of the most popular digital communication system, so this bandwidth-time product may be of interest also. We examine the hard and soft decision decoding performance for the different $BT$. Figure 5.7 depicts the BLER curves for the systems defined by $BT=0.3$ in the cases of: no-coding; (63,47) Reed-Solomon coding (up to 7 symbol error correction); and Reed-Solomon coding with the errors and erasures decoder. The $E_b/N_o$ corresponding to acceptable performance for each of three cases listed above are 18 dB (no-coding), 12.4 dB (Reed-Solomon coding) and 11.4 dB.
(errors and erasures decoding). The $E_b/N_o$ in each of the three cases is greater than those observed in Figure 5.1 by about 4.5 dB. This verifies that the narrowband spectrum is achieved at the expense of introducing severe intersymbol interference (ISI) into the baseband waveform of the FM modulator input. In addition, it is seen that ISI is inversely proportional to the $BT$ product. The coding gain for Reed-Solomon code with the error-correction only decoder is 5.6 dB while the coding gain for Reed-Solomon code with the errors-erasures decoder is 6.6 dB. Thus, a 1 dB improvement is realized with the errors and erasures decoder. The BER plots for each of the coding approaches are compared in Figure 5.8.

**Figure 5.7** BLERs of Three Cases ($BT=0.3$ and AWGN)
5.2 Flat Rayleigh Fading

Fading is one of the most significant problems in wireless mobile radio communications. Deep fading can corrupt, or even destroy, the information signal in the wireless mobile environment. A mobile receiver at 50 km/hr may pass through several, or more, fades in one second. Many factors associated with the radio communications channel influence fading such as: the speed of the mobile receiver; multipath propagation; and the transmission bandwidth of the signal. The CDPD system specification provides the performance standard for mobile vehicles moving at speeds of 100 km/hr, 50 km/hr
and 8 km/hr in flat Rayleigh fading conditions. The Doppler frequencies associated with the different vehicle speeds affect the transmission frequency. As indicated in the CDPD specification, a 5% BLER is the standard block error rate (referred to the receiver RF sensitivity) for the fading environment. This section presents the simulation results obtained in simulating the CDPD system under Rayleigh fading channels.

**5.2.1 Rayleigh Fading Channels with \( v = 100 \) km/hr and \( BT = 0.5 \)**

The BLER values obtained from simulating the CDPD system under the Rayleigh fading condition with a vehicle speed of 100 km/hr is shown in Figure 5.9. The 5% BLER occurs at \( E_b/N_o = 30 \) dB for the system without error control coding and \( E_b/N_o = 18.4 \) dB for the system with (63,47) Reed-Solomon coding, respectively. This provides a coding gain of 11.6 dB. This is in contrast to the AWGN case, where the seven symbol error correction implementation of Reed-Solomon code yielded a gain of only 4.8 dB.
Figure 5.9 BLER of No-Coding and Error Correction only Coding in Rayleigh Channels ($v=100$ km/hr and $BT=0.5$)

As discussed in Chapter 4, the soft decision decoder of the (63,47) Reed-Solomon code is implemented by the optimum decision threshold and errors and erasures correction decoder. The errors and erasures correction decoder follows the relationship between the number of symbol errors and erasures in one Reed-Solomon code block and the minimum distance of the code, i.e. $2t + f + 1 \leq d_{\text{min}}$. The optimum decision threshold actually consists of two thresholds which are symmetric with respect to the x-axis. For each $E_b/N_o$, the thresholds vary over a predetermined range in small increments. The threshold which provides the minimum BLER for a particular $E_b/N_o$ is selected.
Therefore, the optimum threshold varies as the $E_b/N_o$ changes. However, the simulation results presented earlier indicate that the optimum threshold varies over a small interval.

Figure 5.10 depicts the BLER obtained using the (63,47) Reed-Solomon code with errors and erasures decoding given Rayleigh fading channels, $v=100$ km/hr and $BT=0.5$. An $E_b/N_o$ of 15.9 dB is needed for the 5% BLER and the coding gain realized approximately equal to 14.1 dB. Thus, when compared to the case corresponding to error-correction only, the errors-erasures decoder yields a 2.5 dB improvement in coding gain. In addition, it is seen that the Reed-Solomon code in conjunction with the errors and erasures decoder is much more effective in the Rayleigh fading channel (14.1 dB coding gain), than in the AWGN channel (5.35 dB). Figure 5.11 contains each of the plots for comparison purpose.

Table 5.2 shows the the optimum thresholds for the different $E_b/N_o$, while the threshold is adjusted from 0 to 0.5 in the increment of 0.05.

<table>
<thead>
<tr>
<th>$E_b/N_o$ (dB)</th>
<th>0-5</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
</tr>
</thead>
<tbody>
<tr>
<td>Threshold</td>
<td>--</td>
<td>0.4</td>
<td>0.3</td>
<td>0.2</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td>$E_b/N_o$ (dB)</td>
<td>--</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td>19</td>
<td>20</td>
<td>21</td>
<td>22</td>
<td>23</td>
<td>24</td>
</tr>
<tr>
<td>Threshold</td>
<td>--</td>
<td>0.1</td>
<td>0.2</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.3</td>
<td>0.2</td>
<td>0.1</td>
<td>0.2</td>
</tr>
</tbody>
</table>
Figure 5.10 BLER of No-Coding and Error-Erasure Decoding in Rayleigh Channels ($v=100$ km/hr and $B T=0.5$)

Figure 5.11 BLER of Three Decoding Algorithms in Rayleigh Channels ($v=100$ km/hr and $B T=0.5$)
The BER plot for the CDPD system with no-coding and with Reed-Solomon error control coding is shown in Figure 5.12. At $E_b/N_0 = 18.4$ dB, the coded BER is about $2 \times 10^{-3}$ and the uncoded BER is $10^{-2}$. Figure 5.13 shows the BER ($6 \times 10^{-4}$) for the CDPD system using the Reed-Solomon code with errors and erasures decoder. The three BER plots are combined in Figure 5.14.

![Figure 5.12 BER of No-Coding and Error Correction only Coding in Rayleigh Channels ($v=100$ km/hr and $BT=0.5$)](image.png)

"Figure 5.12 BER of No-Coding and Error Correction only Coding in Rayleigh Channels ($v=100$ km/hr and $BT=0.5$)"
Figure 5.13 BER of No-Coding and Error-Erasure Decoding in Rayleigh Channels ($v=100$ km/hr and $BT=0.5$)

Figure 5.14 BER of Three Decoding Algorithms in Rayleigh Channels ($v=100$ km/hr and $BT=0.5$)
5.2.2 Rayleigh Fading Channel with \( v = 50 \text{ km/hr} \) and \( BT = 0.5 \)

Keeping the bandwidth-time product at \( BT = 0.5 \), the vehicle speed is changed to \( v = 50 \text{ km/hr} \) and the CDPD system simulated again using a Rayleigh fading channel. Figure 5.15 shows the simulation results, in terms of BLER, for the system employing the three different coding algorithms. The \( E_b/N_o \) necessary to obtain a BLER of 0.05 is found to be 19 dB for the CDPD system using (63,47) Reed-Solomon code. The coding gain in this case is 10.5 dB. Referring to Figure 5.15, there is a coding gain of 12.5 provided by the Reed-Solomon code with errors and erasures decoding. Thus, there is a coding gain difference of 2 dB between the hard-decision decoding algorithm and the soft-decision with the error-erasure decoding. Figure 5.16 contains the BERs associated with the three different cases given a Rayleigh fading channel under the conditions \( v = 50 \text{ km/hr} \) and \( BT = 0.5 \).

5.2.3 Rayleigh Fading Channel with \( v = 8 \text{ km/hr} \) and \( BT = 0.5 \)

The BLER and BER plots corresponding to the 8 km/hr fading case are shown in Figure 5.17 and 5.18. The coding gain of 3.6 dB for the error correction only decoding algorithm is achieved at an \( E_b/N_o \) of 22.4 dB. A coding gain of 5.5 dB, given the errors and erasures decoding algorithm, is obtained at an \( E_b/N_o \) of 20.5 dB.
Figure 5.15 BLER of Three Decoding Algorithms
in Rayleigh Channels ($v=50$ km/hr and $BT=0.5$)

Figure 5.16 BER of Three Decoding Algorithms
in Rayleigh Channels ($v=50$ km/hr and $BT=0.5$)
The $E_b/N_0$ levels listed above are the largest among each of the cases examined given the fading channel ($v=100, 50$ and $8$ km/hr). The performance, in terms of $E_b/N_0$, deteriorates at lower vehicle speeds. This is attributed to the fact that a mobile receiver moving at a lower speed operates for a longer period in the severely faded portion of the communication channel. Given the lower vehicle speed, an accurate simulation is impractical in terms of computation time. Therefore, the results displayed in Figure 5.17 and 5.18 should be interpreted only as general indicators of behavior in the low speed case.

![Figure 5.17 BLER of Three Decoding Algorithms in Rayleigh Channels ($v=8$ km/hr and $BT=0.5$)]
5.2.4 Rayleigh Fading Channels for $BT=0.3$ Cases

Figure 5.19 and 5.20 contain the BLER and BER plots at $v=100$ km/hr and $BT=0.3$. Larger coding gains are realized in this case (15.7 dB for the error-correction only and 18.4 dB for the errors and erasures decoding) than were seen in the $BT=0.5$, $v=100$ km/hr case (11.4 dB for the error-correction only and 14.1 dB for the errors and erasures decoding). However, it is noted that these gains occur only at higher $E_b/N_0$ levels.
Figure 5.19 BLER of Three Decoding Algorithms in Rayleigh Channels ($v=100$ km/hr and $BT=0.3$)

Figure 5.20 BER of Three Decoding Algorithms in Rayleigh Channels ($v=100$ km/hr and $BT=0.3$)
BLER and BER plots corresponding to the case \(v=50 \text{ km/hr}\) and \(BT=0.3\) are shown in Figures 5.21 and 5.22. We desire to compare the results obtained under the bandwidth-time product (\(BT=0.5\) and \(BT=0.3\)). Tables 5.1 and 5.1 contain a summary of the results obtained in the \(BT=0.5\) and \(BT=0.3\) cases, respectively. We have found through extensive simulation that additional coding gain of 0.5 - 1 dB are possible for an AWGN channel and additional 1.9 - 2.7 dB are possible for a fading channel, using errors and erasures decoding of the Reed-Solomon. Slightly greater coding gain are possible for small bandwidth-time products, because of the increased role of ISI in producing errors. Performance for fading channels is superior at higher vehicle speeds because of the randomizing effect which this has an error statistics.

![Figure 5.21 BLER of Three Decoding Algorithms in Rayleigh Channels (\(v=50 \text{ km/hr}\) and \(BT=0.3\))](image-url)
Figure 5.22 BER of Three Decoding Algorithms

in Rayleigh Channels ($v=50$ km/hr and $BT=0.3$)

Table 5.3 Simulation results for $BT=0.5$

<table>
<thead>
<tr>
<th></th>
<th>SNR for 5% BLER</th>
<th>Coding Gain</th>
<th>Difference (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Uncoded (dB)</td>
<td>Coded(h) (dB)</td>
<td>Coded(s) (dB)</td>
</tr>
<tr>
<td>AWGN</td>
<td>12.5</td>
<td>7.7</td>
<td>7.15</td>
</tr>
<tr>
<td>100 km/hr</td>
<td>30</td>
<td>18.4</td>
<td>15.9</td>
</tr>
<tr>
<td>50 km/hr</td>
<td>29.5</td>
<td>19</td>
<td>17</td>
</tr>
<tr>
<td>8 km/hr</td>
<td>26</td>
<td>22.4</td>
<td>20.5</td>
</tr>
</tbody>
</table>
Table 5.4 Simulation results for $BT=0.3$

<table>
<thead>
<tr>
<th></th>
<th>SNR for</th>
<th>5% BLER</th>
<th>Coding</th>
<th>Gain</th>
<th>Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Uncoded (dB)</td>
<td>Coded (dB)</td>
<td>Coded (s) (dB)</td>
<td>Error only</td>
<td>Errors &amp; erasures</td>
</tr>
<tr>
<td>AWGN</td>
<td>18</td>
<td>12.4</td>
<td>11.4</td>
<td>5.6</td>
<td>6.6</td>
</tr>
<tr>
<td>100 km/hr</td>
<td>37.5</td>
<td>21.8</td>
<td>19.1</td>
<td>15.7</td>
<td>18.4</td>
</tr>
<tr>
<td>50 km/hr</td>
<td>37.5</td>
<td>22.6</td>
<td>20.7</td>
<td>14.9</td>
<td>16.8</td>
</tr>
</tbody>
</table>
Chapter 6 Conclusion

6.1 Summary

CDPD is a wireless packet data communication system which provides packet data connectivity in the wireless mobile radio environment. The CDPD system utilizes existing cellular infrastructure equipment and allows a mobile user to access unused cellular radio channels by the frequency hopping technique. This research focused on simulating the airlink of CDPD system and investigating the performance of an advanced decoding algorithm.

The results of simulations for the CDPD system in AWGN are grouped by the $BT$ product associated with GMSK modulation. Given a $BT$ of 0.5, the $(63,47)$ Reed-Solomon code with hard-decisions and an error-correction only decoding algorithm yields a 4.8 dB block error rate coding gain. This result agrees quite well with those published in current literature as well as Elson’s result [Els96]. Given the same conditions, the errors and erasures decoder in conjunction with $(63,47)$ Reed-Solomon code contributes an extra coding gain of 0.55 dB. The BERs corresponding to those two decoding algorithms and with respect to $E_b/N_o = 7.7$ dB are on the order of $10^{-3}$ and $10^{-4}$, respectively (slightly better performance in the case of the errors and erasures decoder). For a $BT$ of 0.3, the simulation results indicate a 1 dB difference in coding gain between the hard-decision decoder (5.6 dB) and the soft-decision decoder (6.6 dB). However, the results also show
that a higher signal strength ($E_b/N_o = 4.5 \text{ dB}$) is needed in the $BT=0.3$ case than in the $BT=0.5$ case. This verifies the inverse relationship between the $BT$ product and ISI and the greater signal strength necessary for the same BLER specification. In other words, a narrow spectrum for GMSK modulation is achieved at the expense of introducing severe ISI into the baseband signal of the FM modulator input.

The Rayleigh fading channel was also examined in this research project. According to the CDPD system specification, a mobile receiver moving at speeds of 100, 50 and 8 km/hr was simulated. The simulation results of the CDPD system under the condition of the vehicle speed of 100 km/hr produce a 2.5 dB coding difference between the error-correction only decoder (11.6 dB) and errors and erasures decoder (14.1 dB). This implies that the required signal strength can be reduced by using an errors and erasures decoder with optimum varying thresholds instead of using an error-correction only decoder with a fixed threshold under the discussed condition. Therefore, the Reed-Solomon code with errors and erasures decoder provides substantial improvement over fading and bursty channels. The BERs achieved in this case ($E_b/N_o = 18.4 \text{ dB}$) were on the order of $10^{-3}$ (error-correction only decoding) and $10^{-4}$ (errors and erasures decoding) for the two decoding algorithms.

The simulation results indicate that the BLERs of both decoding algorithms tend to be worse at lower vehicle speeds than at higher vehicle speeds. This is expected since a lower speed vehicle will remain in a deep fading situation longer than a higher speed vehicle (given a specific length of a Reed-Solomon code block). The longer fading time in
each Reed-Solomon block will create longer bursts of symbol errors and erasures, which will lead to the higher BLERs for both decoding algorithms in the lower speed case.

6.2 Further Work

The work undertaken in this project yields a fundamental result: the Reed-Solomon code with errors and erasures decoder can provide better performance for packet data transmission in a wireless mobile environment, especially in fading and bursty channels. However, more detailed research regarding the determination of an optimum threshold for the soft-decision technique will certainly be needed for the best possible practical application of the errors and erasures decoder. Attempts at deriving analytic results which support the results found by simulation will provide valuable insight into the performance of the CDPD system. In addition, future investigation into the implementation errors and erasures decoding is necessary to make the technique more practical. Finally, variable rate coding schemes should be examined as they may lead to increased capacity in the CDPD system.
References


[Gra95] Gray, F. Coding Theory Class Notes, Virginia Polytechnic Insitute and State University Spring, 1995


Vita

Chenbo Zheng was born in Ningbo, China. He obtained a Bachelor of Electrical Engineering degree in 1982 from Dalian University of Technology, China, a Master of Science in Electrical Engineering degree in 1988 from the Southeast University, China, and a Master of Science in Physics degree in 1993 from Emporia State University, Kansas, USA. He worked as an engineer and a lecturer at Nanjing Power College, China from 1982 to 1991. In August 1994, he enrolled in the Department of Electrical Engineering at Virginia Polytechnic Institute and State University where he was selected as a member of the Honor Society of Phi Kappa Phi and the member of the National Electrical Engineering Honor Society, Eta Kappa Nu. At Virginia Tech, he completed the Master of Science in Electrical Engineering degree in June 1996.