PERFORMANCE EVALUATION OF THE ISDB-T STANDARD FOR MULTIMEDIA SERVICES

by

XINRONG WANG

Bachelor of Engineering, Shanghai Jiao Tong University, Shanghai, China, 1991

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF APPLIED SCIENCE

in

THE FACULTY OF GRADUATE STUDIES

DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING

We accept this thesis as conforming
to the required standard

THE UNIVERSITY OF BRITISH COLUMBIA

June 2002

© Xinrong Wang, 2002
In presenting this thesis in partial fulfilment of the requirements for an advanced degree at the University of British Columbia, I agree that the Library shall make it freely available for reference and study. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by the head of my department or by his or her representatives. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

Department of **Electrical & Computer Engineering**

The University of British Columbia
Vancouver, Canada

Date **August 27, 2002**
Abstract

The Integrated Services Digital Broadcasting - Terrestrial (ISDB-T) standard is one of the three dominant digital television terrestrial broadcasting standards. It employs concatenated coding and OFDM techniques to combat fading and multipath channel impairments. ISDB-T operates in 6, 7, or 8 MHz channel bandwidths, with each channel supporting up to three different services. These services may use different convolutional code rates, modulation schemes, and time interleaving lengths to meet different quality of service requirements.

In this thesis, an ISDB-T baseband simulation model is implemented using SPW™, C, and C++, and validated. This model is built to accommodate all of the possible ISDB-T system configurations.

Since the target BER in ISDB-T is very low, for example, $1 \times 10^{-11}$ for high definition television, it is impractical to obtain the BER directly using software simulation. A Markov chain-based semi-analytic method is used, which agrees well with available simulation results in non-fading environments. The semi-analytic results are quite different from the simulation results in fading channels, especially at very low Doppler frequencies. The channel undergoes deep fades at very low Doppler frequencies, causing a large number of symbol errors during these fading periods. The Markov chain model does not model these kinds of long error bursts very well.

A numerical analysis is provided for very slow Rayleigh fading. The results of the numerical analysis agree quite well with the simulation results.
# Table of Contents

Abstract ................................................................. ii

Table of Contents ......................................................... iii

List of Tables ................................................................ vii

List of Figures ................................................................ viii

Acknowledgment ............................................................. xi

Chapter 1  Introduction .................................................... 1

1.1 Digital Television Terrestrial Broadcasting .............................. 1

1.2 Motivation and Objectives .................................................. 4

1.3 Outline of the Thesis ......................................................... 6

Chapter 2  Overview of ISDB-T ............................................. 7

2.1 Requirements for ISDB-T Transmission ................................ 8

2.2 Main Techniques of ISDB-T ................................................ 10

2.2.1 Concatenated Coding .................................................... 10

2.2.2 Band Segmented Transmission - OFDM .......................... 10

2.2.3 Hierarchical Transmission and Partial Reception ................ 11

2.2.4 Multiplex Frame and OFDM Frame ................................ 14

2.3 ISDB-T Transmission Parameters ..................................... 18

2.4 Block Diagram of ISDB-T .................................................. 20

2.4.1 ISDB-T Transmitter ...................................................... 21

2.4.2 ISDB-T Receiver .......................................................... 32

Chapter 3  Description of the SPW Simulation Model ................. 34

3.1 Choice of Simulation Tool ................................................ 34
3.1.1 Simulation Tool in the Related Works ........................................... 34
3.1.2 Physical Layer Simulation Tool .................................................. 34

3.2 Brief Overview of SPW™ ................................................................. 35
3.2.1 File Manager ................................................................................ 35
3.2.2 Block Diagram Editor ................................................................. 35
3.2.3 Signal Calculator ........................................................................ 36
3.2.4 Simulation Manager .................................................................... 36
3.2.5 Simulation Program Builder Simulator ....................................... 37
3.2.6 SPW™ Libraries .......................................................................... 37

3.3 General Information of the ISDB-T Model ...................................... 38
3.3.1 SPW™ ISDB-T Model ................................................................. 38
3.3.2 Editable Parameters in the ISDB-T Model .................................... 39

3.4 Main Blocks in the ISDB-T Model .................................................. 41
3.4.1 MPEG-2 Source .......................................................................... 41
3.4.2 RS Encoder and Splitter ............................................................. 42
3.4.3 Channel Coding ........................................................................... 42
3.4.4 Modulation .................................................................................. 44
3.4.5 Time Interleaving ........................................................................ 45
3.4.6 Frequency Interleaving ............................................................... 46
3.4.7 OFDM Structure .......................................................................... 46
3.4.8 ISDB-T Channel Model ............................................................. 47
3.4.9 Guard Remove and FFT ............................................................... 53
3.4.10 Channel Compensation ............................................................. 53
List of Tables

Table 2.1 Number of Interfaced TSPs in One Multiplex Frame .................................................15
Table 2.2 Number of Transmission TSPs in One OFDM Frame (using 1-, 2-, 7-, 13-segment systems as examples) .................................................................16
Table 2.3 Segment Parameters for ISDB-T (6 MHz) .................................................................18
Table 2.4 Transmission Parameters for ISDB-T (6 MHz) .........................................................19
Table 2.5 Information Rates of OFDM Segment for ISDB-T (6 MHz channel bandwidth) .................................................................20
Table 2.6 Delay Adjustment for Byte-wise Interleaving and Deinterleaving .........................24
Table 2.7 Puncturing Pattern and Transmitted Sequence .........................................................25
Table 2.8 Delay Adjustment for Bit Interleaving and Deinterleaving ..................................29
Table 2.9 Delay Adjustment for Time Interleaving and Deinterleaving ............................31
Table 3.1 Five Channel Models ..............................................................................................48
Table B.1 FFT Size for Number of Segments ............................................................................105
Table C.1 Segment Parameters for ISDB-T (7 MHz) .............................................................106
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Block Diagram of DTTB System</td>
<td>2</td>
</tr>
<tr>
<td>2.1</td>
<td>Block Diagram of ISDB-T Transmitter</td>
<td>8</td>
</tr>
<tr>
<td>2.2</td>
<td>Illustration of Hierarchical Transmission and Partial Reception in ISDB-T</td>
<td>13</td>
</tr>
<tr>
<td>2.3</td>
<td>MPEG-2 TSP and ISDB-T Transmission TSP</td>
<td>14</td>
</tr>
<tr>
<td>2.4</td>
<td>Structures of Multiplex Frame and OFDM Frame</td>
<td>17</td>
</tr>
<tr>
<td>2.5</td>
<td>Block Diagram of ISDB-T</td>
<td>21</td>
</tr>
<tr>
<td>2.6</td>
<td>Scrambler / Descrambler Schematic Diagram</td>
<td>22</td>
</tr>
<tr>
<td>2.7</td>
<td>Conceptual Diagram of the Byte-wise Interleaver and Deinterleaver</td>
<td>23</td>
</tr>
<tr>
<td>2.8</td>
<td>Mother Convolution Code of Rate 1/2</td>
<td>25</td>
</tr>
<tr>
<td>2.9</td>
<td>-Shift DQPSK Bit Interleaving and Modulation</td>
<td>26</td>
</tr>
<tr>
<td>2.10</td>
<td>Phase Calculation and -Shift DQPSK Mapping</td>
<td>26</td>
</tr>
<tr>
<td>2.11</td>
<td>QPSK Bit Interleaving, Mapping and the Corresponding Bit Patterns</td>
<td>26</td>
</tr>
<tr>
<td>2.12</td>
<td>16QAM Bit Interleaving, Mapping, and the Corresponding Bit Patterns</td>
<td>26</td>
</tr>
<tr>
<td>2.13</td>
<td>64QAM Bit Interleaving, Mapping, and the Corresponding Bit Patterns</td>
<td>28</td>
</tr>
<tr>
<td>2.14</td>
<td>Structure of Data Segment</td>
<td>30</td>
</tr>
<tr>
<td>2.15</td>
<td>Configuration of Frequency Interleaver</td>
<td>32</td>
</tr>
<tr>
<td>2.16</td>
<td>Block Diagram of ISDB-T Receiver</td>
<td>33</td>
</tr>
<tr>
<td>3.1</td>
<td>Soft Decision Variables in QPSK, 16QAM and 64QAM</td>
<td>59</td>
</tr>
<tr>
<td>3.2</td>
<td>Block Diagram for Time Interleaving Validation</td>
<td>65</td>
</tr>
<tr>
<td>3.3</td>
<td>Performance of the ISDB-T Model with No Coding in A Non-fading Channel</td>
<td>67</td>
</tr>
<tr>
<td>3.4</td>
<td>Performance of the ISDB-T Model with No Coding in A Non-fading Channel</td>
<td>69</td>
</tr>
</tbody>
</table>
Figure 3.5  BER Performance of the ISDB-T Model with Convolutional Coding in A Non-fading Channel. (Solid curves are simulation results obtained in this study, and dashed curves are results from [6].) ..........70

Figure 4.1  Simple Block Diagram of ISDB-T .................................................................71

Figure 4.2  Three-State Markov Chain Model ..............................................................73

Figure 4.3  Five-state Markov Chain Model .................................................................74

Figure 4.4  Distribution of the Number of Symbol Errors in the RS Codewords at the Input of the RS Decoder in A Non-fading Channel...........................................79

Figure 4.5  Distribution of the Number of Symbol Errors in the RS Codewords at the Input of the RS Decoder in A Two-path Frequency Selective Non-fading Channel.................................80

Figure 4.6  Distribution of the Number of Symbol Errors in the RS Codewords at the Input of the RS Decoder in A Ricean Channel.........................................................81

Figure 4.7  Distribution of the Number of Symbol Errors in the RS Codewords at the Input of the RS Decoder in A Two-path Rayleigh Channel (Fd = 200 Hz).................................................................81

Figure 4.8  Distribution of the Number of Symbol Errors in the RS Codewords at the Input of the RS Decoder in A Two-path Rayleigh Channel (Fd = 70 Hz and Fd = 7 Hz).........................................................82

Figure 5.1  Comparison of BER at the Output of Viterbi Decoder with BER at the Output of the RS Decoder in Different Modulation Schemes in a Non-fading Channel. (Solid curves are the results at the output of the Viterbi decoder, and dashed curves are the results at the output of the RS decoder.) .........................84

Figure 5.2  Comparison of BER for with and without Convolutional Interleavings in A Non-fading Channel.................................................................85

Figure 5.3  BER, SER, and WER Performance in A Non-fading Channel. (Solid curves are the simulations. Dashed curves are the semi-analytic results.) .........................................................86

Figure 5.4  Comparison of BER for Different Convolutional Coding Rates in A Two-path Frequency Selective Non-fading Channel. (Solid curves are the simulation results, and dashed curves are the semi-analytic results.) ........................................87
Figure 5.5 Comparison of BER for Different Time Interleaving Lengths. ...............88

Figure 5.6 Comparison of the Semi-analytic and the Simulation Results in A Ricean Channel. .................................................................89

Figure 5.7 Comparison of Different Doppler Frequencies in A Two-path Rayleigh Channel.................................................................90

Figure 5.8 Comparison of the Semi-analytic and the Simulation Results in A Two-path Rayleigh Channel.................................................91

Figure 5.9 Comparison of Different Number of Segments in A Two-path Rayleigh Channel.................................................................92

Figure 5.10 Comparison of the Semi-analytic and the Simulation Results in A Rayleigh Channel..........................................................93

Figure 5.11 Comparison of the Semi-analytic and the Simulation Results in A Rayleigh Channel..........................................................93

Figure 5.12 BER Performance in A Non-fading Channel.................................95

Figure 5.13 PDF of the Central Chi-square Degree Two. ...............................95

Figure 5.14 Comparison of the Numerical Analysis and the Simulation Results in a Very Slow Rayleigh Fading Channel. .......................96
Acknowledgment

I would like to express my sincerest gratitude to my thesis advisor, Professor Cyril S. Leung, for his continuous guidance and insight throughout this thesis and my education at UBC. His critical reviews and many constructive suggestions were essential to the completion of this work. He spent precious time each week discussing ideas and details with me, even during his sabbatical leave. Without his guidance, this work could not have been accomplished. This work is supported by the NSERC Grant OGP0001731, and a grant from Cadence Design Systems, Inc.

I would also like to thank my industrial advisor, Mr. Siavash M. Alamouti, who is of Mabuhay Networks, Inc., for his constant support and direction. As well, many thanks to Mr. John Lundell and Mr. Vafa Ghazi-Moghadam, both of whom work for Cadence.

Mr. Masayuki Takada, who is of NHK, and Mr. Cyril Valadon, who is of TTPCom Ltd., helped make this work possible. Many thanks to Mr. Vaibhav Dinesh, Ms. Lisa Beckett, Mrs. Barbara Greenway and Ms. Kirsty Barclay for proofreading. Thanks also go to the members of the exam committee, Dr. William G. Dunford and Dr. Robert Schober, for their time and effort.

Fellow students in the communications group have certainly made my studies at UBC memorable and interesting. I would like to thank Farshid Agharebparast, Jian (Colin) Chen, Jian (Lawrence) Chen, Titus Cheung, Vaibhav Dinesh, Maged Elkashlan, Hossam Fattah, Donghui Gu, Cyril Iskander, Mingzhu Li, Hong Nie, Kassim Olawale, Sergio Valenzuela, Chen Wang, Zhipeng Wang, Lijuan Wu, Kaikuan Xie, Fei Yu, Youli Zhuang, and others.

Finally, I would like to thank my large family, especially my wife Xiaoming Zhang. Her true love and continued encouragement played a significant role in the completion of this work.
Chapter 1 Introduction

For ages past, people could only dream of the possibility of transmitting pictures over great distances. The advent of the electronic age turned this dream into practical reality [1]. The world's very first black and white television service was introduced in Berlin, Germany, in March 1935. In 1954, a color television system was launched in the United States by the National Television System Committee (NTSC). With developments in computer and image processing technologies, the digital storage and transmission of television is now a reality. After a decade of intense research and development, Digital Television Terrestrial Broadcasting (DTTB) has finally reached the implementation stage. Currently, there are three main DTTB transmission standards [2], including Integrated Services Digital Broadcasting - Terrestrial (ISDB-T), which was developed in Japan and is the subject of this thesis.

1.1 Digital Television Terrestrial Broadcasting

Digital Television (DTV) transmission involves digitally sending audio and video components to a destination. It can provide high definition television (HDTV) in a 16:9 TV set, as well as Dolby digital surround sound. There are three commonly used methods for DTV transmission: satellite broadcasting, cable delivery, and terrestrial broadcasting (DTTB). DTTB is the newest of the three digital television transmission methods.

DTTB is composed of a transmission side and a reception side. Figure 1.1 depicts the composition of the transmission side. There are two major parts in the transmission side of a DTTB system: source coding and channel coding. Source coding includes two subsystems: source encoding and multiplexing. Channel coding includes error correction encoding and modulation.
The three dominant DTTB transmission standards are these:

(1) The Advanced Television Systems Committee - Digital Television (ATSC-DTV) standard, developed in the United States [3].

(2) The Digital Video Broadcasting - Terrestrial (DVB-T) standard, developed in Europe [4].

(3) The Integrated Services Digital Broadcasting - Terrestrial (ISDB-T) standard, developed in Japan [5].

The ATSC-DTV, which employs Trellis-coded 8-level Vestigial Side-Band (8-VSB) modulation, was adopted as the DTTB standard by the United States Federal Communications Committee in December 1996. The ATSC-DTV system was originally designed to transmit HDTV over a 6 MHz channel where it could reliably deliver 19.4 Mbps of data throughput. Although the system was developed and tested with a 6 MHz channel, it can be scaled to any of
the television channel bandwidths (6, 7, or 8 MHz) with corresponding scaling in the data capacity. For terrestrial broadcasting, the system is designed to allow the allocation of an additional digital transmitter for each existing NTSC transmitter, with comparable coverage and minimum disturbance to the existing NTSC services in terms of both area and population coverage.

The DVB-T system, which employs Coded Orthogonal Frequency Division Multiplexing (COFDM) modulation, was developed by Digital Video Broadcasting Projects (DVB) in Europe, which is an industry-led consortium with over 300 members in 35 countries. This system was approved by the European Telecommunications Standards Institute (ETSI) as the DTTB standard in February 1997. It was designed for digital video and digital audio distribution, and for the transport of forthcoming multimedia services with an 8 MHz television channel. The 8 MHz channel can support a net bit rate from 4.98 to 31.67 Mbps. However, it can be scaled to any channel bandwidths (6, 7, or 8 MHz) with corresponding scaling in the data capacity. The DVB-T system allows fixed, portable, and mobile receptions, and can withstand the static and dynamic multipath distortion that is associated with long delay.

The ISDB-T system, which employs Band Segmented Transmission Orthogonal Frequency Division Multiplexing (BST-OFDM) modulation, was developed by the Association of Radio Industries and Businesses (ARIB) in Japan. The final draft specifications for ISDB-T was fixed in September 1998, and was adopted by Japan as the new type of broadcasting standard for multimedia services. Originally, it was designed for a 6 MHz television channel, but it can be scaled to any channel bandwidth (6, 7, or 8 MHz). This system was designed to have enough flexibility to deliver digital television and sound programs, and to offer multimedia services in
which various types of digital information, such as video, audio, data and computer programs, are integrated. In addition to typical home-use TV sets, it also aims at providing stable reception through compact, light, and inexpensive mobile receivers.

1.2 Motivation and Objectives

DTTB is now in the implementation stage. DTTB testing services have been available in Europe and North America since November 1998. Many countries have compared the above three systems and announced their final choices, whereas others are still conducting field trials. The ATSC-DTV standard has been chosen by the United States (December 1996), Canada (November 1997), South Korea (November 1997), Taiwan (May 1998), and Argentina (October 1998). The DVB-T standard has the best market penetration in Europe, Niger, Egypt, South Africa, Turkey, Singapore, Hong Kong, India, New Zealand, and Australia. Japan and Brazil have selected ISDB-T as their DTTB standards.

Each country has its own geographic profile, population distribution, spectrum resource, coverage and type of service requirements, and reception conditions. DTTB computer simulation and field trials need to be carried out before a decision on the DTTB standard can be made. Those countries who have made their decisions still need to do simulation and field trials to finalize the system parameters for each city or specific region. Since DTV has a large potential market, as end-user acceptance, many electronic companies, such as Motorola, Sony, Hitachi, and Philips, have started to develop DTV baseband chips.

DTTB system simulation tools are in high demand. It is extremely expensive and time consuming for each researcher or engineer to build his or her own system level simulation model. In addition, there are only a couple of DTTB system simulation tools currently available. One is
Chapter 1 Introduction

Digital Video Broadcasting (DVB-T) Design Library, developed by Cadence Design Systems. The other is Digital TV Transmission Design Library, developed by Agilent Technologies. The 8-VSB modulation scheme used in ATSC-DTV is still facing controversy. Although the techniques used in DVB-T and ISDB-T are quite similar, ISDB-T can support up to three different services in one channel and is, therefore, more sophisticated than DVB-T. This thesis focuses on ISDB-T and builds the ISDB-T System Design Library using the Cadence Signal Processing Worksystem (SPW™), C, and C++. This ISDB-T simulation model has a user friendly interface, and it is suitable for different system configurations.

Although the ISDB-T system is in the implementation stage, only a few ISDB-T performance evaluations and analysis studies are found in the literature. Some computer simulation results and laboratory measurements appear in [6]. To the best of the author's knowledge, there are neither computer simulation results for coherent modulation schemes under Ricean fading, nor any Rayleigh fading simulation results. Some Bit Error Rate (BER) performances of 64QAM-OFDM Viterbi decoding in AWGN and multipath channels, are found in [7]. This thesis provides not only the computer simulation results of ISDB-T for Ricean and Rayleigh, but also some other simulation results which are helpful in evaluating the ISDB-T system performance.

ISDB-T employs a concatenated coding scheme and OFDM. It uses Reed-Solomon coding as the outer coding, and convolutional coding as the inner coding, with convolutional interleaving in between. The target BER for HDTV is $1 \times 10^{-11}$. Such a low value makes it impractical to obtain the BER performance directly using computer simulation. Since no theoretical analysis method exists in the literature that can be applied to the ISDB-T performance evaluation, a semi-analytic method is applied in this study, which is based on the simulation results of
the inner decoding. The semi-analytic results of non-fading and two-path frequency selective non-fading channel agree closely with the simulation results. However, the semi-analytic results of the Ricean channel, and two-path Rayleigh channel, differ by 0.5 dB to 2.5 dB, according to the simulation results (at BER of $1\times10^{-6}$).

### 1.3 Outline of the Thesis

In Chapter 2, an overview of the ISDB-T standard is presented. In Chapter 3, the SPW™ simulation tool is briefly reviewed, and an implementation of the ISDB-T standard using SPW™, C, and C++ is described and validated. A semi-analytic method applied in the performance evaluation is described in Chapter 4. Some results, based on the ISDB-T simulation model and semi-analytic method, are presented and compared in Chapter 5. The error probability performance of ISDB-T is also evaluated. The main contributions and some suggestions for future work are summarized in Chapter 6.
Chapter 2  Overview of ISDB-T

The digital transmission scheme for ISDB-T, designed for flexibility and expandability, was proposed by NHK Science and Technical Research Laboratories in Japan [8]. ISDB-T provides new multimedia broadcasting services for terrestrial networks. It can accommodate HDTV, Standard Definition Television (SDTV), Digital Sound Broadcasting (DSB), and mobile multimedia in a 6 MHz channel bandwidth. ISDB-T is sufficiently resistant to multipath and fading interference to make portable and mobile reception possible. The ISDB-T standard was adopted in September 1998 by ARIB.

Figure 2.1 illustrates a block diagram of the ISDB-T transmitter. Layer A, B, and C correspond to different services.
2.1 Requirements for ISDB-T Transmission

The Moving Picture Experts Group (MPEG) is a working group of ISO/IEC, in charge of the development of standards for the coded digital representation of audio and video. MPEG-2 (ISO/IEC 13818) is a standard for the source coding of television signals. The interface of the ISDB-T transmission system is the MPEG-2 Transport Stream (MPEG-2 TS). The input signals to the ISDB-T system, and the output signals from the ISDB-T system, conform to the MPEG-2
Since ISDB-T accommodates different kinds of services, the system must cover a wide range of requirements that may differ from one service to another. For example, a large transmission capacity is required for HDTV services, whereas mobile reception ability is required for sound and data services that only need a small transmission capacity. The main requirements for ISDB-T transmission include the following:

1. Prior to ISDB-T, two digital transmission standards, ISDB-Satellite (ISDB-S) and ISDB-Cable (ISDB-C), already existed. ISDB-T is designed to be as compatible as possible with ISDB-S and ISDB-C. This ensures that ISDB-T receiver technology has as much similarity as possible with that of satellite and cable. This may cost less for the manufacturers.

2. Since HDTV, SDTV, DSB and data are services with different data rates and Quality of Service (QoS) requirements, ISDB-T should be flexible enough to accommodate different service configurations, and ensure the flexible use of transmission capacity.

3. ISDB-T should have an optimum area coverage for stationary reception with a rooftop antenna, and should be sufficiently resistant to multipath and fading so that portable and mobile reception is possible.

4. ISDB-T should have separate receivers dedicated to television, sound, and data, as well as fully integrated receivers.

5. ISDB-T signals should be transmitted in terrestrial single frequency networks (SFN). SFN is a frequency synchronous broadcast and covers a large service area with a single frequency.

6. ISDB-T should be suitable for 6, 7, and 8 MHz television channels, and be expandable
Chapter 2 Overview of ISDB-T

In order to meet the above requirements, ISDB-T uses the following main techniques.

2.2 Main Techniques of ISDB-T

2.2.1 Concatenated Coding

In MPEG-2 coding, a very high compression ratio, normally higher than 50:1, is required in order to accommodate more information in the limited channel bandwidth. This means that the digital signal is more vulnerable to disturbances in the transmission path. The target BER for HDTV is $1 \times 10^{-11}$. Therefore, a strong channel coding scheme is required for forward error correction.

Concatenated coding is a technique that combines relatively simple codes to form a powerful coding system for achieving high performance, and large coding gain with low decoding complexity. [10] and [11] show that concatenated coding can yield a high coding gain. The ISDB-T standard employs a concatenated coding scheme in which the inner code is a punctured convolutional code ($k = 7, G_1 = 171_{OCT}, G_2 = 133_{OCT}$), and the outer code is a shortened Reed-Solomon (RS) code ($204, 188, t = 8$). The decoding of the concatenated coding is carried out in two stages: the soft or hard decision decoding of the inner code, and the hard decision decoding of the outer code.

2.2.2 Band Segmented Transmission - OFDM

A terrestrial broadcasting channel is prone to multipath propagation. Reflections of the transmitted signal from obstacles are superimposed asynchronously on the directly received
Chapter 2 Overview of ISDB-T

signal. These reflected signals are time delayed and can cause harmful interference. OFDM is a multi-carrier modulation scheme, and its basic principle is to split a high-rate data stream into a number of low-rate data streams, which are transmitted simultaneously over a number of subcarriers. Since the symbol duration increases for low-rate parallel subcarriers, the relative amount of time dispersion caused by multipath delay spread is decreased. Inter-symbol interference is eliminated almost completely by introducing a guard interval between OFDM symbols. In the guard interval, the OFDM symbol is cyclically extended to avoid inter-carrier interference. OFDM has excellent characteristics for resisting multipath reflections [12] [13] in order that an effective SFN can be constructed [14].

A Band Segmented Transmission - OFDM (BST-OFDM) transmission scheme is used in ISDB-T. BST refers to dividing a 6, 7, or 8 MHz TV channel bandwidth into 13 OFDM segments, with each OFDM segment having a bandwidth of 6/14, 7/14, or 8/14 MHz, respectively. The rest of the bandwidth is used for the gaps between TV channels. Hereafter, the OFDM segment is simply referred to as a segment.

2.2.3 Hierarchical Transmission and Partial Reception

There are up to 13 segments in an ISDB-T channel. Each segment can be independently assigned transmission parameters, such as the modulation scheme for OFDM subcarriers (DQPSK, QPSK, 16QAM, or 64QAM); the coding rate of an inner convolutional code (1/2, 2/3, 3/4, 5/6, or 7/8); and the length of time interleaver. Hierarchical transmission allows the transmission of different contents with different QoS. This is achieved by transmitting segment groups that have different transmission parameters in a channel. A maximum of three segment groups can be transmitted simultaneously in a channel. Each segment group, also
referred to as a layer, corresponds to a different service, for example, HDTV, SDTV, DSB, and data.

Figure 2.2 illustrates the hierarchical transmission and partial reception in ISDB-T. A wideband ISDB-T system contains up to 13 segments and supports HDTV, SDTV, and multimedia services, whereas a narrowband ISDB-T system contains only one or three segments and carries DSB and data services.

The service intended for partial reception is always allocated in the centre of the channel (segment number 0), and it is treated as one layer. By limiting the range of frequency interleaving for partial reception within that segment, it is possible to separate that segment independently from the remaining segments in the transmitted signal. In such a way, the partial reception of services contained in a transmission channel can be obtained by using a narrowband receiver, which has a bandwidth of one or three segments.

The top left subfigure shows one service that occupies the whole bandwidth. Frequency interleaving is carried out within 13 segments. The top right subfigure shows two services. Frequency interleaving is done independently for these two services, since Layer A is intended for partial reception. The bottom left subfigure shows three services. Frequency interleaving for Layer A is carried out separately. The two bottom right subfigures show the narrowband services, which can be received by both wideband and narrowband receivers.

A wideband receiver can decode both a wideband signal and a narrowband signal. A narrowband receiver can pick up the central segment signal in a case where the central segment is dedicated to partial reception.
Chapter 2 Overview of ISDB-T

Figure 2.2 Illustration of Hierarchical Transmission and Partial Reception in ISDB-T

(Examples: 13 segments for television services, 1 or 3 segments for audio services)
2.2.4 Multiplex Frame and OFDM Frame

The input to the ISDB-T system is the MPEG-2 Transport Stream Packet (MPEG-2 TSP). After the RS encoder, it is changed to ISDB-T Transmission Transport Stream Packet (Transmission TSP), which has a length of 204 bytes. Figure 2.3 shows their structures. Null TSP has the same size as that of Transmission TSP. Null TSPs are inserted into a stream of Transmission TSPs to form a stream of Interfaced TSPs. Null TSPs are discarded in the processing (right after the splitter in Figure 2.1) and are not transmitted.

<table>
<thead>
<tr>
<th>Sync</th>
<th>MPEG-2 transport MUX data</th>
<th>Parity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 byte</td>
<td>187 bytes</td>
<td>16 bytes</td>
</tr>
</tbody>
</table>

MPEG-2 TSP ISDB-T Transmission TSP

Figure 2.3   MPEG-2 TSP and ISDB-T Transmission TSP

There are two frame concepts in ISDB-T: multiplex frame and OFDM frame. Multiplex frame is a frame structure for a stream of Interfaced TSPs. OFDM frame is defined in the time domain and has a duration of 204 symbols' duration plus guard interval. The number of Interfaced TSPs in a multiplex frame can be calculated by

\[
\text{Number of Interfaced TSPs in a Multiplex Frame} = 2^{n-1} \times (1 + R_y)
\]  

(2.1)

where \(n\) is the order of FFT size and \(R_y\) is the Guard Interval Ratio. The derivation of the above equation is found in Appendix A. The relationship of the FFT size and the number of segments is found in Appendix B.
Table 2.1 shows the number of Interfaced TSPs in one multiplex frame. The size of a multiplex frame depends on the mode, guard interval ratio, and the order of the FFT size. The FFT size depends on the number of segments in a system. Four examples are shown, which are 1-segment, 2-segment, 7-segment and 13-segment ISDB-T systems. ISDB-T can operate in three modes, Mode 1, Mode 2 and Mode 3. Different modes correspond to different numbers of subcarriers in a segment. These are 108, 216 and 432 subcarriers, respectively. From Table 2.1, it is clear that the multiplex frame sizes are the same for 7-segment and 13-segment systems, since they have the same FFT size.

Table 2.1  Number of Interfaced TSPs in One Multiplex Frame

<table>
<thead>
<tr>
<th>Guard Interval Ratio (Ry)</th>
<th>Number of Interfaced TSPs in 1-segment System (Mode 1/2/3)</th>
<th>Number of Interfaced TSPs in 2-segment System (Mode 1/2/3)</th>
<th>Number of Transmission TSPs in 7-segment System (Mode 1/2/3)</th>
<th>Number of Transmission TSPs in 13-segment System (Mode 1/2/3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/4</td>
<td>160/320/640</td>
<td>320/640/1280</td>
<td>1208/2560/5120</td>
<td>1280/2560/5120</td>
</tr>
<tr>
<td>1/8</td>
<td>144/288/576</td>
<td>288/576/1152</td>
<td>1152/2304/4608</td>
<td>1152/2304/4608</td>
</tr>
<tr>
<td>1/16</td>
<td>136/272/544</td>
<td>272/544/1088</td>
<td>1088/2176/4352</td>
<td>1088/2176/4352</td>
</tr>
</tbody>
</table>

Table 2.2 shows the number of Transmission TSPs actually transmitted in one OFDM frame. In order to simplify the receiver, ISDB-T has only one system clock, which is for both FFT sampling and Interfaced TSPs counting. The duration of the multiplex frame is synchronous with the OFDM frame by counting the Interfaced TSP (in bit) using a clock that is four times faster than that used for FFT sampling.
Table 2.2 Number of Transmission TSPs in One OFDM Frame (using 1-, 2-, 7-, 13-segment systems as examples)

<table>
<thead>
<tr>
<th>Carrier Modulation</th>
<th>Convolutional Code Rate</th>
<th>Number of Transmission TSPs in 1-segment System (Mode 1/2/3)</th>
<th>Number of Transmission TSPs in 2-segment System (Mode 1/2/3)</th>
<th>Number of Transmission TSPs in 7-segment System (Mode 1/2/3)</th>
<th>Number of Transmission TSPs in 13-segment System (Mode 1/2/3)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1/2</td>
<td>2/3</td>
<td>3/4</td>
<td>5/6</td>
</tr>
<tr>
<td>DQPSK/QPSK</td>
<td>1/2</td>
<td>12/24/48</td>
<td>16/32/64</td>
<td>18/36/72</td>
<td>20/40/80</td>
</tr>
<tr>
<td></td>
<td>2/3</td>
<td>24/48/96</td>
<td>32/64/128</td>
<td>36/72/144</td>
<td>40/80/160</td>
</tr>
<tr>
<td></td>
<td>3/4</td>
<td>84/168/336</td>
<td>112/224/448</td>
<td>126/252/504</td>
<td>140/280/560</td>
</tr>
<tr>
<td></td>
<td>5/6</td>
<td>156/312/624</td>
<td>208/416/832</td>
<td>234/468/936</td>
<td>260/520/1040</td>
</tr>
<tr>
<td></td>
<td>7/8</td>
<td>312/624/1248</td>
<td>312/624/1248</td>
<td>312/624/1248</td>
<td>312/624/1248</td>
</tr>
<tr>
<td>16QAM</td>
<td>1/2</td>
<td>24/48/96</td>
<td>48/96/192</td>
<td>168/336/672</td>
<td>312/624/1248</td>
</tr>
<tr>
<td></td>
<td>2/3</td>
<td>32/64/128</td>
<td>64/128/256</td>
<td>224/448/896</td>
<td>416/832/1664</td>
</tr>
<tr>
<td></td>
<td>3/4</td>
<td>36/72/144</td>
<td>72/144/288</td>
<td>252/504/1008</td>
<td>468/936/1872</td>
</tr>
<tr>
<td></td>
<td>5/6</td>
<td>48/96/192</td>
<td>80/160/320</td>
<td>320/640/1120</td>
<td>520/1040/2080</td>
</tr>
<tr>
<td></td>
<td>7/8</td>
<td>72/144/288</td>
<td>144/288/576</td>
<td>576/1152/2304</td>
<td>1152/2304/4606</td>
</tr>
<tr>
<td>64QAM</td>
<td>1/2</td>
<td>36/72/144</td>
<td>72/144/288</td>
<td>252/504/1008</td>
<td>468/936/1872</td>
</tr>
<tr>
<td></td>
<td>2/3</td>
<td>48/96/192</td>
<td>96/192/384</td>
<td>336/672/1334</td>
<td>624/1248/2496</td>
</tr>
<tr>
<td></td>
<td>5/6</td>
<td>60/120/240</td>
<td>120/240/480</td>
<td>420/840/1680</td>
<td>780/1560/3120</td>
</tr>
<tr>
<td></td>
<td>7/8</td>
<td>63/120/252</td>
<td>126/252/504</td>
<td>441/882/1764</td>
<td>819/1638/3276</td>
</tr>
</tbody>
</table>

For a specific system mode, different services have different transmission parameters (number of segments, modulation scheme, and convolutional code rate); hence, they have a different number of Transmission TSPs in an OFDM frame. The duration of an OFDM frame is fixed for a specific mode and a guard interval setting. Since the duration of a multiplex frame is the same as that of an OFDM frame, it is necessary to insert Null TSPs into a stream of Transmission TSPs in order to form a multiplex frame. The difference between the numbers in Table 2.1 and the numbers in Table 2.2 correspond to the number of Null TSPs.

As an example, Figure 2.4 illustrates the structures of a multiplex frame and an OFDM frame for a two-segment system.
Chapter 2 Overview of ISDB-T

(a) Multiplex Frame Structure (16QAM modulation, 2/3 convolutional coding rate)

(b) Multiplex Frame Structure (64QAM modulation, 7/8 convolutional coding rate)

(c) Same OFDM Frame Structure for (a) and (b)

Figure 2.4 Structures of Multiplex Frame and OFDM Frame
2.3 ISDB-T Transmission Parameters

ISDB-T was developed and tested with a 6 MHz channel bandwidth, but it can be scaled to any other channel bandwidths (7 or 8 MHz) with corresponding variations in the data capacity. Table 2.3 (Table 1-1 in [5]) shows the segment parameters for ISDB-T with 6 MHz. Table 2.4 (Table 1-2 in [5]) shows the transmission parameters for ISDB-T with 6 MHz, whereas Table 2.5 (Table 1-3 in [5]) shows the information rates per segment for ISDB-T with 6 MHz. The tables of the transmission parameters for 7 and 8 MHz channel bandwidths are found in [5]. There are some mistakes in Table 1-4 Segment Parameters for ISDB-T (7 MHz) in [5]. The amendatory table is provided in Appendix C. Appendix D provides some explanations to the above quoted tables.

Table 2.3  Segment Parameters for ISDB-T (6 MHz)

<table>
<thead>
<tr>
<th>Mode</th>
<th>Mode 1</th>
<th>Mode 2</th>
<th>Mode 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bandwidth</td>
<td>6000/14 = 428.57 kHz</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Carrier Spacing</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>108</td>
<td>108</td>
<td>216</td>
</tr>
<tr>
<td>Data</td>
<td>96</td>
<td>96</td>
<td>192</td>
</tr>
<tr>
<td>SP</td>
<td>9</td>
<td>0</td>
<td>18</td>
</tr>
<tr>
<td>CP</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>TMCC</td>
<td>1</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>AC1</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>AC2</td>
<td>0</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>Carrier Modulation</td>
<td>QPSK, 16QAM, 64QAM</td>
<td>DQPSK</td>
<td>QPSK, 16QAM, 64QAM</td>
</tr>
<tr>
<td>Symbols per Frame</td>
<td>204</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Effective Symbol Duration</td>
<td>252μs</td>
<td>504μs</td>
<td>1008μs</td>
</tr>
<tr>
<td>FFT Sample Clock</td>
<td>512/63 = 8.126984 MHz</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Table 2.4 Transmission Parameters for ISDB-T (6 MHz)

<table>
<thead>
<tr>
<th>Mode</th>
<th>Mode 1</th>
<th>Mode 2</th>
<th>Mode 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Segments</td>
<td>$N_s \leq 13$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bandwidth (kHz)</td>
<td>$6000/14 \times N_s + 6000/(14 \times 108)$</td>
<td>$6000/14 \times N_s + 6000/(14 \times 216)$</td>
<td>$6000/14 \times N_s + 6000/(14 \times 432)$</td>
</tr>
<tr>
<td>Segments for DQPSK</td>
<td>$n_d$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Segments for Coherent Modulation</td>
<td>$n_s \ (n_s + n_d = N_s)$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Carrier Spacing</td>
<td>$6000/(14 \times 108) = 3.968kHz$</td>
<td>$6000/(14 \times 216) = 1.9841kHz$</td>
<td>$6000/(14 \times 432) = 0.99206kHz$</td>
</tr>
<tr>
<td>Number of Carriers</td>
<td><strong>Total</strong> $108 \times N_s + 1$</td>
<td><strong>Total</strong> $216 \times N_s + 1$</td>
<td><strong>Total</strong> $432 \times N_s + 1$</td>
</tr>
<tr>
<td></td>
<td><strong>Data</strong> $96 \times N_s$</td>
<td><strong>Data</strong> $192 \times N_s$</td>
<td><strong>Data</strong> $384 \times N_s$</td>
</tr>
<tr>
<td></td>
<td><strong>SP</strong> $9 \times n_s$</td>
<td><strong>SP</strong> $18 \times n_s$</td>
<td><strong>SP</strong> $36 \times n_s$</td>
</tr>
<tr>
<td></td>
<td><strong>CP</strong> $n_s + 1$</td>
<td><strong>CP</strong> $n_s + 1$</td>
<td><strong>CP</strong> $n_s + 1$</td>
</tr>
<tr>
<td></td>
<td><strong>TMCC</strong> $n_s + 5 \times n_d$</td>
<td><strong>TMCC</strong> $2 \times n_s + 10 \times n_d$</td>
<td><strong>TMCC</strong> $4 \times n_s + 20 \times n_d$</td>
</tr>
<tr>
<td></td>
<td><strong>AC1</strong> $2 \times N_s$</td>
<td><strong>AC1</strong> $4 \times N_s$</td>
<td><strong>AC1</strong> $8 \times N_s$</td>
</tr>
<tr>
<td></td>
<td><strong>AC2</strong> $4 \times N_d$</td>
<td><strong>AC2</strong> $9 \times N_d$</td>
<td><strong>AC2</strong> $19 \times N_d$</td>
</tr>
<tr>
<td>Carrier Modulation</td>
<td>DQPSK, QPSK, 16QAM, 64QAM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Symbols per Frame</td>
<td>204</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Effective Symbol Duration</td>
<td>252$\mu$s</td>
<td>504$\mu$s</td>
<td>1008$\mu$s</td>
</tr>
<tr>
<td>Guard Interval</td>
<td>$63\mu$s (1/4), $31.5\mu$s (1/8), $15.75\mu$s (1/16), $7.875\mu$s (1/32)</td>
<td>$126\mu$s (1/4), $63\mu$s (1/8), $31.5\mu$s (1/16), $15.75\mu$s (1/32)</td>
<td>$252\mu$s (1/4), $126\mu$s (1/8), $63\mu$s (1/16), $31.5\mu$s (1/32)</td>
</tr>
<tr>
<td>Frame Duration</td>
<td>$64.26\text{ms (1/4)}, 57.834\text{ms (1/8)}, 54.621\text{ms (1/16), 53.015ms (1/32)}</td>
<td>$128.52\text{ms (1/4)}, 115.668\text{ms (1/8), 109.242ms (1/16), 106.029ms (1/32)}</td>
<td>$257.04\text{ms (1/4)}, 231.336\text{ms (1/8), 218.464ms (1/16), 212.058ms (1/32)}</td>
</tr>
<tr>
<td>Inner Code</td>
<td>Convolutional Code (1/2, 2/3, 3/4, 5/6, 7/8)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Outer Code</td>
<td>Reed-Solomon Code (204, 188)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Table 2.5 Information Rates of OFDM Segment for ISDB-T (6 MHz channel bandwidth)

<table>
<thead>
<tr>
<th>Carrier Modulation</th>
<th>Convolutional Code Rate</th>
<th>Information Rates (kbps)</th>
<th>Guard Interval Ratio 1/4</th>
<th>Guard Interval Ratio 1/8</th>
<th>Guard Interval Ratio 1/16</th>
<th>Guard Interval Ratio 1/32</th>
</tr>
</thead>
<tbody>
<tr>
<td>DQPSK / QPSK</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1/2</td>
<td>12/24/48</td>
<td>280.85</td>
<td>312.06</td>
<td>330.42</td>
<td>340.43</td>
<td></td>
</tr>
<tr>
<td>2/3</td>
<td>16/32/64</td>
<td>374.47</td>
<td>416.08</td>
<td>440.56</td>
<td>453.91</td>
<td></td>
</tr>
<tr>
<td>3/4</td>
<td>18/36/72</td>
<td>421.28</td>
<td>468.09</td>
<td>495.63</td>
<td>510.65</td>
<td></td>
</tr>
<tr>
<td>5/6</td>
<td>20/40/80</td>
<td>468.09</td>
<td>520.10</td>
<td>550.70</td>
<td>567.39</td>
<td></td>
</tr>
<tr>
<td>7/8</td>
<td>21/42/84</td>
<td>491.50</td>
<td>546.11</td>
<td>578.23</td>
<td>595.76</td>
<td></td>
</tr>
<tr>
<td>16QAM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1/2</td>
<td>24/48/96</td>
<td>561.71</td>
<td>624.13</td>
<td>660.84</td>
<td>680.87</td>
<td></td>
</tr>
<tr>
<td>2/3</td>
<td>32/64/128</td>
<td>748.95</td>
<td>832.17</td>
<td>881.12</td>
<td>907.82</td>
<td></td>
</tr>
<tr>
<td>3/4</td>
<td>36/72/144</td>
<td>842.57</td>
<td>936.19</td>
<td>991.26</td>
<td>1021.30</td>
<td></td>
</tr>
<tr>
<td>5/6</td>
<td>40/80/160</td>
<td>936.19</td>
<td>1040.21</td>
<td>1101.40</td>
<td>1134.78</td>
<td></td>
</tr>
<tr>
<td>7/8</td>
<td>42/84/168</td>
<td>983.00</td>
<td>1092.22</td>
<td>1156.47</td>
<td>1191.52</td>
<td></td>
</tr>
<tr>
<td>64QAM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1/2</td>
<td>36/72/144</td>
<td>842.57</td>
<td>936.19</td>
<td>991.26</td>
<td>1021.30</td>
<td></td>
</tr>
<tr>
<td>2/3</td>
<td>48/96/192</td>
<td>1123.43</td>
<td>1248.26</td>
<td>1321.68</td>
<td>1361.74</td>
<td></td>
</tr>
<tr>
<td>3/4</td>
<td>54/108/216</td>
<td>1263.86</td>
<td>1404.29</td>
<td>1486.90</td>
<td>1531.95</td>
<td></td>
</tr>
<tr>
<td>5/6</td>
<td>60/120/240</td>
<td>1404.29</td>
<td>1560.32</td>
<td>1652.11</td>
<td>1702.17</td>
<td></td>
</tr>
<tr>
<td>7/8</td>
<td>63/126/252</td>
<td>1474.50</td>
<td>1638.34</td>
<td>1734.71</td>
<td>1787.28</td>
<td></td>
</tr>
</tbody>
</table>

### 2.4 Block Diagram of ISDB-T

Figure 2.5 shows a basic block diagram of ISDB-T. The ISDB-T standard specifies the transmitter of the baseband transmission system in detail, in order to allow compatibility between pieces of equipment developed by different manufactures. However, only a block diagram of the receiver is provided because the receiver is left open to different implementation solutions.
2.4.1 ISDB-T Transmitter

Figure 2.1 illustrates a block diagram of the ISDB-T transmitter. This section describes the functional blocks in the transmitter and the algorithms used in ISDB-T.

- **Multiplexing.** This block accepts MPEG-2 TSPs as its input, then multiplexes them in a predetermined way so that the receiver can generate the same MPEG-2 TSPs. The Null TSPs (without the Reed-Solomon parity part yet) are inserted into this block.

- **Reed-Solomon Encoder.** The Reed-Solomon (204, 188, t = 8) shortened code is used as the outer encoder. It is applied to each MPEG-2 TSP to generate Transmission TSP. See Figure 2.3 for the TSP illustration. The Reed-Solomon code can correct up to eight-byte errors in a received 204-byte word. The field generator polynomial is \( p(x) = x^8 + x^4 + x^3 + x^2 + 1 \). The code generator polynomial is \( g(x) = (x - 1)(x - \alpha)(x - \alpha^2)\ldots(x - \alpha^{15}) \).

- **Splitter.** The stream of 204-byte Interfaced TSPs is sorted in the splitter block. According to the packet identification information, the Transmission TSPs are routed to Layer A, B, or C while the Null TSPs are discarded. Layer A, B, and C corresponds to different services.
• **Energy Dispersal.** In order to ensure adequate binary transitions, the data is scrambled in accordance with the configuration depicted in Figure 2.6. The polynomial for the Pseudo Random Binary Sequence (PRBS) generator is $p(x) = x^{15} + x^{14} + 1$. The loading of the sequence “100101010000000” into the PRBS registers is initiated at the start of every OFDM frame. The SYNC byte (47\text{HEX}) is not scrambled.

![Figure 2.6 Scrambler / Descrambler Schematic Diagram](image)

• **Byte-wise Interleaving (Convolutional Interleaving).** Figure 2.7 depicts the convolutional byte-wise interleaving where depth $I = 12$, which is then applied to the data stream. The interleaver is composed of 12 branches. Each branch $j$ ($j = 0, 1, 2, ..., 11$) is a First-in First-out (FIFO) shift register, with depth $j \times m$ cells, where $m = 17 = 204 / I$. The cell of FIFO contains one byte. The input and output switches are synchronized. The deinterleaving has the same branches, but in the reverse order.
- *Delay Adjustment for Byte-wise Interleaving (Convolutional Interleaving).* Some delay is caused by byte-wise interleaving in the transmitter and byte-wise deinterleaving in the receiver. This delay is 11 Transmission TSPs. Since different layers have their own transmission parameters, delay adjustment is needed in each layer in order to align the different services. The total delay for each layer is adjusted to one multiplex frame. Table 2.6 shows the delay adjustment for byte-wise interleaving and deinterleaving in terms of the number of Transmission TSPs. (N stands for the number of segments used in the corresponding layer.)
- **Convolutional Encoder.** ISDB-T allows a range of punctured convolutional codes that are based on a rate of 1/2 mother code with a constraint length of $k = 7$. The generator polynomials of the mother code are $G_1 = 171_{OCT}$ for $X$ output and $G_2 = 133_{OCT}$ for $Y$ output. Figure 2.8 (Fig. 3-6 in [5]) shows the mother convolutional code with a rate of 1/2. In addition to this code rate, ISDB-T also allows punctured rates of 2/3, 3/4, 5/6, and 7/8. The puncturing patterns are given by Table 2.7 (Table 3-2 in [5] where “1” means transmitted, and “0” means punctured).
• **DQPSK Bit Interleaving and Signal Mapping.** The serial bit sequence is converted to complex data, according to Figure 2.9 (Fig. 4-3 in [5]). The phase calculation and $\pi/4$ shift DQPSK mapping is shown in Figure 2.10 (Fig. 4-4 in [5]).
**QPSK Bit Interleaving and Signal Mapping.** The serial bit sequence is converted to complex data, according to Figure 2.11.
• **16QAM Bit Interleaving and Signal Mapping.** The serial bit sequence is converted to complex data, according to Figure 2.12 (Fig. 4-8 in [5]).

![Diagram of 16QAM Bit Interleaving and Mapping](image)

Figure 2.12 16QAM Bit Interleaving, Mapping, and the Corresponding Bit Patterns

• **64QAM Bit Interleaving and Signal Mapping.** The serial bit sequence is converted to complex data, according to Figure 2.13 (Fig. 4-10 in [5]).
• **Delay Adjustment for Bit Interleaving.** Bit interleaving and deinterleaving cause the delay of 120 complex data. By adding the appropriate delay, as shown in Table 2.8 (Table 4-1 in [5]), the total delay in the transmitter and the receiver is adjusted to the amount of two OFDM symbols.
Table 2.8 Delay Adjustment for Bit Interleaving and Deinterleaving

<table>
<thead>
<tr>
<th>Modulation</th>
<th>Mode 1</th>
<th>Mode 2</th>
<th>Mode 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>DQPSK / QPSK</td>
<td>384 x N – 240</td>
<td>768 x N – 240</td>
<td>1536 x N – 240</td>
</tr>
<tr>
<td>16QAM</td>
<td>768 x N – 480</td>
<td>1536 x N – 480</td>
<td>3072 x N – 480</td>
</tr>
<tr>
<td>64QAM</td>
<td>1152 x N – 720</td>
<td>2304 x N – 720</td>
<td>4608 x N – 720</td>
</tr>
</tbody>
</table>

- **Synthesis of Hierarchical Burst Stream.** After signal mapping, complex data of each layer are inputted to pre-assigned data segments of each symbol (in the time domain). A data segment is one part of an OFDM segment. The data segment part carries data, whereas the other part carries pilots and system information. If there is a partial reception service, one segment is assigned to the partial reception layer. If there is a differential modulation service, a certain number of segments are assigned to the differential modulation layer. The remaining segments are assigned to the coherent modulation layer. The procedure of synthesis is carried out according to Figure 4-12 Synthesis of Layer-Data Streams in [5].

- **Time Interleaving and Frequency Interleaving.** Time interleaving is carried out immediately following synthesis, which is then followed by frequency interleaving. Time interleaving and frequency interleaving are executed on the data segment, which is defined as a table of addresses for complex data. The data segment corresponds to the data portion of the OFDM segment. The data segment has two dimensions, one for the time domain and the other for the frequency domain. The details of the data segment is shown in Figure 2.14.
Time interleaving is symbol-wise time interleaving. The integer $I$ is a parameter of time interleaving, which can be specified for each layer. The length of time interleaving $L$ for each carrier is given by

$$L = n \times I$$

where $n$ is the number of OFDM subcarriers for each mode. For Modes 1, 2, and 3, $n$ is 96, 192, and 384, respectively.

This figure shows the structure of the data segment, with different segments and symbols interleaved in both the frequency and time domains. The interleaving process is described in detail in the text.
\[ L_j = (5 \times j \times I) \mod 96 \] (2.2)

where \(0 \leq j \leq 95\) for Mode 1; \(0 \leq j \leq 191\) for Mode 2; and \(0 \leq j \leq 383\) for Mode 3.

Since there are some delays in time interleaving and deinterleaving, a delay adjustment is required for each layer. The delay adjustment for each layer is determined so that the total delay is adjusted to an integral number of OFDM frames. Table 2.9 shows the delay adjustment and the total delay in terms of the number of OFDM frames.

Table 2.9 Delay Adjustment for Time Interleaving and Deinterleaving

<table>
<thead>
<tr>
<th>Mode</th>
<th>Number of Symbols for Delay Adjustment</th>
<th>Number of OFDM Frames to be Delayed by Delay Adjustment, Time Interleaving and Deinterleaving</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mode 1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>(2 = \frac{4 \times 95 + 28}{204})</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>(4 = \frac{8 \times 95 + 56}{204})</td>
</tr>
<tr>
<td></td>
<td>16</td>
<td>(8 = \frac{16 \times 95 + 112}{204})</td>
</tr>
<tr>
<td>Mode 2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>(1 = \frac{2 \times 95 + 14}{204})</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>(2 = \frac{4 \times 95 + 28}{204})</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>(4 = \frac{8 \times 95 + 56}{204})</td>
</tr>
<tr>
<td>Mode 3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>(1 = \frac{1 \times 95 + 109}{204})</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>(1 = \frac{2 \times 95 + 14}{204})</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>(2 = \frac{4 \times 95 + 28}{204})</td>
</tr>
</tbody>
</table>

The block diagram of frequency interleaving is shown in Figure 2.15. Inter-segment frequency interleaving is carried out among the segments that have the same modulation scheme (differential modulation or coherent modulation), whether they are in the same layer or different layer. The allocation of complex data before and after inter-segment interleaving is shown in
Figure 4-16 in [5]. Intra-segment frequency interleaving is carried out according to Figure 4-17 in [5]. Intra-segment carrier randomization is shown in Table 4-4 in [5].

![Diagram](image-url)

Figure 2.15 Configuration of Frequency Interleaver

### 2.4.2 ISDB-T Receiver

Figure 2.16 illustrates the block diagram of an ISDB-T receiver. ISDB-T does not specify the details in the receiver.
Chapter 2 Overview of ISDB-T

Figure 2.16 Block Diagram of ISDB-T Receiver
Chapter 3  Description of the SPW Simulation Model

DTTB is a complicated communication system. It is unwise to implement and test a DTTB system right after the design stage, as a small mistake in this stage may lead to a costly loss in the implementation stage. Software simulation seems to be a perfect candidate for solving this problem. A good simulation model can help designers optimize their solutions in order to meet the unique requirements. This chapter describes an implementation of the ISDB-T standard, using SPW™, C, and C++. This simulation model is used to evaluate ISDB-T performance.

3.1 Choice of Simulation Tool

There are many computer languages and simulation tools available. Good tool selection is very important in ISDB-T software implementation. Two factors affect tool selection. The first is to discover which tool is used in the related work as the simulation tool. The second is to find a tool that is suitable for the ISDB-T baseband simulation.

3.1.1 Simulation Tool in the Related Works

The main techniques used in DVB-T are the same as those used in ISDB-T, such as Reed-Solomon code, convolutional code, coherent modulation schemes, and OFDM. Cadence Design Systems has implemented DVB-T using its own simulation tool, SPW™.

3.1.2 Physical Layer Simulation Tool

The ISDB-T, which is implemented, is the specification of channel coding, frame structure, and modulation scheme. It specifies the baseband transmission system and focuses on the physical layer in the communication system. SPW™ is a suitable tool for physical layer
simulation, and easily integrates C and C++ models.

3.2 Brief Overview of SPW™

Signal Processing Worksystem (SPW™) by Cadence is an integrated framework for developing Digital Signal Processing (DSP) and communications products. SPW™ provides all the tools needed to interactively capture, simulate, test, and implement a broad range of DSP designs. Typical applications include digital communication systems, image processing, radar systems, digital audio, and HDTV. SPW™ is used to evaluate various architectural approaches to a design and to develop, simulate, and fine tune DSP algorithms.

The basic SPW™ system consists of seven modules. They are File Manager, Block Diagram Editor (BDE), Signal Calculator (SigCalc), Simulation Manager, Simulation Program Builder (SPB) Simulator, DSP Library, and Design Database. In addition to the basic components, SPW™ also has Optional Tools, Optional Libraries, and Optional Verification Environments. We only provide some introduction to the parts related to our implementation.

3.2.1 File Manager

File Manager is a unified tool that one can use to create and manage SPW™ libraries, to have access to all types of SPW™ data files, and to invoke the various SPW™ tools such as BDE, SigCalc, etc.

3.2.2 Block Diagram Editor

Block Diagram Editor (BDE) is the basic design environment of SPW™. From the BDE, one can place graphical blocks and link them by wires on the screen. Each block is a symbol that
represents an operation, and the interconnecting wires symbolize the flow of signals between blocks. Using BDE, one can concentrate on the high level aspects of designing a signal processing system, rather than on the coding details of the design.

Using BDE, one can also create a design by adding library blocks and connecting them into a signal flow network. This is done graphically, like drawing block diagrams on a piece of paper. Block diagrams of complex systems can be constructed in a hierarchical fashion from the bottom up or from the top down. One can use hierarchy to divide design tasks, to hide the complexity of a design, and to simplify debugging. Careful planning of the design hierarchy helps to build reusable blocks. ISDB-T is a complicated system. This methodology is used frequently throughout the implementation of ISDB-T.

3.2.3 Signal Calculator

Signal Calculator (SigCalc) provides a tool for manipulating digital signals. With the SigCalc, one can create, display, edit, process, and analyze all types of signal waveforms. Signal waveforms can be saved as files to be used as input to a simulation program. One can also read in, display, and analyze the signals generated by the Simulation Program Builder.

3.2.4 Simulation Manager

Simulation Manager can be invoked from File Manager or BDE. It simulates the operation of a signal flow system designed with the BDE. Given a signal processing block diagram and a set of input signals, the simulator determines the output signals of the system over a specified interval. It writes the results into a set of signal files that one can display and analyze in the SigCalc.
3.2.5 Simulation Program Builder Simulator

There are two forms of Simulation Program Builder (SPB): SPB-Interpreted (SPB-I) and SPB-Compiled (SPB-C). SPB-I is the standard simulator provided for all SPW™ users, whereas SPB-C is an optional simulation accelerator. The two simulators are similar in operation and produce the same results when used to simulate the same block diagram.

SPB-I works like an interpretive computer language. The simulator executes blockcode without any precompiling of the program code. The simulation is run either on the local node or a compatible node in the same network. SPB-C works by creating, compiling, and running a C program. A system design is executed considerably faster with SPB-C than with SPB-I. The SPB-C program is run on the local node, another node in the same network, or on any platform with a C compiler.

3.2.6 SPW™ Libraries

SPW™ provides a DSP Library and Optional Libraries. The DSP Library contains blocks which are used in the BDE to build a signal processing system. These blocks are grouped according to their functions. Optional Libraries provide enhanced analysis capabilities and support specific applications. These libraries include a communications library, an interactive simulation library, and a radar and radio frequency library.

There are two main categories of DSP blocks: the functional blocks and the custom-coded blocks. The functional blocks are supplied by the DSP Library. The custom-coded blocks are built by the SPW™ user. If the system includes elements that cannot be modeled effectively by existing blocks in the libraries, the user must create a custom-coded block. To incorporate a
custom-coded block, template files in C/C++ are automatically created in SPW™ by the custom-coded block function. The user can edit the source code to define the functionality of the block. After adding the compiled code to a simulation kernel, a corresponding symbol block is created. One can use the custom-code block the same way as standard library blocks.

3.3 General Information of the ISDB-T Model

The ISDB-T baseband transmission system is implemented in SPW™ 4.6, and the custom-coded blocks are built in C/C++. The entire ISDB-T simulation model is tested in SPW™ 4.6 and SPW™ 4.7, and it is put into the library named isdbt.

3.3.1 SPW™ ISDB-T Model

The SPW™ ISDB-T model is built according to [5]. Everything specified in this standard is included in this model, except for the TMCC information. The TMCC is used to inform the receiver about the transmission parameters, such as the modulation scheme, convolutional coding rate, and so forth. Even if the TMCC is included, the receiver is not able to really use it because of the limitations of SPW™. It is assumed that the receiver knows the TMCC. The ISDB-T model is built with an ideal receiver, meaning it has perfect channel estimation and perfect time and frequency synchronization. Five channels are included in the model: non-fading, two-path frequency selective non-fading, Ricean, Rayleigh, and two-path Rayleigh.

ISDB-T can support up to three layers. Each layer has its own modulation scheme: differential or coherent modulation. The frequency interleaving varies for different modulation schemes. SPW™ is not able to support a zero-length vector implementation. Therefore, in order
to simplify the implementation and to try to accommodate as many service combinations as possible, four SPW™ ISDB-T simulation systems are built into the model, as follows:

- \textit{isdbt/isdbt\_1.system} is for one layer of differential modulation or one layer of coherent modulation. The number of segments can be up to 13.

- \textit{isdbt/isdbt\_2.system} is for two layers of differential modulation or two layers of coherent modulation. The total number of segments can be up to 13.

- \textit{isdbt/isdbt\_3.system} is for three layers of differential modulation or three layers of coherent modulation. The total number of segments can be up to 13.

- \textit{isdbt/isdbt\_3pdc.system} is for three layers. One layer is for partial reception, another for differential modulation, whereas the third one is for coherent modulation. The total number of segments can be up to 13.

### 3.3.2 Editable Parameters in the ISDB-T Model

ISDB-T is a flexible and extendable standard and can support thousands of system configurations. Throughout the setting of the top-level system parameters, the model can meet the above requirements. There are two kinds of top-level system parameters: editable and non-editable. All of the non-editable parameters are changed automatically according to the editable parameters. The editable parameters are listed below.

- \textit{Transmission Mode of ISDB-T} specifies the mode of the system. The value can only be set to 1, 2, or 3.
• **Guard Interval (Ratio)** sets the denominator of the guard interval, and the value can only be 4, 8, 16, or 32. The guard interval is automatically set to 1/4, 1/8, 1/16, or 1/32, respectively.

• **System Bandwidth** can be set to 6, 7, or 8 MHz by this parameter depending on the country.

• **Decoding** is for the viterbi decoding, and can be configured to *soft* or *hard*.

• **Carrier to Noise Ratio** specifies the $C/N$ in dB.

• **Number of Segment(s)** can be set independently in each layer. In the *isdbt/isdbt_1.system* it can be from 1 to 13. In the *isdbt/isdbt_2.system*, it can be from 1 to 12, and the total number of segments in these two layers cannot exceed 13. In the *isdbt/isdbt_3.system* and the *isdbt/isdbt_3pdc.system*, the segments can be from 1 to 11, and the total number of segments in these three layers cannot exceed 13.

• **Modulation** can be set independently in each layer where it specifies the modulation scheme for that layer. The value can either be DQPSK, QPSK, 16QAM, or 64QAM.

• **Convolutional Coding Rate** can be set independently in each layer, with a coding rate of either 1/2, 2/3, 3/4, 5/6, or 7/8.

• **Time Interleaving Length** independently specifies the time interleaving length for each layer. The value can either be 0, 4, 8, or 16 for the Mode 1 system; 0, 2, 4, or 8 for the Mode 2 system; and 0, 1, 2, or 4 for the Mode 3 system.

• **Channel Type** specifies the transmission channel. The channel can be set to non-fading, two-path frequency selective non-fading, Ricean, Rayleigh, or two-path Rayleigh.
• **Delay Spread** is used only for two-path frequency selective non-fading, Ricean, Rayleigh, and two-path Rayleigh channels. It sets the delay time of the second path in micro seconds.

• **Doppler Frequency** is used only for the Ricean, Rayleigh, and two-path Rayleigh channels.

• **Delayed Path Relative Power** is only for the Rayleigh and two-path Rayleigh channels. It sets the ratio of the power of the non-delayed path to the power of the delayed path in dB.

• **Reflected Arm Gain** is only for the two-path frequency selective non-fading and Ricean channels. It sets the ratio of the power of the direct path to the power of the reflected path in dB.

### 3.4 Main Blocks in the ISDB-T Model

The block diagram of the SPW™ ISDB-T model is accessible on the Web. In this section, the SPW™ implementation of the main blocks in the ISDB-T model is discussed. Each one of the following blocks are used in all of the four SPW™ ISDB-T systems (isdbt/isdbt_1.system, isdbt/isdbt_2.system, isdbt/isdbt_3.system, and isdbt/isdbt_3pdc.system), except those otherwise mentioned.

#### 3.4.1 MPEG-2 Source

ISDB-T is used for multimedia services. The interface for ISDB-T is the MPEG-2 Transport Stream with the length of an MPEG-2 Transport Stream Packet (TSP) of 188 bytes. The MPEG-2 source (isdbt/mpeg2_source) block generates an MPEG-2 TSP one at a time.

To reflect the randomness of the source, a noise generator (spb/wng) block is used to produce a sequence of random binary numbers. By adding the SYNC byte \(47_{\text{HEX}} = 71_{10}\) to this random sequence, a vector of 188 bytes is generated.
3.4.2 RS Encoder and Splitter

The RS encoder and splitter (isdbt/rsencoder_splitter) block does vector Reed-Solomon (204, 188, \( t = 8 \)) encoding, then routes the RS encoded Transmission TSPs to the appropriate layers. The RS encoding is accomplished by the vector Reed-Solomon encoder (comm/vrsencoder) block. In one multiplexing frame, the numbers of Transmission TSPs belonging to Layer A, B, and C are defined by the parameters in this block. These parameters inherit the values from the upper level file. Since each Transmission TSP is divided at the end of the synchronization byte, the SYNC byte is moved to the end of each Transmission TSP. This block also outputs the sources of Layer A, B, and C, that are used for calculating the system BERs.

This block is only used in the isdbt/isdbt_3.system and the isdbt/isdbt_3pdc.system. The similar block which is used in isdbt/isdbt_2.system is the RS encoder and splitter - 2 (isdbt/rsencoder_2) block.

3.4.3 Channel Coding

The functions of the channel coding (isdbt/channel_coding) block include energy dispersal, delay adjustment for byte-wise interleaving, byte-wise interleaving, convolutional coding with a rate of 1/2, and puncturing. This block accepts Transmission TSPs as its input, then outputs a convolutional encoded data stream to the modulation (isdbt/modulation) block.

The energy dispersal (isdbt/energy_dispersal) block is a hierarchical block. This block generates Pseudo Random Binary Sequence (PRBS), and then applies them to the input sequence. The initial values of the PRBS registers are loaded at the start of every OFDM frame. Since different system settings may have a different number of TSPs in one OFDM frame, we need a parame-
After layer_tsp_per_frame to control the loading time. This is the parameter in the clock (spb/clock) block. The other clock in the energy dispersal block disables the randomization whenever the SYNC byte appears.

The delay adjustment for byte-wise interleaving and deinterleaving is implemented using a bulk delay (spb/zdel) block. The delay in this block is given by (3.1), which satisfies each system configuration shown in Table 2.6.

\[ D = 1632 \times (12 \times N \times \log_2 S \times R \times 2^{M-1} - 11) \]  

(3.1)

where \( N \) is the number of segments in this layer; \( S \) is the constellation in this layer (\( S = 4 \) for DQPSK and QPSK, \( S = 16 \) for 16QAM, and \( S = 64 \) for 64QAM); \( R \) is the convolutional code rate in this layer; and \( M \) is the system mode (\( M = 1 \) for Mode 1, \( M = 2 \) for Mode 2 and \( M = 3 \) for Mode 3).

Byte-wise interleaving is implemented using an interleave/deinterleave (comm_lib/interleave) block. A convolutional encoder is implemented using a vector convolutional coder (comm_lib/vconv_coder) block. The generator polynomials of the mother code \( G_1 = 171_{OCT} \) for \( X \) output and \( G_2 = 131_{OCT} \) for \( Y \) output, are specified in the ASCII file isdbt/gen_217.

The convolutional encoder puncturing pattern and transmitted sequence is given in Table 2.7. Since the puncturing cannot be implemented based on the existing DSP blocks in SPW™, a C/C++ program is written to form a custom-coded punctured (isdbt/punctured) block. This custom-coded block is specified as puncturing or depuncturing by a parameter pd_mode. It is also used in any coding rate of \( 1/2, 2/3, 3/4, 5/6, \) or \( 7/8 \) by changing the parameter rate.
3.4.4 Modulation

The Modulation (isdbt/modulation) block performs the delay adjustment for bit interleaving, bit interleaving, and signal mapping. Accepting a convolutional coded data stream as its input, this block outputs complex data signals.

The number of bits for the delay adjustment is shown in Table 2.8. This is implemented by a bulk delay (spb/zdel) block. The delay in this block is given by (3.2), which satisfies each system configuration shown in Table 2.8.

\[ D = (2 \times 96 \times N \times 2^M - 1 - 120) \times B \]  
(3.2)

where \( N \) is the number of segments in this layer; \( M \) is the system mode (\( M = 1 \) for Mode 1, \( M = 2 \) for Mode 2, and \( M = 3 \) for Mode 3); and \( B \) is the number of bits represented by a constellation point (\( B = 2 \) for DQPSK and QPSK, \( B = 4 \) for 16QAM, and \( B = 6 \) for 64QAM). The total delay (including the delay in bit interleaving, the delay in bit deinterleaving, and the delay adjustment) is adjusted to the amount of two OFDM symbols.

Bit interleaving in each branch is carried out by a down sample (multirate/downsample) block and a bulk delay block. After including three scalars joined into vector (spb/sjoiv) blocks, two vector join (spb/vjoiv) blocks, and one vector to scalar (multirate/vec2scal) block, this bit interleaving part accommodates all of the schemes shown in Figure 2.9, Figure 2.11, Figure 2.12, and Figure 2.13.

Coherent modulation is implemented using a number to symbol mapping (comm_lib/num2sym) block. The type of coherent modulation scheme (QPSK, 16QAM, or 64QAM) is
specified by the parameter *constellation file*. The constellation files (*isdbt/qpsk*, *isdbt/16qam*, and *isdbt/64qam*) contain the constellation information for coherent modulation schemes (QPSK, 16QAM, and 64QAM, respectively).

Differential modulation, as depicted in Figure 2.9 and Figure 2.10, is implemented using a hierarchical block *isdbt/num2pi4dqpsk*. The output signal of this block at time $i$ is described by

$$out(i) = out(i - 96 \times 2^{M-1} \times N) \times in(i)$$

(3.3)

where $M$ is the system mode ($M = 1$ for Mode 1, $M = 2$ for Mode 2, and $M = 3$ for Mode 3); $N$ is the number of segments in this layer; $out(i - 96 \times 2^{M-1} \times N)$ is the output signal at the time of one OFDM symbol period before; and $in(i)$ is the input signal (in QPSK format) at time $i$.

There is a *complex switch in (spb/cmplx_sw_in)* block at the end of the *modulation* block. This switch selects the coherent or differential signal according to the parameter *modulation in this layer's* value. Note that the power of the complex signal is normalized to 1.

### 3.4.5 Time Interleaving

The time interleaving (*isdbt/time_interleaving*) block accommodates the delay adjustment for time interleaving and deinterleaving, and time interleaving.

The delay adjustment is implemented according to Table 2.9. The delay is specified by a *constant (mdk/const)* block, which inherits the appropriate parameter from the upper level file.

There are up to 13 segments in an ISDB-T system. Inside each segment, each OFDM carrier may have time interleaving with different lengths. For Mode 3, there are 384 OFDM
carriers. Consequently there will be up to 4992 (13 x 384) branches for Mode 3. Using the existing DSP blocks in SPW™ to implement time interleaving appears to be too complicated and impractical. Therefore, a polymorphic custom-coded block *isdb/t ime_interleaver* is built to fulfil this function. This custom-coded block is used in time interleaving or time deinterleaving, as specified by the parameter *id_mode*.

### 3.4.6 Frequency Interleaving

As shown in Figure 2.15, for different system services the configuration of the frequency interleaver may vary. Two hierarchical frequency interleaving blocks, *isdb/frequency_interleaving_3pdc* block for the ISDB-T system *isdb/isdb_3pdc.system*, *isdb/frequency_interleaving* block for the ISDB-T systems *isdb/isdb_1.system*, *isdb/isdb_2.system* and *isdb/isdb_3.system* are implemented.

The inter-segment interleaver, intra-segment carrier rotation, and intra-segment carrier randomization are implemented by three polymorphic custom-coded blocks (*isdb/inter_segment*, *isdb/carrier_rotation*, and *isdb/carrier_random* respectively) according to Figure 4-16, Figure 4-17, and Table 4-4 in [5].

### 3.4.7 OFDM Structure

The *OFDM structure* (*isdb/ofdm_structure_3pdc*, *isdb/ofdm_structure*) blocks form OFDM structure, and perform Inverse Fast Fourier Transform (IFFT). They also add a guard interval in the time domain.

The OFDM segment-frame structure is defined in Section 4.8 of [5]. In the implementation, it is assumed that the receiver knows the TMCC information and has perfect frequency
synchronization. Also, an ideal channel estimation is used. Therefore, the real TMCC, the real scatter pilots (SP), and the continuous pilots (CP) information is not included in the OFDM structure blocks. All of that information is added into the next stage model.

Since there are two different frequency interleaving blocks, two different OFDM structure blocks are implemented: the isdbt/ofdm_structure_3pdc block for the isdbt/isdbt_3pdc.system; and the isdbt/ofdm_structure block for the isdbt/isdbt_1.system, the isdbt/isdbt_2.system, and the isdbt/isdbt_3.system.

The isdbt/ofdm_frame block is a custom-coded block and should integrate TMCC, SP and CP to form the standard defined OFDM structure. Currently, it simply fills those positions with random values.

The Complex IFFT (spb/cmplx_fft) block boosts the input signal power by a factor of $\text{fft\_size}$. In order to keep the output signal power the same as the input signal power, the input complex data is scaled by a factor of $1 / \sqrt{\text{fft\_size}}$.

### 3.4.8 ISDB-T Channel Model

Three types of channel models for DTTB are described in [15], which take non-fading, fading, and multipath conditions into account. Non-fading, Ricean, and two-path Rayleigh channel models are used in [6]. This thesis evaluates the performance of ISDB-T in five different channel models: non-fading, two-path frequency selective non-fading, Ricean, Rayleigh, and two-path Rayleigh. Ideally, the reception conditions within the service area are described by a non-fading channel model, which is based on a direct signal path from transmitter to receiver, overlaid with Additive White Gaussian Noise (AWGN). In order to include the impairments caused by
echoes, the reception conditions are described by two-path frequency selective non-fading, Ricean, Rayleigh, and two-path Rayleigh channel models. Table 3.1 shows the information of these five channel models.

**Table 3.1 Five Channel Models**

<table>
<thead>
<tr>
<th>Channel Name</th>
<th>Path 1</th>
<th>Path 2</th>
<th>Overlaid by AWGN</th>
</tr>
</thead>
<tbody>
<tr>
<td>non-fading</td>
<td>direct path, no fading</td>
<td>--</td>
<td>yes</td>
</tr>
<tr>
<td>two-path frequency</td>
<td>direct path, no fading</td>
<td>echo path, delayed, no fading</td>
<td>yes</td>
</tr>
<tr>
<td>selective non-fading</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ricean</td>
<td>direct path, no fading</td>
<td>echo path, delayed, Rayleigh fading</td>
<td>yes</td>
</tr>
<tr>
<td>Rayleigh</td>
<td>echo path, Rayleigh fading</td>
<td>--</td>
<td>yes</td>
</tr>
<tr>
<td>two-path Rayleigh</td>
<td>echo path, Rayleigh fading</td>
<td>echo path, delayed, Rayleigh fading</td>
<td>yes</td>
</tr>
</tbody>
</table>

The implementation of the above five channel models are elaborated on as follows. The *ISDB-T channel model (isdbt/isdbt_channel)* block is implemented in the time domain, meaning that the channel amplitude and phase are applied to the time domain samples of the ISDB-T model after IFFT. The parameter *channel type* in this block specifies the type of channel. This block performs the following two tasks:

- It multiplies the incoming signal at the input port *in* by the appropriate complex channel gain.

  It then sends the results to the output port *out*.

- It outputs the channel complex weights, *w*₁ and *w*₂, to the appropriate ports.

The output of the *ISDB-T channel model* block at time *i* is given by

\[
out(i) = w_1 \times in(i) + w_2 \times in(i - \Delta) \tag{3.4}
\]

where *in(i)* is the input to the channel at time *i*; *w*₁ is the complex channel weight for the direct
path; \( w_2 \) is the complex channel weight for the second path; and \( \Delta \) is the channel delay for the second path.

- **Non-fading Channel.** In the case of the non-fading channel, the first switch in the complex tri-switch (isdbt/cmplx_sw_in3) block is selected. The input is then sent directly to the output. The channel weights, \( w_1 \) and \( w_2 \), are set to 1 and 0 respectively. Hence, for the non-fading channel

\[
\text{out}(i) = \text{in}(i). \quad (3.5)
\]

- **Two-path frequency selective non-fading Channel.** The output of the two-path frequency selective non-fading channel (isdbt/rummler_vd) block is described by

\[
y(i) = w_{\text{dir}} \times x(i) + w_{\text{del}} \times x(i - \Delta) \quad (3.6)
\]

where \( x(i) \) is the input to this block at time \( i \); \( w_{\text{dir}} \) is the complex channel weight for the direct path; \( w_{\text{del}} \) is the complex channel weight for the second path (delayed path); and \( \Delta \) is the channel delay (in samples) for the second path. The delay value \( \tau \) is given as a parameter in \( \mu \text{s} \). The delay \( \Delta \) in the samples is calculated by multiplying \( \tau \) by the sampling frequency, and rounding up the result to the nearest integer. The weights are calculated as follows

\[
w_{\text{dir}} = \frac{1}{\sqrt{1 + \beta^2}} \quad (3.7)
\]

\[
w_{\text{del}} = \frac{-\beta}{\sqrt{1 + \beta^2}} \quad (3.8)
\]

where \( \beta \) is the reflected arm gain. Note that the channel has been normalized to have
\[ |w_{dir}|^2 + |w_{del}|^2 = 1. \]  

(3.9)

Therefore, in the case of the two-path frequency selective non-fading channel, the output is given by

\[ \text{out}(i) = w_{dir} \times \text{in}(i) + w_{del} \times \text{in}(i - \Delta) \]  

(3.10)

where \( w_{dir} \) and \( w_{del} \) are defined by (3.7), (3.8), and (3.9).

- **Ricean Channel.** The output of the Ricean channel \((\text{isdbt/ricean})\) block is described by

\[ y(i) = w_{dir} \times x(i) + w_{del} \times x(i - \Delta) \]  

(3.11)

where \( x(i) \) is the input to this block at time \( i \); \( w_{dir} \) is the complex channel weight for the direct path; \( w_{del} \) is the complex channel weight for the second path (delayed path); and \( \Delta \) is the channel delay (in samples) for the second path. The delay value \( \tau \) is given as a parameter in \( \mu s \). The delay \( \Delta \) in samples is calculated by multiplying \( \tau \) by the sampling frequency and rounding up the result to the nearest integer. The weights are calculated as follows

\[ w_{dir} = \frac{1}{\sqrt{1 + \beta^2}} \]  

(3.12)

\[ w_{del} = \frac{-\beta}{\sqrt{1 + \beta^2}} \times w_{ray} \]  

(3.13)

where \( \beta \) is the reflected arm gain; \( w_{ray} \) is generated by the Rayleigh flat fade \((\text{comm_lib/rayleigh_flat})\) block and is normalized to 1. Note that the channel is normalized to have
Therefore, in the case of the Ricean channel, the output

\[ \text{out}(i) = w_{\text{dir}} \times \text{in}(i) + w_{\text{del}} \times \text{in}(i - \Delta) \]  

(3.15)

where \( w_{\text{dir}} \) and \( w_{\text{del}} \) are defined by (3.12), (3.13), and (3.14).

- Two-path Rayleigh Channel. The output of the two-path Rayleigh channel (isdbt/rayleigh_sel_fixed) block is also described by

\[ y(i) = w_{\text{dir}} \times x(i) + w_{\text{del}} \times x(i - \Delta) \]  

(3.16)

where \( x(i) \) is the input to this block at time \( i \); \( w_{\text{dir}} \) is the complex channel weight for the direct path; \( w_{\text{del}} \) is the complex channel weight for the second path (delayed path); and \( \Delta \) is the channel delay (in samples) for the second path. The channel weights, \( w_{\text{dir}} \) and \( w_{\text{del}} \), are generated by the Rayleigh flat fade (comm_lib/rayleigh_flat) block. Note that the channel is also normalized to have

\[ E[|w_{\text{dir}}|^2] + E[|w_{\text{del}}|^2] = 1. \]  

(3.17)

The parameter, sampling factor for fixed duration of channel, is used to keep the channel fixed for a given interval. For instance, if the value of this parameter is \( M \), then the channel coefficients for samples from 0 to \( M - 1 \) are set to the value of sample 0. The channel coefficients for the samples from \( M \) to \( 2M - 1 \) are set to have the value of sample \( M \), and so on. This is accomplished using a complex down sample (multirate/cmplx_dnsamp) block and a complex repeat (multirate/cmplx_repeat) block. This feature is useful for testing the performance of the
channel estimator, as explained later in the section about channel compensation.

- **Rayleigh Channel.** A Rayleigh channel model can be obtained from the two-path Rayleigh channel (isdbt/rayleigh_sel_fixed) block. This is accomplished by setting the delayed path relative power parameter to a very low value, such as −100 dB.

- **Adding Complex AWGN.** Complex AWGN is added to the system immediately after the ISDB-T channel model block, which is generated by a complex white noise (comm/cmplx_w_noise) block. Its mean is $\mu = 0$, and its variance, $\sigma^2$, is calculated as follows.

Let $C/N$ denotes the carrier-to-noise ratio. Since the complex noise mean is $\mu = 0$, the variance, $\sigma^2$, specifies the power of the real part of complex noise, and also specifies the power of the imaginary part. The real and imaginary parts of complex noise are independent, so that the complex noise power is

$$N = 2 \times \sigma^2. \tag{3.18}$$

The average power of the data carriers (not including the pilots) is normalized to 1 in the implementation. The standard specifies that the power of the pilots is boosted to $(4/3)^2$ times that of the data carriers. Since, for different system configurations, the ratio of the number of data carriers to the number of pilots is always $96/12$, the average power of the carriers at the front end is given by

$$C = \frac{96 \times 1 + 12 \times (4/3)^2}{108}. \tag{3.19}$$

The ratio, `carrier_to_noise`, which specifies the carrier to noise ratio in dB, is a parameter in the
system. It can be specified as any value, and is defined by

\[
\text{carrier\_to\_noise} = 10 \times \log_{10}(C/N).
\]  

(3.20)

Substituting (3.19) to (3.20), then (3.20) to (3.18), the expression for the complex noise variance \( \sigma^2 \) can be obtained as

\[
\sigma^2 = 0.5 \times 10^{\left(\frac{\text{carrier\_to\_noise}}{10}\right)} \times \frac{96 \times 1 + 12 \times (4/3)^2}{108}.
\]  

(3.21)

3.4.9 Guard Remove and FFT

The guard remove and FFT (isdbt/guard_remove_fft) block removes the guard interval in the time domain and performs the FFT.

The Complex FFT (isdbt/cmplx_fft) block affects the signal power by a factor of \( 1/(\text{fft\_size}) \). In order to keep the output and input signal power the same, one must scale the input signal to the complex FFT block by a factor of \( \sqrt{\text{fft\_size}} \).

3.4.10 Channel Compensation

Channel compensation is done after the FFT, in the receiver side of the frequency domain. To obtain the ideal channel estimation, the channel complex weights are provided as outputs of the ISDB-T channel block. The weights, together with the delay of each path, are used to calculate the impulse response of the channel. The frequency response of the channel needed for frequency domain compensation is the FFT of the impulse response.

The Channel compensation (isdbt/channel\_comp) block is a hierarchical block containing
two major blocks: the *ideal two-path OFDM channel estimator* (isdbt/ideal_fft_est) block, and the *element-wise vector multiply* (spb/complx_vpm) block. The *ideal two-path OFDM channel estimator* block builds the ideal channel estimation vector with $N$ elements (the FFT size). The *element-wise vector multiply* block performs an element-wise multiplication of the channel estimation vector and the output of the FFT (received signal in frequency domain). The outputs of the channel compensation block are the channel compensated received vector ($\text{out}$) and the channel estimation vector ($\omega$).

The *ideal two-path OFDM channel estimator* block is a hierarchical block. Its output is a vector representation of the channel response in the frequency domain. Each element of the vector represents the response at each of the OFDM FFT bins. The frequency response is obtained by taking the FFT of the channel’s impulse response. The impulse response for the two-path channel is an impulse at time 0 (sample 0) with a complex weight of $\delta_1$, and another impulse at time $\tau$ (sample $\Delta$) with the complex weight of $\delta_2$.

The time domain channel coefficients are fed into the ideal channel estimator for every sample. The ideal channel estimator needs to build the frequency response of the channel from the time domain samples. The first step is to build the impulse response vector with $\delta_1$ as its first element and $\delta_2$ as its $\Delta$th element. All other elements are 0, as shown below:

$$\delta = [\delta_1, 0, 0, ..., 0, \delta_2, 0, 0, ..., 0].$$  \hspace{1cm} (3.22)

The size of the vector is the same as the FFT size $N$. The frequency response vector is the FFT of the above vector.
For every frequency domain sample there is a time domain sample; hence, for every FFT there are \( N \) sets of channel coefficients; however, there is only one impulse response per FFT. Therefore, \( \delta_1 \) and \( \delta_2 \) must be selected according to the \( N \) values of \( w_1 \) and \( w_2 \). In fact, due to the guard interval insertion, one must consider \( M \) sets of coefficients where \( M \) is the total number of time domain samples per FFT operation, including the guard interval. Fortunately, the channel changes very slowly, so the channel coefficients are almost constant for the \( M \) samples. Therefore, the simple assumption is made that \( \delta_1 = w_1(0) \) and \( \delta_2 = w_2(0) \), where \( w_1(0) \) and \( w_2(0) \) are the channel coefficients at the start of the FFT interval. Therefore, in the model, the following is assumed

\[
\delta = [w_1(M/2), 0, 0, ... , w_1(M/2), 0, 0, ... , 0].
\]

If the channel does not change slowly, the estimator would be far from ideal. To test how far the receiver is from an ideal receiver, the ISDB-T channel parameter sampling interval for fixed duration of channel (sampling_factor), explained in the section of ISDB-T channel model, is set to \( M \). This ensures that the channel remains constant during the \( M \) time domain samples in one FFT interval.

The channel coefficients, \( w_1 \) and \( w_2 \), are fed into the estimation block, and are separately down-sampled by a factor of \( M \). The down sample blocks are set to select the middle sample \((M/2)\). The selected \( w_1 \) is placed in element 0 of the impulse response vector. The next \((\Delta_2 - 1)\) elements are then set to 0, followed by \( w_2 \) placed in element \( \Delta_2 \) of the vector. Finally, the rest of the elements are set to 0.
The channel impulse response vector is passed to the FFT to obtain the channel frequency response vector, and is then outputted from the output port (w).

3.4.11 OFDM Destructure

The OFDM destructure (isdb/ofdm_destructure_3pdc, isdb/ofdm_destructure) blocks perform OFDM destructuring. The isdb/ofdm_destructure_3pdc block is for the isdb/isdb_3pdc.system, whereas the isdb/ofdm_destructure block is for the isdb/isdb_1.system, the isdb/isdb_2.system, and the isdb/isdb_3.system.

These blocks are hierarchical blocks. The complex vector split vector (spb/cmplx_vsplv) blocks extract data segments from OFDM segments, and then output them.

3.4.12 Frequency Deinterleaving

Frequency deinterleaving is the counterpart of frequency interleaving. In correspondence to two frequency interleaving blocks, one has two frequency deinterleaving blocks. The isdb/frequency_deinterleaving_3pdc block is for the isdb/isdb_3pdc.system, whereas the isdb/frequency_deinterleaving block is for the isdb/isdb_1.system, the isdb/isdb_2.system, and the isdb/isdb_3.system. The major blocks in these hierarchical blocks are the same as those in frequency interleaving, except that these custom-coded blocks are set to work in the deinterleaving mode.

3.4.13 Time Deinterleaving

The time deinterleaving (isdb/time_deinterleaving) block is the inverse of the time interleaving block. The main block in it is the time interleaver (isdb/time_interleaver) block, which is working in the deinterleaving mode.
3.4.14 Demodulation

The demodulation (isdbt/demod_ml) block performs signal demapping, bit deinterleaving, and more delay adjustment. By accepting complex data from a time deinterleaving block, this block outputs hard or soft decision information which is used for Viterbi decoding in the channel decoding block.

**Signal Demapping** For signal demapping, there are two major hierarchical blocks: the coherent demodulation (isdbt/gen_demod) block and the \( \pi/4 \) shift DQPSK demodulation (isdbt/pi4dqpsk_slice) block. The appropriate block is selected automatically, according to the system parameter setting.

In the coherent demodulation block, the input complex signal is demodulated into bits. There are two demodulation modes: hard and soft decision. In the hard decision mode, this block generates hard bits (1s and -1s) which correspond to the best (maximum likelihood) decision, selecting the constellation point with the minimum distance from the received complex signal. These constellation points are shown in Figure 2.11, Figure 2.12, and Figure 2.13 for QPSK, 16QAM, and 64QAM, respectively. Note that a ‘zero’ bit is represented by ‘1’ and a ‘one’ bit is represented by ‘-1’. This is the requirement of the Viterbi decoder in order to perform soft decision decoding.

In the soft decision mode, instead of assigning a ‘1’ or ‘-1’ to a particular bit, this block generates a soft decision variable, producing the correct soft decision for that particular bit when compared with a zero threshold level. It then outputs the soft information.

Here, the 16QAM constellation is selected to demonstrate the soft decision algorithm. The
Chapter 3 Description of the SPW Simulation Model

input complex signal shall be

\[ r = r_R + j r_I \]  \hspace{1cm} (3.24)

where \( r_R \) and \( r_I \) are the real and imaginary parts of \( r \), respectively. The soft decision variables for bits \( b_0, b_1, b_2, \) and \( b_3 \) are as follows:

- \( b_0 = r_R \): when compared with a zero threshold, \( b_0 \) is 0 when \( r_R > 0 \), and \( b_0 \) is 1 when \( r_R < 0 \).
- \( b_1 = r_I \): when compared with a zero threshold, \( b_1 \) is 0 when \( r_I > 0 \), and \( b_1 \) is 1 when \( r_I < 0 \).
- \( b_2 = |r_R| - 2 \): when compared with a zero threshold, \( b_2 \) is 0 when \( (|r_R| - 2) > 0 \), and \( b_2 \) is 1 when \( (|r_R| - 2) < 0 \).
- \( b_3 = |r_I| - 2 \): when compared with a zero threshold, \( b_3 \) is 0 when \( (|r_I| - 2) > 0 \), and \( b_3 \) is 1 when \( (|r_I| - 2) < 0 \).

Similarly, we can derive the soft decision variables for the bits in QPSK and 64QAM. Figure 3.1 summarizes the soft decision variables in QPSK, 16QAM, and 64QAM.

For both hard and soft decision modes, the estimated channel coefficient for a given subcarrier is used in the demodulation process. \( \alpha \) is the channel estimation for a given subcarrier; all the constellation points (for a hard decision mode) and the threshold levels (for a soft decision mode) must be multiplied by \( |\alpha|^2 \).

Similar to the coherent modulation block, the \( \pi/4 \) shift DQPSK demodulation block delivers either hard bits or soft decision information. The first part of this block decodes QPSK from DQPSK signals, using the following algorithm
**Chapter 3 Description of the SPW Simulation Model**

**QPSK Constellation**
Soft Decision Variables:
- \( b_0 = r_R \)
- \( b_1 = r_i \)

**16QAM Constellation**
Soft Decision Variables:
- \( b_0 = r_R \)
- \( b_1 = r_i \)
- \( b_2 = |r_R| - 2 \)
- \( b_3 = |r_i| - 2 \)

**64QAM Constellation**
Soft Decision Variables:
- \( b_0 = r_R \)
- \( b_1 = r_i \)
- \( b_2 = |r_R| - 4 \)
- \( b_3 = |r_i| - 4 \)
- \( b_4 = |r_R| - 4 - 2 \)
- \( b_5 = |r_i| - 4 - 2 \)

---

**Figure 3.1** Soft Decision Variables in QPSK, 16QAM and 64QAM
\[ z(i) = in^*(i - 96 \times 2^{M-1} \times N) \times in(i) \] (3.25)

where \( M \) is the system mode (\( M = 1 \) for Mode 1, \( M = 2 \) for Mode 2 and \( M = 3 \) for Mode 3); \( N \) is the number of segments in this layer; \( in^*(i - 96 \times 2^{M-1} \times N) \) is the conjugate of the input signal at the time of one OFDM symbol period before; and \( in(i) \) is the input signal at time \( i \).

The second part of the \( \pi/4 \) shift DQPSK demodulation block uses \( z(i) \) as its input, then generates hard bits or soft decision information based on the same algorithm, as in the QPSK case.

**Bit Deinterleaving** Bit deinterleaving is the counterpart of bit interleaving. The algorithm it uses is straightforward.

**More Delay Adjustment** As described in Section 3.4.4, the total delay (including the delay in bit interleaving, the delay in bit deinterleaving, and the delay adjustment) is adjusted to the amount of two OFDM symbols. The input to the vector Viterbi decoder (\texttt{comm\textunderscore lib/vviterbi}) block is a vector with the size of 3264 bits. The total bits for the period of time for two OFDM symbols (including the bits accommodated in the two OFDM symbols and the depunctured bits) may not be integral times of 3264 bits. Therefore, in order to let the vector Viterbi decoder block work well, more delay adjustment is needed. To simplify the implementation, more delay (equivalent to 202 OFDM symbols) is added here, as shown in (3.26), to make the whole delay one OFDM frame (equivalent to 204 OFDM symbols).

\[ D = 202 \times 96 \times N \times 2^{M-1} \times B \] (3.26)

where \( N \) is the number of segments in this layer; \( M \) is the system mode (\( M = 1 \) for Mode 1, \( M = 2 \) for Mode 2, \( M = 3 \) for Mode 3).
for Mode 2, and $M = 3$ for Mode 3); and $B$ is the number of bits represented by a constellation point ($B = 2$ for DQPSK and QPSK, $B = 4$ for 16QAM, and $B = 6$ for 64 QAM).

### 3.4.15 Channel Decoding

The functions of the *channel decoding* (*isdbt/channel_decoding*) block include depuncturing, Viterbi decoding, byte-wise deinterleaving, energy redispersal, and Reed-Solomon decoding. The input to this block is the hard or soft decision information from the *demodulation* block. Likewise, there are two outputs from this block. One port (*out*) outputs the final RS decoded bits, which is the final system output. The other port (*rs*) outputs the bits just before the RS decoder, which is used for testing the system BER before the RS decoder.

**Depuncturing** The major block in this part is the custom-coded *punctured* (*isdbt/punctured*) block, which is working in the *depunctured* mode. This block sets the value of the positions, which were originally occupied by the punctured bits, to 0s.

**Viterbi Decoding** The vector *Viterbi decoder* (*comm_lib/vviterbi*) block performs the vector soft Viterbi decoding.

**Byte-wise Deinterleaving** The implementation of *byte-wise deinterleaving* is the same as that of *byte-wise interleaving*, except that the *interleaver/deinterleaver* block works in the *deinterleave* mode.

**Energy Redispersal** The *energy redispersal* (*isdbt/energy_re_dispersal*) block descrambles the input data stream. This block is almost the same as the *energy dispersal* block, except that there is a delay for the PRBS in this block. The delay amount (*in frames*) is the total system delay in this layer, which is given by
\[ D_{\text{total}} = D_{\text{byte}} + D_{\text{bit}} + D_{\text{time}} \]  \hspace{1cm} (3.27)

where \( D_{\text{byte}} \) is the delay related to byte-wise interleaving (1 OFDM frame); \( D_{\text{bit}} \) is the delay related bit interleaving (1 OFDM frame); and \( D_{\text{time}} \) is the delay related to time interleaving (the appropriate number of OFDM frames, given by Table 2.9). This ensures that the PRBS is applied to the valid bits of multimedia services.

**RS Decoding** RS decoding is implemented by using the vector Reed-Solomon decoder (comm/vrsdecoder) block.

### 3.4.16 Display Results

BER versus Carrier-to-Noise ratio (\( C/N \)) is the most frequently-used quality criterion in the performance evaluation of DTTB. Symbol Error Rate (SER) and Word Error Rate (WER) versus \( C/N \) are also useful in the evaluation. Therefore, BER, SER, and WER at the input and at the output of the RS decoding are recorded in the simulation.

To reach a relatively stable BER for a specific system configuration, the simulation time may vary from several hours to several days. Simply recording the BER at the end of a simulation, however, does not convince one that this is the proper BER. By taking advantage of the SPW™ interactive simulation library, one can build a hierarchical display results (isdbt/display_result) block. This block takes the outputs from channel decoding and RS encoder & splitter as its input, then displays the real-time BERs on the screen.

The major blocks in display results are two hierarchical error counter (isdbt/error_count) blocks and some interactive simulation library blocks. Error counter compares the decoded data
stream and the reference source data stream, then outputs the BER/SER/WER, total error bits/symbols/words and total valid bits/symbols/words. The delay parameter in display results is also given by (3.27).

3.5 Running of the Simulation

The ISDB-T simulation model can be run in both SPW™ 4.6 and SPW™ 4.7. In each ISDB-T simulation system (isdbt/isdbt_1.system, isdbt/isdbt_2.system, isdbt/isdbt_3.system, or isdbt/isdbt_3pdc.system), there is a parameter window. The editable parameters can be changed according to the standard or be specified by the user. By reconfiguring the editable parameters, this simulation model accommodates all of the possible system configurations specified by the standard.

Using the SPB-C simulation engine, and setting the “Number of Samples” to a large value, the simulation keeps running and displaying the real-time simulation results until the number of samples runs out, or the simulation is interrupted by the user. The real-time simulation results include:

- **Final Data** - the RS decoded bit in a layer at time \( t \)
- **Delay Bits** - the total delay of bits in a layer
- **Valid Bits** - the total valid RS decoded bits in a layer up to time \( t \)
- **Bit Errors before RS** - the total error bits before RS decoding in a layer up to time \( t \)
- **BER before RS** - the BER before RS decoding in a layer at time \( t \)
- **SER before RS** - the SER before RS decoding in a layer at time \( t \)
• **WER before RS** - the WER before RS decoding in a layer at time \( t \)

• **Final Bit Errors** - the total error bits after RS decoding in a layer up to time \( t \)

• **Final BER** - the BER after RS decoding in a layer at time \( t \)

• **Final SER** - the SER after RS decoding in a layer at time \( t \)

• **Final WER** - the WER after RS decoding in a layer at time \( t \)

### 3.6 Validation of the ISDB-T Model

In order to ensure that the simulation model truly reflects the performance of the communication system, it is very important that the simulation model is set up properly and accurately.

#### 3.6.1 Validation of the Building Block

ISDB-T is a complicated communication system, and the number of building blocks in this simulation model is huge. It is necessary to make sure that each block works correctly, especially the custom-coded blocks.

The existing blocks in the SPW™ library are examined before they are added to the simulation model.

For the custom-coded blocks, much more attention must be paid to ensure that they reflect the ISDB-T specification perfectly. Figure 3.2 shows the diagram that is used to validate the *time interleaving* (isdbt/time_interleaving) block. The *source* block generates a random binary data stream, and *sink_1* and *sink_2* record the outputs from *source* and *time interleaving*, respectively.
SigCalc is used to verify that the bits from source are located in the right positions after time interleaving. Moreover, the comparison block gives the BER of this simple transmission system. The BER of 0 in a high enough C/N means that both time interleaving and time deinterleaving work fine. This method is also applied to the validation of the following custom-coded blocks: intra-segment carrier randomization (isdbt/carrier_random), intra-segment carrier rotation (isdbt/carrier_rotation), inter-segment frequency interleaving (isdbt/inter_segment), OFDM frame (isdbt/ofdm_frame), and punctured (isdbt/punctured).

![Block Diagram for Time Interleaving Validation](image)

**Figure 3.2** Block Diagram for Time Interleaving Validation

### 3.6.2 Validation of the Structure of ISDB-T Model

The ISDB-T model is flexible. It has 14 editable parameters and supports up to three layers. Each layer has its own transmission scheme. Therefore, this model supports all of the possible system configurations. Different layers may also have different data rates and different delays (in bits). Since there are splitters and syntheses in the data stream, different layers must operate synchronously to ensure that the entire model works properly. There are a lot of hierarchical blocks in this model. Each hierarchical block has its parameters, and each basic block in a
hierarchical block also has its own parameters. These parameters inherit their values from the parameters in the upper level blocks, and all of them are related to the editable parameters. An error within any parameter may lead to strange system performance. In the worst case, a simulation cannot run.

Testing each possible parameter combination is impossible. The method of testing used takes one editable parameter at a time. One editable parameter is selected, while the other editable parameters are set randomly. After this, one can set $C/N$ to a high enough value, then check to see if the BER is at 0 for each possible value of that chosen editable parameter. Using this method, all editable parameters can be tested. Therefore, it is assumed that the structure of the ISDB-T model is properly built.

3.6.3 Performance of the ISDB-T Model with No Coding

To further validate the ISDB-T model, the performances of the ISDB-T model with no coding in a non-fading channel are presented, and compared with the published theoretical results in this section.

In [16], the BER of QPSK with Gray encoding is given exactly by

$$\text{BER}_{QPSK} = \frac{1}{2} \text{erfc}\left( \frac{E_b}{\sqrt{N_0}} \right)$$

(3.28)

where $E_b$ is the transmitted signal energy per bit, and $N_0/2$ is the noise power spectral density.

For QPSK, $C/N = 2(E_b/N_0)$, then
where $C/N$ is the carrier to noise ratio. [16] says that DQPSK is approximately 2.3 dB poorer in performance than QPSK. The theoretical approximate BER for DQPSK, therefore, becomes

$$BER_{DQPSK} = \frac{1}{2} \text{erfc} \left( \sqrt{\frac{C}{2N}} \times 10^{-0.23} \right).$$

Turning off the RS coding and convolutional coding gives the performances of the ISDB-T model with no coding for DQPSK and QPSK modulation schemes. Figure 3.3 illustrates three theoretical curves of (3.28), (3.29), and (3.30), plus two simulation results of the ISDB-T model for DQPSK and QPSK.
From Figure 3.3, one can conclude that the performances of the ISDB-T model with no coding for DQPSK and QPSK are the same as the theoretical results.

In [17], the theoretical BER performance of coherently-detected Gray encoding 16QAM in the non-fading channel environment is given by

$$\text{BER}_{16\text{QAM}} = \frac{3}{8} \text{erfc} \left( \sqrt{0.4 \frac{E_b}{N_o}} \right) - \frac{9}{64} \text{erfc}^2 \left( \sqrt{0.4 \frac{E_b}{N_o}} \right). \quad (3.31)$$

For 16QAM, \( C/N = 4(E_b/N_o) \), then

$$\text{BER}_{16\text{QAM}} = \frac{3}{8} \text{erfc} \left( \sqrt{0.1 \frac{E_b}{N_o}} \right) - \frac{9}{64} \text{erfc}^2 \left( \sqrt{0.1 \frac{E_b}{N_o}} \right). \quad (3.32)$$

For the case of 64QAM, \( C/N = 6(E_b/N_o) \), from the Equation (18) in [18], the BER performance of 64QAM in the non-fading channel environment is given by

$$\text{BER}_{64\text{QAM}} = \frac{7}{24} \sum_{i=1}^{4} \text{erfc} \left( (2i-1) \sqrt{\frac{C}{42N}} \right). \quad (3.33)$$

By turning off the RS coding and the convolutional coding, the performances of the ISDB-T model result with no coding for both 16QAM and 64QAM modulation schemes. Figure 3.4 illustrates three theoretical curves of (3.31), (3.32), and (3.33), plus two simulation results of the ISDB-T model for both 16QAM and 64QAM.

From Figure 3.4, it is clear that the performances of the ISDB-T model with no coding for 16QAM and 64QAM are the same as the theoretical results.
In summary, the performances of the ISDB-T model with no coding are exactly the same as the theoretical results.

### 3.6.4 Performance of the ISDB-T Model with Convolutional Coding

Some ISDB-T performance simulation results (before RS decoder) appear in [6]. In order to further validate the ISDB-T simulation model, the results from this simulation model are compared with those in [6]. Figure 3.5 illustrates the comparison in different modulation schemes.

[6] does not describe its simulation model in detail. It is impossible to know which demodulation scheme it uses, which convolutional decoding scheme it uses, and so on. To determine which curve is better is unfair.
Chapter 3 Description of the SPW Simulation Model

System Parameters:
Mode: 1
Guard Interval: 1/8
System Bandwidth: 6 MHz
Number of Segments: 1
Convolutional Code Rate: 3/4
Time Interleaving Length: $I = 0$
Channel: non-fading

Figure 3.5 BER Performance of the ISDB-T Model with Convolutional Coding in A Non-fading Channel. (Solid curves are simulation results obtained in this study, and dashed curves are results from [6].)

In Figure 3.5, the solid curves are the simulation results used in this thesis, and the dashed curves are the results from [6]. The slopes of the curves from the ISDB-T simulation model used in this thesis with high $C/N$ are almost the same as those in [6], while those used in this thesis with low $C/N$ are deeper than those in [6]. The cause of this difference is the assumption made on the TMCC information in this thesis. In [6], the TMCC information is transmitted and used to inform the receiver. After decoding the TMCC, the receiver knows which transmission scheme the transmitter is using. If the receiver has a decoding error in the TMCC, the BER would be higher. However, in the simulation model used here, it is assumed that the receiver knows the TMCC information. Therefore, in the lower $C/N$ part, the slopes of the curves in this study are deeper, since there is no TMCC decoding errors in this simulation model.
Chapter 4  Performance Analysis of ISDB-T

An important issue in evaluating a communication system is to consider its error performance. Performance evaluation techniques usually fall into three categories: software simulation method, theoretical analysis method, and field trial. Chapter 3 described the SPW™ simulation model for ISDB-T. In this chapter, a semi-analytic method for the ISDB-T performance evaluation is provided.

A simple block diagram of ISDB-T is illustrated by Figure 4.1. ISDB-T employs a concatenated coding scheme. Reed-Solomon coding is used as its outer coding, and convolutional coding is used as its inner coding, with convolutional interleaving in between. Also, following channel coding, it employes bit-wise interleaving, time interleaving, and frequency interleaving. All of these interleavings are finite.

![Simple Block Diagram of ISDB-T](image)

Figure 4.1  Simple Block Diagram of ISDB-T

The performance evaluation of ISDB-T applying simulation method is impractical when the BER is in the order of $1 \times 10^{-11}$. The theoretical analysis of ISDB-T is not an easy task, especially when finite interleavings are used.
Some methods of theoretical analysis for the concatenated coding system are proposed. An analytic upper bound on the performance of concatenated coding schemes with a small interleaver size in a non-fading channel appears in [19]. However, the upper bound in [19] is 1 to 2 dB differing from the simulation result. A semi-analytic approach, based on a Markov chain model, for the performance evaluation of concatenated coding schemes in a non-fading channel, is proposed in [20]. It shows that the semi-analytical results are within 0.1 dB from the simulation results. Therefore, a similar approach is applied in this thesis to evaluate the performance of ISDB-T.

4.1 Markov Chain Models

The finite interleavings shown in Figure 4.1 make it very difficult to theoretically analyze the performance of ISDB-T. A Markov chain model is used to model the dash-line box in Figure 4.1. The errors, produced by the Viterbi decoder (in the absence of convolutional interleaving), or the errors at the output of the convolutional deinterleaver, are modeled using a Markov chain. The transition probabilities of the Markov chain are calculated using the results from the simulation of the dash-line box component. Once the Markov chain is characterized, the performance of ISDB-T can be evaluated analytically.

Figure 4.2 shows the three-state Markov chain model, which is used to represent the correlation of the symbol errors at the input of the RS decoder. The symbol errors at the output of the Viterbi decoder are bursty. The three states are listed below:

- "G" is the state in which a symbol is correctly decoded.
- "B1" is the state in which a symbol is incorrectly decoded right after a previous correct symbol.
• “B2” is the state for the second and consecutive error symbols after a correct symbol.

A decoding error occurs in both “B1” and “B2” states. No error occurs in the “G” state. With two bad states, it is possible to model the fact that the probability of a symbol being incorrectly decoded depends not only on whether the previous symbol was correctly decoded, but also on the number of previous consecutive errors. The three-state Markov chain model requires the estimation of only three out of six parameters (\(P_{gg}, P_{gb1}, P_{blg}, P_{blb2}, P_{b2g}, P_{b2b2}\)).

![Three-State Markov Chain Model](image)

In order to provide better matches to the error patterns at the input of the RS decoder, a five-state Markov chain model is introduced (see Figure 4.3). The five states are listed below:

• “G1” is the state in which a symbol is correctly decoded immediately following a previous error symbol.

• “G2” is the state for the second and consecutive correct symbols following an error symbol.

• “B1” is the state in which a symbol is incorrectly decoded immediately following a previous correct symbol.

• “B2” is the state for the second error symbols.
"B3" is the state for the third and consecutive error symbols following a correct symbol.

The five-state Markov chain model requires the estimation of only five out of 10 parameters (Pg2g2, Pg2b1, Pg1g2, Pg1b1, Pb1g1, Pb1b2, Pb2g1, Pb2b3, Pb3g1, Pb3b3).

\[
T = \begin{bmatrix}
    P_{gg} & P_{gb1} & 0 \\
    P_{bg1} & 0 & P_{b1b2} \\
    P_{b2g} & 0 & P_{b2b2}
\end{bmatrix}
\]

\text{(4.1)}

4.2 Performance Analysis of ISDB-T

The RS decoder in ISDB-T can correct up to eight symbol errors in a codeword with 204 symbols. For the RS decoder, the probability of incorrect decoding or decoding failure can be calculated when the distribution of the number of symbol errors in the RS codewords is known; the distribution is obtained using the algorithm proposed in [21]. This algorithm is applied in the analysis of ISDB-T. The technique is described below.

For the three-state Markov chain model, the transition matrix is
Chapter 4 Performance Analysis of ISDB-T

The steady state probabilities of being in the good state ("G", i.e. \( P_G \)), and the bad states ("B1" and "B2", i.e. \( P_{B1} \) and \( P_{B2} \)) are calculated as

\[
\begin{bmatrix}
P_G \\
P_{B1} \\
P_{B2}
\end{bmatrix} = \begin{bmatrix}
1 & 1 & 1 \\
P_{gb1} & -1 & 0 \\
0 & P_{b1b2} & -P_{b2g}
\end{bmatrix}^{-1} \times \begin{bmatrix}
1 \\
0 \\
0
\end{bmatrix}.
\] (4.2)

Whenever in a bad state, a symbol error occurs. In order to keep track of the number of times the system enters the bad states, a counter \( D \) is appended to the corresponding transition probabilities. The modified transition matrix becomes

\[
T(D) = \begin{bmatrix}
P_{gg} & P_{gb1} \times D & 0 \\
P_{b1g} & 0 & P_{b1b2} \times D \\
P_{b2g} & 0 & P_{b2b2} \times D
\end{bmatrix}.
\] (4.3)

The symbolic matrix, \( T(D) \), is raised to the power of 204, resulting in the matrix

\[
[T(D)]^{204} = \begin{bmatrix}
Q_{gg}(D) & Q_{gb1}(D) & Q_{gb2}(D) \\
Q_{b1g}(D) & Q_{b1b1}(D) & Q_{b1b2}(D) \\
Q_{b2g}(D) & Q_{b2b1}(D) & Q_{b2b2}(D)
\end{bmatrix}
\] (4.4)

where, for example, the coefficient of \( D^i \) in \( Q_{gb2}(D) \) represents the probability of starting from the state "G", and ending in the state "B2", after 204 symbols with \( i \) symbol errors. Therefore, the probability of having exactly \( i \) symbol errors in an RS codeword of 204 symbols, is given by the coefficient of \( D^i \) in the polynomial \( Z(D) \), which is defined as
Chapter 4  Performance Analysis of ISDB-T

\[ Z(D) = [P_G \ P_{B1} \ P_{B2}] \times \begin{bmatrix} Q_{gg}(D) + Q_{gb1}(D) + Q_{gb2}(D) \\ Q_{b1g}(D) + Q_{b1b1}(D) + Q_{b1b2}(D) \\ Q_{gb2}(D) + Q_{b2b1}(D) + Q_{b2b2}(D) \end{bmatrix}. \]  \hspace{1cm} (4.5)

For the five-state Markov chain model, the transition matrix is given as

\[
T = \begin{bmatrix}
P_{g2g2} & 0 & P_{g2b1} & 0 & 0 \\
P_{g1g2} & P_{g1b1} & 0 & 0 & 0 \\
0 & P_{b1g1} & 0 & P_{b1b2} & 0 \\
0 & P_{b2g1} & 0 & 0 & P_{b2b3} \\
0 & P_{b3g1} & 0 & 0 & P_{b3b3}
\end{bmatrix}. \]  \hspace{1cm} (4.6)

The steady state probabilities of being in the good states ("G1" and "G2"), and the bad states ("B1", "B2", and "B3") are calculated as

\[
P_{G2} = \begin{bmatrix}
1 & 1 & 1 & 1 & 1 \\
-P_{g2b1} & 1-P_{g1b1} & 0 & 0 & 0 \\
P_{g2b1} & P_{g1b1} & -1 & 0 & 0 \\
0 & 0 & P_{b1b2} & -1 & 0 \\
0 & 0 & 0 & P_{b2b3} & -P_{b3g1}
\end{bmatrix}^{-1} \begin{bmatrix}
1 \\
0 \\
0 \\
0 \\
0
\end{bmatrix}. \]  \hspace{1cm} (4.7)

In compliance with the three-state model, a counter \( D \) is appended to record the number of times the system enters any of the bad states. The modified transition matrix is written as

\[
T(D) = \begin{bmatrix}
P_{g2g2} & 0 & P_{g2b1} \times D & 0 & 0 \\
P_{g1g2} & 0 & P_{g1b1} \times D & 0 & 0 \\
0 & P_{b1g1} & 0 & P_{b1b2} \times D & 0 \\
0 & P_{b2g1} & 0 & 0 & P_{b2b3} \times D \\
0 & P_{b3g1} & 0 & 0 & P_{b3b3} \times D
\end{bmatrix}. \]  \hspace{1cm} (4.8)
The symbolic matrix $T(D)$ is raised to the power of 204 resulting in the following matrix

$$[T(D)]^{204} = \begin{bmatrix}
Q_{g2g2}(D) & Q_{g2g1}(D) & Q_{g2b1}(D) & Q_{g2b2}(D) & Q_{g2b3}(D) \\
Q_{g1g2}(D) & Q_{g1g1}(D) & Q_{g1b1}(D) & Q_{g1b2}(D) & Q_{g1b3}(D) \\
Q_{b1g2}(D) & Q_{b1g1}(D) & Q_{b1b1}(D) & Q_{b1b2}(D) & Q_{b1b3}(D) \\
Q_{b2g2}(D) & Q_{b2g1}(D) & Q_{b2b1}(D) & Q_{b2b2}(D) & Q_{b2b3}(D) \\
Q_{b3g2}(D) & Q_{b3g1}(D) & Q_{b3b1}(D) & Q_{b3b2}(D) & Q_{b3b3}(D)
\end{bmatrix}, \quad (4.9)$$

and the corresponding polynomial $Z(D)$ is given by

$$Z(D) = \left[ P_{G2} P_{G1} P_{B1} P_{B2} P_{B3} \right] \left[ \begin{bmatrix}
Q_{g2g2}(D) + Q_{g2g1}(D) + Q_{g2b1}(D) + Q_{g2b2}(D) + Q_{g2b3}(D) \\
Q_{g1g2}(D) + Q_{g1g1}(D) + Q_{g1b1}(D) + Q_{g1b2}(D) + Q_{g1b3}(D) \\
Q_{b1g2}(D) + Q_{b1g1}(D) + Q_{b1b1}(D) + Q_{b1b2}(D) + Q_{b1b3}(D) \\
Q_{b2g2}(D) + Q_{b2g1}(D) + Q_{b2b1}(D) + Q_{b2b2}(D) + Q_{b2b3}(D) \\
Q_{b3g2}(D) + Q_{b3g1}(D) + Q_{b3b1}(D) + Q_{b3b2}(D) + Q_{b3b3}(D)
\end{bmatrix} \right]. \quad (4.10)$$

Knowing $Z(D)$, the ISDB-T performance can be evaluated. The probability of the RS codeword error, $P_w$, is shown by the following equation

$$P_w = \sum_{i=9}^{188} z_i \quad (4.11)$$

where $z_i$ is the coefficient of $D^i$ in the polynomial $Z(D)$; $z_i$ represents the probability of having $i$ symbol errors in an RS codeword. Assuming the worst case scenario where an incorrectly decoded word has eight more symbol errors than the received word, the probability of symbol error, $P_s$, is given as

$$P_s = \sum_{i=9}^{180} \left( \frac{z_i \times (i + 8)}{188} \right). \quad (4.12)$$
Assuming that one symbol error corresponds to three incorrect bits, the probability of bit error, $P_b$, is given as

$$P_b = \sum_{i = 9}^{180} \left( \frac{z_i \times (i + 8)}{188} \times \frac{3}{8} \right).$$ (4.13)

### 4.3 Validation of the Markov Chain Models

To provide the validation of the three-state Markov chain model in a non-fading channel, Figure 4.4 presents the probability of having $n$ error symbols in an RS codeword. The dashed lines correspond to the results from the Markov chain model (the coefficient of $D^i$ in the polynomial $Z(D)$), and the graphical symbols correspond to the simulation results at the input of the RS decoder. The figure indicates that the three-state Markov chain model provides an excellent match to the simulation results. Other simulations show that the three-state Markov chain model is still valid for other system parameters in a non-fading channel.
System Parameters:
Mode: 1
Guard Interval: 1/8
System Bandwidth: 6 MHz
Number of Segments: 1
Modulation: 16QAM
Convolutional Code Rate: 1/2
Channel: non-fading

Figure 4.4 Distribution of the Number of Symbol Errors in the RS Codewords at the Input of the RS Decoder in A Non-fading Channel.

The case of a two-path frequency selective non-fading channel is presented in Figure 4.5. Similar to the non-fading case, the results based on the three-state Markov chain model matches closely with the simulation results. It is also valid for other system parameters.
Figure 4.5 Distribution of the Number of Symbol Errors in the RS Codewords at the Input of the RS Decoder in A Two-path Frequency Selective Non-fading Channel.

Since the three-state Markov chain model does not provide a good match to the simulation results in the Ricean, Rayleigh, and two-path Rayleigh channels, a five-state Markov chain model is introduced. However, the five-state Markov chain model provides only a slight improvement over the three-state Markov chain model.

To provide the validation of the five-state Markov chain model, Figure 4.6 and Figure 4.7 present the probability of having $n$ error symbols in a RS codeword for Ricean and two-path Rayleigh channels, respectively. The dashed lines correspond to the results from the Markov chain model, and the solid lines correspond to the simulation results. These figures indicate that the five-state Markov chain model does not provide an excellent match to the simulation results.
Chapter 4 Performance Analysis of ISDB-T

System Parameters:
Mode: 1
Guard Interval: 1/8
System Bandwidth: 6 MHz
Number of Segments: 1
Modulation: DQPSK
Convolutional Code Rate: 1/2
Time Interleaving Length: 1 = 16
Channel: Ricean
Delay Spread: 15 μs
Doppler Frequency: 70 Hz
Ratio of Main to Delayed Signal: 3 dB

Figure 4.6 Distribution of the Number of Symbol Errors in the RS Codewords at the Input of the RS Decoder in A Ricean Channel.

Figure 4.7 Distribution of the Number of Symbol Errors in the RS Codewords at the Input of the RS Decoder in A Two-path Rayleigh Channel (Fd = 200 Hz).

Figure 4.8 shows the validation curves for other two-path Rayleigh channel cases having
the same system parameters as Figure 4.7, but a different Doppler frequency. The figures show that, as the Doppler frequency decreases, the analytic and simulation curves drift further apart. Similar behavior is observed in the Ricean and Rayleigh cases.

![Graph showing the distribution of the number of symbol errors in the RS codewords at the input of the RS decoder in a two-path Rayleigh channel.](image)

**Figure 4.8** Distribution of the Number of Symbol Errors in the RS Codewords at the Input of the RS Decoder in a Two-path Rayleigh Channel (Fd = 70 Hz and Fd = 7 Hz).

At very high Doppler frequencies, the channel varies dramatically with small fade duration. This results in small bursts of symbol errors at the input of the RS decoder (as in a non-fading channel). The bad states in the three- or five-state Markov chain model can easily capture these kinds of error bursts. However, at very low Doppler frequencies, the channel undergoes slow variations, and the average fade duration is extremely long, causing a large number of symbol errors during the fading periods. The bad states in the three- or five-state Markov chain model are not appropriate for modeling these kinds of long bursts of symbol errors.
Chapter 5 Numerical Results

Based on the SPW™ ISDB-T simulation model described in Chapter 3, and the semi-analytic method discussed in Chapter 4, the performance of ISDB-T in five different channel conditions is evaluated in this chapter. BER, SER, and WER are the metrics used in performance evaluation, as BER is the most frequently-used metric in this thesis. Comparisons of the semi-analytic results with the simulation results are provided.

Simulation results show that for different system modes and different system bandwidths, the system error probability performances are the same. They also reflect that for a different number of segments in a system, the system error probability performances are the same level as for a non-fading channel.

5.1 Comparison of BER at the Output of the Viterbi Decoder with BER at the Output of the RS Decoder with Different Modulation Schemes in a Non-fading Channel

The BER at the output of the Viterbi decoder, and the BER at the output of the RS decoder, are observed in order to assess the advantage of concatenated coding.

Figure 5.1 shows the curves obtained for different modulation schemes. For each modulation scheme, the BER decreases appreciably as $C/N$ increases beyond a certain value. At a BER of $1 \times 10^{-5}$, the curves for the concatenated coding systems show a coding gain of 3 dB when compared to the one with only convolutional coding. Since the target BER of a DTTB system for HDTV is $1 \times 10^{-11}$, the coding gain is expected to be large at the BER of $1 \times 10^{-11}$. 

83
Chapter 5 Numerical Results

System Parameters:
Mode: 1
Guard Interval: 1/8
System Bandwidth: 6 MHz
Number of Segments: 1
Convolutional Code Rate: 3/4
Time Interleaving Length: 1 = 0
Channel: non-fading

As the modulation scheme changes from 64QAM to 16QAM, from 16QAM to DQPSK or QPSK, the BER performance improves, but the system data rate falls.

5.2 Effect of Convolutional Interleaving in A Non-fading Channel

The task of convolutional interleaving is to scatter the symbol error bursts introduced by the Viterbi decoder, resulting in the errors spreading out to the other RS codewords. In the absence of convolutional interleaving, the error bursts in one RS codeword easily exceeds the error correcting capability of the RS code.

Figure 5.2 illustrates the simulation and the semi-analytic results. The curves show that the semi-analytic results are within 0.1 dB when compared with the simulation results. Other simulations performed under different system parameter settings also show that the differences
between the semi-analytic and the simulation results are within 0.1 dB in a non-fading channel.

The above figure indicates that the convolutional interleaving in this case provides approximately 1 dB improvement at the BER of $1 \times 10^{-11}$. This, however, introduces some delay in the system.

Figure 5.3 shows the BER, SER, and WER performance with convolutional interleaving in a non-fading channel. The semi-analytic results are within 0.1 dB from the simulation results.
5.3 Effect of Convolutional Code Rate in a Two-path Frequency Selective Non-fading Channel

A convolutional coding scheme based on a rate of 1/2 mother code is used in ISDB-T. It also allows punctured rates of 2/3, 3/4, 5/6, and 7/8. The performance with different code rates in a two-path frequency selective non-fading channel is illustrated by Figure 5.4.

The curves indicate that the semi-analytic results are within 0.1 dB of the simulation results for each convolutional code rate. Similar observations are made with simulations for different system parameter settings in a two-path frequency selective non-fading channel.

From the figure, it is clear that as the code rate decreases from 7/8 to 1/2, the BER performance improves. For example, at a BER of $1 \times 10^{-11}$, the coding gain improvements are 2
Chapter 5 Numerical Results

5.4 Effect of Time Interleaving Length in A Ricean Channel

Time interleaving is used to spread error bursts in the time domain. Time interleaving does not affect the BER performance of the system in non-fading and two-path frequency selective non-fading channels since these channels are time invariant. However, time interleaving affects the system performance in Ricean, Rayleigh and two-path Rayleigh channels.

Figure 5.5 illustrates the system performance for different time interleaving lengths. As expected, the deeper the time interleaving, the better the BER performance, and the longer the delay. In this case, the compensations for time interleavings of \( I = 0, 4, 8, \) and 16 are 0, 2, 4, and 8 OFDM frames, respectively.
Chapter 5 Numerical Results

System Parameters:
Mode: 1
Guard Interval: 1/8
System Bandwidth: 6 MHz
Number of Segments: 1
Modulation: 64QAM
Convolutional Code Rate: 5/6
Channel: Ricean
Delay Spread: 15 μs
Doppler Frequency: 70 Hz
Ratio of Main to Delayed: 1 dB

Figure 5.5 Comparison of BER for Different Time Interleaving Lengths.

Figure 5.6 illustrates a comparison of semi-analytic and simulation results in a Ricean channel. At a BER of $1 \times 10^{-6}$, the semi-analytic and the simulation results differ by 0.2 dB. As discussed in Section 4.3, the five-state Markov chain model does not represent the input to the RS decoder perfectly. Therefore, it is clear from the curves that the difference becomes larger as $C/N$ increases.
5.5 Effect of Doppler Frequency in A Two-path Rayleigh Channel

Doppler frequency is an important parameter in Ricean, Rayleigh and two-path Rayleigh channels. The channel varies much more frequently as the Doppler frequency increases. Hence, the deep fading period does not last as long.

Figure 5.7 illustrates the effect of Doppler frequency in a two-path Rayleigh channel. The curves show that the error probability performance improves as the Doppler frequency increases. At a BER of $1 \times 10^{-4}$, the 200 Hz case outperforms the 7 Hz case by 2.5 dB.
System Parameters:

- Mode: 1
- Guard Interval: 1/8
- System Bandwidth: 6 MHz
- Number of Segments: 1
- Modulation: DQPSK
- Convolutional Code Rate: 1/2
- Time Interleaving Length: I = 16
- Channel: two-path Rayleigh
- Delay Spread: 15 μs
- Ratio of Main to Delayed: 0 dB

Figure 5.7 Comparison of Different Doppler Frequencies in a Two-path Rayleigh Channel.

Figure 5.8 depicts the comparison of semi-analytic and simulation results for different Doppler frequencies. At a BER of $1 \times 10^{-6}$, for Doppler frequencies of 7 Hz, 25 Hz, 70 Hz, and 200 Hz, the corresponding curves differ by 2.5 dB, 1.5 dB, 0.85 dB, and 0.25 dB, respectively.

From the figures, it is clear that the semi-analytic results do not agree well with the simulation results in a two-path Rayleigh channel, especially at smaller Doppler frequencies.
Chapter 5 Numerical Results

System Parameters:
Mode: 1
Guard Interval: 1/8
System Bandwidth: 6 MHz
Number of Segments: 1
Modulation: DQPSK
Convolutional Code Rate: 1/2
Time Interleaving Length: 1 = 16
Channel: two-path Rayleigh
Delay Spread: 15 µs
Ratio of Main to Delayed: 0 dB

Figure 5.8 Comparison of the Semi-analytic and the Simulation Results in A Two-path Rayleigh Channel.

5.6 Effect of Number of Segments in A Two-path Rayleigh Channel

The ISDB-T system can have 1 to 13 segments. Inter-segment frequency interleaving is carried out within the segments, which have the same modulation type (differential modulation or
coherent modulation). The more segments a system has, the deeper the inter-segment frequency interleaving is carried out. The number of segments in a system does not affect the BER performance in a non-fading channel; however, it does affect the performance in fading channels.

Figure 5.9 illustrates the effect of the number of segments in a two-path Rayleigh fading channel. The BER performance of a 13-segment system is slightly better than that of a one-segment system.

![Comparison of Different Number of Segments in A Two-path Rayleigh Channel](image)

**Figure 5.9** Comparison of Different Number of Segments in A Two-path Rayleigh Channel.

### 5.7 Effect of Doppler Frequency in A Rayleigh Channel

The effect of Doppler frequency in a Rayleigh channel is similar to that in a two-path Rayleigh channel. Figure 5.10 illustrates the simulation results and the semi-analytic results with a Doppler frequency of 1 Hz, whereas Figure 5.11 illustrates the simulation results and the semi-analytic results with a Doppler frequency of 70 Hz.
Chapter 5 Numerical Results

Figure 5.10 Comparison of the Semi-analytic and the Simulation Results in A Rayleigh Channel.

Figure 5.11 Comparison of the Semi-analytic and the Simulation Results in A Rayleigh Channel.

It is clear that the semi-analytic results do not agree with the simulation results in both cases. As the Doppler frequency decreases from 70 Hz to 1 Hz, the difference between the simula-
tion results and the semi-analytic results becomes further apart.

5.8 Numerical Analysis for Very Slow Rayleigh Fading

As seen in the previous sections, the Markov chain-based semi-analytic method does not provide accurate results with the simulations for fading channels, especially at low Doppler frequencies. A numerical analysis for a very slow Rayleigh fading channel is presented in this section.

The channel varies slowly in the very slow Rayleigh fading environment. It is assumed that the channel is fixed during the RS codeword length (after all interleavings). For each $C/N$ of the very slow Rayleigh fading channel, the probability of the actual $C/N$ being a certain value is given by the following central chi-square degree two distribution

$$P_{C/N}(y) = \frac{1}{2\sigma^2} \exp\left(\frac{-y}{2\sigma^2}\right)$$  \hspace{1cm} (5.1)

where $2\sigma^2$ is the received signal power. If $N$ is normalized to unity, then $2\sigma^2 = C$.

Figure 5.12 shows the BER performance in a non-fading channel. As expected, the semi-analytic results agree with the simulation results quite well. In the numerical analysis, the BER values from the simulations are used when $C/N \leq 20$ dB, whereas the values from the semi-analytic method are used when $C/N > 20$ dB.

The pdf of the central chi-square degree two is given in Figure 5.13. Each curve corresponds to the appropriate $C/N$ in the very slow Rayleigh fading channel.
Chapter 5 Numerical Results

System Parameters:
Mode: 1
Guard Interval: 1/8
System Bandwidth: 6 MHz
Modulation: 64QAM
Convolutional Code Rate: 7/8
Time Interleaving Length: I = 0
Channel: non-fading

Based on the BER performance in a non-fading channel, and the appropriate central chi-square distribution, the numerical analysis of the BER performance for the corresponding $C/N$
can be obtained. Figure 5.14 illustrates the numerical analysis and the simulation results.

![Figure 5.14](image)

**Figure 5.14** Comparison of the Numerical Analysis and the Simulation Results in a Very Slow Rayleigh Fading Channel.

The solid line represents the BER results obtained using numerical analysis (by averaging the non-fading channel BER over the central chi-square degree two pdf). The dashed line corresponds to the simulation results. These results match closely. The slight difference is due to the step size and the accuracy of the BER performance in a non-fading channel.
Chapter 6 Conclusions

In this chapter, the main contributions of this thesis are summarized, and some suggestions for future work are provided.

6.1 Main Contributions

The goal of this thesis is to evaluate the performance of the ISDB-T standard for multimedia services. Two methods are used: software simulation and semi-analytic.

An ISDB-T simulation model is built using SPW™, C, and C++. This model accommodates all the different system configurations defined in the ISDB-T standard by the appropriate choice of the editable parameters.

To overcome the problem of very time-consuming software simulations, a Markov chain-based semi-analytic method is used. The Markov chain model is obtained from the simulation results of the input to the RS decoder.

The BER performances for a number of different system configurations are obtained by using both the simulation and semi-analytic method. It is found that the semi-analytic method provides accurate results in non-fading channels. However, the results are inaccurate in fading channels, especially at low Doppler frequencies. At low Doppler frequencies, a large number of symbol errors occur during the deep fades, and the Markov chain model does not provide appropriate modeling for these bursty error patterns.

Based on the non-fading BER performance, a numerical analysis for the very slow Rayleigh fading case is performed, and the results from which are in good agreement with those
obtained from simulations.

6.2 Suggestions for Future Work

Several aspects of the model are idealized and could be made more realistic. For example,

- Perfect bit, symbol, frame, and frequency synchronization is assumed in the model.
- The control channel specified in the standard could be incorporated in the model.
- A more realistic channel model could be useful.

Future performance studies could include:

- The influence of the guard interval parameter
- The effect of the incorrect decoding of the TMCC information
- Performance in the more realistic channel

The semi-analytic method yields poor results for time varying channels, especially for low Doppler frequencies. A more accurate analytic method could be very useful.
## Glossary

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>8-VSB</td>
<td>Trellis-coded 8-level Vestigial Side-Band</td>
</tr>
<tr>
<td>AC</td>
<td>Auxiliary Channel</td>
</tr>
<tr>
<td>ARIB</td>
<td>Association of Radio Industries and Businesses</td>
</tr>
<tr>
<td>ATSC-DTV</td>
<td>Advanced Television Systems Committee - Digital Television</td>
</tr>
<tr>
<td>AWGN</td>
<td>Additive White Gaussian Noise</td>
</tr>
<tr>
<td>BDE</td>
<td>Block Diagram Editor</td>
</tr>
<tr>
<td>BER</td>
<td>Bit Error Rate</td>
</tr>
<tr>
<td>bps</td>
<td>Bits per Second</td>
</tr>
<tr>
<td>BST-OFDM</td>
<td>Band Segmented Transmission OFDM</td>
</tr>
<tr>
<td>C/N</td>
<td>Carrier to Noise Ratio</td>
</tr>
<tr>
<td>COFDM</td>
<td>Coded Orthogonal Frequency Division Multiplexing</td>
</tr>
<tr>
<td>CP</td>
<td>Continual Pilot</td>
</tr>
<tr>
<td>DQPSK</td>
<td>Differential Quadri Phase Shift Keying</td>
</tr>
<tr>
<td>DSB</td>
<td>Digital Sound Broadcasting</td>
</tr>
<tr>
<td>DSP</td>
<td>Digital Signal Processing</td>
</tr>
<tr>
<td>DTTB</td>
<td>Digital Television Terrestrial Broadcasting</td>
</tr>
<tr>
<td>DTV</td>
<td>Digital Television</td>
</tr>
<tr>
<td>DVB-T</td>
<td>Digital Video Broadcasting - Terrestrial</td>
</tr>
<tr>
<td>ETSI</td>
<td>European Telecommunications Standards Institute</td>
</tr>
<tr>
<td>FFT</td>
<td>Fast Fourier Transform</td>
</tr>
<tr>
<td>HDTV</td>
<td>High Definition Television</td>
</tr>
<tr>
<td>IFFT</td>
<td>Inverse Fast Fourier Transform</td>
</tr>
<tr>
<td>ISDB-C</td>
<td>Integrated Services Digital Broadcasting - Cable</td>
</tr>
<tr>
<td>ISDB-S</td>
<td>Integrated Services Digital Broadcasting - Satellite</td>
</tr>
<tr>
<td>ISDB-T</td>
<td>Integrated Services Digital Broadcasting - Terrestrial</td>
</tr>
<tr>
<td>Acronym</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>ISO/IEC</td>
<td>International Organization for Standardization / International Electrotechnical Commission</td>
</tr>
<tr>
<td>MPEG</td>
<td>Moving Picture Experts Group</td>
</tr>
<tr>
<td>MUX</td>
<td>Multiplex / Multiplexing</td>
</tr>
<tr>
<td>NTSC</td>
<td>National Television System Committee</td>
</tr>
<tr>
<td>OFDM</td>
<td>Orthogonal Frequency Division Multiplexing</td>
</tr>
<tr>
<td>PRBS</td>
<td>Pseudo Random Binary Sequence</td>
</tr>
<tr>
<td>QAM</td>
<td>Quadrature Amplitude Modulation</td>
</tr>
<tr>
<td>QoS</td>
<td>Quality of Service</td>
</tr>
<tr>
<td>QPSK</td>
<td>Quadri Phase Shift Keying</td>
</tr>
<tr>
<td>SDTV</td>
<td>Standard Definition Television</td>
</tr>
<tr>
<td>SER</td>
<td>Symbol Error Rate</td>
</tr>
<tr>
<td>SFN</td>
<td>Single Frequency Network</td>
</tr>
<tr>
<td>SigCalc</td>
<td>Signal Calculator</td>
</tr>
<tr>
<td>SP</td>
<td>Scattered Pilot</td>
</tr>
<tr>
<td>SPB</td>
<td>Simulation Program Builder</td>
</tr>
<tr>
<td>SPB-C</td>
<td>Simulation Program Builder - Compiled</td>
</tr>
<tr>
<td>SPB-I</td>
<td>Simulation Program Builder - Interpreted</td>
</tr>
<tr>
<td>SPW™</td>
<td>Signal Processing Worksystem</td>
</tr>
<tr>
<td>TMCC</td>
<td>Transmission and Multiplexing Configuration Control</td>
</tr>
<tr>
<td>TSP</td>
<td>Transport Stream Packet</td>
</tr>
<tr>
<td>WER</td>
<td>Word Error Rate</td>
</tr>
</tbody>
</table>
Bibliography


Appendix A  Number of Interfaced TSPs

The following is the derivation for the equation of the Number of Interfaced TSPs in a Multiplexing frame:

\[ M \] denotes the system mode \((M = 1, 2, \text{ or } 3 \text{ for mode type } 1, 2, \text{ or } 3 \text{ respectively})\);

\[ n \] denotes the order of FFT;

\( B \) denotes the bandwidth of the system;

\( F_s \) denotes the FFT sampling rate;

\( R_y \) denotes the guard interval ratio in the time domain; and

\( T_s \) denotes the effective symbol duration.

Since the carrier spacing is \( \frac{B}{14 \times 108 \times 2^{(M-1)}} \), then the effective symbol duration is

\[ T_s = \frac{14 \times 108 \times 2^{(M-1)}}{B}. \quad (A.1) \]

Since the FFT size is \( 2^n \), then the FFT sampling rate is

\[ F_s = \frac{2^n}{T_s} = \frac{2^n \times B}{14 \times 108 \times 2^{(M-1)}}. \quad (A.2) \]

There are 204 symbols in one OFDM frame. The duration of one OFDM frame is the same as that of one multiplex frame, which is \((204 \times T_s \times (1 + R_y))\). Since the counting speed of the interfaced TSPs in bits is 4 times faster than the FFT sampling rate, the total bits in one multiplex frame is \((204 \times T_s \times (1 + R_y) \times 4 \times F_s)\). It is obvious that there are 1632 bits in one TSP.
Therefore, the number of Interfaced TSPs in one multiplex frame is

\[
\text{Number of Interfaced TSPs in one Multiplex frame} = \frac{204 \times T_s \times (1 + R_y) \times F_s}{1632}.
\]  

(A.3)

Substituting (A.1) and (A.2) into (A.3), the equation for the number of Interfaced TSPs in one multiplex frame becomes

\[
\text{Number of Interfaced TSPs in One Multiplex Frame} = 2^{(n-1)}(1 + R_y).
\]  

(A.4)
Appendix B  Amendatory Table for Table 2-1 in [5]

Some mistakes appear in Table 2-1, Number of FFT Size for Number of Segments in [5]. They are in the “FFT Sampling Rate” rows. Table B.1 is the amended table. The changes are marked in **Bold and Italics**.

Table B.1 FFT Size for Number of Segments

<table>
<thead>
<tr>
<th>Mode</th>
<th>Number of Segments</th>
<th>Number of Carrier</th>
<th>FFT Size</th>
<th>FFT Order (n)</th>
<th>FFT Sampling Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>109</td>
<td>256</td>
<td>8</td>
<td>1.016 MHz</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1024</td>
<td>9</td>
<td>2.032 MHz</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>8</td>
<td>10</td>
<td>4.063 MHz</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>8.127 MHz</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>217</td>
<td>512</td>
<td>9</td>
<td>1.016 MHz</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1024</td>
<td>10</td>
<td>2.032 MHz</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4.063 MHz</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>8.127 MHz</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>433</td>
<td>649</td>
<td>10</td>
<td>1.016 MHz</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>865</td>
<td>11</td>
<td>2.032 MHz</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4.063 MHz</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>8.127 MHz</td>
</tr>
</tbody>
</table>

The calculation of FFT Sampling Rate for Mode 3 with a FFT Size of 8192 is as follows:

\[
\text{FFT Sampling Rate} = \frac{\text{FFT Size}}{\text{Effective Symbol Duration}} = \frac{8192}{1008 \mu s} = 8.127 \text{ MHz}.
\]

Note that the Number of Carrier in the above table includes one additional continual pilot (CP).
Appendix C  Amendatory Table for Table 1-4 in [5]

Some mistakes appear in Table 1-4, Segment Parameters for ISDB-T (7 MHz) in [5]. They are in the rows of “Carrier Spacing” and “Number of Carriers”. Table C.1 is the amended table. The changes are marked in **Bold and Italics**.

Table C.1  Segment Parameters for ISDB-T (7 MHz)

<table>
<thead>
<tr>
<th>Mode</th>
<th>Mode 1</th>
<th>Mode 2</th>
<th>Mode 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bandwidth</td>
<td>7000/14 = 500kHz</td>
<td>7000 / (14 x 216) = 2.3148 kHz</td>
<td>6000 / (14 x 432) = 0.99206kHz</td>
</tr>
<tr>
<td>Carrier Spacing</td>
<td>7000 / (14 x 108) = 4.629kHz</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Number of Carriers</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>108</td>
<td>108</td>
<td>216</td>
</tr>
<tr>
<td><strong>Data</strong></td>
<td>96</td>
<td>96</td>
<td>192</td>
</tr>
<tr>
<td><strong>SP</strong></td>
<td>9</td>
<td>0</td>
<td>18</td>
</tr>
<tr>
<td><strong>CP</strong></td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td><strong>TMCC</strong></td>
<td>1</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td><strong>AC1</strong></td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td><strong>AC2</strong></td>
<td>0</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>Carrier Modulation</td>
<td>QPSK, 16QAM, 64QAM</td>
<td>DQPSK</td>
<td>QPSK, 16QAM, 64QAM</td>
</tr>
<tr>
<td>Symbols per Frame</td>
<td></td>
<td></td>
<td>204</td>
</tr>
<tr>
<td>Effective Symbol Duration</td>
<td>216µs</td>
<td>432µs</td>
<td>864µs</td>
</tr>
<tr>
<td>Guard Interval</td>
<td>54µs (1/4), 27 µs (1/8), 13.5µs (1/16), 6.75 µs (1/32)</td>
<td>108µs (1/4), 54 µs (1/8), 27µs (1/16), 13.5 µs (1/32)</td>
<td>216µs (1/4), 108 µs (1/8), 54µs (1/16), 27 µs (1/32)</td>
</tr>
<tr>
<td>Frame Duration</td>
<td>55.08ms (1/4), 49.572ms (1/8), 46.818ms (1/16), 45.441ms (1/32)</td>
<td>110.16ms (1/4), 99.144ms (1/8), 93.636ms (1/16), 90.882ms (1/32)</td>
<td>220.32ms (1/4), 198.288ms (1/8), 187.272ms (1/16), 191.762ms (1/32)</td>
</tr>
<tr>
<td>FFT Sample Clock</td>
<td>256 / 27 = 9.481 MHz</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Inner Code</td>
<td>Convolutional Code (1/2, 2/3, 3/4, 5/6, 7/8)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Outer Code</td>
<td>Reed-Solomon Code (204, 188)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Appendix D  Explanation to Table 2.3, Table 2.4 and Table 2.5

Carrier spacing in Table 2.3. One OFDM segment has $1/14$ th of the system bandwidth. For the Mode 1 system, there are 108 carriers in one OFDM segment. Therefore, the carrier spacing for the Mode 1 system is

$$\frac{6000}{(14 \times 108)} = 3.968 \text{ kHz}. \quad (D.1)$$

Bandwidth in Table 2.4. $N_s$ is the number of segments in the system. Each system has one extra CP for the higher edge of the whole bandwidth. Therefore, the bandwidth for the Mode 1 system includes two parts: one for all of the segments, and the other for that extra CP. This is

$$\text{Mode 1 system bandwidth} = \frac{6000}{14} \times N_s + \frac{6000}{(14 \times 108)} \text{ kHz}. \quad (D.2)$$

Number of Transmitting TSPs in Table 2.2. Assume that the system has a parameter setting of 6 MHz, Mode 3, 64QAM, and 7/8 convolution code rate. The calculation for the number of transmitting TSPs per one OFDM frame is as follows (from Table 2.2, it is 252 TSPs)

$$\text{Number of Transmitting TSPs per One OFDM Frame} = \frac{(204 \text{ symbols}) \times (384 \text{ data carriers/symbols}) \times (6 \text{ bits/carrier}) \times (7/8)}{(204 \text{ bytes} \times 8 \text{ bits/byte}) / (1 \text{ TSPs})} = 252 \text{ TSPs}. \quad (D.3)$$

Information Rate in Table 2.2. Let the system be the same as the above one (Guard 1/32). The rate is

$$\frac{(252 \text{ TSPs/OFDM frame}) \times (188 \text{ bytes/TSPs}) \times (8 \text{ bits/byte})}{(212.058 \text{ ms/OFDM frame})} = 1787.28 \text{ kbps}.\quad (D.4)$$