License-Free Communication using Digital Mobile Radio Standards and Spread Spectrum

by

Colter James Alexander McQuay
B.Eng., University of Victoria, 2011

A Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of

MASTER OF APPLIED SCIENCE

in the Department of Electrical and Computer Engineering

License-Free Communication using Digital Mobile Radio Standards and Spread Spectrum

by

Colter James Alexander McQuay
B.Eng., University of Victoria, 2011

Supervisory Committee

Dr. Peter Driessen, Supervisor
(Department of Electrical and Computer Engineering)

Dr. Stephen Harrison, Unit Member
(Department of Electrical and Computer Engineering)

Dr. Belaid Moa, Unit Member
(Department of Electrical and Computer Engineering)

Supervisory Committee

Dr. Peter Driessen, Supervisor
(Department of Electrical and Computer Engineering)

Dr. Stephen Harrison, Unit Member
(Department of Electrical and Computer Engineering)

Dr. Belaid Moa, Unit Member
(Department of Electrical and Computer Engineering)

ABSTRACT

The concept of using spread spectrum and open radio standards to provide license free, short range Peer-to-Peer (P2P) communication is explored. This research makes use of the Time Compression Overlap Add (TC-OLA) algorithm to transparently spread the spectrum of the Digital Mobile Radio (DMR) standard; this allows for reuse of existing hardware, software, and expertise relating to this well established protocol. Initial high level hardware designs of a communication device established the need to implement a proof of concept system which could be validated against Radio Frequency (RF) regulations. This proof of concept system was constructed using a hardware implementation of DMR processed through custom TC-OLA blocks in GNU Radio (GR). A spectral and performance analysis of this system was performed, showing that this approach has several benefits over existing license free communication options.

Contents

List of Tables

List of Figures

Acronyms

3G 3rd Generation

4FSK 4 level Frequency Shift Keying

4PAM 4 Level Pulse Amplitude Modulation

ADC Analog to Digital Converter

AM Amplitude Modulation

API Application Programming Interface

ASCII American Standard Code for Information Interchange

AWGN Additive White Gaussian Noise

BER Bit Error Rate

BPSK Binary Phase Shift Keying

CB Citizens Band

CML-DE9945E CML Microcircuits PMR Common Platform Demonstration board

CMX-7341 CML-CMX7341 PMR/LMR Baseband Protocol Processor

D-STAR Digital Smart Technologies for Amateur Radio

DAC Digital to Analog Converter

dB Decibels

DMR Digital Mobile Radio

dPMR Digital Private Mobile Radio

DSSS Direct Sequence Spread Spectrum

DTD Device to Device

DUSA Downsample Upsample Shift Add

FCC Federal Communication Commission

FDMA Frequency Division Multiple Access

FEC Forward Error Correction

FHSS Frequency Hopping Spread Spectrum

FM Frequency Modulation

FPGA Field Programmable Gate Array

FRS Family Radio Services

FSK Frequency Shift Keying

FSPL Free Space Path Loss

GMRS General Mobile Radio Services

GR GNU Radio

GUI Graphical User Interface

IoT Internet of Things

ISI Intersymbol Interference

ISM Industrial Scientific and Medical

ITU International Telecommunications Union

LCD Liquid Cystal Display

LFM Linear Frequency Modulation

LMR Land Mobile Radio

LoRa Long Range

LOS Line of Sight

LPWAN Low-Power Wide-Area network

LTE Long Term Evolution

MURS Multi User Radio Services

NB Narrowband

NFC Near Field Communication

NXDN NXDN (NEXEDGE)

OBW Occupied Bandwidth

OFDM Orthogonal Frequency Division Multiplexing

P25 Apco 25/Project 25

P2P Peer-to-Peer

PDF Probability Density Function

PM Phase Modulation

PMR Private Mobile Radio

PRBS Pseudo Random Bit Sequence

RBW Resolution Bandwidth

RC Raised Cosine

RF Radio Frequency

RFID Radio Frequency Identification

RRC Root Raised Cosine

SDR Software Defined Radio

SNR Signal to Noise Ratio

TC-OLA Time Compression Overlap Add

TDMA Time Division Multiple Access

TETRA Terrestrial Trunked Radio

USRP Universal Software Radio Peripheral

VBW Video Bandwidth

WB Wideband

ACKNOWLEDGEMENTS

I would like to thank:

Dr. Peter Driessen, for all of the support, guidance and great theoretical conversations/explanations over the years.

My committee members, for providing the questions and feedback during my defense.

Dr. Stephen Harrison, for providing insight and understanding of TC-OLA through conversations and his dissertation.

All sources of funding, including the NSERC CGSM, Jarmila Vlasta Von Drak Thouvenelle Graduate Scholarship, Dr. Peter Driessen and the companies I was fortunate enough to work for while completing this degree.

My wife Holly McQuay, for your undying love and support and for reading my thesis so many times and catching all of my mistakes.

My mother Iris Rich, for all of your words of wisdom and encouragement sourced from your own academic journeys.

Grandma Ines, for inspiring my academic journey and always calling me *Professor Colter*. I'm not one yet, but at least one step closer.

The rest of my family, for all of your support in this and every endeavor that I take on.

Nicholas Bruce and Ahmed Youseff, for rooftop lunches, great conversations and comedic relief. Thanks for making ELW402 such a great place to work.

This document was typeset in LaTeX, on Overleaf, with Mendeley integration for references.

Simulations and experiments were performed using python and GNU Radio on Ubuntu Linux. Python notebooks using NumPy, SciPy and matplotlib were used to generate experiment figures. Many technical drawings and plots were created using Tikz and Pgfplots within the LaTeXenvironment on Overleaf.

Chapter 1

Introduction

The motivation behind this research is the direct result of my own personal interest and love for activities combined with the astute suggestions of my supervisor, Dr. Peter Driessen. The specific motivating example, which is discussed in more detail in Chapter 2, was derived from my many hours spent cycling with my friends. The aim of this research was to design a license-free RF communication device, to improve communication and safety while cycling. In discussions with my supervisor, he advised using TC-OLA to spread an existing and established Land Mobile Radio (LMR) protocol until the signals were compliant with regulations, instead of implementing a custom communications protocol over a commodity, license free RF link. This inversion of the problem focused the direction of this research on the feasibility of reusing existing LMR technology within the Industrial Scientific and Medical (ISM) band using TC-OLA to transparently spread the bandwidth of the signals. LMR protocols have been designed and developed to address many of the common challenges of an RF communication system such as:

- Synchronization
- Data payloads
- Packet framing, addressing and routing
- Group calls (or talk groups)
- P2P calling

- Multiple user access
- Encryption

Since the aforementioned items are already addressed by the LMR protocol, these items were not explored in the research completed for this thesis.

1.1 Claims

1. A method comprising: using spread spectrum to increase the bandwidth of a radio signal.
2. The method of claim 1, further comprising using TC-OLA to increase the bandwidth of a radio signal.
3. The method of claim 1, further comprising of the use of spread spectrum to increase the bandwidth of a Narrowband (NB) radio signal.
4. The method of claim 3, wherein the spread spectrum signals are transmitted in an unlicensed radio band.
5. The method of claim 4, wherein the unlicensed radio band is the ISM band.
6. The method of claim 3, wherein the NB radio signal is generated by a standard LMR protocol.
7. The method of claim 6, wherein the LMR protocol used is DMR, Digital Private Mobile Radio (dPMR), Private Mobile Radio (PMR), Apco 25/Project 25 (P25), NXDN (NEXEDGE) (NXDN) or Terrestrial Trunked Radio (TETRA).

I make *four* claims which my thesis validates:

The new approach to providing short range P2P communication introduced in this thesis provides the following advantages over existing technology:

1. Reuse of open standards and protocols
2. Reduced spread spectrum synchronization requirements
3. Increased channel resilience to noise
4. Increased capacity for simultaneous users

1.1.1 Contributions

The academic contributions of this research come in the form of several Github repositories. The first is *gr-tcola*[1] which provides an optimized and efficient real time implementation of the TC-OLA algorithm built for use in the GR runtime. The second *pybergr*[2] is a python based package whic provides a generalized BER simulation framework for generating BER curves. Other contributions come from the experimentation and analysis of spread spectrum LMR in the context of the ISM band and its regulations.

1.2 Agenda

The following is a high level outline of the chapters in this thesis:

Chapter 1 contains a statement of the claims, which will be proved by this thesis followed by an overview of the structure of the document itself.

Chapter 2 introduces the motivating example and requirements then proceeds to review challenges in RF design and explores the landscape of existing technology.

Chapter 3 describes the methodology and approach behind this research: introducing the TC-OLA algorithm, describing an efficient implementation, and discussing the LMR technology selected.

[1] https://github.com/mistic-lab/gr-tcola
[2] https://github.com/mistic-lab/pybergr

Chapter 4 describes the experiments conducted in this research.

Chapter 5 evaluates the data and measurements obtained from the experiments conducted.

Chapter 6 discusses areas of future work.

Chapter 7 provides a conclusion of the research described in this thesis.

Chapter 2

Problem Definition

In this chapter, the challenges regarding the design of license-free mobile communication devices will be discussed in detail. It will start with a real world example which will be used as the context and motivation for further discussion of technical merit. Based on the motivating example, general application requirements will be outlined and used for discussing the merits of existing technology that address this problem. A brief background on RF design considerations and a review of existing technology will be given to provide the reader with context for design decisions made in Chapter 3.

2.1 Motivating Example

Cycling is, and continues to grow as, a popular form of transportation and recreation. Effective communication amongst cyclists riding together can be challenging. The following section describes the challenge of communication in cycling and illustrates a situation where a smarter communications device could improve not only the enjoyment but also the safety of cycling in a group.

Cyclists can save significant amounts of energy and allow for muscle recovery by riding in slip stream of another rider, a technique known as drafting. In both recreational and competitive cycling, groups of riders will align in tightly packed formations to take advantage of the aerodynamics of the group. The formations of these groups are designed to use the efforts of each rider as efficiently as possible,

allowing a formation of cyclists to ride faster for longer than an individual could on their own. Figure 2.1 depicts the single paceline formation in which the rider in the front will rotate to the back of the paceline to rest when they become tired allowing the next fresh rider to take over in their place. This type of rotation is common in cycling and allows each rider an opportunity rest after expending effort at the front of the group. Though these formations are extremely efficient, they can also be quite dangerous. Since each rider must sacrifice their field of view to effectively draft those in front of them, cyclists within these formations are responsible for communicating hazards and dangers to the riders behind them. Usually this is done by yelling and/or pointing to hazards on either side of the group. This information must propagate backwards to the rest of the group before they reach the hazard, which is further complicated by high speeds and environmental noise.

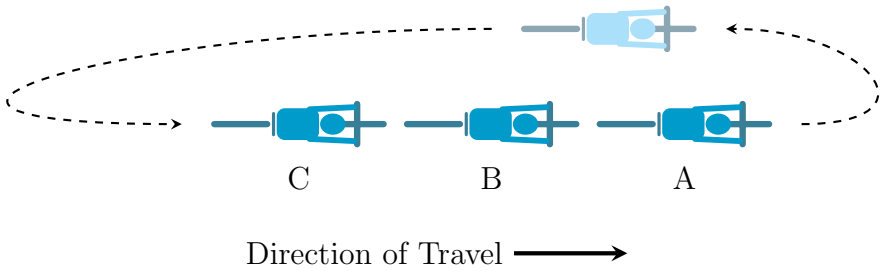


Figure 2.1: Single paceline cycling formation

Consider the example shown in Figure 2.1. Given that the cyclists are under the constraint that everyone must face forward (for safety reasons), the following situation arises:

- **A** will be heard by **B** and **C**
- **B** will be heard by **C** but not **A**
- **C** won't be heard by either **B** or **A**

What's particularly unfortunate about this situation is that the leader has the most unobstructed view of the road ahead but also has the lowest chance of being heard by the rest of the group.

In addition to the safety aspect of cycling, there is also the case for general communication among riders that may not be riding within the group formation. For instance, if many riders of different fitness levels are riding together, the riders in the front of the group may be several kilometers ahead of the riders in the back. Riders may want to notify the rest of the group of a change in route, mechanical or physical difficulty, or an unplanned stop. Improving communication in cycling offers many benefits from the safety and logistical perspective, but also provides a strategic advantage to teams within the context of a race.

Cycling teams are very common in races and introduce a lot of strategy to the sport of cycling. Teams generally work together in an attempt to get a particular member of the team to win. Teammates will attack and counter attack other teams in attempts to force rivals to expend more energy. Improved communication amongst team members would allow the team to coordinate their efforts more effectively and precisely. Mobile communication systems are heavily used in professional cycling, however, there does not yet seem to be a cost effective and robust offering for the amateur cyclist.

The research that follows in this thesis is the result of a high level design of a wireless communications device, depicted in Figure 2.2. The goal of the device is not to replace a smartphone, but instead provide a license-free platform for smartphones that provides Device to Device (DTD) communication. An RF communication channel with a Bluetooth-based software interface would increase the portability and adoption of such a device. In the case of this thesis, the general requirements were derived to help amateur cyclists solve this communication problem but could also be applied to many other areas where short range wireless communication provides value.

2.2 General Application Requirements

The initial requirements can be gathered from the capabilities (*i.e.* range, talk time, group calling, data transmission, user capacity, and cost) of existing devices that this product aims to replace. In order to keep the design centered around user value, these capabilities will be evaluated in conjunction with the examples in Section 2.1.

Figure 2.2: High-level RF communication hardware concept

2.2.1 Range

The range of the design should be comparable to that of a FRS device. Typically, these devices operate within a range 1 km to 8 km.

2.2.2 Talk Time

The designed device should attempt to at least equal the battery/talk time of existing hand held FRS or General Mobile Radio Services (GMRS) devices. The hope is that device run-time/talk-time can be increased by removing the need for extra peripheral components like screens, buttons and speakers. Instead these functions will be offloaded to the smartphone that this device will be paired with.

2.2.3 Group calling

The goal of this device is to allow individuals to communicate more easily using devices they already own in ways that they are already familiar. By removing a technology barrier (*i.e.* a separate hand held device), the use of shorter range communications might become more approachable for individuals doing activities where a separate hand held device is not practical. Ideally, a user should be able to talk with a number of other people at the same time, however, at this point in time full duplex is not a

hard requirement as it increases the complexity of the device significantly.

2.2.4 Data Transmission

In addition to transmitting voice through this P2P communication device, it would be valuable to accommodate data transmission through the channel. This data could be used for control signals to interact with applications on the connected mobile devices. These control signals could help create a more cognitive radio network where the devices could coordinate frequency or modulation changes to avoid interference. If the audio transmission is digital, then data transmission becomes a generalization of the communication layer, allowing for frames of application data to be sent along with the audio frames.

2.2.5 User Capacity

The system design should support multiple users and allow for distinct and unique channels between many different sets of users. The FRS band contains 22 distinct channels (Appendix B); however, the examples given in Section 2.1 describe what might be a slightly higher requirement in the order of 100s of separate channels.

2.2.6 Cost

The cost should be comparable to cost of a FRS or a GMRS device. However, since this device is being designed with features that are not supported by FRS, it seems reasonable that the device would cost more.

2.2.7 Summary of Requirements

Table 2.1 distills the previous sections into general requirements.

Designing RF communication devices to fulfill these requirements is a challenging endeavor full of many trade offs and design decisions. The next section will provide some background on the constraints and considerations that must be evaluated when designing an RF system.

Capability	Requirement
License Type	*License-Free*
Range	1-6+ km
Run Time	4-6+ hours
Payload Type	digital data and voice
Capacity	100s of channels / users
Cost	< $200

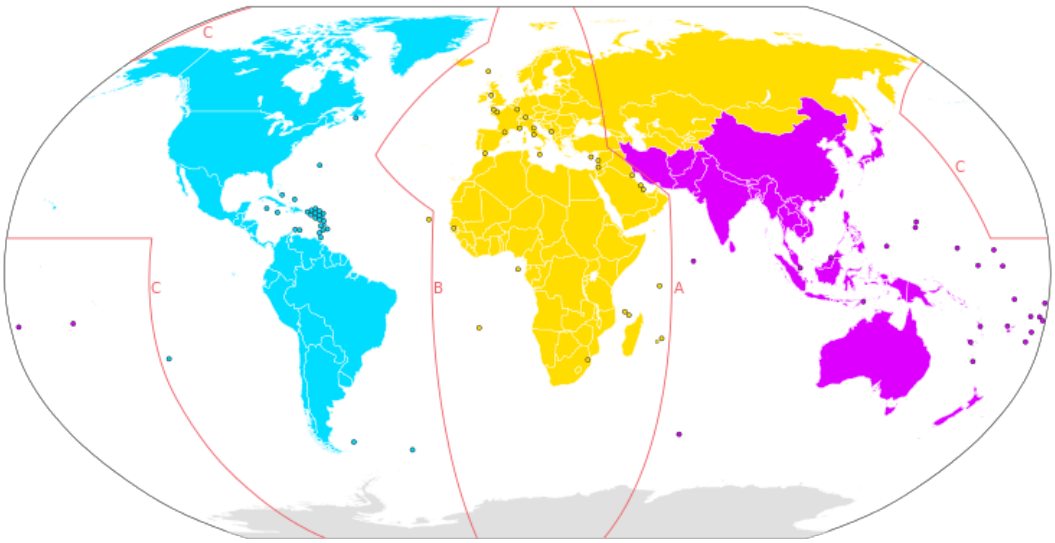
Table 2.1: Application requirements derived from example

2.3 RF Design Considerations

RF design is a challenging field with many decisions and trade offs to be made. RF device design must strike a balance between application requirements and regulations. In this chapter, the regulations of license-free RF bands will be explored, followed by discussion of some challenges introduced by RF propagation in the real world and some ways that modulation can help mitigate them.

2.3.1 Regulations

The radio spectrum is an extremely valuable and extremely regulated resource. As such, any device used for transmitting RF signals must comply with the regulations set forth by regulating bodies such as the Federal Communication Commission (FCC) or the International Telecommunications Union (ITU). These regulating bodies are responsible for portioning the available radio spectrum into what is known as a frequency allotment plan. Within these plans, ranges of frequencies, called bands, are allocated and set aside for specific applications and purposes. These band plans are designed to minimize interference and maximize the spectrum resources available to applications. The basis of these band plans is a mixture of application requirements, environmental factors, and the frequency dependent physical behavior of electromagnetic radiation, as described in Section 2.3.2.1. Each allocation in the band plan comes with its own specific set of regulations that govern that portion of the spectrum, for example restrictions on power and emission types (see Appendix A), spectral density requirements, and whether or not use of the band requires a license. These regulations are necessary to preserve and protect the integrity of the available radio



Region 1: ■ | Region 2: ■ | Region 3: ■

Figure 2.3: Dörrbecker, M. (2016). ITU regions and the dividing lines between them. [online] Available at: https://upload.wikimedia.org/wikipedia/commons/9/9b/International_Telecommunication_Union_regions_with_dividing_lines.svg [Accessed 9 Apr. 2019].

spectrum for all of the various applications that use it. Licenses are required in order to operate within a large number of the radio bands and range from Amateur radio licenses to commercial broadcasting or telecommunications licenses. The requirements for obtaining these licenses vary, but without one, an individual is not permitted to transmit within the associated band. The ITU created 3 geographical zones (see Figure 2.3) called ITU regions in which the band plans and regulations are more or less homogeneous for that region. For example, the FRS band within North America (ITU Region 2) provides an unlicensed band for short range voice communication is actually used for emergency services within Europe (ITU Region 1). There can also be country specific exceptions, and as such, devices are usually designed and built for a particular ITU region or country based on the appropriate regulations.

Unlicensed bands have been crucial for the success of consumer RF products and technologies, since requiring a consumer to first have an appropriate license to operate a device would be a significant barrier for adoption. Some examples of license free bands include the aforementioned FRS band and the ISM band.

2.3.1.1 Family Radio Services (FRS) Band

The FRS band was allocated for license free analog voice communication and at first seemed like an excellent choice for a possible solution for Section 2.2. However, investigating the regulations further, it was found that restrictions on emission types limit the use of this band considerably [12] (see Table 2.2).

Code	Meaning
F3E	One channel of analog voice information modulated with Frequency Modulation (FM)
G3E	One channel of analog voice information modulated with Phase Modulation (PM)
F2D	One channel of digital telemetry information modulated with FM using a sub carrier
G2D	One channel of digital telemetry information modulated with PM using a sub carrier

Table 2.2: Permitted emission type codes for FRS band

From these emission type restrictions, it can be seen that transmissions within this band are limited to NB analog modulations. The FRS band does support digital payloads; however, upon further investigation of the regulations, they are limited considerably by the context in which they may be transmitted:

- Digital payloads may be either a location or a brief text message
- Data transmissions must be initiated by the operator with an exception being that a unit may automatically respond to a location interrogation
- Digital transmissions may not exceed 1 s
- Digital transmissions may not be sent more frequently than one transmission within a 30 s time period (with the exception of location interrogation)

Digital payloads must be a location or text payload and initiated directly by the user (at most once every 30 s) or in response to a location request [12, § 95.531]. This band is governed by some basic rules, and as such, the analog protocol is fairly straightforward. The regulations do not discuss the format of the digital information, so in order for cross device compatibility, the format of this digital information would need to be made standard across manufacturers.

2.3.1.2 Industrial Scientific and Medical (ISM) Band

The ISM band is a collection of a few different frequency ranges:

- from 902 MHz to 928 MHz
- from 2.4 GHz to 2.4835 GHz
- from 5.725 GHz to 5.850 GHz

Contrary to the FRS band (see Section 2.3.1.1), the ISM band is more general in terms of emission types. The ISM band does not place restrictions on the emission types or the type of information transmitted; however, the regulations require that all transmissions within the band meet the power limit and spectral density requirements. The power limits within the ISM band are 1 W with a minimum spectral density defined in terms of the 6 dB bandwidth, which must be at least 500 kHz [11]. This means that signals transmitted within the ISM band must be spread spectrum modulation that is robust against interference and noise present within this band.

2.3.2 Channel Effects

Losses and effects experienced by propagating radio waves are a complicated problem that must be overcome to establish a reliable and robust RF communication channel. In order to sufficiently compensate for these effects, mathematical models have been formulated [2] and fall into three categories:

- Path loss
- Shadowing
- Multipath Fading

Figure 2.4 shows a visualization of how these three channel loss models are used to model the overall loss of a wireless channel with respects to distance. These losses and how they are modeled will be discussed in the following sections to aid in understanding the challenges of designing a robust RF system.

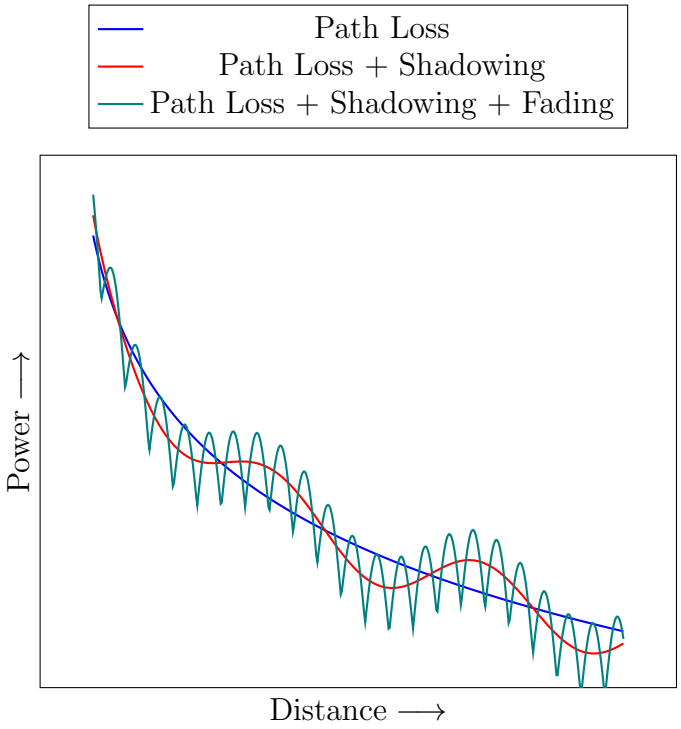


Figure 2.4: Visualization of different channel losses

2.3.2.1 Path Loss and Shadowing

When considering the general propagation of RF waves, the Free Space Path Loss (FSPL) model provides a convenient way of estimating the losses due to free space. The FSPL model is derived from the Friis transmission equation [14]:

$$\frac{P_r}{P_t} = \left(\frac{A_r A_t}{d^2 \lambda^2}\right), \tag{2.1}$$

where P_r is the power received, P_t is the power transmitted, A_t and A_r are the effective aperture of the transmitter and receiver antennas respectively, d is distance between the receiver and transmitter, and λ is the wavelength of the RF radiation. This formula represents the gain of the signal as seen by the receiver. If both antennas are assumed to be isotropic ($A_{isotr.} = \frac{\lambda^2}{4\pi}$), then the formula is further simplified to Equation 2.2.

$$\frac{P_r}{P_t} = \left(\frac{\lambda}{4\pi d}\right)^2. \tag{2.2}$$

With loss being the reciprocal of gain; inverting Equation 2.2 and replacing λ with $\frac{c}{f}$ leads to the commonly known FSPL model seen in Equation 2.3:

$$Path\ Loss\ = \left(\frac{4\pi df}{c}\right)^2. \tag{2.3}$$

This model gives the estimated loss (*i.e* the ratio of $\frac{P_t}{P_r}$), due to the dissipation of RF energy, for a given wavelength λ at a distance d in free space conditions. In these free space conditions, it is assumed that there are no obstructions or obstacles and that the RF waves are propagating along a Line of Sight (LOS) path. It is often more convenient to convert losses and gains into Decibels (dB) so that they may be added and subtracted accordingly (see Section 2.3.3). Converting Equation 2.3 into power and using units of frequency in MHz and distance in km yields

$$\text{Path Loss (dB)} = 20\log_{10}(d) + 20\log_{10}(f) - 32.45. \tag{2.4}$$

Looking at the basic FSPL formula in Equation 2.3, it can be seen that power of a transmitted signal decreases with the square of distance (d) and frequency (f). Therefore, lower frequencies will offer better range with the same amount of output power, as demonstrated in Figure 2.5.

In general, the free space model is rarely representative of real world applications and is only applicable where LOS is the only path from transmitter to receiver. As a result, more general path models have been developed [28] to take into account the effect of shadowing on the overall path loss. Shadowing is an effect caused by large obstructions in between the transmitter and receiver and results in losses due to absorption of RF energy by the obstruction. Path loss and shadowing are considered to be large-scale channel effects [2], since they do not change significantly with respect to time relative to the period ($T = \frac{1}{f}$) of the RF waves. A more general path loss model called the log-distance path loss model takes into account the shadowing effect and is given by

$$\text{Path Loss} = 10n\log\left(\frac{d}{d_{ref}}\right) + L_{ref}, \tag{2.5}$$

where d is the distance between transmitter and receiver, L_{ref} is the free space path loss in dB at a distance d_{ref}, and n is the path loss exponent and represents the

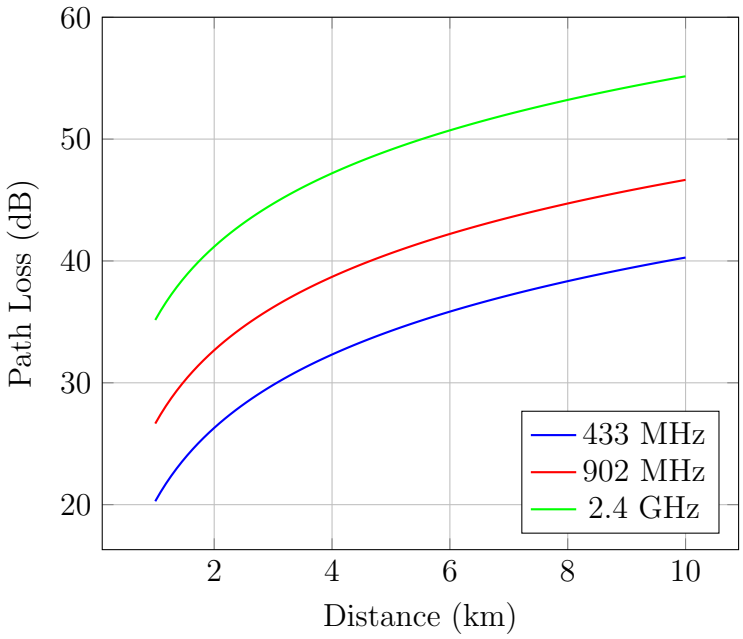


Figure 2.5: Free Space Path Loss vs distance for various frequencies

severity of obstruction between the transmitter and receiver. The value of the path loss exponent can vary between 2 and 6, where 2 represents free space conditions and 6 represents heavy obstruction.

So far, only large scale channel effects, (*i.e* effects that change slowly relative to symbol time T_s) have been explored. Models for these effects provide analytical representation of the dissipation of RF energy over distance as well as the consequences of shadowing introduced by obstructions. Small scale channel effects like fading change quickly, relative to the symbol time T_s, and are the consequence of the aggregation of reflections and multiple propagation paths of an RF signal.

2.3.2.2 Multipath Fading

In Section 2.3.2.1, it was seen that the received power of an RF wave is attenuated by path loss and shadowing effects. In the real world, there are many objects and obstacles between the transmitter and receiver that cause reflection and refraction of the original signal, resulting in what are known as multipath effects (see Figure 2.6). These additional signal paths, referred to as rays, will all vary in delay, phase shift, and attenuation as a result of the path each ray took to reach the receiver.

Calculating the effect of every one of these rays is computationally prohibitive, and as a result, statistical fading models have been developed to represent these effects. A simple model commonly used to describe multipath fading effects is the *Rayleigh* fading model.

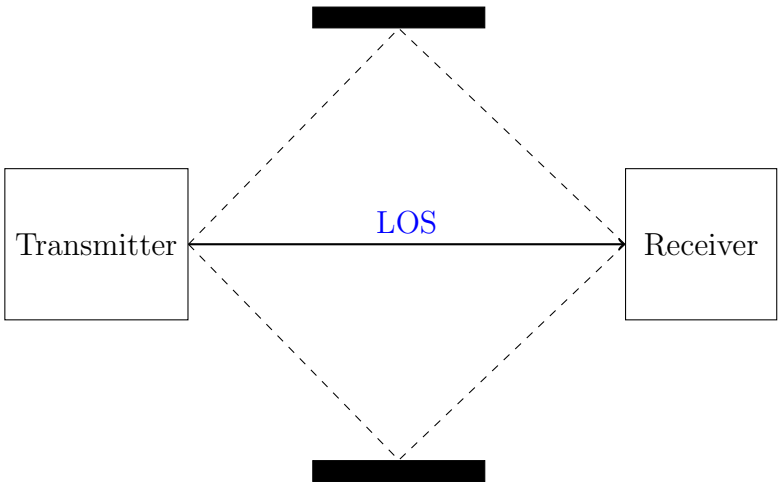


Figure 2.6: Depiction of simplified multipath scenario

Rayleigh Fading

In the *Rayleigh* fading model, the received signal can be thought to be a vector sum of all of the rays generated in a multipath environment [29]. Assuming that the channel experiences *frequency flat* fading, meaning that all frequencies within the transmitted signal will experience the same fading, allows the transmitted signal to be simplified to a single sinusoid with a frequency f_m. As stated previous, a signal experiencing *Rayleigh* fading $s_{Ra}(t)$ is the vector sum of all of the N multipath components:

$$s_{ray}(t) = \sum_{i=1}^{N} a_i \cos(2\pi f_m t + \phi_i), \tag{2.6}$$

where a_i and ϕ_i are the amplitude and phase of the i^{th} multipath component respectively. This model can be further evaluated using trigonometric identities to break Equation 2.6 into its inphase and quadrature components yielding

$$s_{ray}(t) = \cos(2\pi f_m t) \sum_{i=1}^{N} a_i \cos(\phi_i) - \sin(2\pi f_m t) \sum_{i=1}^{N} a_i \sin(\phi_i). \tag{2.7}$$

If the following substitutions are made in Equation 2.7

$$X = \sum_{i=1}^{N} a_i \cos(\phi_i), \;\; Y = \sum_{i=1}^{N} a_i \sin(\phi_i),$$

it gives

$$s_{ray}(t) = X \cos(2\pi f_m t) - Y \sin(2\pi f_m t). \tag{2.8}$$

Assuming that the number of multipath components, N, is very large and that the environment from which they are generated is completely random, then X and Y will be Gaussian random variables with zero mean, which have independent but identical distributions.

The amplitude envelope of $s_{ray}(t)$, equal to

$$A = \sqrt{X^2 + Y^2},$$

will therefore have a *Rayleigh* distribution with a Probability Density Function (PDF) of

$$f_{ray}(a) = \frac{a}{\sigma^2} e^{\left(-\frac{a^2}{2\sigma^2}\right)}, \;\; a \geq 0, \tag{2.9}$$

with the phase of $s_{ray}(t)$ being

$$\Theta = \tan^{-1}\left(\frac{X}{Y}\right),$$

and since X and Y are both zero mean Gaussian random variables, the PDF of the phase can be obtained as:

$$f_{ray}(\theta) = \frac{1}{2\pi}, 0 \leq \theta \leq 2\pi,$$

showing that the phase has a uniform distribution within 0 and 2π. The fact that X and Y are both independent random variables yields that the PDF for phase $f_{ray}(\theta)$ and envelope $f_{ray}(a)$ are also independent. Intuitively, this means that in *Rayleigh*

fading, the phase and power fluctuate independently.

2.3.3 Link Budget

In RF systems, the link budget is a concept that accounts for all gains and losses over a RF communications channel. In Section 2.3.2, path losses were explored and are indeed one factor that would be included in a link budget. An example of a link budget (in dB) can be seen in Equation 2.10.

$$P_{rx} = P_{tx} + G_{tx} - L_{tx} - L_{path} + G_{rx} - L_{rx}. \tag{2.10}$$

P_{rx}	received power (dBm)
P_{tx}	transmitted power (dBm)
G_{tx}	transmitter antenna gain (dBi)
L_{tx}	transmitter losses (dB)
L_{path}	path losses (dB)
G_{rx}	receiver antenna gain (dBi)
L_{rx}	receiver losses (dB)

In practice, there could be more or less items included in the link budget depending on the losses or gains within the system. The link budget is used to calculate the projected received power, P_{rx} (in dB). This received power can then be compared to the receiver sensitivity. This receiver sensitivity, usually measured in dB, describes the minimum Signal to Noise Ratio (SNR), or in this case P_{rx}, at which the receiver can operate. The difference between the receiver sensitivity and the P_{rx} is essentially extra power available to the receiver and provides a buffer against channel fading effects and is hence known as the *Fade Margin*. The ratio between the P_{rx} and the power level of the noise in the channel is known as the SNR and can be related to the capacity of the channel via the *Shannon Capacity* formula [30]

$$C = W \log_2(1 + \frac{S}{N}), \tag{2.11}$$

where C is the channel capacity in bits per second, W is the bandwidth in Hz, and S and N are the signal and noise power levels respectively. The SNR (in dB) can be found by taking the result of the link budget calculation (in dB) and the noise

power within the channel (in dB). This SNR dB value must be converted back to a ratio before being used in Equation 2.11. It can be seen from this equation that the channel capacity can be increased by either increasing the bandwidth, W, or increasing the SNR. Conversely, increasing a signal's bandwidth can yield the same theoretical capacity with a lower SNR. Equation 2.11 provides an important result for RF and band plan design, since traditionally noisy bands (such as the ISM band) can use Wideband (WB) signals to counteract the additional noise present in this band. NB and WB signals have different characteristics and provide yet another tradeoff that can be made in the RF design. In the next section, NB and WB signals and their advantages/disadvantages will be discussed further.

2.3.4 Narrowband and Wideband Signals

The duality between the time and frequency domains forces tradeoffs in RF design, between data rates and spectrum usage. To increase data rates, either more bits have to be sent per symbol or more symbols have to be sent per time period. Increasing the number of bits per symbol decreases the amount of space between symbols and the decision boundaries used to discern one symbol from another. This generally leads to increased susceptibility to noise and other channel effects. On the other hand, increasing the number of symbols sent per time period requires that the symbol time is decreased. As signals become shorter in the time domain, spectrum requirements increase in the frequency domain with the extreme being the δ function representing a single impulse in the time domain which results in equal power across all frequencies.

The terms NB and WB refer to the amount of space occupied by a signal in the frequency domain. Regulating bodies, such as the FCC, maintain a band plan dividing the available spectrum into sections known as bands. Regulations differ for each band in the plan and specify the qualities of signals that are transmitted in the band. These regulations stipulate, among many parameters, the bandwidth requirements and thus if the signals must be NB or WB.

NB signals are used to split a band into multiple independent channels that can be squeezed next to each other without interference from other channels. It is however, much easier for malicious users to purposely interfere or jam NB signals, as all of the information is contained within a small section of the spectrum. WB signals, on the other hand, consume much more bandwidth and are also permitted in certain

frequency bands. From the time-frequency duality discussed earlier, increasing the bandwidth of a signal can increase the overall data capacity of the channel. Spread spectrum techniques use this extra capacity to add redundancy to the data being transmitted, which makes the transmitted signals robust against malicious jammers or environmental interference. This also follows with Equation 2.11, since by holding the data capacity, C, constant and increasing the bandwidth, W, the required SNR can be lowered. This is why WB signals are required in bands such as the 2.4 GHz band, since this band contains a large amount of interference, produced by microwave ovens, and thus NB channels would not reliably function. In addition to suppressing interference, WB signals are often used to obfuscate the presence of transmission by spreading the bandwidth of a transmitted signal out until the signal power is below the noise floor.

It should be noted that a given message signal can be modulated and transmitted a myriad of ways resulting in either NB or WB signals. For instance, in radio broadcasting the two common modulations are Amplitude Modulation (AM) and FM, which both carry audio signals. AM signals are, by nature, the same bandwidth of the message signal. On the contrary, FM has a parameter called frequency deviation that is used to control the overall bandwidth of the modulated signal. This increase in bandwidth results in signal redundancy and is why FM radio stations provide higher quality audio transmissions than their AM counterparts. In RF design, WB signal modulations are referred to as spread spectrum modulations and will be explored further in the next section.

2.3.5 Spread Spectrum

Spread spectrum describes various modulation techniques that transform NB signals into new signals with higher bandwidth requiring higher sampling rates. There are many reasons for doing this:

- Increasing User Capacity
- Increasing immunity to interference
- Decreasing spectral density
- Obfuscating signals

There are many different techniques used to spread the spectrum of a given signal, but in practice, the most heavily used systems at this time are Frequency Hopping Spread Spectrum (FHSS), Direct Sequence Spread Spectrum (DSSS), and Orthogonal Frequency Division Multiplexing (OFDM). It should be noted here, that OFDM is not considered a spread spectrum modulation as it does not introduce signal redundancy; however, OFDM is used in Long Term Evolution (LTE) networks as it provides a very flat spectrum and high data rates. OFDM is included here for reference, as it does have the capability to create a very flat and broad spectrum from a NB signal.

2.3.5.1 Frequency Hopping Spread Spectrum (FHSS)

FHSS transforms the input signal into a new signal with higher bandwidth by iterating through a predefined pattern of frequencies and transmitting a NB modulation of the input signal for a very short period of time (see Figure 2.7). The time spent occupying a single frequency is extremely short; therefore any interference or fading effects present at that particular frequency will only be experienced for a small amount of time. FHSS requires that the transmitter and receiver are synchronized in both time and frequency in order for the receiver to be tuned into to the correct hopping frequency at the correct time.

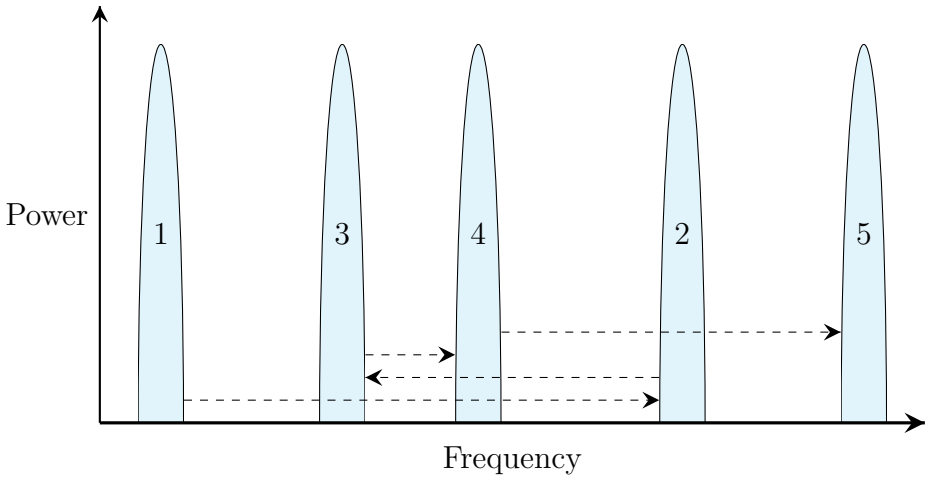


Figure 2.7: Visualization of Frequency Hopping Spread Spectrum

2.3.5.2 Direct Sequence Spread Spectrum (DSSS)

DSSS is a form of spread spectrum that leverages the duality between the time and frequency domains discussed in Section 2.3.4. In order spread the energy of the signal across more spectrum, the original signal must be transformed into a signal with a higher sampling rate. DSSS does this by multiplying the input signal with what is known as a spreading waveform [31]. This spreading waveform, commonly referred to as a chip code, has a much higher sampling rate than the original data signal. Therefore, when the spreading code is multiplied by the input signal, each bit of the input signal is transformed into several bits, as shown in Figure 2.8.

The redundancy and processing gain, in this case, is introduced by the chip code and will vary based on the length of said code. Another side-effect is that the transmitted signal's spectrum takes on the spectral shape of the spreading sequence. Since the spreading sequence is essentially an arbitrary and random sequence, the resulting processed signal looks like noise to anyone without the spreading sequence. In order to receive a DSSS signal, the receiver uses the chip code as a matched filter on the received signal to reconstruct the original message. In order to construct the message, the matched filter must be perfectly aligned with the received signal in order for all of the samples in the chip code to correlate correctly. Sophisticated synchronization mechanisms have been developed [22] to account for channel effects and maintain sequence alignment for each received message.

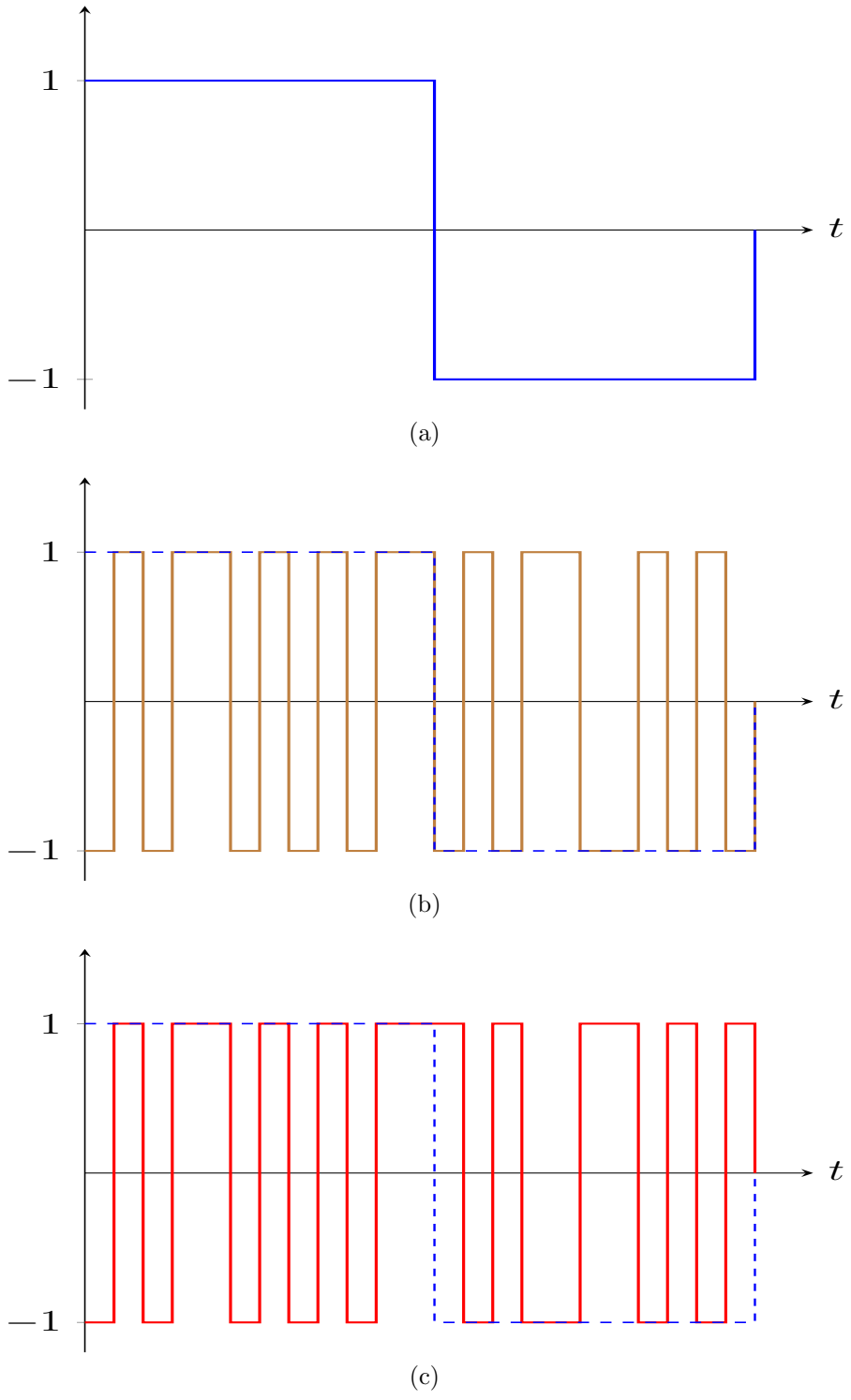


Figure 2.8: (a) binary message modulation, (b) spreading sequence (chip code), (c) message multiplied by spreading sequence

2.3.5.3 Orthogonal Frequency Division Multiplexing (OFDM)

As the name implies, this modulation is a form of frequency multiplexing where symbols are transmitted across a number of orthogonal frequencies, referred to as sub carriers, simultaneously (see Figure 2.9). This modulation is not actually considered spread spectrum, however, it results in signal content being distributed over wide amounts of spectrum and has been included here for reference. Each of the sub carrier frequencies are spaced orthogonally, are generally modulated using a NB modulation technique and responsible for carrying one constellation point. OFDM combines a great number of these NB sub carriers simultaneously, and results in a very flat WB spectrum. The success of OFDM depends on accurate time and frequency synchronization between the transmitter and receiver [21] to demodulate the many sub carriers that are used to transmit data.

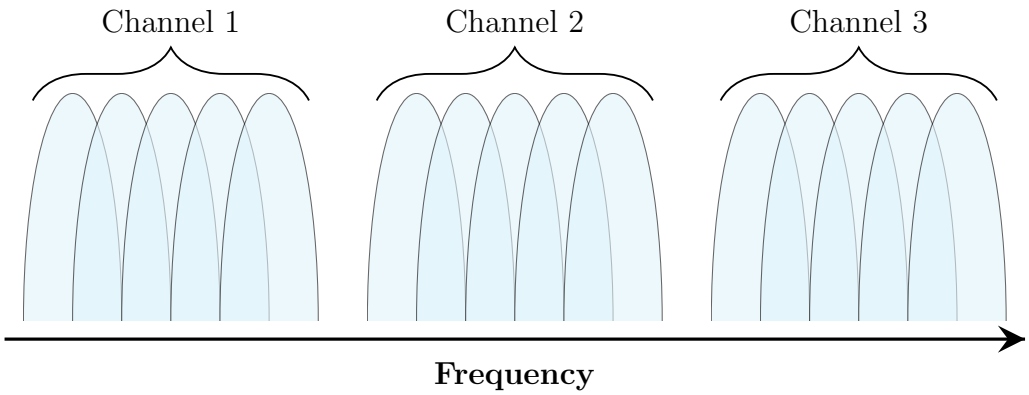


Figure 2.9: Visualization of Orthogonal Frequency Division Multiplexing

2.4 Existing Technology

This section explores the landscape of existing technology that might be used to provide solutions for the example given in Section 2.1.

2.4.1 Smartphones

Smartphones have undoubtedly transformed the way that the people work, communicate, and manage their lives. New devices are being developed to take advantage

of mobile networks that don't yet exist, and in the same vein, new networks are being developed for devices that have yet to be designed. Consumer expectations and demands for faster networks, better battery life, and intuitive integration with the devices around them have driven the innovation and evolution of the hardware and components within these devices. Consumer applications are continually being developed and updated to take advantage of the latest hardware capabilities. Current smartphones are designed with a myriad of communication technologies, such as Near Field Communication (NFC), Bluetooth, WiFi, and cellular data networks of various generations (3rd Generation (3G), LTE, *etc.*), in attempts to provide a hardware platform suitable for the problems that people most commonly face. NFC provides support for interacting with Radio Frequency Identification (RFID) chips and allows devices to communicate small amounts of data over very short distances (1 cm to 3 cm). Bluetooth provides connections to devices and sensors using relatively low data rates [25] within fairly short ranges (up to 20 m). WiFi provides much higher data links over slightly longer ranges than Bluetooth (up to 40 m). Cellular data connections provide data rates comparable to WiFi but over much larger distances. Consider these existing technologies in terms of their functional ranges shown in Figure 2.10.

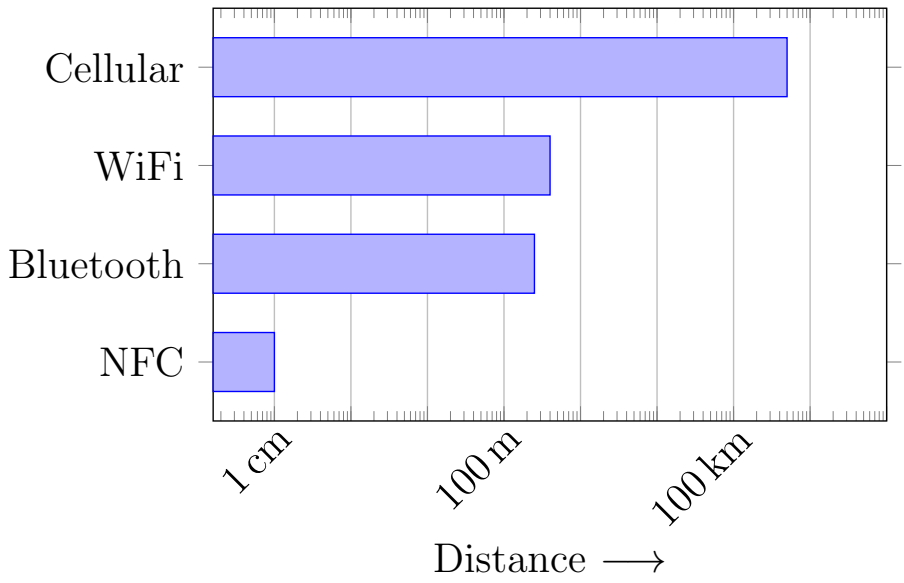


Figure 2.10: Functional ranges of smartphone wireless technologies

It can be seen in Figure 2.10 that there is abundant hardware support for shorter distance (< 40 m) DTD communication. Cellular towers are capable of providing cov-

erage for many 10s of kms, providing network connectivity to users out of range of WiFi. However, DTD communication is limited by the range of device WiFi or Bluetooth. Investigations have been done by various teams [13] [5] into the feasibility of P2P communication architectures, including discussion of potential benefits like increased spectral efficiency, decreased load on cellular systems, and improved coverage. Though these architectures show promise, they would still require centralized coordination of the spectrum usage and thus would only be available with cellular network coverage. This complement of hardware support is sufficient for supporting real time DTD communication, if one can assume cellular coverage is always 100%. In reality, this is not the case. In the absence of cellular data networks, the range of device communication is limited to that of WiFi or Bluetooth. Since smartphones lack the hardware capability for long range inter-device communication within a disconnected environment, separate devices must be used to facilitate this need. These devices such as hand held radios, Bluetooth intercoms, or other proprietary transceiver technology must be certified and adhere to the regulations set forth by regulating bodies for radio transmissions.

2.4.2 Mobile Radio

Mobile radio describes a family of devices and protocols that provide standalone communication channels for audio and data over vast distances. The communication channels between these devices can be P2P or also centrally coordinated for increased range and spectral efficiency. Many modern mobile radio systems use digital protocols from open standards that are designed to be agnostic of the frequency band that they are transmitted within. Many handheld radios allow users to reprogram the device to use different frequency bands giving more flexibility as to where it can be used. Each of these frequency bands are, however, governed by regulations that dictate the parameters under which transmissions are allowed.

2.4.2.1 Radio Standards and Protocols

In order for users to be able to communicate with one another using a wireless link, the mode of transmission must be standardized and known to all of those who would like to participate. Open radio standards, which have been designed by standards bodies in compliance with regulations, provide platforms on which communication

devices, applications, and communities can be built. In the analog world, a radio standard would describe the type of modulation being used at the transmitter and receiver such as FM or AM radio. Analog protocols are more or less dictated by the regulations for the band that they operate within.

Digital radio standards offer many advantages over their analog counter parts. Digital communications can be augmented with Forward Error Correction (FEC) to extend range and offer higher sound quality and clarity at the expense of digital overhead and slightly slower data rates. Digital standards allow for packet based communication, which can be routed and encrypted at the application level using standard means. Digital communication provides more flexibility to those designing these digital radio standards, since the permutations of packet shapes and modulations are limitless. Fortunately, many companies within this domain have helped develop open standards for digital radio communication such as DMR [7][9][10], dPMR [8], P25 [27], NXDN [26], or TETRA [6]. These open standards give manufacturers and hardware designers a spectrally efficient and regulation compliant blueprint from which to base their implementations, allowing design and delivery of standards compliant devices to be faster and more reliable.

2.4.3 License free RF devices

Hand held radios have long been the customary device for communicating off the grid. Some examples of these devices include hand held radios which can operate over unlicensed (FRS) or licensed (Citizens Band (CB), GMRS) frequency bands. New products have begun to emerge over the last several years designed to operate within unlicensed bands to accommodate the communication needs of users and organizations operating within disconnected environments.

2.4.3.1 Bluetooth Intercom Devices

There are many devices on the market offering a solution to this problem using the Bluetooth Intercom Profile. This allows for up to 3 devices to be connected together via Bluetooth for a group phone call. Some of the devices on the market today offer up to 6 devices to be connected together with a range from 100 m to 1600 m. This range is achieved by increasing the signal strength of the transmitted Bluetooth signal

to more closely match the upper limits allowed by regulations within the ISM band. The Bluetooth protocol offers full duplex audio with the expense of added overhead and therefore decreased range. The limitations of this technology are the numbers of devices that can communicate at the same time, and the increased overhead of the Bluetooth protocol imposes limits on the range

2.4.3.2 GoTenna

GoTenna [20] is a product designed to provide a long range RF link to allow users to send text messages, images, and locations over large distances. The consumer version of goTenna uses a proprietary mesh networking protocol built on top of their proprietary hardware which provides users with a wireless communication link via a smartphone app that is capable of transmitting and receiving data messages over distances unfeasible with a single RF link. Due to technical constraints of the hardware, and protocol, the goTenna mesh network is not capable of supporting real-time voice communication.

The limitations of this technology are the lack of real-time support for communication, though the mesh-networking capabilities are compelling and allow the range of the product to extend as the user base is increased.

2.4.3.3 Beartooth

Beartooth shares similarities to goTenna in that it provides consumers with a means of communicating through its proprietary hardware via a smartphone application. Beartooth is built on top of Long Range (LoRa) which is an emerging networking protocol that provides low rate data communication over large distances. LoRa, was designed to compete with other Low-Power Wide-Area network (LPWAN) solutions that are needed to support Internet of Things (IoT) devices. As such, LoRa is able to provide quite a long range channel at the expense of lower data rates. Beartooth has implemented a proprietary communication protocol within software, using the data channel provided by LoRa to transmit the packets.

This technology attempts to solve the real-time communication problem by building out voice and data protocols on top of a generic wireless data solution. In its current form, there are two visible limitations of this approach. The first limitation

is a consequence of the chosen hardware solution. The LoRa protocol was designed in an effort to provide solutions for the IoT market and as such lacks the necessary bandwidth to allow this device to operate effectively for real-time communications at any great distance. The other obstacle with this approach is that it requires designing, developing, and maintaining a custom real-time communication protocol from scratch which can be expensive and likely encumbered by more overhead than using off-the-shelf protocols already designed.

2.4.4 Summary

Table 2.3 gives a summary of existing license free communication solutions:

Technology	**Pros**	**Cons**
FRS	Good range and real-time voice	No data transmission allowed
Bluetooth Intercom	Full duplex audio and reasonable range	Only a handful of connections possible
Gotenna	Good range	No real-time voice
Beartooth	Real time voice support	Proprietary communications protocol

Table 2.3: Summary of pros and cons of existing technology

Chapter 3

Approach

In Chapter 2, existing mobile communication technologies were explored in the context of providing solution to the scenario described Section 2.1 (*i.e.* real time communication among cyclists). The difficulty in designing unlicensed RF communication devices comes not only from constraints in the physical world but also from the regulatory restrictions applied to these unlicensed RF bands. Because the FRS band is too restrictive in terms of digital transmissions and is only available within ITU Region 2, the ISM band was chosen as the target band of operation. Choosing the ISM band allows a product design that would be capable of the widest adoption. Within the ISM, several different approaches and designs for consumer RF communication devices were investigated, as summarized in Table 2.3.

The approach of this research was to investigate the feasibility and design of an RF communication device that could use spread spectrum to transparently transform an off-the-shelf digital radio protocol into a signal capable of being transmitted in the unlicensed ISM band, as depicted in Figure 3.1. Transparently applying spread spectrum to existing digital radio standards allows for the reuse of these standards in frequency bands that would otherwise not support them. Spread spectrum techniques like DSSS or FHSS require precise synchronization of the transmitter and receiver, in order to successfully decode the received signals. Unfortunately, this synchronization requirement introduces overhead that makes a transparent spreading mechanism more difficult to achieve. TC-OLA is a relatively new algorithm that has been shown to provide excellent spectrum spreading results with the possibility of minimum synchronization requirements. TC-OLA has been used to enhance Linear

Frequency Modulation (LFM) radar systems without modifying the underlying LFM radar hardware or algorithms [34] and the technique of wrapping existing radar hardware in TC-OLA served as some inspiration for the use of TC-OLA with existing digital radio protocols. In this chapter, the TC-OLA algorithm will be explored in further detail, followed by a description of an efficient TC-OLA implementation for GR and how it was used to transparently spread a hardware implemented open digital radio protocol into signals capable of being transmitted within the ISM radio band.

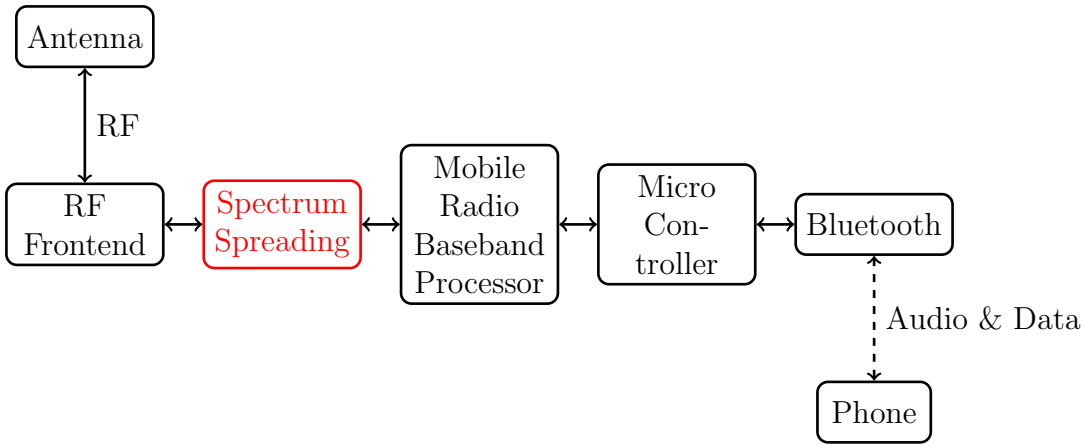


Figure 3.1: Block diagram of hardware device

3.1 Time Compression Overlap Add (TC-OLA)

TC-OLA is a spread spectrum technique that was developed to take advantage of the ever increasing bandwidths of Software Defined Radio (SDR) systems [17]. TC-OLA was inspired by the techniques from well a known audio signal processing unit called a Phase Vocoder [35]. The Phase Vocoder and its overlap and add pattern has been widely used for many audio effects such as pitch shifting and time scaling [32], and is the basis of the TC-OLA algorithm, which has been used to create resilient wideband RF channels [18]. The TC-OLA process is composed of a Time Compression in the transmitter followed by an Overlap and Add process in the receiver.

3.1.1 Time Compression

The Time Compression portion of the TC-OLA process is responsible for spreading the spectrum of the input signal before transmission. Overlapping windows of size M (samples) are taken of the input signal. The amount of overlap between windows is dictated by the hop size, R (samples), between each subsequent window. It can be seen from Figure 3.2 that the overlap is equal to $M - R$, and thus the smaller the value of R the more window overlap and redundancy in the transmitted signal.

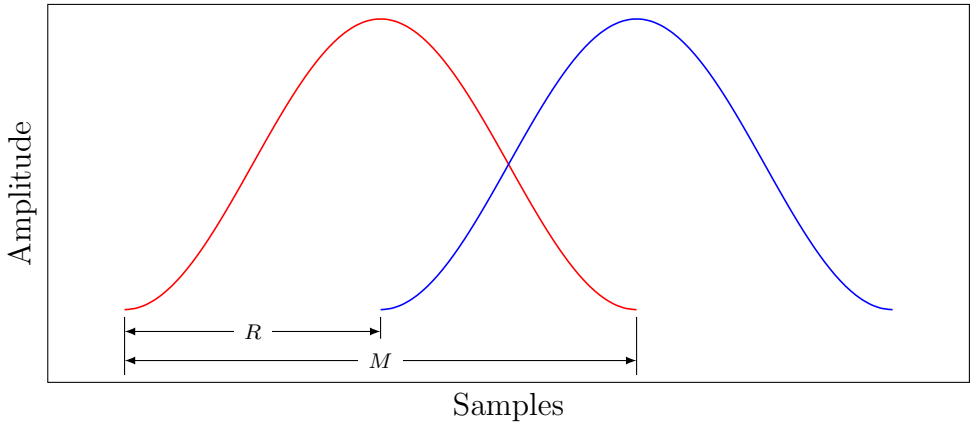


Figure 3.2: Overlapping windows with window size M and hop size R

Figure 3.3 shows a depiction of the Time Compression process for $M = 4$ and $R = 1$. It can be seen that the Time Compression process will introduce an initial delay of M samples, as the algorithm needs M samples to output. However, after the initial M samples, one window of M samples are placed at the output for every R new samples that arrive at the input. This process, from a sampling rate perspective, is an interpolation by a factor of $\frac{M}{R}$ and results in an output sampling rate given by:

$$F_{s_{out}} = F_{s_{in}} * \frac{M}{R}.$$

This increase in sampling rate exactly equals the spectral spreading factor of $\frac{M}{R}$. Since the windows are overlapping by $M - R$ samples, this process produces $\frac{M}{R}$ redundant copies of the input signal; this redundancy creates what is known as processing gain. This redundancy is similar to that of DSSS as discussed in Section 2.3.5.2 with one major difference. Recall that with DSSS the random chip code was multiplied by each bit resulting in a signal that could be received using a perfectly

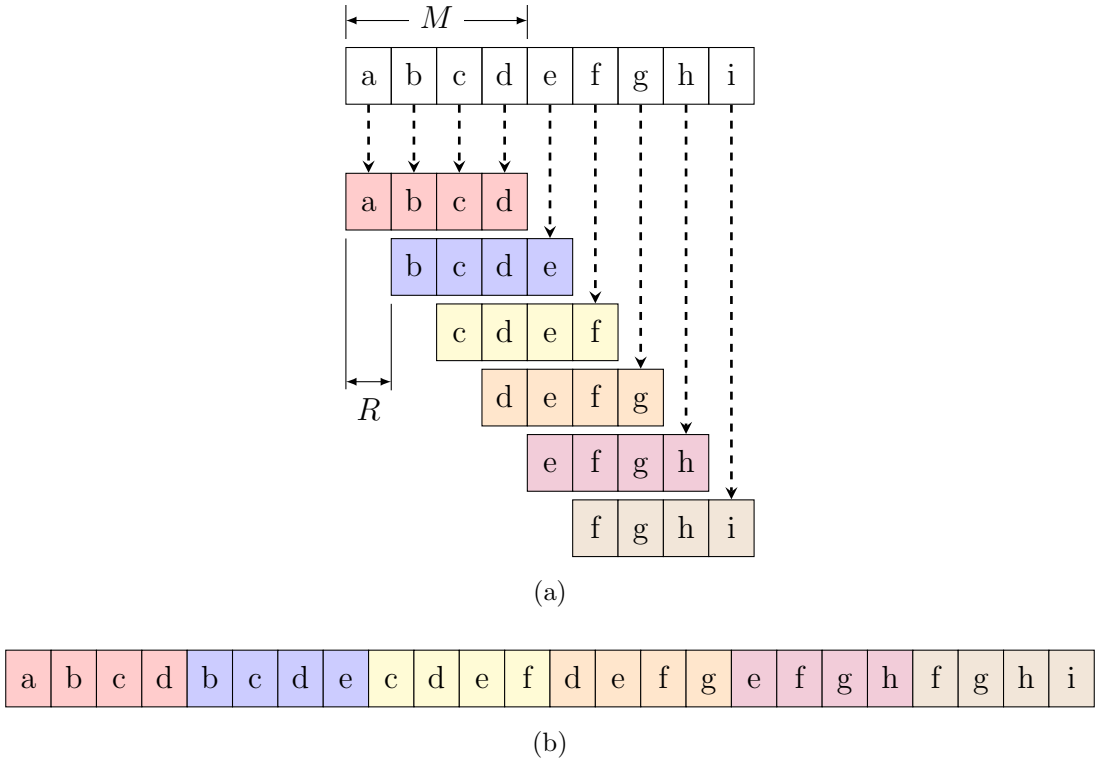


Figure 3.3: Time Compression process ($M = 4$, $R = 1$) (a) overlapping windows of a message signal producing (b) Time Compressed message signal

synchronized chip code in the receiver. In the case of TC-OLA instead of using a random code to produce redundancy, the message itself is time compressed and used.

It should be noted that the TC-OLA algorithm can be applied, without modification, to both analog and digital signals alike. It can be seen in Figure 3.3b that discontinuities are introduced at the edges of each window or every M samples. Discontinuities result in spurious spectral artifacts and harmonics and can be limited by multiplying a window function to the M input samples prior to emitting them.

3.1.1.1 Windowing

Figure 3.3 shows the Time Compression process using rectangular windows for demonstration purposes; however, in practice, a windowing function of size M is multiplied by the M input samples to reduce the spurious spectral emissions caused by discontinuities in the time domain signal. These discontinuities occur at the edges of each window due to the hop of R samples, and is therefore advantageous to have a

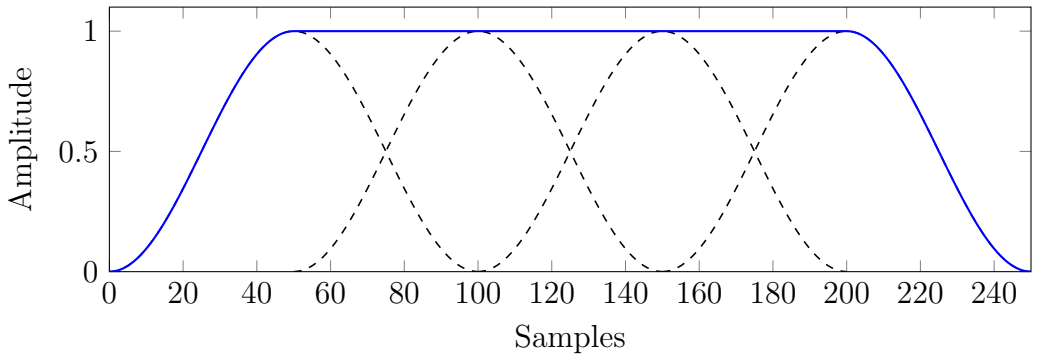
windowing function that tapers to zero at the beginning and end of the window. In addition to limiting discontinuities at window edges, it is also advantageous to select a windowing function that has the constant overlap add property [15]:

$$p = \sum_{k=-\infty}^{\infty} w(n - kR), \tag{3.1}$$

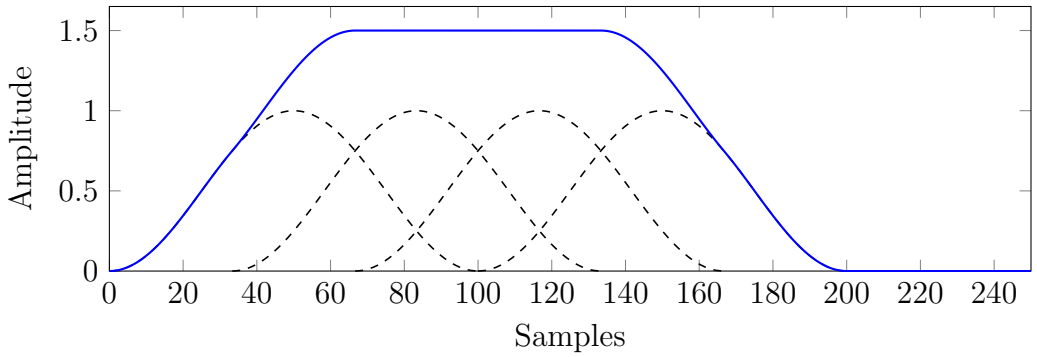
where p is a constant, w is the windowing function and R is the hop size between windows. Windowing functions that satisfy Equation 3.1 allow for simple normalization in the receiver, as the effect of the windowing function can be equalized via a scaling factor, p. Any windowing function can be used; however, based on its tapered shape and constant overlap add property (see Figure 3.4), the Hann window given by Equation 3.2 serves as an excellent choice for TC-OLA.

$$w_{hann}(n) = \frac{1}{2}\left(1 - \cos\left(\frac{2\pi n}{M-1}\right)\right). \tag{3.2}$$

To see the effect of the Time Compression process it is useful to visualize the impact the operation has on the spectrum of a signal. In the next section, a noise signal is shaped into a recognizable spectrum and used for visualizing the effect of the Time Compression with varying parameter values.



(a)



(b)

Figure 3.4: Result of overlap and add of 4 Hanning windows with (a) $\frac{1}{2}$ window overlap (b) $\frac{1}{3}$ window overlap

3.1.1.2 Time Compressed Spectrum

An empirical investigation of the spectral spreading effects of Time Compression can be seen clearly by using an input signal with a recognizable spectral shape. To produce a recognizable spectrum, white noise was shaped using a band-pass (from 500 Hz – 5.5 kHz) and a band-stop (from 1.7 kHz – 3.5 kHz) filter. This input signal was sampled at 11.1 kHz, which is precisely the *Nyquist rate* of this band limited signal (see Figure 3.5a). It can be seen from Figures 3.5b–3.5g that the spectral envelope of the Time Compressed signal is a $\frac{M}{R}$ point approximation of the original message signal's spectral envelope. Though the spectral envelope retains the shape of the original signal, the overall bandwidth is increased by the aforementioned spreading factor ($\frac{M}{R}$). To ensure that the same behavior holds true for different values of R, the same input signal was processed with TC-OLA while holding $M = 32$ and varying R to produce Figures 3.6a–3.6d.

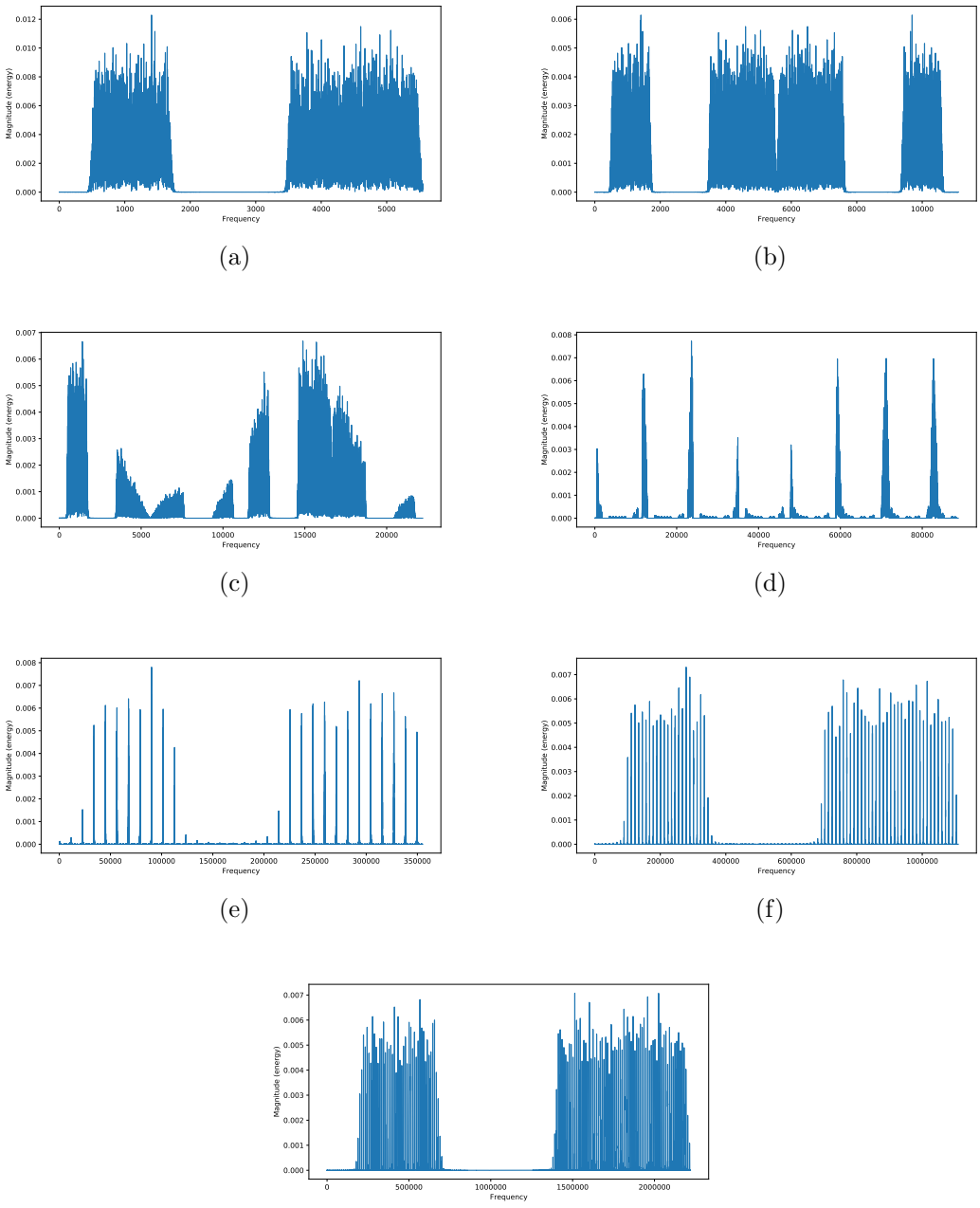
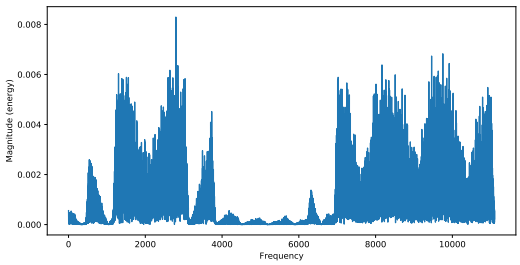
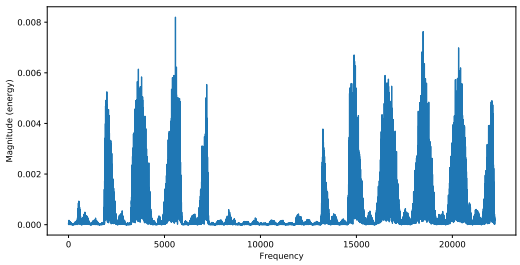


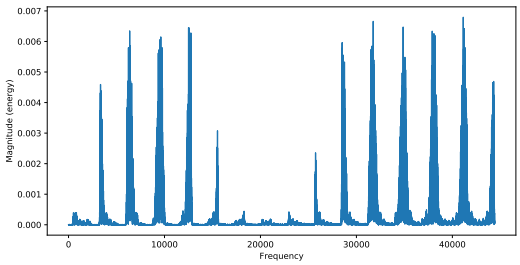
Figure 3.5: (a) Original signal spectrum ($F_s = 11\,\text{kHz}$) vs Time Compressed signals (b) M=2, R=1 (c) M=4, R=1 (d) M=16, R=1 (e) M=64, R=1 (f) M=200, R=1 (g) M=400, R=1



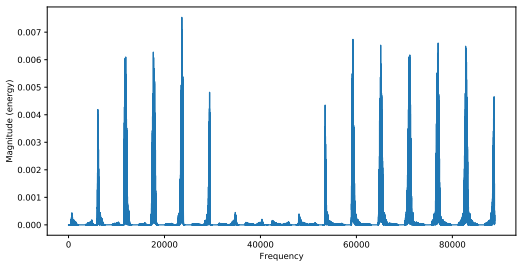
(a)



(b)



(c)



(d)

Figure 3.6: Time Compression of signal in Figure 3.5a with varying R value (a) M=32, R=16 (b) M=32, R=8 (c) M=32, R=4 (d) M=32, R=2

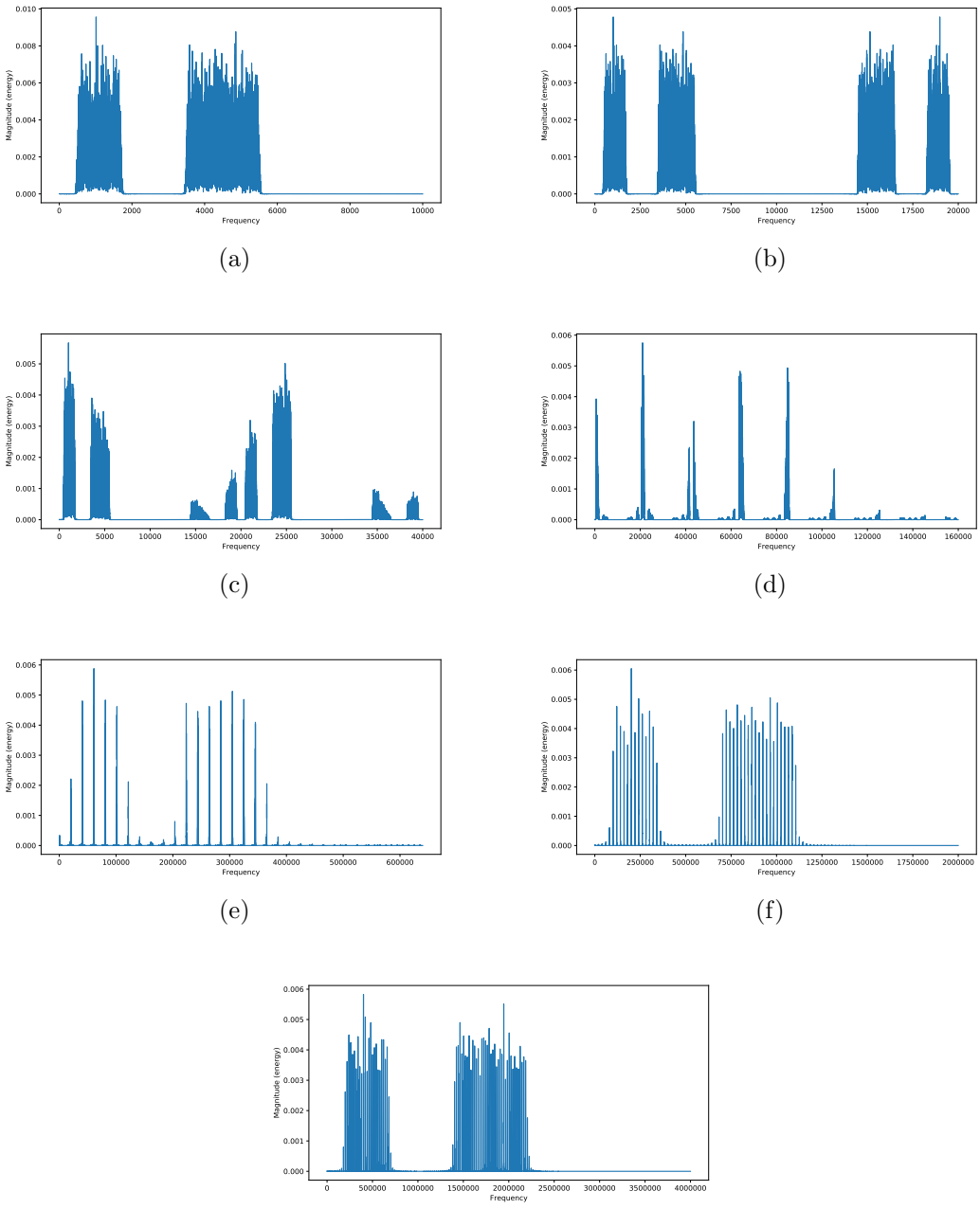


Figure 3.7: (a) Original signal spectrum ($F_s = 20\,\text{kHz}$) vs Time Compressed signals (b) M=2, R=1 (c) M=4, R=1 (d) M=16, R=1 (e) M=64, R=1 (f) M=200, R=1 (g) M=400, R=1

In Figure 3.5, the band limited input signal was sampled at the *Nyquist Frequency*; the spectrum was fully utilized to represent the message signal. The effect of oversampling within the TC-OLA processing chain was explored within Figures 3.7b–3.7g. The same shaped noise signal from Figure 3.5a was sampled at 20 kHz, well above the *Nyquist rate* of 11 kHz. Since the input noise signal is band limited, this increased sampling rate provides no new information and simply expands the maximum frequency in the spectral representation shown in Figure 3.7a. The spectrum of the oversampled signal is no longer fully utilized as shown by the zero portions of the spectrum outside of the message signals bandwidth. Time Compression on the oversampled signal results in two undesirable consequences:

- The overall sampling rate of the Time Compressed signal is almost double that of the perfectly sampled signal, requiring more processing power to handle the output sampling rate
- The spectral inefficiency is multiplied by a factor of $\frac{M}{R}$ since the Time Compression spectrum exhibits the same spectral envelope of the message signal

This result from oversampling (shown in Figure 3.7), shows that for performance, both computationally and spectrally, the sampling rate should be selected to be as close to the Nyquist rate of the input message signal as possible.

3.1.1.3 Overlap and Add

At the receiver, each of the windows of length M within the received signal are overlapped and added using the same hop size R used in the transmitter, as shown in Figure 3.8. From a sampling rate perspective, this can be viewed as a decimation by a factor of $\frac{M}{R}$. There is however an initial delay ($\frac{M^2}{R}$ samples) caused by the redundancy introduced by the Time Compression process and requires that $\frac{M}{R}$ windows be acquired before the original signal begins to appear at the output of the Overlap and Add process. Once the algorithm has reached steady state, the output of the Overlap and Add process will be the same as the input message signal. Figure 3.8 shows a simplified version of the Overlap and Add process using rectangular windows and a scaling factor of $\frac{M}{R}$. The steady state output can be seen in the middle of the output window in bold (**d**,**e**,**f**).

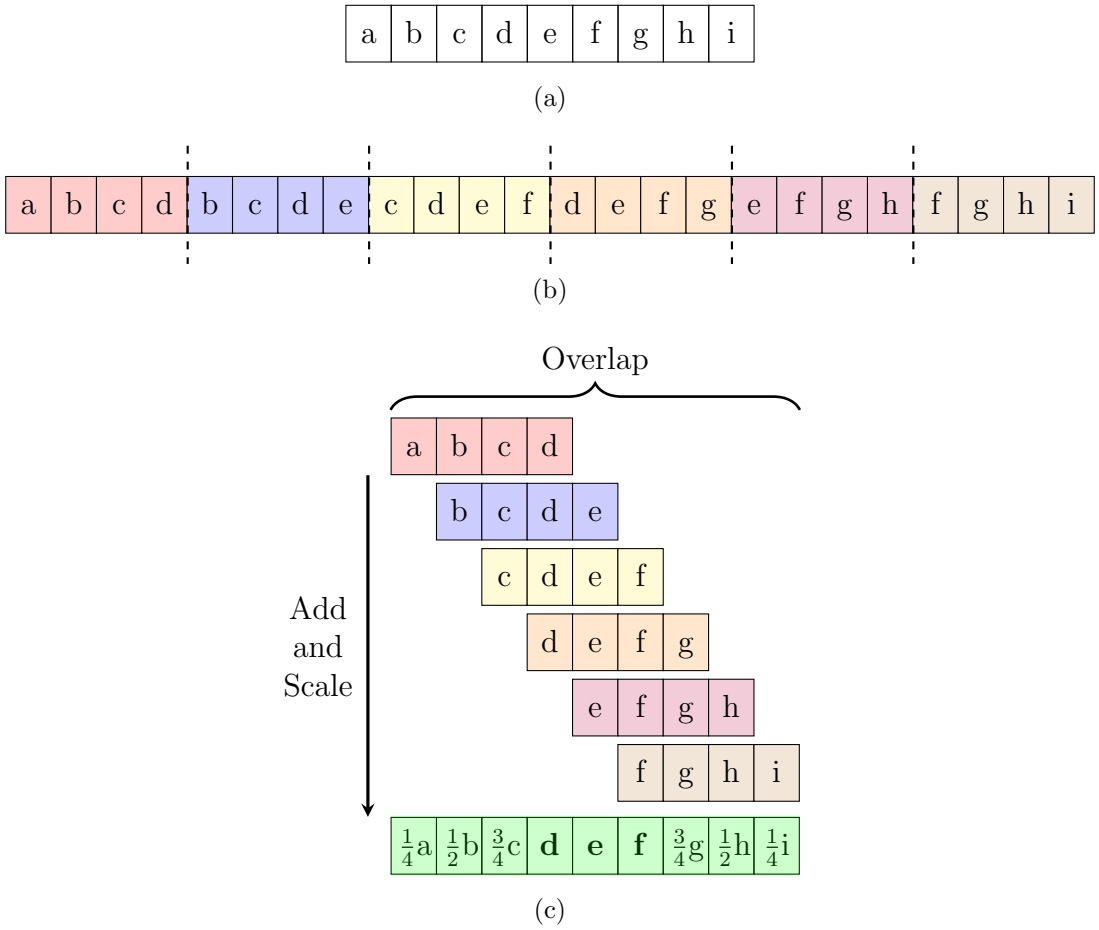


Figure 3.8: (a) Message signal, (b) Time Compressed signal, (c) Overlap and Add process and result ($M = 4$, $R = 1$)

It is important that the Overlap and Add process starts each window at the correct position of the received signal to reconstruct the message. If the receiver windowing process is off by even a single sample, the received window will be comprised of samples from two transmitted windows. Unlike other spread spectrum modulations, TC-OLA can still recover much of the message signal even when running out of sync. When using the TC-OLA algorithm in practice, R is fixed at a value of 1, which means that adjacent transmitted windows will share $M-1$ of the same samples. Even when receiver is out of sync by N samples, there are still $M-N$ samples in each received window that Overlap and Add to recreate a distorted version of the original message (see Figure 3.9). The degree to which the received signal is distorted will depend on the number of samples, N, by which the receiver is out of sync. It can be seen in Figure 3.9, that the source of this distortion is the addition of samples from the adjacent window. In practice however, since these samples are from the edge of the adjacent window, they will be attenuated by the tapered tail of the window function (see Section 3.1.1.1). This means that TC-OLA offers a graceful degradation with respect to discrepancies in receiver synchronization and greatly relaxes the synchronization requirements [16, 2.3.3]. In contrast to this graceful degradation, recall that DSSS (see Section 2.3.5.2) requires perfect synchronization of the chip code within the receiver, otherwise no message can be recovered from the received signal.

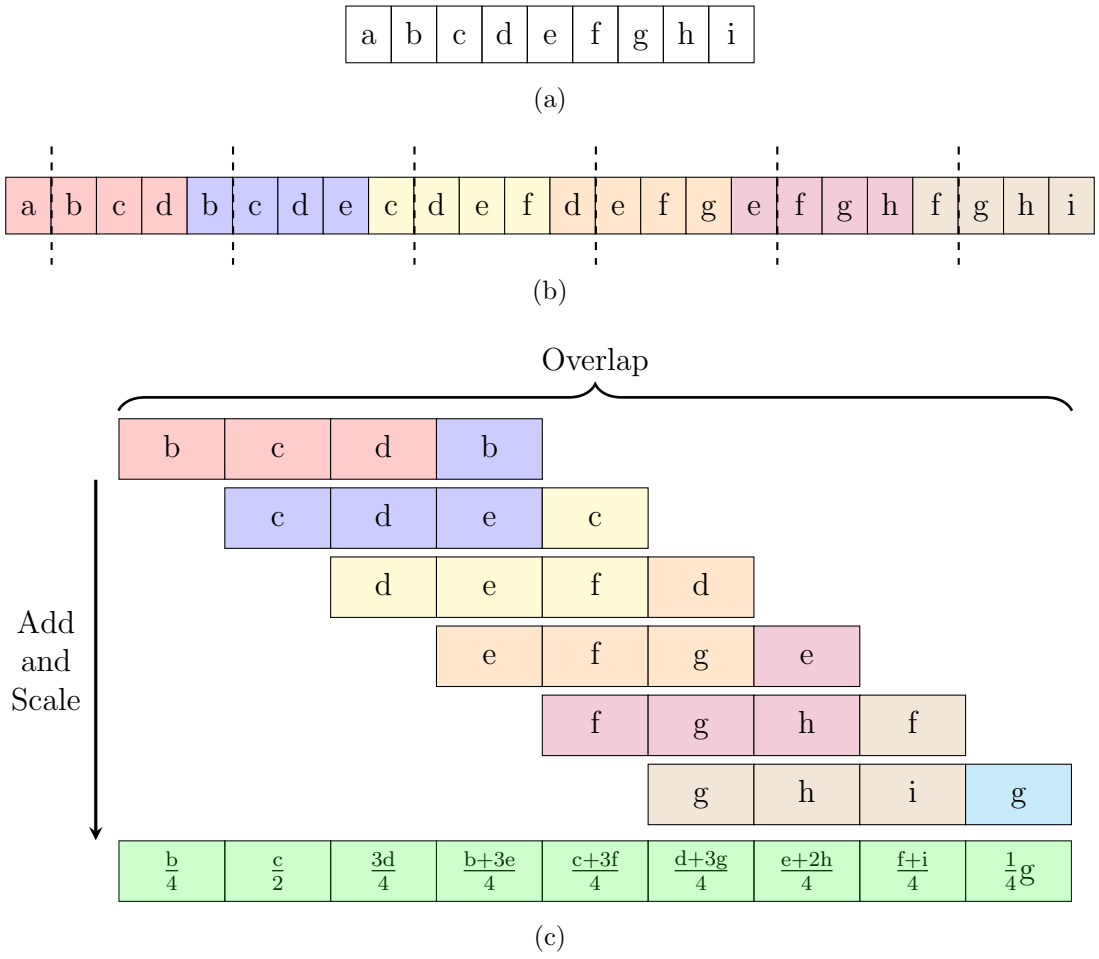


Figure 3.9: (a) Message signal, (b) Time Compressed signal, (c) Out of sync Overlap and Add process and result ($M = 4$, $R = 1$)

3.2 GR-TCOLA - GNU Radio Block

Using the experience and knowledge gained from developing the gr-flex block, which provided a complex baseband source block for the Flex radio system (see Appendix D), it was decided that implementing TC-OLA blocks for GR would allow for faster iteration and experimentation. The result of this development was the *gr-tcola* project [1] which consists of two highly optimized GR processing blocks: the Time Compression block and the Overlap and Add block. The two blocks were written in C++ to provide the best signal processing performance possible. The blocks were developed using a test driven approach; the tests were initially written against the original python based algorithm and then used to verify the C++ implementation as it took shape. The algorithms were then further simplified by leveraging the built in signal processing infrastructure provided by GR. This makes the implementation of these algorithms less portable, however, the scheduling and buffer management could likely be abstracted in order to leverage the same implementation for use outside of GR.

3.2.1 GNU Radio (GR)

GR provides a signal processing platform that allows developers to design real-time signal processing blocks without concern about the plumbing necessary for using these blocks in an application. Blocks written for GR can be easily wired together, into what's called a Flowgraph, using either Python directly, or by using the GNU Radio Companion application, which provides a simple to use Graphical User Interface (GUI). GR is a multithreaded run-time with a scheduling subsystem that keep sample buffers moving between blocks within a Flowgraph. As with any framework or run-time, GR comes with many built in primitives which give block developers a solid foundation from which to build their processing units. These primitives are available as base classes in GR which automatically provide many of the necessary settings that the scheduler uses to optimize the flow of samples in and out of the block (see Table 3.1).

For the TC-OLA algorithm, the interpolator and decimator base classes were used to build the Time Compression and Overlap and Add blocks respectively.

[1] Repository located at https://github.com/mistic-lab/gr-tcola

Block	Samples (In:Out)	Description
Synchronous	$1:1$	A block that produces one output for every input sample
Interpolator	$1:N$	A block optimized for producing N output samples for every 1 input sample
Decimator	$N:1$	A block optimized for producing 1 output sample for every N input samples
General	$N:M$	General block with no scheduling optimizations. Can be used to create blocks with any input/output sample ratio needed.

Table 3.1: Types of GNU Radio Blocks

3.2.2 TC-OLA Blocks

This section describes the implementations of the Time Compression block and the Overlap and Add block in further detail. Throughout this section, the parameters for Window Size and Hop Size will be represented with M and R respectively.

3.2.2.1 Time Compression Block

As discussed in Section 3.1.1, the Time Compression algorithm spreads a signal by concatenating overlapping windows of the input to form the output signal, effectively increasing the sampling rate by a factor of $\frac{M}{R}$. This process is an interpolation, which produces M samples of output for every R samples of input; therefore this block [23, *include/tcola/time_compression.h*] inherited from the base Interpolator block with an interpolation factor of $\frac{M}{R}$. This forces M to be divisible by R for this implementation; however, in practice this has not proven to be of large consequence since R is usually fixed at 1. In addition to the base scheduler optimizations provided by the Interpolator block, the properties shown in Table 3.2 were also set to optimize the scheduling for the Time Compression block [23, *lib/time_compression_impl.cc*].

The *Output Multiple* optimization constrains the sample flow through the Time Compression block and forces GR to only ever request multiples of M samples to be produced by the block. Since Time Compression block is an interpolator, the *Output*

Setting	Value	Description
Output Multiple	M	How many output samples to request from the block
History	$M - R + 1$	How many previous samples to provide to the block to process

Table 3.2: Time Compression Scheduler Settings

Multiple ensures that the block will only ever be given multiples of R new samples to process. Using the *history* optimization in GR, blocks allow GR's scheduler to efficiently manage buffers to each block throughout the Flowgraph in an optimal way. The default *history* for a block is 1, which represents that only the new samples will be passed to the block. In the case of Time Compression, the block will already receive R new samples to process, and a *history* of $M - R + 1$ ensures that a total of M samples will be passed into the block for processing for every R samples of input. This means that the GR scheduler is effectively implementing the entire hopping window portion of the algorithm. With the hopping window portion taken care of by GR, the Time Compression algorithm simplifies to multiplying a window function to M input samples and copying them to the output for every R new samples at the input.

The implementation of *gr-tcola* blocks support custom window coefficients [23, *lib/tcola_base.cc*], which may be passed in as parameters. These window coefficients must be the same for the Time Compression block and Overlap and Add block in order for the signals to be reconstructed correctly. For reasons discussed in Section 3.1.1.1 and in [18], the Hann window is an excellent choice for TC-OLA. Using the square root Hann window allows for the response of the Hann window to be split between the transmitter and receiver, the total response of these filters in cascade becomes a normal Hann window. Splitting a filter between transmitter and receiver is common practice in RF communications helping minimize spectral leakage in the transmitted signal and suppress noise at the receiver.

3.2.2.2 Overlap and Add Block

The Overlap and Add algorithm is used to despread the Time Compressed signal back into its original form. The block takes M samples and overlaps and adds them with previous $\frac{M}{R}$ windows of M samples producing R samples, effectively reduc-

ing the sampling rate by a factor of $\frac{M}{R}$. This process, which is a decimation by a factor of $\frac{M}{R}$, was implemented using the Decimation block as a base class [23, *include/tcola/overlap_add.h*]. In addition to the optimizations implemented by the base class, the properties shown in Table 3.3 were adjusted to further optimize the block.

Setting	**Value**	**Description**
Output Multiple	R	Hint for Scheduler to make requests for multiples of full windows

Table 3.3: Overlap and Add Scheduler Settings

Since the block is implementing the Decimation base class and is also requesting that multiples of R samples to be produced, GR will ensure that the block will always be given $R * \frac{M}{R} = M$ samples as input. Therefore, the block will always be receiving full windows of samples that can then be overlapped with previous windows.

The Overlap and Add process is done using an internal buffer, of length M, within the block [23, *lib/overlap_add_impl.cc*]. This buffer stores the previous result of the Overlap and Add. When M samples arrive at the input of the block, they are windowed and then overlapped and added with the internal buffer, which is initially full of zeros. After the Overlap and Add process, R samples are taken from the front of the buffer, normalized and transferred to the block's output. Thus, M input samples produce R output samples. Once the samples are copied from the internal buffer to the output, the buffer is rotated by R samples, effectively removing R samples from the front and adding R zeros to the end of the buffer, while retaining the original buffer size; after which, the process repeats itself for the next M samples. The normalization of the output signal is necessary to account for the processing gain introduced by overlapping and adding $\frac{M}{R}$ copies of the same signal, as visualized in Figure 3.8. The Overlap and Add process requires $\frac{M}{R}$ windows of M samples before the internal overlap and add buffer reaches steady state and begins to return the original message signal thus introducing a delay of $\frac{M}{R}$ samples (at the message signal's sampling rate).

3.2.2.3 Normalization and Processing Gain

As discussed in Section 2.3.5, spread spectrum techniques spread signals by introducing redundancy that increases the overall sampling rate. This redundancy leads to

increased signal power in the received signal, known as processing gain. In DSSS, the processing gain is achieved when the chip code is correlated against a correctly synchronized signal. In TC-OLA, the processing gain is achieved when overlapping windows of signal are added together in the receiver. The amount of processing gain at the receiver will depend on the size of the window, the amount of overlap, and the windowing function used. Recall the effect of adding overlapping Hann windows, shown in Figure 3.4 from Section 3.1.1.1. Normalization is the process of scaling the signal back down to its original amplitude by dividing the received signal by the processing gain. This normalization process attenuates the received signal along with any noise that was introduced within the RF channel, increasing the SNR and making the RF channel more robust against noise.

As stated earlier, the window used within the TC-OLA blocks is a parameter that can be set arbitrarily. The only requirement is that it is a vector of floating point numbers that is of length M. This flexibility, however, poses a challenge when it comes to accounting for the processing gain, since the processing gain is related to not only M and R but also the type of window being used at both ends. To overcome this, the Overlap and Add block calculates the gain produced by the windowing process. It does this when the Flowgraph initially starts, by instantiating an internal instance of the Overlap and Add block with the exact same parameters as itself [23, *lib/overlap_add_impl.cc*]. The block then allocates a buffer to hold $\frac{M}{R}$ window functions, which is essentially mimicking the output of the Time Compression block being fed a buffer of ones. These $\frac{M}{R}$ windows are fed into the internal Overlap and Add block, and the output signal is analyzed to find the peak value (*i.e.*the processing gain, used to scale the output of the Overlap and Add process).

3.3 Mobile Radio

Over the years, a great number of digital LMR protocols and standards have been designed and developed such as DMR [7], dPMR [8], NXDN [26], TETRA [6], Digital Smart Technologies for Amateur Radio (D-STAR) [19] [19], and P25 [27]. As discussed in Section 2.4.2.1, a digital radio protocol consists of the digital protocol (*e.g.* packet structure, headers, *etc.*) and the modulation used. This allows RF chip manufacturers to build chips that can be used for encoding/decoding packets for many different protocols. With intention of designing a production piece of hardware, it was deemed

that a hardware chip should be sourced and used for providing protocol support for the experiments. After investigating different options for digital radio chips, the CML-CMX7341 PMR/LMR Baseband Protocol Processor (CMX-7341) [3] was found which can support analogue PMR, DMR, dPMR, NXDN, and P25. The CMX-7341 requires additional supporting infrastructure (*e.g.* micro-controller, RF front end, amplifiers) in order to be fully operational; therefore, the CML Microcircuits PMR Common Platform Demonstration board (CML-DE9945E) [4] was purchased to provide a digital mobile radio platform based on the selected CMX-7341 chip. The CML-DE9945E comes with an onboard ARM processor and RF frontend to provide a completely standalone, mobile radio implementation based on the CMX-7341 chip.

After investigating many of the different existing open radio standards, it was decided that the DMR standard would be used for the purpose of this research. DMR is a very common communication standard [7] used by many on amateur radio bands. It supports many features such as channels, talk groups, general data transmission, calling, and also optional support for centralized trunking [10]. DMR's air interface [9] defines the modulation scheme used to transmit the digital signal as an analog radio signal.

In the next Chapter, experiments involving the *gr-tcola* blocks and the CML-DE9945E were developed to determine feasibility of using TC-OLA to spread DMR into signals that are spectrally compatible with the regulations for the ISM band.

Chapter 4

Experiments

4.1 Overview

Using the *gr-tcola* blocks and the CML-DE9945E, the following experiments are intended to investigate the feasibility of using TC-OLA DMR within the ISM band. The experiments were broken into several different phases: data acquisition, TC-OLA processing and validation, spectral analysis, and performance analysis.

4.2 Data Acquisition

In order to provide deterministic results, DMR test signals were acquired by recording the output generated from the CML-DE9945E using the setup shown in Figure 4.1. Though the DMR protocol was selected for this experiment, the following should also apply equally to other protocols such as dPMR or even analog signals.

The CML-DE9945E acquired for this project was designed to be a standalone RF device that could operate within the PMR446 band (see Section C). This board was acquired to provide support for encoding and decoding the DMR signals, however, since it was designed for standalone operation, it introduced an unnecessary frequency conversion (*i.e.* to/from PMR446). This conversion was performed by a Universal Software Radio Peripheral (USRP) since the focus of this research was the feasibility of using TC-OLA to transparently spread and despread DMR into ISM compatible signals. In a production device however, signals would be converted from the ISM

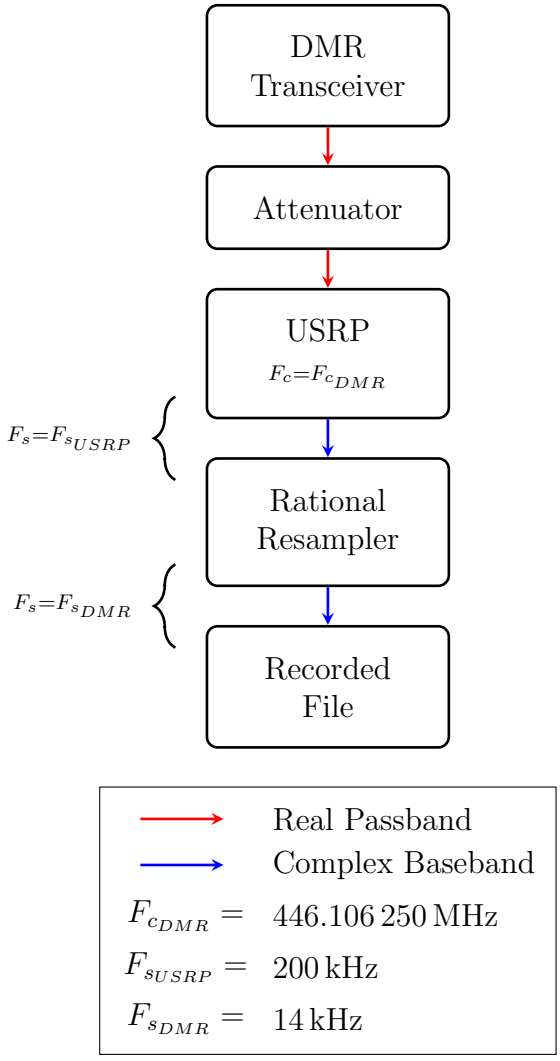


Figure 4.1: Data collection setup

into baseband and processed directly by the baseband processing chip. Test signals were generated by the CML-DE9945E and captured/recorded using a USRP N210 using GR. The USRP and the CML-DE9945E were both set to use Channel 1 of DMR446 (see C.2) and the CML-DE9945E was then placed in the respective testing mode to produce the test signals in Table 4.1. The only exception was the voice signal, which was created by playing a recorded audio clip into the audio input of the CML-DE9945E. In the case of audio, each audio frame is processed using the CML-DE9945E's on-board AMBE-2 codec to reduce the bit rate significantly.

Signal	**Description**
Test Tone	1 KHz Test Tone
Data Message 1	"DE9945 Data Message using rate 1/2 data"
Data Message 2	"Hello World from the DE9945"
Voice	Clip from Tom's Diner
PRBS	Pseudo-random stream of bits

Table 4.1: Acquired Test Signals and descriptions

For each test signal, the USRP was responsible for translating the received signal from passband at $F_{s_{DMR}}$ to the complex baseband. The resulting signals captured by the N210 were recorded using the GNU Radio *File Sink* block to store the complex numbers in a binary data file. The minimum sampling rate for the N210, which is based on the master clock and minimum decimation factor, is around 200 kHz. The bandwidth of a DMR signal is 12.5 kHz with a *Nyquist rate* of 25 kHz. As such, it became necessary to reduce the sampling rate and bandwidth of the complex baseband signal before recording. To do this, the rational resampler was used to resample the complex signal down to a rate of 14 kHz. At this point, an astute reader may notice that the sampling rate of 14 kHz is less than the required *Nyquist rate* of 25 kHz for a 12.5 kHz signal. This would be true if real signals were being recorded. However, when recording complex baseband signals, every sample consists of two floating point numbers representing the real and imaginary components; therefore, the effective sampling rate in terms of floating point numbers is actually 28 kHz, well above the required 25 kHz required.

A larger sampling rate could have been used here but maintaining a low sampling rate has several benefits. As discussed in Section 3.1.1, the Time Compression process is effectively an interpolation process, and as such the sampling rate after the Time Compression process is the input sampling rate multiplied by $\frac{M}{R}$. The higher the sampling rate, the more processing power or clock speed is required to move the buffers of samples through the system. Thus the lower the input sampling rate, the higher the $\frac{M}{R}$ ratio that can be used with the same amount of processing power. The second benefit of a lower sampling rate is explored in Section 3.1.1.2. Recall that when a signal is Time Compressed, the resulting spectrum is an $\frac{M}{R}$ point approximation of the original spectrum's envelope, spread across a $\frac{M}{R}$ times larger bandwidth. Since the DMR signal is a band-limited signal, increasing the sampling rate leads to wasted spectrum beyond the bandwidth of DMR signal. This wasted spectrum is then increased in bandwidth by $\frac{M}{R}$ as shown in Figure 3.7. After the test signals were acquired from the CML-DE9945E, the next step was to use these test signals as input to the TC-OLA blocks and confirm that the resulting signals could be correctly decoded by the CML-DE9945E without any issue.

4.3 TC-OLA Processing and Validation

In this phase of the experiment, the test signals generated in Section 4.2 were processed using the *gr-tcola* blocks developed (see Section 3.2) and validated using the CML-DE9945E. Using the setup shown in Figure 4.2, the test signals were used as complex baseband input to the TC-OLA processing chain. These complex baseband signals are the IQ streams that would be produced directly from the CMX-7341 [3] baseband processor chip in a production piece of hardware. This complex baseband signal is processed by Time Compression block with a specified window size and hop size denoted by M and R respectively. Channel modeling was not performed in this phase of the experiment, as clean signals were needed for accurate spectral analysis. However, a dashed placeholder block has been included in Figure 4.2 for a reference location. After optionally passing through a channel model, the "received" signal is then Overlapped and Added using the same values of M and R. After the Overlap and Add, the signal was resampled to the minimum sampling rate of the USRP (see Section 4.2) and sent to the CML-DE9945E to validate the signal integrity. Since DMR is a digital protocol, if the signal received by the CML-DE9945E is corrupted,

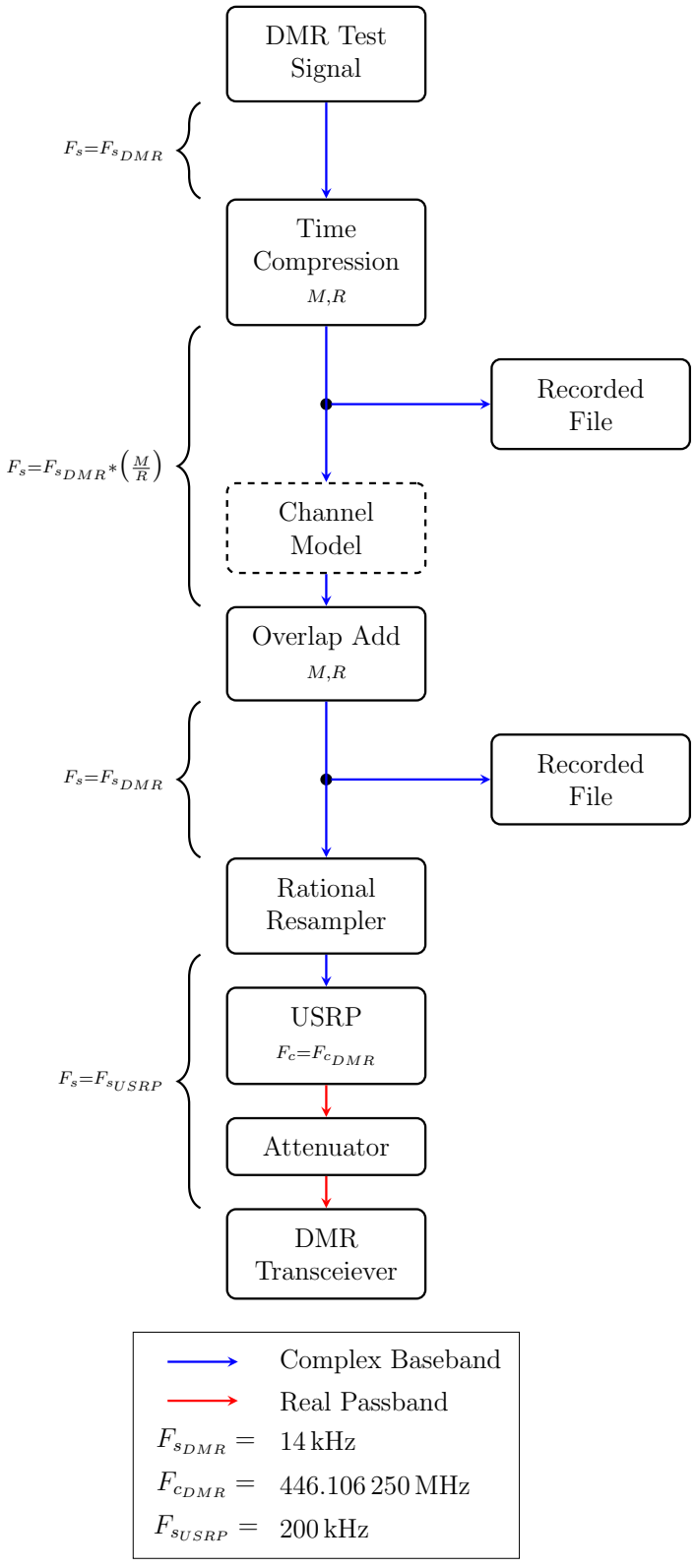


Figure 4.2: Experiment setup

it will fail to decode the frame and will output an error message. For successfully received audio frames, the CML-DE9945E outputs the audio out from the onboard speaker. For successfully received data frames, the CML-DE9945E reads the data packet and writes the American Standard Code for Information Interchange (ASCII) representation of the frame on the onboard Liquid Cystal Display (LCD). It can be seen from Figure 4.2, that recordings of the signal were created after the Time Compression and Overlap Add processing blocks. These recordings provide the signals necessary for performing more deterministic spectral density and channel modeling analysis. For each of the test signals, recordings were made for the M and R values shown in Table 4.2.

Window Size (M)	**Hop Size (H)**
32	1
128	1
512	1

Table 4.2: Spreading parameters used for experimental TC-OLA processing

The results of the validation are either pass or fail due to the binary nature of the decoding process on the CML-DE9945E. Table 4.3 below shows the results of this validation process.

Signal	$\frac{M}{R} = 32$	$\frac{M}{R} = 128$	$\frac{M}{R} = 512$
Test Tone	Pass	Pass	Pass
Data Message 1	Pass	Pass	Pass
Data Message 2	Pass	Pass	Pass
Voice	Pass	Pass	Pass

Table 4.3: Results of signal validation

4.4 Spectral Analysis and Occupied Bandwidth

In order for a device to legally be allowed to transmit within the ISM band, it must meet the regulations set forth by the FCC within Part 15 [11], which define not only

the types of emissions allowed within the band but also the characteristics of the transmitted spectrum, power constraints, and emission limitations. The ISM band is designated for license free use by devices that are both robust against interference and do not cause excessive interference within the band. The regulations for this band provide alternate rules for FHSS and non-FHSS digital modulations. The portion of the regulation that enforces the spectral quality for non-FHSS modulation states the following:

> "Systems using digital modulation techniques may operate in the 902–928 MHz, 2400–2483.5 MHz, and 5725–5850 MHz bands. The minimum 6 dB band-width shall be at least 500 kHz." [11, §15.247]

Following the procedures outlined in the ANSI ISM compliance procedure [1], the 99% Occupied Bandwidth (OBW) and 6 dB bandwidth were measured using an Avantest R3131 spectrum analyzer, which has built in utilities for measuring both the **X% OBW** and the **X**dB bandwidth. The **X% OBW** is defined as the bandwidth that contains **X%** of the total signal power as shown in Figure 4.3. The **X**dB **bandwidth** is the bandwidth of the signal within 6 dB of the peak power level as shown in Figure 4.4. Based on the procedures, the 99% OBW had to be measured in order to determine the correct values to be used for the Resolution Bandwidth (RBW) and Video Bandwidth (VBW) parameters of the spectrum analyzer.

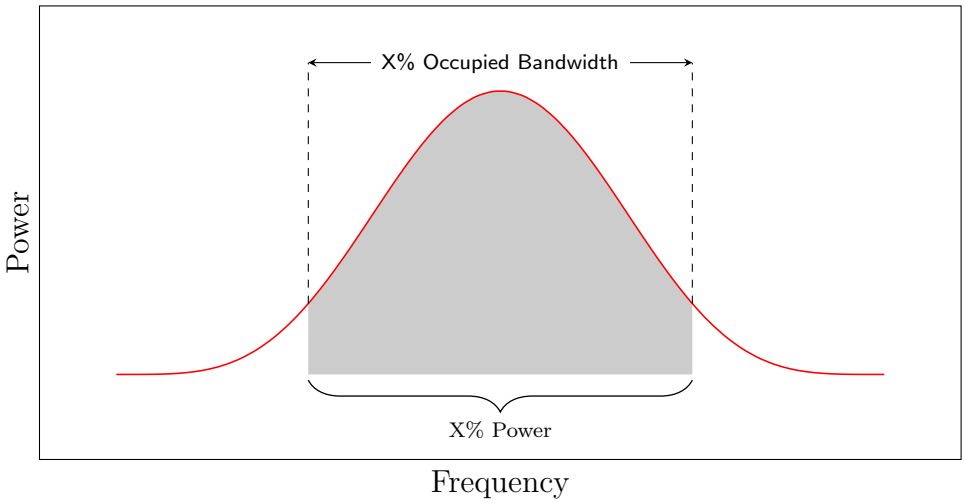


Figure 4.3: X% Occupied Bandwidth measurements

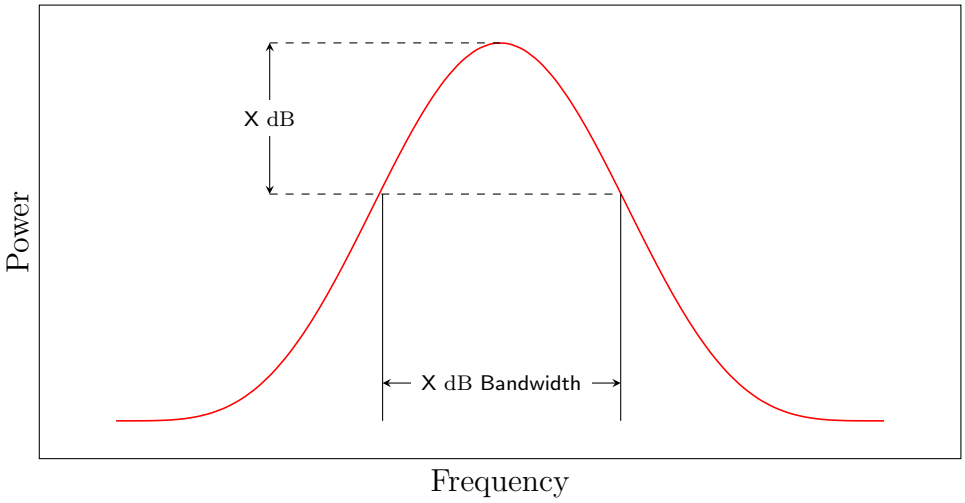


Figure 4.4: X-dB bandwidth measurements

In order to get an accurate reading of the OBW and the 6 dB bandwidth, the max hold tracing option needed to be used. The reason for this is that the original DMR signal is a Time Division Multiple Access (TDMA) modulation with a finite transmission duration. The spectrum analyzer has a sweep time that is based on the selected RBW and VBW parameters and is generally larger than that of the TDMA window. Using the peak hold tracing allows the spectrum analyzer to produce the maximum overall envelope of the signal over a larger portion of time. Measurements for the 99% OBW and the 6 dB were recorded once the spectrum analyzers output had settled to a constant value. Figure 4.5 shows a depiction of the setup used for performing the spectral analysis on the Time Compressed signals. The only reason for the Rational Resampler in this setup was for analyzing the original DMR signals, as their bandwidth was lower than the USRP's minimum sampling rate.

In this case, the signals used to perform the spectral analysis were the same signals that were recorded in Section 4.3. Three representative test signals (Voice, Data Message 1, and Data Message 2) were chosen for spectral analysis as they provided cases for sending data and audio using the Time Compressed DMR. The 99% OBW and 6 dB bandwidth of the original DMR modulated signal were also measured so that the empirical $\frac{M}{R}$ could be calculated by comparing the width of the Time Compressed spectrums to the original spectrum.

In addition to measuring the 99% OBW and 6 dB bandwidth of the original DMR signals, each of the recorded Time Compressed signals for the parameters in Table 4.2

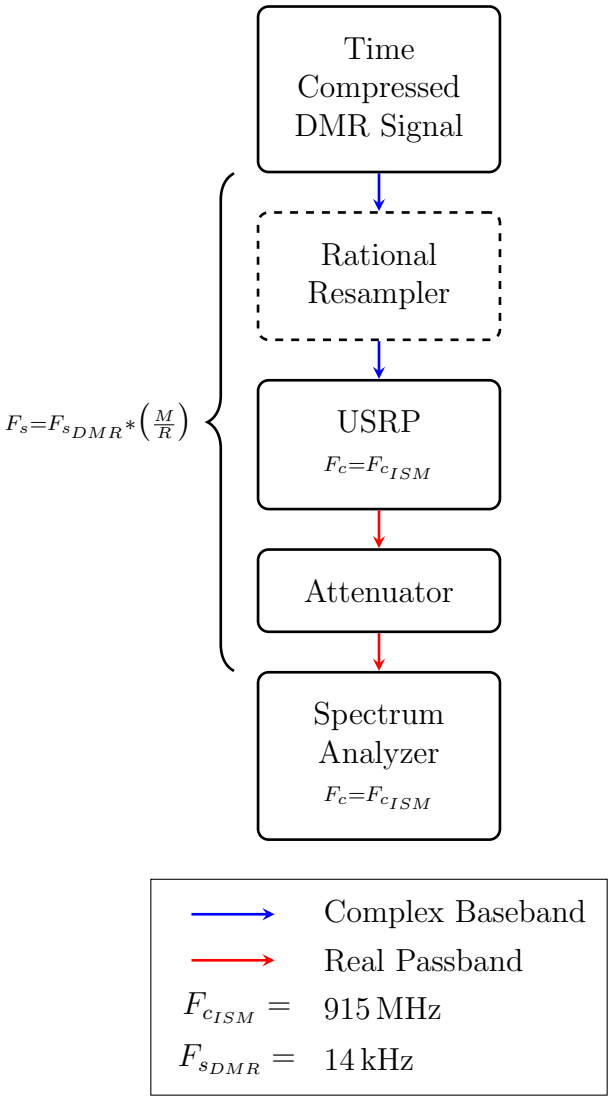


Figure 4.5: Spectral analysis setup

were also analyzed. For each of the signals analyzed, three separate measurements were performed on the spectrum analyzer to avoid any statistical anomalies.

4.5 TC-OLA DMR Performance

Before collecting metrics on performance of TC-OLA DMR, it is useful to first understand the basic structure of the DMR protocol so that an accurate model of performance may be selected. Protocols are often described as stacks and are made up of layers of services, each of which is built on top of the services provided by the layer below to provide an additional layer of abstraction for a particular domain of an application. Figure 4.6 shows the basic structure of DMR protocol stack.

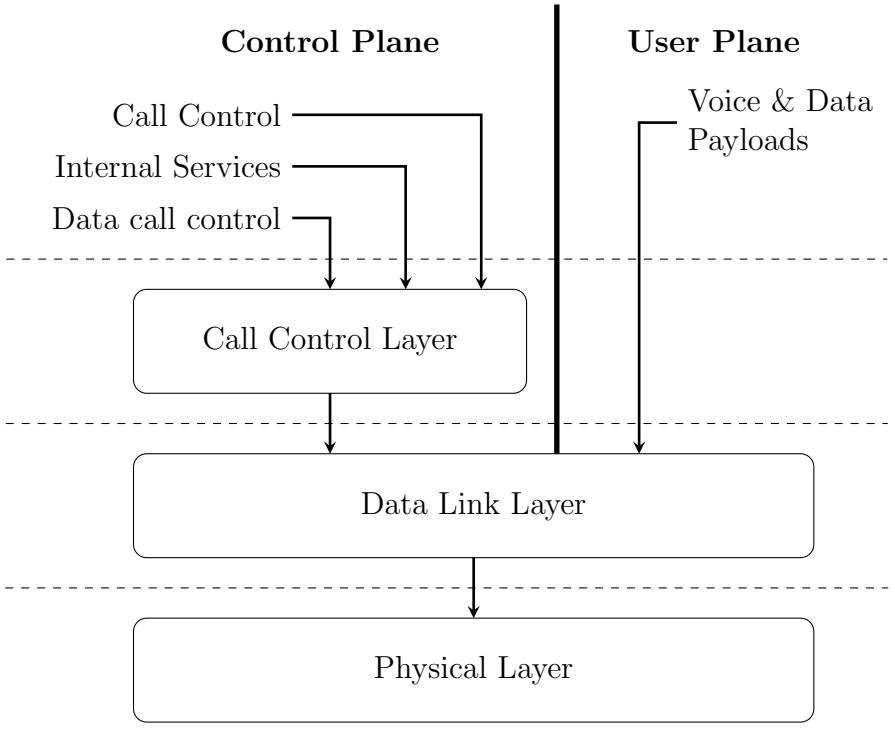


Figure 4.6: DMR Protocol Stack

The Call Control layer provides the call management services for setting up and managing calls, but is built upon the Data Link layer, which provides data and packet routing services. The physical layer, which is often referred to as the air interface, is responsible for providing the RF link by providing synchronization and modulation to the binary information being transmitted.

The ideal performance tests for this experiment would be done through the use of a DMR test set, which is a physical piece of hardware designed to measure statistics and performance metrics particular to a digital radio protocol. These test sets are useful for exercising and validating the various layers of the protocol and are used for testing and validating hardware designed to support a particular protocol. The test sets can be used for evaluating protocol specific metrics like packet errors, Frequency Shift Keying (FSK) error, symbol deviation, clock errors, and frequency drift. Unfortunately, a test set could not be acquired for this experiment, and as such it was decided that acquiring BER measurements of the physical layer modulation through TC-OLA would serve as an accurate proxy to the performance of DMR through the same process.

4.5.1 DMR Physical Layer

The physical layer of DMR defines [9] the modulation at a rate of 4800 symbols/s with each symbol consisting of 2 bits. The modulation used by DMR is 4 level Frequency Shift Keying (4FSK) with a maximum frequency deviation of 1944 kHz. Table 4.4 shows the mapping between bits, symbols, and the resulting frequency deviation.

Bit 1	**Bit 0**	**Symbol**	**4FSK deviation**
0	0	+1	+648 kHz
0	1	+3	+1944 kHz
1	0	-1	-648 kHz
1	1	-3	-1944 kHz

Table 4.4: Bit Symbol mapping to 4FSK deviation

The 4FSK modulation defined in the DMR protocol specification [9, 10.2.2], uses a pulse shaping filter to change the shape of the symbol stream from impulses into a more spectrally efficient shape before they enter the frequency modulator. In the case of DMR, and in many other digital modulations, the RC filter is used to shape the symbol stream as it provides excellent Intersymbol Interference (ISI) suppression. A Common practice in communications is to break a filter response in half and use it in both the transmitter and receiver so that the cascaded response of the two filters creates the desired response, as first shown in Section 3.2 with the square root Hann

window. The Root Raised Cosine (RRC) filter, when cascaded together, produces the response of a single RC and is therefore used in both the transmitter and receiver of the DMR physical layer. An overview of RC filters and their role in pulse shaping will be presented so that the DMR physical layer can be more accurately understood and modeled for BER tests.

4.5.1.1 Raised Cosine Filtering

ISI occurs as a result of symbols bleeding into one another and decreases the probability of receiving the correct symbol in the receiver. To understand why the RC filter is so effective at suppressing ISI it is useful to investigate its time and frequency domain representations, the mathematical descriptions of which can be seen in Equations 4.1 and 4.2 respectively.

$$h(t) = \begin{cases} \frac{\pi}{4T}\,\mathrm{sinc}\left(\frac{1}{2\beta}\right), & t = \pm\frac{T}{2\beta} \\ \frac{1}{T}\,\mathrm{sinc}\left(\frac{t}{T}\right)\frac{\cos\left(\frac{\pi\beta t}{T}\right)}{1-\left(\frac{2\beta t}{T}\right)^2}, & \text{otherwise} \end{cases}, \tag{4.1}$$

$$H(f) = \begin{cases} 1, & |f| \leq \frac{1-\beta}{2T} \\ \frac{1}{2}\left[1+\cos\left(\frac{\pi T}{\beta}\left[|f|-\frac{1-\beta}{2T}\right]\right)\right], & \frac{1-\beta}{2T} < |f| \leq \frac{1+\beta}{2T} \\ 0, & \text{otherwise} \end{cases}. \tag{4.2}$$

In Equation 4.1 and 4.2, T represents the symbol duration and β is a parameter known as the roll off factor. Through inspection, one can see that as β approaches 0, the frequency response in Equation 4.2 approaches that of an ideal rectangular response. The time frequency duality forces a trade off between the ideal frequency response and filter length. In the case of $\beta = 0$, the RC filter would need infinite taps to represent the rectangular frequency response without oscillations in the frequency domain at the discontinuities. Figure 4.7 shows the impulse response and frequency response of the RC filter for various β values. It can be seen in Figure 4.7a that $h(t)$ is periodic with zero crossings at $t = nT$, where n is any integer value. This property is what makes the RC filter so effective at suppressing ISI since the peak energy of a symbol is centred at the zero crossings of the symbols surrounding it.

Figure 4.8 shows the result of pulse shaping on a train of 5 4-level symbols. It

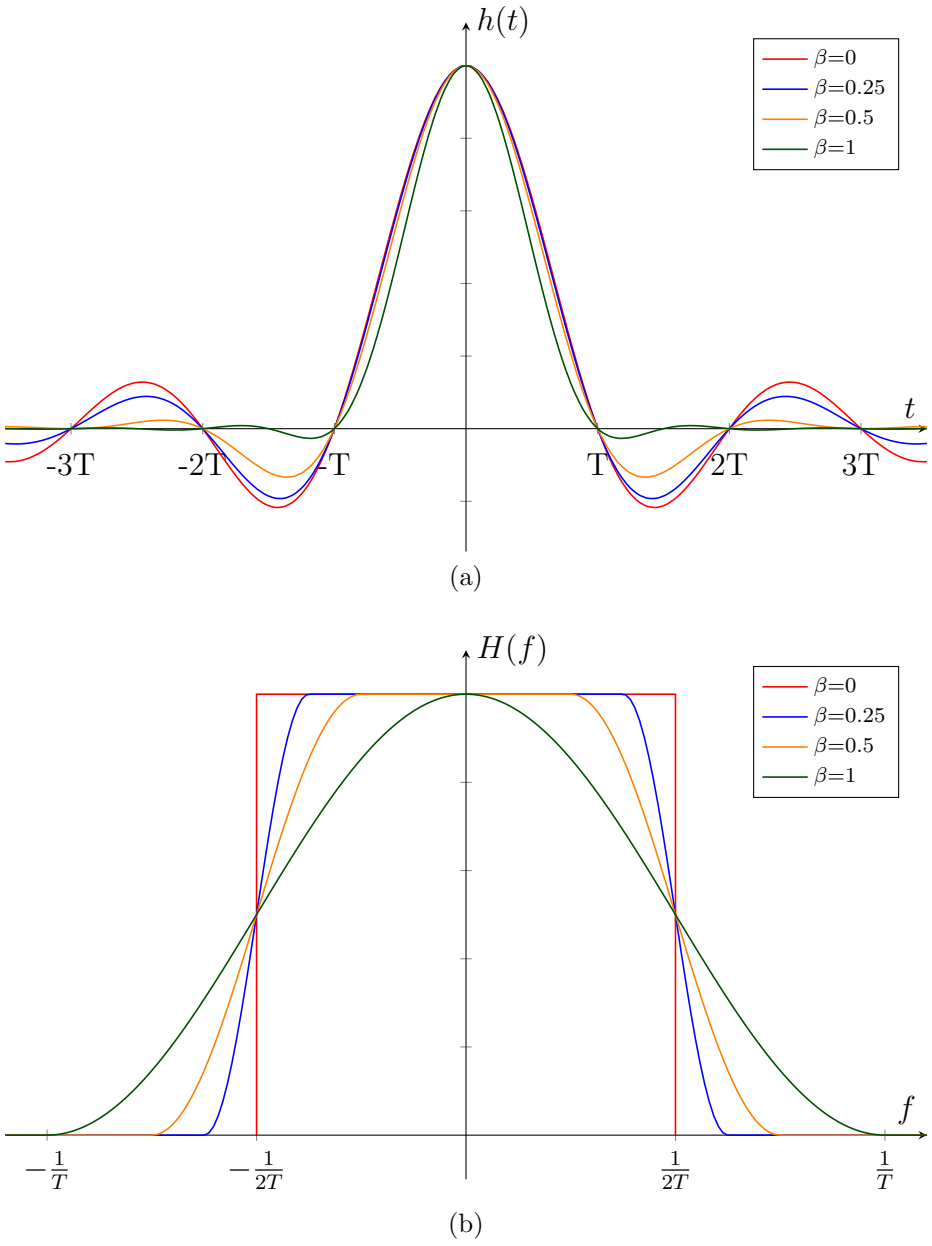


Figure 4.7: The RC function in the (a) Time Domain and the (b) for various values of β.

should be noted that in this figure the RC filter responses have been time shifted to be centered on the symbols to show the relationship to each impulse but would normally be delayed by half the filter length.

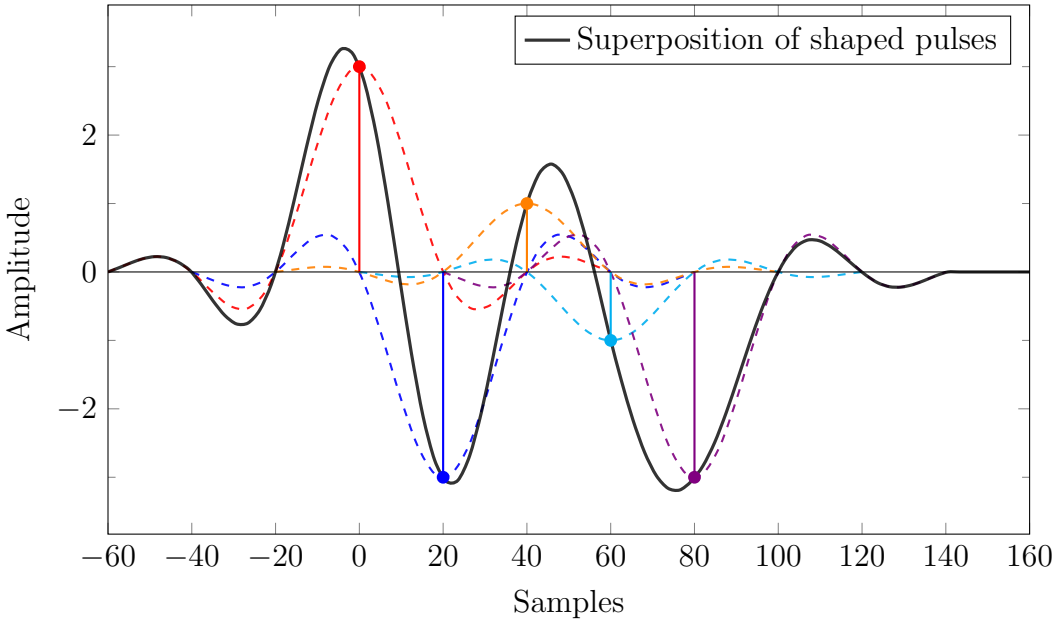
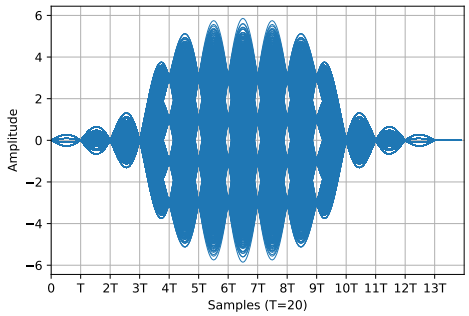


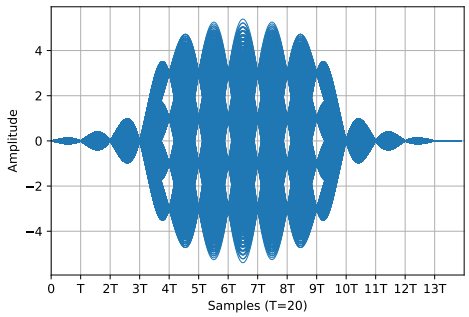
Figure 4.8: The result of RC pulse shaping on a train of 4-level symbols, with T_{symbol} = 20 samples and filter length of $6T_{symbol}$ samples

Eye diagrams are often used to show the impact of pulse shaping on the reception of symbols at the receiver. Eye diagrams are constructed by graphing every permutation of symbol levels for a particular number of symbols and find their namesake from the eye shapes that are produced by the absence of signal. These openings in eye diagrams show the not only the optimal location to sample the pulses for most accurate reception, but also give an indication of how difficult it will be to receive a symbol via the size of the opening in the diagram. In the case of of 4FSK, there are 4 potential symbols being transmitted. In order to view the eye diagram at steady state, it is required to generate permutations for a number of symbols greater than the number of potential levels. Figure 4.9 shows the eye diagram for 4 symbol levels with varying levels for the RC filter's β parameter.

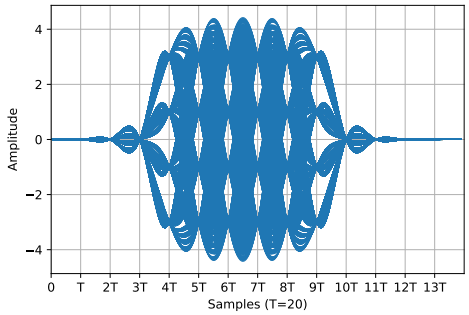
All digital radio protocols eventually use a digital modulation for transmitting the symbols or bits over the air. It was important to determine the impact of TC-OLA on the performance of digital modulations. A generalized BER analysis framework was developed for measuring the BER of digital modulations with and without TC-OLA.



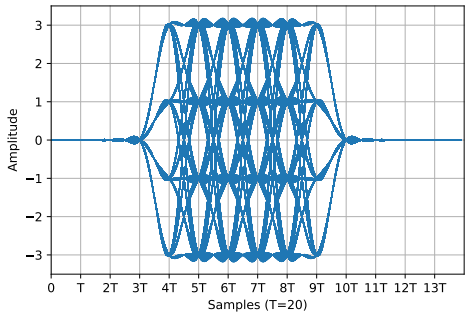
(a)



(b)



(c)



(d)

Figure 4.9: 5 symbol eye diagrams for 4 level modulation (a) $\beta = 0$ (b) $\beta = 0.25$ (c) $\beta = 0.5$ (d) $\beta = 1.0$

4.5.2 Bit Error Rate (BER) Measurements

BER refers to the rate at which errors occur in the resulting decoded bit stream within an RF communication system. BER's are generally compared using graphs that show BER vs $\frac{E_b}{N_0}$, where E_b is the energy per bit and N_0 is the power spectral density of the noise signal. This $\frac{E_b}{N_0}$ is a normalized axis that makes it possible to compare digital modulations with different bandwidths and bits per symbol since amplitude of the noise is normalized in terms of E_b. BER is calculated by comparing a received bit stream with the original stream of bits calculating the rate of error between the two streams. It is therefore, necessary to account for any delay introduced by a modulation's processing chain in order to have both bit streams synchronized. In addition to accounting for modulation delay, the E_b must be calculated in order to determine the value of N_0 and subsequently the amplitude of the Additive White Gaussian Noise (AWGN) being added in the channel. Both the delay and E_b must be accounted for to generate an accurate BER curve for any modulation (see Figure 4.10 for simulation block diagram). The TC-OLA process presents challenges as it injects additional delay into the processing chain as well as significantly alters the E_b of the transmitted signal. This change in E_b is due to redundancy introduced by the Time Compression and the window coefficients used in the process. Each symbol will be repeated $\frac{M}{R}$ times but also scaled by the window coefficient coinciding with the symbols location in each window. Before the BER simulations could be run, a framework needed to be established that would allow for the analysis and simulation of a generalized modulation.

4.5.2.1 Simulation Infrastructure

In order to support the simulating and analyzing various modulations with and without TC-OLA, a generalized transceiver interface was constructed using GR (see Figure 4.11). Using this generalized block interface, an efficient multi-core simulation framework was developed[1] to efficiently calculate BER curves for general modulation transceivers.

This generalized block structure allowed for a more general approach to determining the inherent delay and energy per bit (E_b) of any modulation via measurements in place of theoretical calculations. The delay measurement Flowgraph is depicted

[1] https://github.com/mistic-lab/pybergr

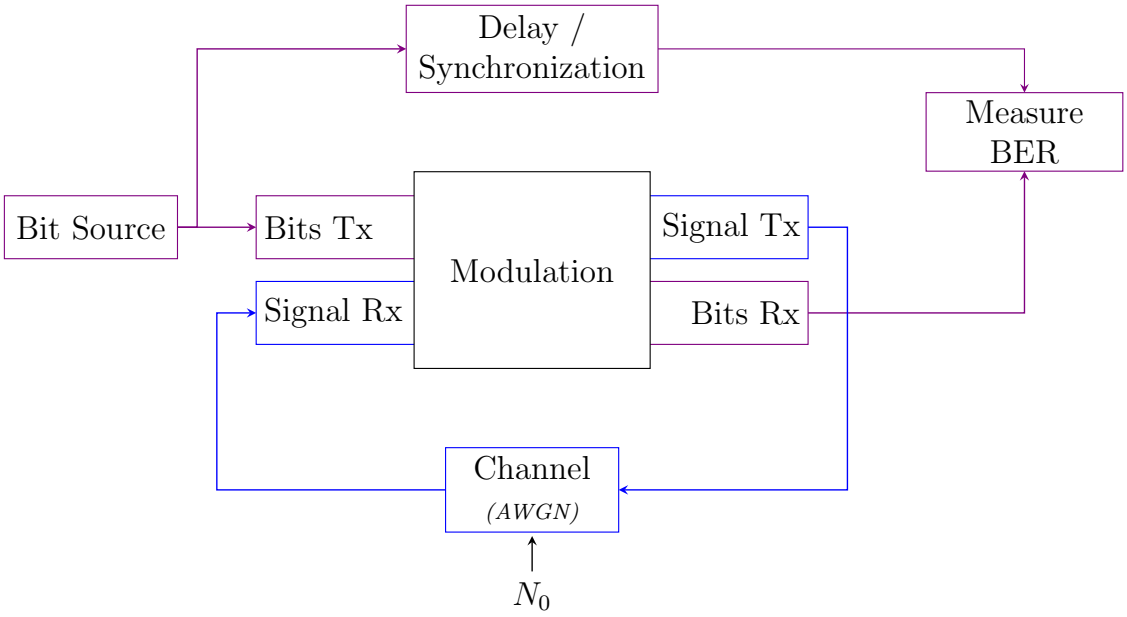


Figure 4.10: Generalized BER measurement Flowgraph

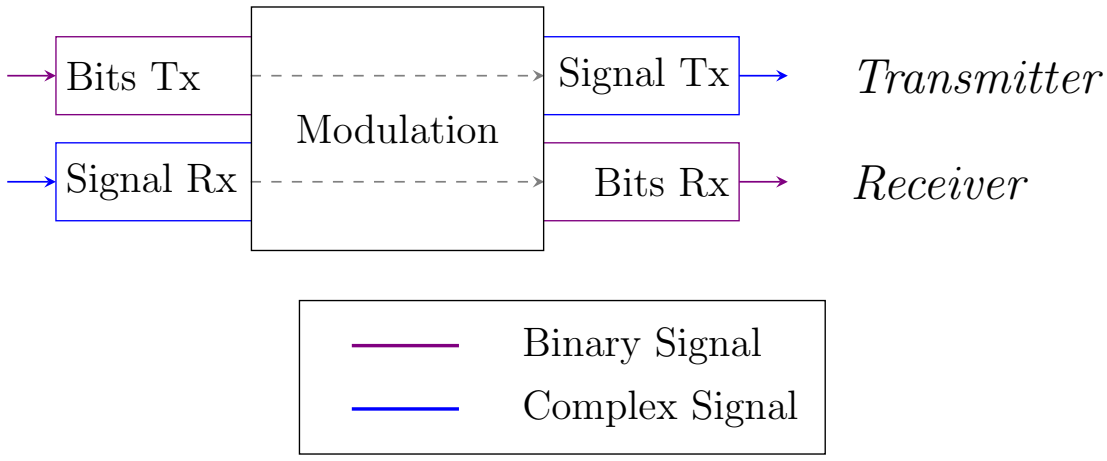


Figure 4.11: Generalized modulation structure

in Figure 4.12a and involves passing a known bit stream as input to the modulation and calculating the time difference between the transmitted and received bits [24, *mod/flowgraphs/DelayMeasurement.py*].

After measuring the overall delay of the modulation block, the E_b is calculated by using the Flowgraph depicted in Figure 4.12b. This measurement involves passing a known number of bits into the modulation and measuring the power of the transmitted signal which is then integrated to get total energy and divided by the number of bits transmitted for E_b [24, *mod/flowgraphs/EbMeasurement.py*].

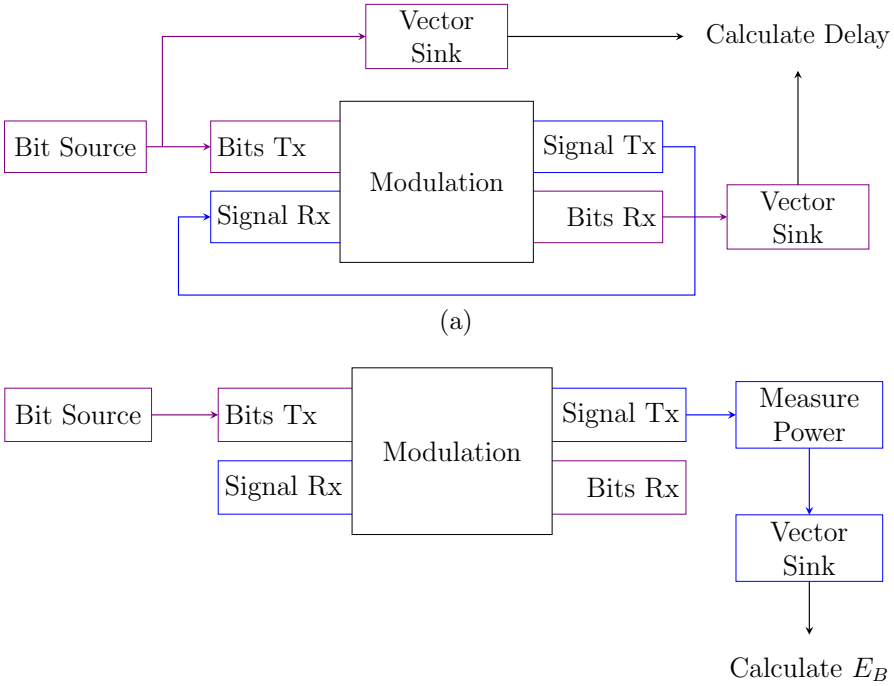


Figure 4.12: Generalized Flowgraphs for measuring (a) delay and (b) energy per bit (E_b)

This simulation infrastructure was used to evaluate the performance impact of using TC-OLA to transparently spread several common digital modulations [24, *mod/flowgraphs/TCOLAModWrapper.py*].

4.5.2.2 Simulations

Several modulation transceiver blocks were constructed to determine the impact TC-OLA has on digital modulation performance through an AWGN channel.

BER curves were generated for Binary Phase Shift Keying (BPSK) and 4 Level Pulse Amplitude Modulation (4PAM) using both rectangular and RRC pulses with 8 samples per symbol. Each of these modulations were simulated both with and without TC-OLA using $\frac{M}{R}$ parameters shown in Table 4.5.

M	R
16	1
32	1
64	1

Table 4.5: M and R Parameters for BER simulations

Chapter 5

Results and Discussion

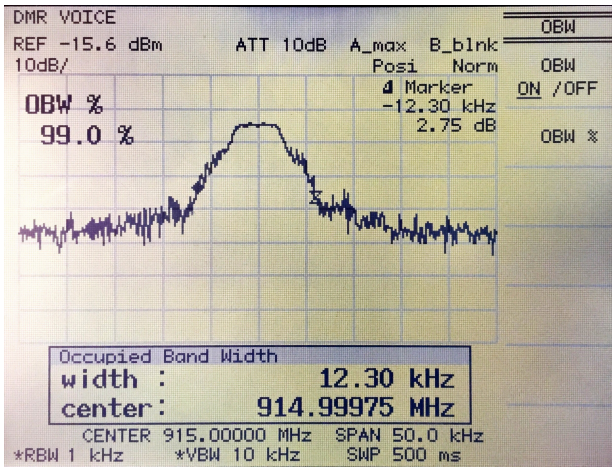
In Chapter 4 several experiments were performed to test the feasibility of using TC-OLA to spread DMR signals and make them compatible with the regulations for the ISM band. In this chapter, results from those experiments are introduced and discussed.

5.1 Spectral Analysis

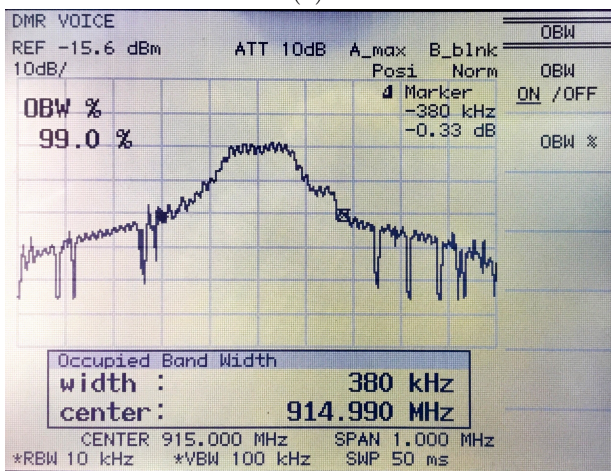
Using the method described in Section 4.4, OBW and 6 dB measurements were taken with the Avantest digital spectrum analyzer. Example output of the OBW and 6 dB measurements for the Voice test signal can be seen in Figure 5.1 and Figure 5.2 respectively.

As stated in Section 4.4, these measurements were performed for the Voice, Data Message 1, and Data Message 2 signals using the parameters in Table 4.2 and also for the unprocessed DMR signals. The results from the measurements of each of the test signals are given in Table 5.1.

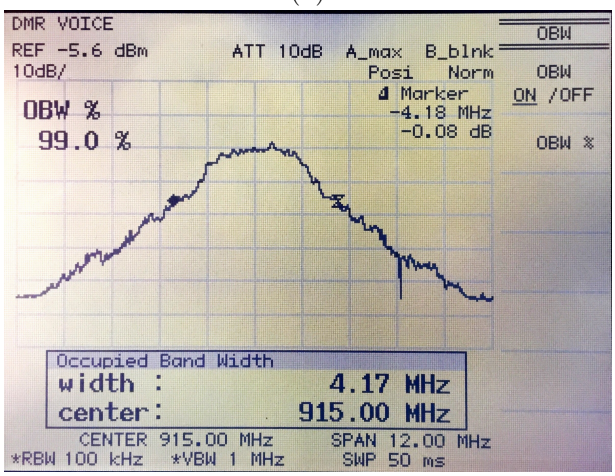
OBW and 6 dB bandwidth measurements of the original signal were used for calculating the *Measured* $\frac{M}{R}$ to compare against the theoretical spreading factor which can be seen in Table 5.1. The error figure is quite large but consistent among the various $\frac{M}{R}$ values. Some possible reasons for this might have to do with the limitations of the spectrum analyzer while performing measurement of the OBW and 6 dB bandwidth of the original signal. The RBW and VBW options on the spectrum



(a)

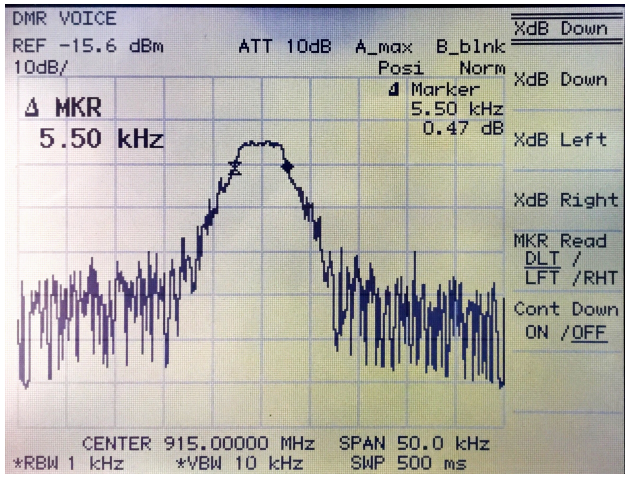


(b)

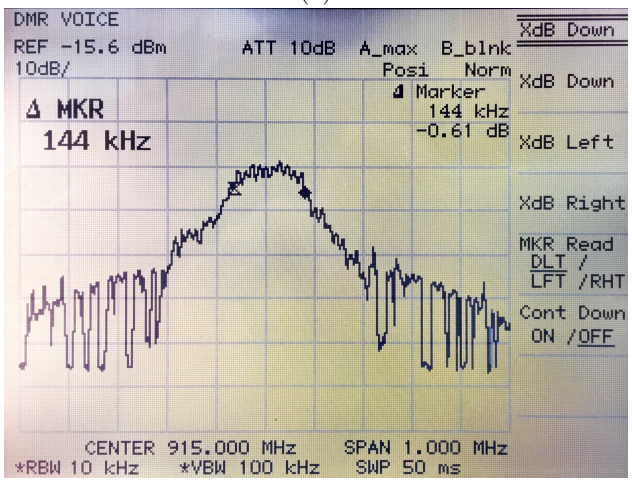


(c)

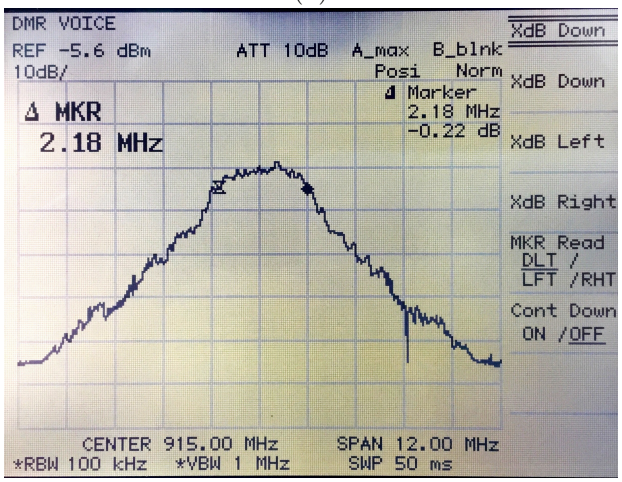
Figure 5.1: Spectrum analyzer OBW measurements for (a) DMR voice signal, (b) TC-OLA DMR voice ($M = 32$, $R = 1$) and (c) TC-OLA DMR voice ($M = 512$, $R = 1$)



(a)



(b)



(c)

Figure 5.2: Spectrum analyzer 6 dB bandwidth measurements for (a) DMR voice signal, (b) TC-OLA DMR voice ($M = 32$, $R = 1$) and (c) TC-OLA DMR voice ($M = 512$, $R = 1$)

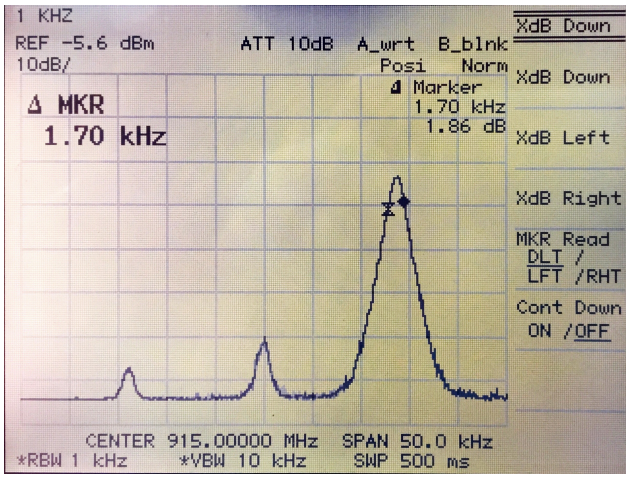
Signal	$\frac{M}{R}$	**Measured OBW (kHz)**	**Measured 6 dB BW (kHz)**	**Measured $\frac{M}{R}$**	**% Error**
Voice	1	12.3	5.5		
Data 1	1	11.6	5.5		
Data 2	1	11.6	5.6		
Voice	32	384	150	27.2	14.92
Data 1	32	338	125	22.7	28.98
Data 2	32	370	113	20.1	37.13
Voice	128	1,507	563	102.3	20.08
Data 1	128	1,392	459	83.4	34.85
Data 2	128	1,299	443	79	38.24
Voice	512	4,253	2,167	393.9	23.06
Data 1	512	4,227	1,440	261.8	48.86
Data 2	512	3,910	1,733	309.5	39.55

Table 5.1: Theoretical $\frac{M}{R}$ versus Measured for test signals

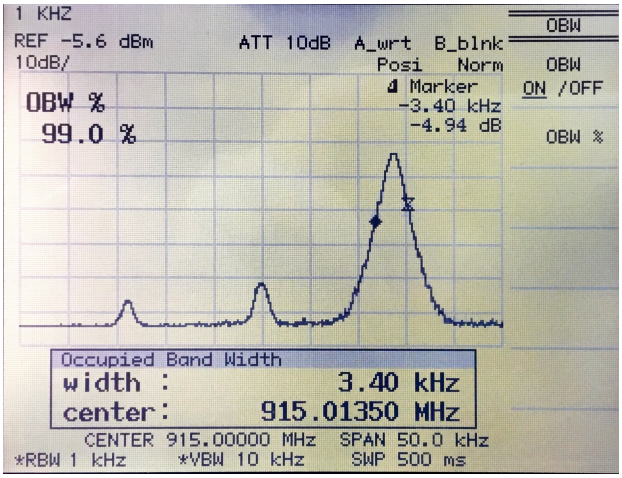
analyzer represent filters that are applied to the signals being fed into the analyzer and output onto the screen. A sanity check was performed to confirm this suspicion, by feeding the spectrum analyzer a single cosine which should theoretically have a single infinitely thin spike on the spectrum analyzer. The results of the OBW and 6 dB measurements of the cosine wave can be seen in Figure 5.3b and Figure 5.3a respectively. These figures show that processing chain shown in Figure 4.5 has a minimum bandwidth capability which leads to less precise measurements of narrowband signals.

Instead of relying on the measurements of the original signal, it was decided to use the measurements of the spread spectrum signals to calculate the average ratio between each step (*i.e.* the bandwidth at $M = 512$ divided by the bandwidth at $M = 128$). To do this, the 6 dB bandwidths of the three test signals (Voice, Data Message 1, and Data Message 2) were averaged for each value of M. These average 6 dB bandwidths were then divided to produce the calculated ratio which was expected to 4 (*i.e* $\frac{128}{32}, \frac{512}{128}$) in all cases. Table 5.2 shows these calculations and the difference between the expected ratio.

This error can be attributed to some of the aforementioned hardware limitations



(a)



(b)

Figure 5.3: Spectral measurements (a) 6 dB down and (b) OBW for 1 kHz cosine

$\frac{M}{R}$	**Expected Ratio**	**Average** 6 dB **BW**	**Calculated Ratio**	**% Error**
32		129.13		
128	4	488	3.78	5.52
512	4	1,780	3.65	8.81

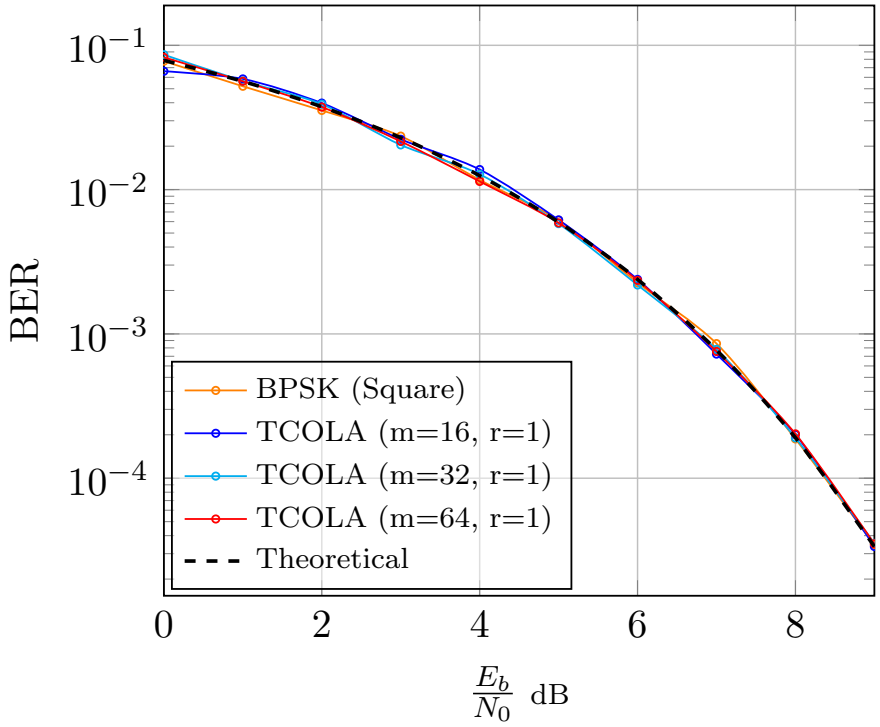
Table 5.2: Calculated ratio between average 6 dB Bandwidth measurements for $\frac{M}{R}$ values

of the spectrum analyzer itself, but also with the test signals being used. DMR signals are TDMA signals and have a time slot size of 30 ms which at some RBW/VBW settings can be short relative to the sweep time. This required that the max hold feature be used on the spectrum analyzer. This feature of the spectrum analyzer persists the maximum amplitude value of the spectrum after every sweep, allowing for an overall picture to be made over the course of many sweeps. In order for a reasonably full picture of the spectrum, the max hold was left on until the OBW and 6 dB bandwidth settled at a value. The test signals in this case were all relatively short in time compared to the amount of time that the spectrum analyzer took to settle; therefore, it was necessary to repeat the test signals for the duration of the spectrum measurement. This repetition accentuates the specific patterns within the test signals hence why the results of multiple test signals were gathered and analyzed for a more general result. These decisions were made based on the capabilities of the CML-DE9945E and could be improved by using a DMR test set to generate more general and representative signals. It is shown from the results in Table 5.1 that even though there are discrepancies between the measured and the theoretical $\frac{M}{R}$, the TC-OLA DMR signals were spread far beyond the 6 dB bandwidth required by the regulations stated by the FCC [11]. In a fully implemented system, $\frac{M}{R}$ would be reduced to the lowest possible value that would meet regulations in order to keep the sampling rate of the overall system to a minimum.

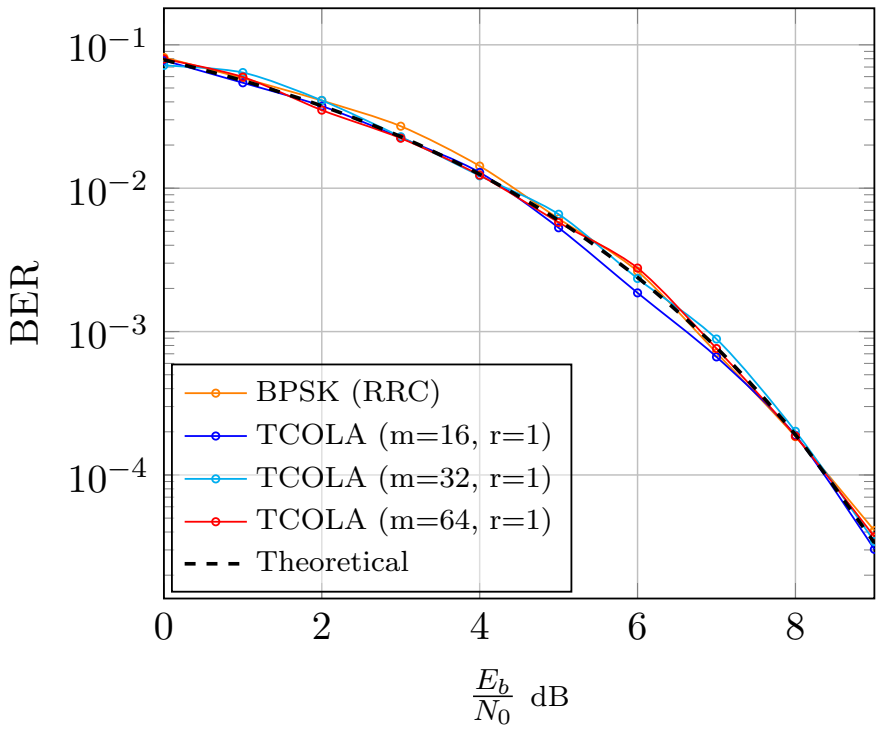
5.2 Bit Error Rate Performance

Using the BER simulation framework discussed in Section 4.5.2.1, BER curves were generated for BPSK (see Figure 5.4) and 4PAM (see Figure 5.5) with and without TC-OLA. Along with the simulated BER curves, the theoretical curve was also produced for reference. It can be seen from these curves that the performance of the digital modulations through TC-OLA is equivilant to that of the original modulation without TC-OLA. This result is an intuitive result considering TC-OLA is being used to provide transparent spectrum spreading around the modulation without actually changing the modulation itself. TC-OLA introduces signal redundancy which increases the E_b of the underlying modulation. When simulating each point of the BER curve, the noise power spectral density N_0 is calculated using the ratio $\frac{E_b}{N_0}$ divided by the E_b of the modulation. With TC-OLA, the additional E_b leads to

higher N_0 which is completely counter acted by the processing gain of the TC-OLA algorithm, resulting in the same performance as the original digital modulation.

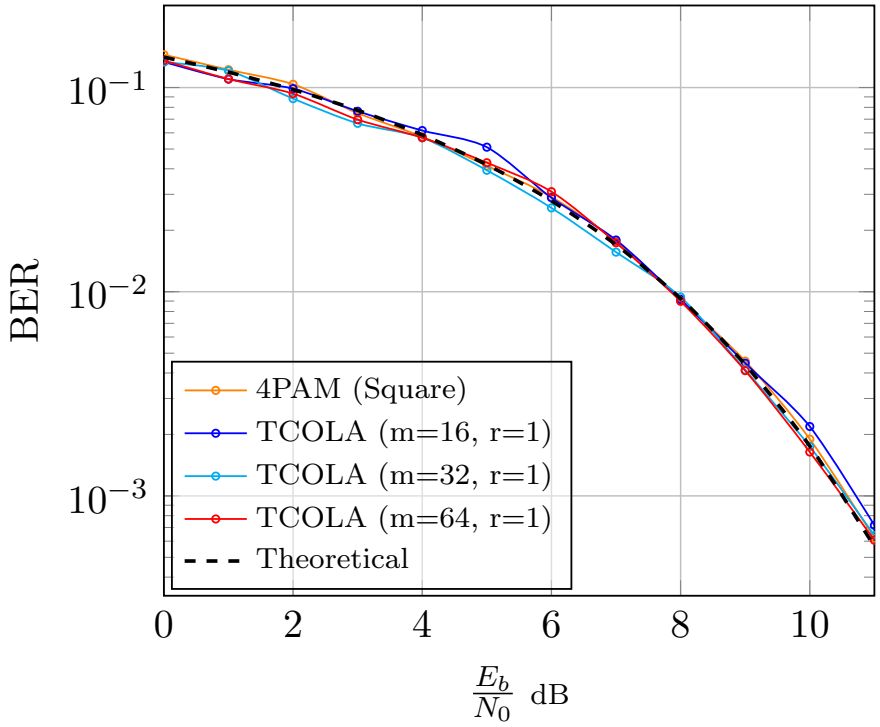


(a)

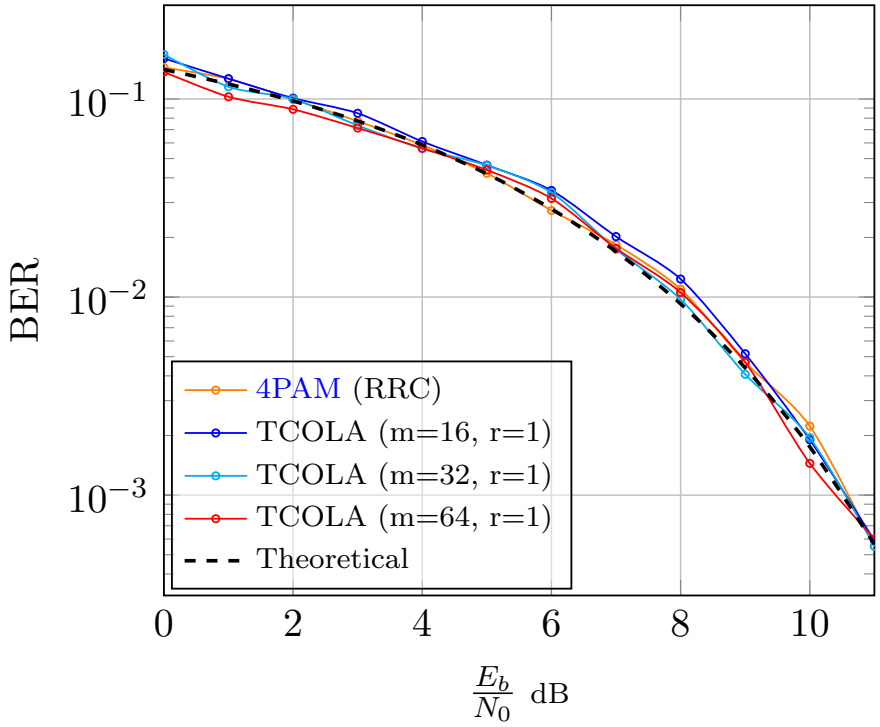


(b)

Figure 5.4: BER curves for BPSK using 8 samples per symbol (a) square pulses (b) RRC pulses



(a)



(b)

Figure 5.5: BER curves for 4PAM using 8 samples per symbol (a) square pulses (b) RRC pulses

Chapter 6

Future Work

This body of research began with the idea of trying to improve communication and overall safety among cyclists (see Section 2.1). After researching existing solutions to this problem, a preliminary high level design was created (see Figure 2.2) from which research began into the feasibility of using TC-OLA to spread an existing LMR protocol (DMR) to regulation compliant bandwidths. In order to fulfill the original goal of creating a piece of communication hardware, there are still many things that would need to be done, including the implementation and testing of Downsample Upsample Shift Add (DUSA), DMR performance tests, multiple access, hardware implementation of TC-OLA or DUSA, and the final design of the communication hardware.

6.1 Implementation of DUSA

DUSA [33] is a more general implementation of the TC-OLA algorithm that does not require the constraint that M be divisible by R. The DUSA algorithm makes use of a multidimensional filter bank with each path including a down-sampling followed by multiplication (when used with slow rate code) followed by an up-sampling operation and finally a shift and add [33, Chapter 6]. The DUSA algorithm has potential to reduce overall processing load with the introduction of the initial down-sampling operation, reducing the overall system clock rate from the beginning of the processing chain. DUSA can be used to to implement TC-OLA, however, it can also be used to produce fractional $\frac{M}{R}$ ratios which cannot be done with TC-OLA. The DUSA

algorithm has many claimed benefits and should be implemented within GR blocks to allow for an easier comparison with TC-OLA from the perspective of computational resource requirements. The implementation of DUSA should be started by forking the *gr-tcola* repository and using test driven development with the existing *gr-tcola* unit tests, since DUSA should be expected to reproduce the exact same output as TC-OLA.

6.2 Further TC-OLA Performance Evaluations

The experiments explored in this research provide a solid foundation for further research into impacts of more adverse channel effects discussed in Section 2.3.2 on signals processed and transmitted with TC-OLA. In addition to evaluating more advanced channel effects, an investigation on the impact of latency, jitter, and carrier drift on the overall performance of digital radio signals processed using TC-OLA would valuable. Finally, a comparison of these performance metrics on a standard set of test signals with the same signals processed using DSSS would provide further insight into the strengths and weaknesses of TC-OLA versus commonly used techniques.

6.3 Mobile Radio Performance Tests

As stated in Chapter 5, the CML-DE9945E purchased for this project provided a simple platform for a feasibility assessment of spread spectrum digital mobile radio using TC-OLA. After completing the experiments described in Chapter 4, it became apparent that a digital radio test set would provide the most ideal environment for running these types of tests. Digital radio test sets provide a myriad of measurements, metrics, and test signals specific to the protocol under test. These test sets are built for validating the implementation of many different digital radio protocols in numerous types of channel environments, and as such, it would be recommended to find performance measurements that are common to each protocol. Each of these measurements should be performed on each protocol both with and without TC-OLA through various channel conditions (*i.e.* AWGN, multipath fading, *etc.*) to allow for the most effective comparison of each protocol. After completing the tests,

the most robust protocol could be determined and selected for use in a hardware implementation. In addition to using a digital radio test set, it would be valuable to perform an analysis by shifting the TC-OLA signals into the RF passband and running them through passband channel models and real world transmissions over the air.

6.4 Multiple Access

Multiple access, or channelization, refers to the mechanism by which multiple clients can use an RF system at the same time without interfering with one another. For NB signals, this is done by simply partitioning up the available spectrum and placing channels far enough away from one another so that no interference is experienced. For WB signals, channelization techniques usually include using orthogonal keys, shared by the receiver and transmitter, which are specific to the type of modulation. For DSSS, this is in the form of an orthogonal chip code, where FHSS uses orthogonal frequency hopping patterns. In the case of TC-OLA, many multiple access methods have been discussed [16, 6.3] and should be investigated within the context of spread spectrum mobile radio. The other interesting aspect to this future work is that many of the existing radio protocols already support multiple access via Frequency Division Multiple Access (FDMA) or TDMA which could optionally be used in conjunction with a TC-OLA-based multiple access scheme. It would be valuable to experiment with multiple access schemes and investigate the number of separate channels that could be accommodated within the ISM band. Implementing multiple access from the TC-OLA level of the transmission chain would also require developing a mechanism by which to estimate and detect carrier offsets and drifts in order to compensate for these effects within the receiver. This would likely be accomplished by introducing a synchronization frame into the TC-OLA processing chain outside of the underlying digital radio protocol being used.

6.5 Hardware Implementation

The largest obstacle preventing the adoption of TC-OLA in hardware applications is the lack of an efficient hardware implementation. It can be seen from the GR

implementation of the TC-OLA blocks that the Time Compression block could potentially be distilled into a floating point shift register: R new samples push out the R samples at the end of the register, then all of the samples are multiplied by the selected window function and sent to the output which in hardware would be have to be a parallel to serial converter running at a clock rate $\frac{M}{R}$ times faster than the clock rate of the original signal. It is likely that the Overlap and Add algorithm could be implemented with a similar approach. Each M new samples of input would be windowed and added to an existing buffer of length M samples. After the addition, R samples would need sent out via a parallel to serial converter running at a clock rate $\frac{M}{R}$ times slower than the input clock rate. It should also be noted that DUSA [33] (discussed in Section 6.1) may lend itself to a hardware implementation more easily than the TC-OLA algorithm used to develop the *gr-tcola* blocks. The first step in implementing TC-OLA or DUSA in hardware would be to use an Field Programmable Gate Array (FPGA), which has the flexibility of software with the speed of hardware. FPGAs allow for faster prototyping and may offer insights into the possible design of a dedicated piece of TC-OLA hardware. This is quite possibly the most important future work influencing the adoption of TC-OLA and/or DUSA by industry.

6.6 Communication Hardware Design

After a hardware implementation of TC-OLA or DUSA has been created, the requirements for these hardware blocks will likely provide many constraints to the design of the overall communications hardware. The multirate nature of TC-OLA will likely determine the minimum clock speed of the processing unit used in the piece of hardware. The increased sampling rate also increases the bandwidth of the signal that needs to be transmitted which will greatly influence the requirements for the RF frontend of the system, mainly the amplifiers which will need to be designed to accommodate the high bandwidth signals produced by the Time Compression operation.

Chapter 7

Conclusion

The basis of this research was a direct result of trying to provide a robust, license free communication solution for cyclists (see Chapter 2). Some of the challenges of license free mobile radio design were investigated and existing solutions to this problem were sought out in an effort to shape the direction of research. It was found that the most promising license free band available is the ISM band. The ISM band has become a very popular RF band for consumer devices, and as such, many different devices have been developed to solve the aforementioned communications problem. Further inquiry revealed, that these existing devices have been built on top of technologies that were designed to support very different uses, leading to a less robust RF channel. In some cases, devices were built using completely custom communication protocols operating over license free RF links using commodity hardware modules such as LoRa. In other devices, Bluetooth has been pushed to its operational limits providing long distance links using the Bluetooth Intercom Profile in conjunction with external amplifiers to increase the range. These devices, though functional, fall short of the initial ideal requirements set forth in Chapter 2.

In Chapter 3, a new way of approaching this problem was introduced. Instead of implementing proprietary communication protocols or trying to overcome limitations of existing ones, this research investigated the use of the TC-OLA algorithm to transparently spread the bandwidth of mobile radio protocols until they could be transmitted within license free radio bands. This approach offloads the complexity of the communications protocol onto existing optimized and power efficient implementations. TC-OLA provides a spread spectrum algorithm that can be used with much

less synchronization overhead and allows for true separation of concerns. An efficient software implementation of the TC-OLA was implemented in C++ for GR and was used for running many different experiments. The CML-DE9945E was purchased as a standalone digital mobile radio transceiver and was used in conjunction with USRPs which allowed for fast prototyping and experimentation in GR.

In Chapter 4, the CML-DE9945E was used to generate a small set of test signals which were then used to verify proper reception of signals after TC-OLA processing. The processed TC-OLA test signals were fed into a spectrum analyzer to confirm that TC-OLA was indeed capable of producing standards compliant signal bandwidths. It was confirmed, in Chapter 5, that TC-OLA could be used to effectively and transparently spread the bandwidth of existing radio protocols to bandwidths required by the ISM radio band. It was also confirmed through simulations that TC-OLA does not impact the BER performance of underlying digital modulations.

TC-OLA has proven to be an extremely useful and powerful spread spectrum technique. An efficient real time implementation was presented in this thesis and used for establishing a working implementation of TC-OLA DMR which was tested to be spectrally compliant with ISM band regulations.

Appendices

Appendix A

Radio emission types

Within the FCC rules and regulations[12], radio transmission types are defined by codes. These emission type codes are in the format:

BBBB MSI DX

where **BBBB** is optional and represents the bandwidth of the signal, **M** is a letter code from Table A.1 representing the type of modulation, **S** is a number from Table A.2 representing the type of modulating signal, **I** is a letter from Table A.3 representing the type of information transmitted, and **D** and **X** are optional and represent details of transmitted information and the method of multiplexing respectively. Table 2.2 provides an example of the permitted emission types for the FRS band along with their meanings derived from the aforementioned code tables.

Character	Description
N	Unmodulated carrier
A	Double-sideband amplitude modulation (e.g. AM broadcast radio)
H	Single-sideband with full carrier (e.g. as used by CHU)
R	Single-sideband with reduced or variable carrier
J	Single-sideband with suppressed carrier (e.g. Shortwave utility and amateur stations)
B	Independent sideband (two sidebands containing different signals)
C	Vestigial sideband (e.g. NTSC)
F	Frequency modulation (e.g. FM broadcast radio)
G	Phase modulation
D	Combination of AM and FM or PM
P	Sequence of pulses without modulation
K	Pulse amplitude modulation
L	Pulse width modulation (e.g. as used by WWVB)
M	Pulse position modulation
Q	Sequence of pulses, phase or frequency modulation within each pulse
V	Combination of pulse modulation methods
W	Combination of any of the above
X	None of the above

Table A.1: Type of modulation

Character	Description
0	No modulating signal
1	One channel containing digital information, no subcarrier
2	One channel containing digital information, using a subcarrier
3	One channel containing analog information
7	More than one channel containing digital information
8	More than one channel containing analog information
9	Combination of analog and digital channels
X	None of the above

Table A.2: Type of modulating signal

Character	Description
N	No transmitted information
A	Aural telegraphy, intended to be decoded by ear, such as Morse code
B	Electronic telegraphy, intended to be decoded by machine (radioteletype and digital modes)
C	Facsimile (still images)
D	Data transmission, telemetry or telecommand (remote control)
E	Telephony (voice or music intended to be listened to by a human)
F	Video (television signals)
W	Combination of any of the above
X	None of the above

Table A.3: Type of transmitted information

Appendix B

Frequency Bands

Table B.1 shows the frequency bands capable of supporting LMR communications.

Band	**Start** (MHz)	**End** (MHz)
Multi User Radio Services (MURS)	151.820	154.600
Family Radio Services (FRS)	4625.5625	462.7250
General Mobile Radio Services (GMRS) (Lower)	462.5500	462.7250
General Mobile Radio Services (GMRS) (Upper)	467.5500	467.7250
Industrial Scientific and Medical (ISM)	902	928
Industrial Scientific and Medical (ISM)	2400	2500
Industrial Scientific and Medical (ISM)	5725	5850
Private Mobile Radio (PMR) (Europe)	446.0	446.2

Table B.1: Frequency bands capable of PMR/LMR communications

Appendix C

PMR446 Channels

The following tables show the frequencies associated with each channel of PMR446 for Analog, DMR and dPMR protocols.

C.1 Analog

Channel	**Frequency** (MHz)
1	446.00625
2	446.01875
3	446.03125
4	446.04375
5	446.05625
6	446.06875
7	446.08125
8	446.09375

Table C.1: Analog PMR446 Channel Frequencies

C.2 DMR446

Channel	**Frequency** (MHz)
1	446.10625
2	446.11875
3	446.13125
4	446.14375
5	446.15625
6	446.16875
7	446.18125
8	446.19375

Table C.2: DMR446 Channel Frequencies

C.3 dPMR446

Channel	**Frequency** (MHz)
1	446.103125
2	446.109375
3	446.115625
4	446.121875
5	446.128125
6	446.134375
7	446.140625
8	446.146875
9	446.153125
10	446.159375
11	446.165625
12	446.171875
13	446.178125
14	446.184375
15	446.190625
16	446.196875

Table C.3: dPMR446 Channel Frequencies

Appendix D

GR Flex - GNU Radio interface to FlexRadio

D.1 GNU Radio implementation of Flex Radio

Mistic lab, run by Dr. Peter Driessen, is a research space focused on communications with special attention given to SDR technology. The lab is always making use of new software radios and trying to implement them in novel ways. This appendix outlines efforts to interface a powerful SDR (FlexRadio-6700) with an open-sourced software package (GNU Radio) over a communications network. All of the work in this appendix was done collaboratively by the members mentioned within D.1.8.

D.1.1 Motivation

There is a FlexRadio-6700 in the Mistic Laboratory but it was only accessible via FlexRadio's proprietary software (only available on Windows). It was found that at the time of development, there was an open source C# Application Programming Interface (API) compatible with the Mono framework, which is a cross platform implementation of .Net. Developing a GNU Radio block built on this API would allow the FlexRadio to be used by any computer running the free GNU Radio software, independent of the computers OS.

D.1.2 Components

D.1.2.1 GNU Radio

GNU Radio is an open-sourced software that supports a wide variety of software radios. It provides a framework for running these radios and doing signal processing on their inputs and outputs. It is primarily used as a user-friendly drag-and-drop Flowgraph built from 'blocks'. These blocks range from signal generators, to data-type converters, to file-system integrations, to SDR interfaces.

D.1.2.2 FlexRadio 6700

The FlexRadio-6700 is an SDR built for amateur radio users. It is able to both receive and transmit on up to two antennas. On the receiver side, it is is able to take each antenna input and split it into eight panadapters (visual displays of both spectrum and waterfall). It makes use of a powerful Analog to Digital Converter (ADC) (250 MSPS) and Digital to Analog Converter (DAC) (500 MSPS) which allow it to generate so many panadapters.

D.1.3 Implementation

The GNU Radio block was developed as an Out-of-Tree module (which means it must be independently maintained) and is build using Python2 (as GNU Radio blocks must be built in either C++ or Python). The FlexRadio API is accessed via a user-developed library called FlexLibMono. In order to use this Mono package in Python, a package called pythonnet was used to dynamically marshal calls between python and .Net libraries at run-time. An overview of this architecture and how the various components work together is shown in Figure D.1.

The majority of work done for this block was in the **gr-flex** layer of Figure D.1, however, some minor fixes were done in the **FlexLibMono** library.

D.1.4 Results

The block functions over either an Ethernet connection and has been tested on both Mac OS and Linux systems. A screenshot of the block being used in a Flowgraph is

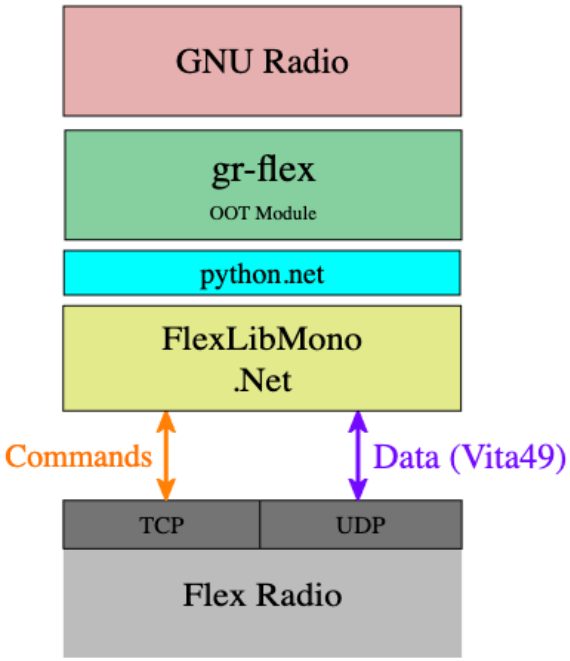


Figure D.1: The architecture of GNU Radio's interface with the Flex Radio.

shown in Figure D.2. The block has parameters which can be changed by the user real-time for center frequency, bandwidth, RX antenna, DAX IQ channel (there are four IQ channels available).

If multiple of the gr-flex blocks are added to a Flowgraph, each will open a panadapter (up to a maximum of eight) on the FlexRadio.

D.1.5 Existing Issues

On the proprietary FlexRadio software, there is a very smooth bandwidth adjustment available for each panadapter. So far the bandwidth adjustment in the block does not work. No serious effort has gone in to understanding why.

D.1.6 Release

The source code and instructions on how to install the block are available on Github via the Mistic Lab Github account github.com/misticlab. Before release, there are some issues that have been fixed within the FlexLibMono library that should be pulled

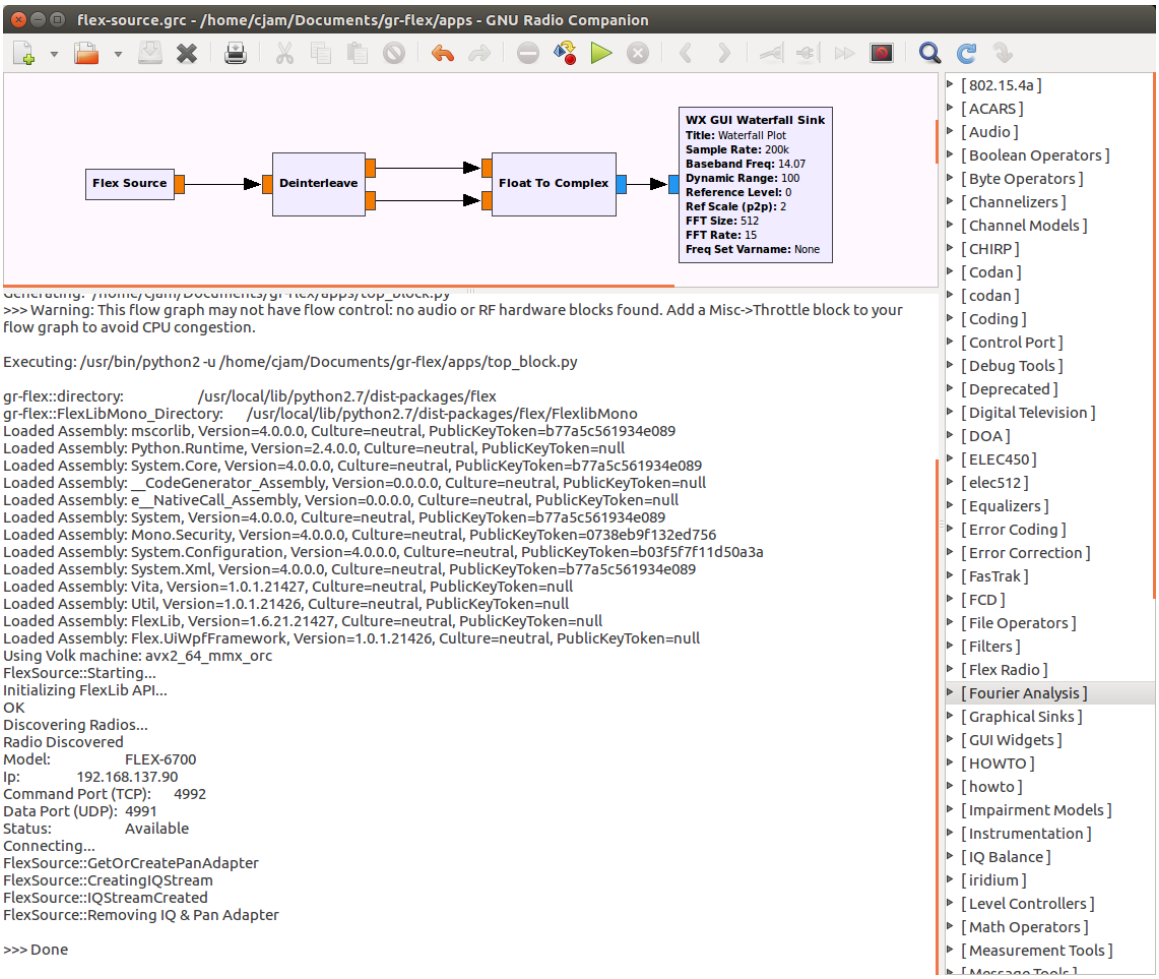


Figure D.2: The gr-flex used to visualize a waterfall and FFT in GNU Radio Companion.

into the original repository. Once those issues have been integrated into the original library, the block will also be announced to the FlexRadio community at large. It is an active community of amateur radio enthusiasts who would no doubt find value in this project.

D.1.7 Future work

Beyond exposing more panadapter controls through the receiving block, a transmission block would be a useful addition to the project. It is hoped that a future student in Mistic Lab will take on this task.

D.1.8 Collaborators

This project was worked on by Colter McQuay and Nicholas Bruce under the suggestion of Dr. Peter Driessen.

Bibliography

[1] American National Standards Institute. American National Standard of Procedures for Compliance Testing of Unlicensed Wireless Devices Accredited Standards Committee C63 ®-Electromagnetic Compatibility C63 ®. Technical report, The Institute of Electrical and Electronics Engineers, Inc., 2013.

[2] Tzi-Dar Chiueh, Pei-Yun Tsai, I-Wei Lai, Tzi-Dar Chiueh, Pei-Yun Tsai, and I-Wei Lai. *Baseband Receiver Design for Wireless MIMO-OFDM Communications*. John Wiley & Sons, Incorporated, New York, SINGAPORE, 2012.

[3] CML Microcircuits. CMX7241/CMX7341 - PMR/LMR Common Platform Processors - CML Microcircuits.

[4] CML Microcircuits. CML Microcircuits DE9945/DE9945E SDR Demonstrator: PMR Common Platform USER MANUAL Advance Information Features Direct Conversion Digital Radio Demonstrator Provides a demonstration platform for: CMX994A-Direct Conversion Receiver. Technical report, CML Microcircuits, 2017.

[5] Klaus Doppler, Mika P. Rinne, P. Janis, Cassio Ribeiro, and Klaus Hugl. Device-to-Device Communications; Functional Prospects for LTE-Advanced Networks. In *2009 IEEE International Conference on Communications Workshops*, pages 1–6. IEEE, 6 2009.

[6] European Telecommunications Standards Institute. TETRA.

[7] European Telecommunications Standards Institute. Electromagnetic compatibility and Radio spectrum Matters (ERM); Digital Mobile Radio (DMR) General System Design. Technical report, European Telecommunications Standards Institute, 2013.

[8] European Telecommunications Standards Institute. Digital Private Mobile Radio (dPMR) using FDMA with a channel spacing of 6,25 kHz. Technical report, European Telecommunications Standards Institute, 2015.

[9] European Telecommunications Standards Institute. Electromagnetic compatibility and Radio spectrum Matters (ERM); Digital Mobile Radio (DMR) Systems; Part 1: DMR Air Interface (AI) protocol. Technical report, European Telecommunications Standards Institute, 2017.

[10] European Telecommunications Standards Institute. Electromagnetic compatibility and Radio spectrum Matters (ERM); Digital Mobile Radio (DMR) Systems; Part 4: DMR trunking protocol. Technical report, European Telecommunications Standards Institute, 2017.

[11] Federal Communication Commission. FCC Title 47 Part 15, 2008.

[12] Federal Communication Commission. FCC FACT SHEET Part 95 Personal Radio Service Reform Report and Order -WT, 2017.

[13] Frank H P Fitzek, Marcos Katz, and Qi Zhang. Cellular Controlled Short-Range Communication for Cooperative P2P Networking. *Wireless Personal Communications*, 48(1):141–155, 1 2009.

[14] H.T. Friis. A Note on a Simple Transmission Formula. *Proceedings of the IRE*, 34:254–256, 5 1946.

[15] Daniel W Griffin A N D Jae S Lim and Senior Member. Signal Estimation from Modified Short-Time Fourier Transform. Technical Report 2, Massachusetts Institute of Technology, 1984.

[16] Stephen Harrison. *Time-Compression Overlap-Add (TC-OLA) for Wireless Communications by*. PhD thesis, University of Victoria, 2009.

[17] Stephen Harrison and Peter F. Driessen. Novel UWB and Spread Spectrum System Using Time Compression and Overlap-Add Techniques. *IEEE Access*, 2:1092–1105, 2014.

[18] Stephen T. Harrison and Peter F. Driessen. Time-compression overlap add: Description and implementation. In *2015 IEEE Pacific Rim Conference on Com-*

munications, Computers and Signal Processing (PACRIM), pages 64–69. IEEE, 8 2015.

[19] Icom. D STAR Info.

[20] GoTenna Inc. goTenna — Distributed Communications Networks.

[21] Thomas Keller and Lajos Hanzo. Adaptive multicarrier modulation: a convenient framework for time-frequency processing in wireless communications. *Proceedings of the IEEE*, 88(5):611–640, 5 2000.

[22] Mohamed Adnan Landolsi. Novel signals for optimized timing synchronization in direct-sequence spread-spectrum communication systems. *International Journal of Communication Systems*, 31(14):e3735, 9 2018.

[23] Colter McQuay. GitHub - mistic-lab/gr-tcola: GNU Radio Module that implements Time Compression Overlap Add (TCOLA) Spread Spectrum.

[24] Colter McQuay. GitHub - mistic-lab/pybergr: A python based framework for running GNU Radio based ber simulations on general modulations.

[25] Ville Niemela, Jussi Haapola, Matti Hamalainen, and Jari Iinatti. An Ultra Wideband Survey: Global Regulations and Impulse Radio Research Based on Standards. *IEEE Communications Surveys & Tutorials*, 19(2):874–890, 2017.

[26] NXDN^{TM} Forum. NXDN Forum Website.

[27] Project 25 Technology Interest Group. Project 25 Technology Interest Group.

[28] T.K. Sarkar, Zhong Ji, Kyungjung Kim, A. Medouri, and M. Salazar-Palma. A survey of various propagation models for mobile communication. *IEEE Antennas and Propagation Magazine*, 45(3):51–82, 6 2003.

[29] P Mohana Shankar. Modeling of Fading and Shadowing. In *Fading and Shadowing in Wireless Systems*, pages 299–520. Springer International Publishing, Cham, 2017.

[30] C.E. Shannon. Communication In The Presence Of Noise. *Proceedings of the IEEE*, 86(2):447–457, 2 1998.

[31] Don Torrieri. Direct-Sequence Systems. In *Principles of Spread-Spectrum Communication Systems*, pages 87–154. Springer International Publishing, Cham, 2018.

[32] Werner Verhelst. Overlap-add methods for time-scaling of speech. *Speech Communication*, 30(4):207–221, 4 2000.

[33] Ahmed Youssef. *Enhancement of Target Detection Using Software Defined Radar (SDR)*. PhD thesis, University of Victoria, 2018.

[34] Ahmed Youssef, Peter F Driessen, Fayez Gebali, and Belaid Moa. A Novel Smeared Synthesized LFM TC-OLA Radar System: Design and Performance Evaluation. *IEEE Access*, 7:18574–18589, 2019.

[35] Udo. Zolzer and Xavier. Amatriain. *DAFX : digital audio effects*. Wiley, 2002.