UTILIZING SOFTWARE DEFINED RADIOS FOR INCREASED BANDWIDTH IN SATELLITE COMMUNICATIONS

THESIS

Brian M. Wright, Captain, USAF

AFIT-ENG-MS-21-M-096

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT-ENG-MS-21-M-096

UTILIZING SOFTWARE DEFINED RADIOS FOR INCREASED BANDWIDTH IN SATELLITE COMMUNICATIONS

THESIS

Presented to the Faculty

Department of Electrical and Computer Engineering

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Master of Science in Electrical Engineering

Brian M. Wright, B.S.E.E

Captain, USAF

March 2021

AFIT-ENG-MS-21-M-096

UTILIZING SOFTWARE DEFINED RADIOS FOR INCREASED BANDWIDTH IN SATELLITE COMMUNICATIONS

Brian M. Wright, B.S.E.E

Captain, USAF

Committee Membership:

Dr. Kenneth M. Hopkinson
Chair

Dr. Robert F. Mills
Member

Maj J. Addison Betances, PhD
Member

AFIT-ENG-MS-21-M-096

Abstract

This research explores the background and future possibilities of using cooperative satellites equipped with Software Defined Radios (SDRs) to combine their bandwidth and increase their signal reliability. Software Defined Radios are a potential solution to realize various software applications that support a reconfigurable and adaptive communication system without altering any hardware devices or features. This benefit, along with others that are offered by SDRs and the ongoing improvements in commercial digital electronics have sparked an interest in developing small satellites for advanced communications. This research effort sets out to prove if a high bandwidth signal can be generated by using low-cost SDRs. The SDR receivers will each receive different sub-bands of the transmitted signal with the goal of the meshing the received signals to form a high bandwidth signal.

Acknowledgments

I would like to express my sincere appreciation to my academic advisor, Dr. Hopkinson, for his guidance, support, and patience throughout my time at AFIT. Your support was a major contributor to me finishing this thesis. I would, also, like to thank Major Betances and Dr. Mills. Major Betances, you were with me through every step in completing this thesis and I appreciate your efforts in ensuring I graduated. For Dr. Mills, thank you for always keeping your door open and pointing me to a solution. To Destinee, Darnell, Natalie, Suresh, Hannah, Austin, Colton, and Jake, I would not have made it through this program without your help. You all are the brightest and most talented people I have worked with in my career. Lastly, but most importantly, I want to thank my wife for the patience and sacrifice she displayed during my time in school. I know this was a challenging time for our family, but your willingness to stand by me drove me to complete this journey.

Brian M. Wright

Table of Contents

Page

Page

Page

List of Figures

Figure Page

List of Tables

List of Acronyms

ADC	Analog to Digital Converter
AFIT	Air Force Institute of Technology
AM	Amplitude Modulation
AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
BFN	Beam Forming Network
BGAN	Broadband Global Area Network
BPSK	Binary Phase Shift Keying
CFO	Carrier Frequency Offset
COTS	Commercial Off the Shelf
CPU	Central Processing Unit
CTTC	Centre Tecnologic de Telecommunications Catalunya
CW	Continuous Wave
DA	Data-Aided
DAC	Digital to Analog Converter
DC	Direct Current
DSP	Digital Signal Processing
DoD	Department of Defense
DUT	Device Under Test
EGNOS	European Global Navigation Overlay Service
EMI	Electromagnetic Interference

ESA	European Space Agency
FCC	Federal Communications Commission
FFT	Fast Fourier Transform
FPGA	Field Programmable Gate Array
GEO	Geostationary Earth Orbit
GLONASS	Global Navigation Satellite System
GMSK	Gaussian Minimum Shift Keying
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
HDL	Hardware Description Language
IARU	International Amateur Radio Union
ICI	Inter-Carrier Interference
ISI	Inter-Symbol Interference
ITU	International Technology Union
LAN	Local Area Network
LEO	Lower Earth Orbit
LO	Local Oscillator
MATLAB	Matrix Laboratory
MEMS	Micro Electro-Mechanical System
MIMO	Multiple-Input Multiple-Output
MUD	Multi-User Domain
NDA	Non-Data Aided
NRZ	Non-Return to Zero

OBD On-Board Demodulator

OBP On-Board Processor

OFDM Orthogonal Frequency Division Multiplexing

OOA/OOD Object Oriented Analysis/Object Oriented Design

OQPSK Offset Quadrature Phase Shift Keying

PC Personal Computer

PDN Power Distribution Network

PM Phase Modulation

PPM Parts Per Million

PRN Pseudo Random Noise

PSD Power Spectral Density

PSK Phase Shift Keying

QPSK Quadrature Phase Shift Keying

RAM Random-Access Memory

RF Radio Frequency

RF-DNA Radio Frequency Distinct Native Attribute

RFID Radio Frequency Identification

RMS Root Mean Square

SATCOM Satellite Communications

SBC Single Board Computer

SCA Software Communications Architecture

SDR Software Defined Radio

SDRRCS Software Defined Radio for Regenerative Communications Satellites

SER	Symbol Error Rate
SMA	SubMiniature
SNR	Signal-to-Noise Ratio
SRRC	Square Root Raised Cosine
SSTL	Surrey Satellite Technologies Ltd
SWaP	Size Weight and Power
TCM8PSK	Trellis Coded Multidimensional 8-PSK
TT&C	Telemetry Tracking and Command
UAV	Unmanned Aerial Vehicle
UHF	Ultra High Frequency
USB	Universal Serial Bus
USRP	Universal Software Radio Peripheral
VHF	Very High Frequency
WSS	Wide-Sense Stationary

UTILIZING SOFTWARE DEFINED RADIOS FOR INCREASED BANDWIDTH IN SATELLITE COMMUNICATIONS

I. Introduction

1.1 Background

For the last 30 years, the aerospace, commercial and defense air industries have depended upon satellite communications (SATCOM) to coordinate civilian passenger travel and global military operations. Due to the exponential growth in data flow and internet applications, the requirement for SATCOM increased ten-fold.

Currently, soldiers and forward operating bases are using more data than ever before. There is also an increased demand from unmanned aerial vehicle (UAV) technology to help carry out military operations. Likewise, the commercial aircraft industry is an ever-increasing need for access to high bandwidth data. The cost required to develop and launch new satellites to support higher frequencies and facilitate this growth in bandwidth is at an all-time high.

1.2 Operational Motivation

Smaller satellites are appealing for aerospace and defense applications due to their decreased development time, more frequent launch opportunities, larger array of mission types, more rapid growth of the technical/scientific knowledge base, and greater engagement with smaller industries and universities. The launch of Minotaur 1 and Dnepr-19 added 63 small satellites to Lower Earth Orbit (LEO). These additions have

caused a burden for licensing and coordinating organizations, such as the Federal Communications Commission (FCC) and the International Amateur Radio Union (IARU). The proliferation of small satellites has created an increasingly congested radio frequency (RF) environment, which has complicated frequency management. The FCC and International Telecommunication Union (ITU) have been restricting the licensing for small satellites, particularly in the very high frequency (VHF) band. Therefore, it is imperative to develop an effective means of using the limited available bandwidth. The existing smaller SATCOM systems are ill-equipped to support these challenges alone [1].

A possible solution for the above issues is equipping satellites with a Software Defined Radio (SDR). An SDR is a radio device that instead of being composed of traditional hardware (e.g., mixers, filters, amplifiers, modulators/demodulators, detectors, etc.), its components are implemented by means of software on a personal computer or embedded system [2]. A simple SDR system will usually comprise of a personal computer that has a sound card, or other analog-to-digital converter (ADC), preceded by some form of radio frequency (RF) front end. Considerable amounts of signal processing are passed through to the general-purpose processor, as opposed to being done by special-purpose hardware [2]. Ideal operation of a SDR would have an incoming signal immediately converted to a digital format where the signal is then processed completely digitally. Conversely for transmission, the signal is generated digitally, then converted to the final analog signal at the antenna. Due to this approach, the radio can be completely reconfigured for a new application by simply changing the software.

Software Defined Radios offer functionalities that can be difficult for air and spacecraft to achieve such as in-flight re-configurability, adaptability, and autonomy

which leads to limited subsystem re-design. These features can be implemented towards a generic SATCOM solution. The benefits offered by SDRs along with increasing advances in commercial digital electronics have bolstered the interest of small satellites in advanced communication systems [3]. The increase in subsystem performance could theoretically allow SATCOM regulators to loosen restrictions on modulation constraints on frequency bands, link conditions, Doppler uncertainties, and data rates at minimum cost, thus making dynamic multiband access and sharing more possible. However, this flexibility and adaptability comes at the expense of power consumption and complexity [1].

1.3 Problem Statement

The aim of this research effort is to transmit, receive, and then stitch together the received signal using software defined radios with the expectation of generating a high bandwidth signal. The total bandwidth of the received signal will be the combined bandwidth of the software defined radio receivers. The transmit signal is assumed to be time-variant. Because of this, multiple SDRs are required to effect simultaneous collection of the wide-band signal. Each SDR will collect a specific portion of the signal.

Software Defined Radios are a light weight, low complexity, and low-cost device. These features underscore their flexibility and portability, but also decrease the ADC bit-rate performance. Low ADC bit rates inhibit SDR instantaneous bandwidth [4]. Due to this issue multiple SDR receivers must collect and store consecutive segments of a wide-band signal until all sub-bands are collected for processing [5].

1.4 Research Objectives

This effort is a continuation of previous research which focused on the reconstruction of a known signal using SDR. That research examined the reconstruction of simultaneous SDR receiver instantaneous bandwidth collections using single, dual, and multiple SDR receivers. The adjacent sub-bands, collectively spanning a transmit signal bandwidth were then autocorrelated with a replica transmit signal to restore frequency and phase offsets. This research looks to expand upon this method, but with the goal of generating a signal bandwidth from a random, unknown received signal.

1.5 Summary

Chapter I described the usefulness of implementing SDR technology with small satellites, provided background on software defined radios, and detailed the research objectives. Chapter II provides the theoretical background needed to prepare the reader for comprehending the research that is conducted along with previous research that has been conducted on this topic. Chapter III describes the methodology of the research effort to include simulation and hardware testing setup. Chapter IV details simulation and test results, while also providing final analysis. Finally, Chapter V concludes with research findings and proposals for future work.

II. Literature Review

The following sections provide the necessary theoretical background and a summary of current research emphasizing the utility in equipping satellites with software defined radios (SDRs) to prepare the reader for this research effort; Section 2.1 details bandwidth demands in satellite communications (SATCOM); Section 2.2 provides an overview of SATCOM and solutions to the bandwidth issue; Section 2.3 details SDR technology, this section also includes summary tables, processing blocks, and their affected parameters; Section 2.4 describes phase-shift keying modulation schemes, bandwidth expansion techniques and signal collection; finally Section 2.5 details previous AFIT research in this area.

2.1 Bandwidth Demands

One of the driving forces for new developments in the SATCOM world is the demand for increased data rates. Military-based SATCOM links have improved from kbps to Mbps data rates, which calls for faster and more efficient data transfer. The increase of unmanned aerial vehicles (UAVs) in both the private and defense sectors have also produced a new platform for SATCOM links [6]. Also, the limitless demand for Internet access and data in the commercial aerospace industry is leading to newer advancements in the K_a-band K_u-band and to support data rates up to 1000 Mbps. Simultaneously, minimizing size, weight, and power while also supporting legacy data links in system

development is pushing the aerospace communications industry to develop flexible architectures and maximum system reuse [6].

2.2 Satellite Communications

Traditionally, SATCOM systems have employed geostationary Earth orbit (GEO) satellites. These are satellites that are relative to the Earth's surface and stay in a fixed position or location. For a satellite to achieve geostationary orbit it must be at an extremely high altitude, over 36,000 km from Earth's surface. The key benefit of achieving such a high orbit is that fewer satellites are required to provide coverage over a large area of ground, also transmitting to the satellite is less challenging because it has known, fixed coordinates. Because of their high launch costs, these systems are designated for long lifecycles, which results in a steady but oftentimes outdated system [6].

Since the satellite is such a great distance from Earth, it will experience significant loss with the user at the ground station, impacting component selection and signal design chain. Longer distance between the satellite and ground station also lead to high latency and longer propagation times between the user and the satellite, which has great effect on communication and data links [6]

In recent years, several replacements or complementing systems to GEO satellites have been proposed, with low Earth orbit (LEO) satellites and UAVs being considered. With lower orbits and operating altitudes, these systems improve a lot of the issues that

affect GEO-based systems, but these alternatives require significantly more satellites or UAVs for similar global coverage [6].

2.2.1 Proposed Solutions

Over the past few decades, telecommunications companies have provided users the opportunity to use their GEO satellites with a K_a-band data link to help with some of the challenges detailed above [6]. A K_a-band is a segment of the microwave portion of the electromagnetic spectrum defined as frequencies in the range of 26.5 to 40 gigahertz (GHz), i.e., wavelengths from slightly longer than one centimeter down to 7.5 millimeters [7]. Alternatively, the K_u-band is the segment of the electromagnetic spectrum of frequencies from 12 to 18 gigahertz (GHz). With a higher frequency, a user can extract more bandwidth from a K_a-band, which results in a higher data transfer rate [8].

From an architectural standpoint, giving users access to the K_a-band provides a solution to bandwidth deficiencies but brings further issues to a design engineer. Figure 1 displays a standard super-heterodyne transmit and receive signal chain for operation in the K_a-band and K_u-band. Normally these systems need two, or three, stages of analog down-conversion and up-conversion. Each stage requires an amplification, synthesizer, and filtering that increases system size, weight, and power (SWaP). However, to operate within the current airliner infrastructure and power distribution system, integrating these types of signal chain for all possible data links may be unsustainable [6].

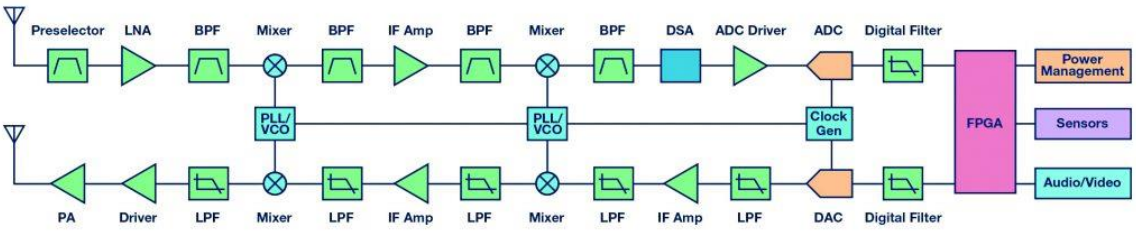

Figure 1. Ka-band/Ku-band super-heterodyne receive and transmit signal chain [10].

Lower Earth orbit satellites may offer some relief to the bandwidth demands of SATCOM. These satellites operate at a much lower altitude, approximately 2 km off earth's surface [9]. At that height, the satellites are not stationary, and they orbit around the Earth's surface approximately every two hours. Lower Earth orbit satellites offer two key benefits: the low altitude reduces launch cost and propagation delay. However, their lifespan is relatively short due to the harsher environment of lower Earth orbit [10]. Also, the main disadvantage with LEO satellites is that they have a small momentary field of view. They are only able to communicate with a fraction of the Earth at a time [10].

Unmanned aerial vehicles (UAVs) have also been considered to address the bandwidth issue with some platforms possibly being a way to increase or extend internet coverage. Unmanned aerial vehicles can offer links with high bandwidth and low latency, like LEOs, but with the added benefit of being comparatively stationary. Still, the coverage vs. cost of using UAVs for bandwidth expansion purposes may be difficult for worldwide operations [11].

2.3 Software Defined Radio

An SDR can be characterized as an open architecture that generates a communication platform by interlinking standardized and modularized flexible hardware building blocks. The radio's software assigns tasks and communicates between the blocks to provide an identity to the system [12]. Figure 2 displays a diagram of an SDR hardware and software. Software defined radios are often described as "Radio that provides software control of a variety of modulation techniques wide and narrow band operation, communication security functions and waveform requirement of current and evolving standards over a broad frequency range [12]."

An SDR systems can be broken into two major parts:

1. Hardware Functional Block
2. Software Functional Block

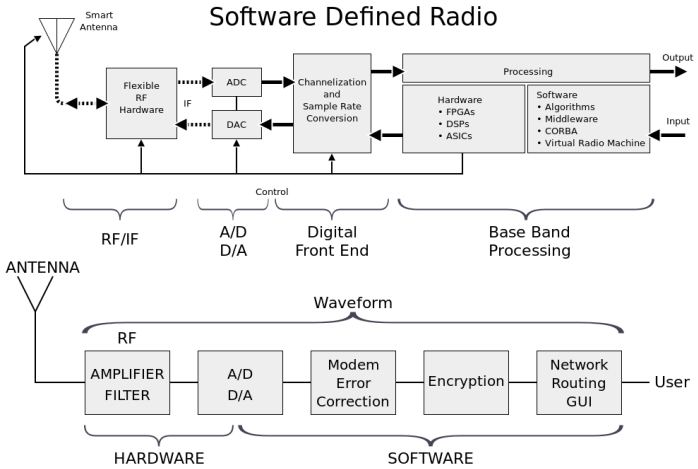

Figure 2. A breakdown of Software Defined Radio [17].

2.3.1 Hardware Functional Block

For the hardware portion an SDR the main blocks are an intelligent antenna, programmable RF module, high performance DAC and ADC. These blocks are interconnected through digital signal processing [13].

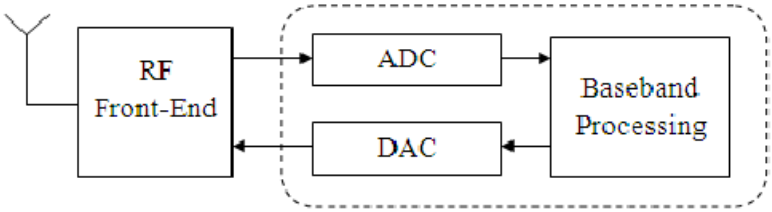

Figure 3. Hardware Decomposition of SDR [17].

Intelligent Antenna Technology

The ideal antenna for an SDR is a self-align, self-adapt and self-restorative antenna, that can adapt to its transmission requirements and required application [12]. Micro Electro-Mechanical System (MEMS) presents some optimism for major developments in broadband reconfigurable antenna design. When using MEMS switches, the antenna can be reconfigured for a new frequency band, therefore it is only required to switch in or out at different slot elements [12].

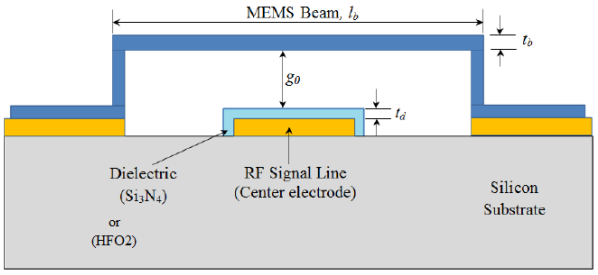

Figure 4. MEMS Switch for Reconfigurable Antennas [19].

Programmable RF Modules

For current SDR systems one of the most used techniques is to use a series of RF modules to span the full frequency band. Due to the efficiency and low loss of MEMS technology, application of high-performance RF devices with a high level of integration circuits including switch have been made possible [12].

Digital-to-Analog Converter (DAC) and Analog to Digital Converter (ADC)

An SDR system's performance is largely dependent on DACs and ADCs components. The distinctive task of these components is to convert between digital and analog and vice versa. By pushing the converter closer to the antenna, the flexibility of SDR can be increased significantly. Conventional electronic converters are pushing the envelope to achieve faster conversion rates with more resolution [12].

Analog-to-Digital-Converter implementation technology that is built on conventional semiconductor methodology is said to achieve 6 bits resolution at 3.2 GS/s (GHZ per second) and 10 bits resolution at 1 GS/s. For DAC, the highest performing approach has a capability of 12 bits at 1.3 GS/s [12].

ADC Technology	Resolution	Speed	Status
Semiconductor Based	6 Bits	3.2 GS/s	Commercially Available
Optical Spacing	8.2 Bits 12 Bits	505 MS/s, 12 Bits	Experimental Proof of Concept
Superconductor (RSFQ)	11 Bits	175 MS/s	Experimental

Table 1. ADC Technology Chart [12].

Digital Signal Processing Techniques

The process-enabling element of SDR is digital signal processing (DSP). To enable all the features of an SDR, a fixed DSP algorithm in the processing engine is required [14] [15]. Noise cancellation, compression, multidimensional filtering, adaptive processing, detection, estimation, and array processing are just some areas that have significant effect on numerous applications. When selecting a DSP engine, the main five-selection criteria to be considered is [12]:

- Programmability
- Level of Integration
- Development Cycle
- Performance
- Power

Interconnect Technology

A key benefit of SDR is the capability to connect several independent building blocks to create a radio link. The following issues must be addressed to achieve a successful interconnect [12]:

- Open standards
- Addressing multiple protocols
- Meeting increasing speed and input requirements
- Connecting to traditional circuit networks

Primarily there are three main interconnect architecture: bus architecture, switch fabric architecture, and tree architecture [12].

	Speed	**Complexity**	**Scalability**	**Application**
Bus	Slow	Low	Low	Medium
Switch Fabric	Medium	Medium	High	High
Tree	Fast	High	Medium	Low

Table 2. Summary of Basic Features and Drawbacks of each Interconnect Architecture [12].

2.3.2 Software Functional Block

An SDR, like any other software program, needs a fast and proficient method for generating, verifying, and validating the required signal-processing algorithm. An SDR is a complex "do everything" radio. Object Oriented Analysis/Object Oriented Design (OOA/OOD) has become the main software methodology for SDR. The improvements are grouped into three areas [16]:

- Significant change in the design process
- Significant growth capability and complexity
- Large increase in the rate at which change is implemented

The performance determining hardware is now digital devices such as single board computers (SBC), digital signal processor (DSP), and data buses. Radio design is now focused on computer engineering instead of electromagnetism and electronics. The complexity and capability of the radio has increased exponentially over the last two decades. Software defined radio can implement a significant number of protocols and channels. Conventional, hardware radios of the pasts were usually only single protocol and single channel devices [12].

2.3.3 Demand for Flexibility, Reconfigurability, and Responsiveness

Many in the satellite industry have already expressed the desire to apply SDR technologies on-board communications satellites. Software defined radios are a promising solution to the urgent demand for reconfigurable, flexible payloads over increasingly longer satellite system lifespans [17].

In the current era of reduced spending and cost-savings, it can be said that satellite operators are concerned with capital investment, flexibility, and reliability as opposed to technological advancement. Space insurance companies are concerned with reliability of newly developed components and newer technologies validate this trend. In-orbit flexibility is needed to adapt to ever-changing business conditions, or to crisis situations [17]. Reconfigurable payloads would eliminate the need for dedicated in-orbit spare satellites. The original concepts of flexibility and reconfigurability have evolved and

merged into a more conventional one: responsiveness. Responsiveness in SATCOM can be defined as the ability to react to different kinds of uncertainty, ranging from technological obsolescence to technical failures to geopolitical operational requirements. It can be theorized operationally introduced SDRs consisting of a collection of software and hardware technologies to reconfigure radios for multiple communication system types, will help overcome technological obsolescence and a lack of flexibility to evolving air interfaces [17].

2.3.4 Software Defined Payload

In space operations, the SDR theory of a truly software-based DSP based software remains restricted to non-complex or low-rate functionalities. When considering advanced telecommunication satellite payloads, the SDR viewpoint is far more challenging. The question a skeptic would likely ask is what the added value of SDR in SATCOM networks is [17]:

- SDR can be exploited in telecom payloads to allow adapting of the on-board processor to new waveform specification
- SDR enables the implementation of reconfigurable satellite payloads, terminals, and gateways
- SDR allows for easier integration of satellite and terrestrial networks.

Currently flexible DSP-based transparent processors are being utilized by geostationary mobile satellite networks. More SDR opportunities will become available with on ground Beam Forming Network (BFN) architectures supporting Multi-User Domain (MUD), Multiple-Input-Multiple-Output (MIMO), and other advanced DSP

techniques that can be implemented at the gateway. Broadband SATCOM network will benefit greatly from flexible (multi-beam) payloads [17]:

- To attain re-configurability of transparent broadband payloads
- To allow reprogrammable and regenerative SDR On-Board Processor (OBP):
  - For professional, governmental, and security application requiring interconnected configurations
  - Potentially high interesting to alleviate "the rigidity" of current on-board processors (the waveform will be frozen for the lifetime of the satellite)
- To support header-based packet switched high throughput transparent processor:
  - Header-only on-board demodulator (OBD) to route packets without regeneration
  - The possibility of having SDR implementation of the header processor.

2.3.5 Space Applications of Software Defined Radio

There are several activities currently being performed in the arena of SDR for on-board satellite operations [18]. The European Space Agency (ESA) has an SDR prototype in development that is being employed on regenerative SATCOM. The Software Defined Radio for Regenerative Communications Satellites (SDRRCS) will have the capability of upgrading and reconfiguring itself and adapt to present and future "waveforms". For this purpose, the platform must be made compliant to the Software

Communications Architecture (SCA), which is a set of regulations and standards (for software and hardware) designed to guarantee manageability of waveforms [17].

The ESA has another activity in development that should be mentioned. The "Evolution of On-Board Processing Applications" is a study that aims at a complete return to the OBP technologies. During this study, alternative technological solutions, and architectural designs were recommended, to including using SDR technology to fix key limitations caused by the inflexible structure of the OBPs. As a result, this should help decrease potential waste of limited satellite assets, resources, and capital [17].

Software defined radios are also finding acceptance into realization of on-board transponders for Telemetry, Tracking and Command (TT&C) [19]. The transponder of the TT&C subsystem is an essential component for most spacecraft. Frequencies of operation, requirements, and signal waveforms are dependent on the type of mission: either LEO, geostationary, or deep space. A key objective of the satellite industry is to design components and equipment that will be reusable for the greatest number of spacecraft models with minimum cost, ultimately, a single, distinctive, and universal TT&C transponder [17].

A prime example of such emerging technology is the COM DEV Europe S-band TT&C transponder. The transponder was developed and certified under the ESA's ARTES 3-4 program. The SDR is connected to a field programmable gate array (FPGA). This connection enables the use of numerous data rates and modulation schemes to provide a flexible method for several different mission types. This transponder has been implemented by Surrey Satellite Technologies Ltd (SSTL) for the first Formosat-7 spacecraft, designed to facilitate global weather forecasting [17].

Figure 5. European SDR TT&C Transponder, Developed by ESA ARTES Program [23].

The ESA also performed a study with the private sector to consider a ultra-high frequency (UHF) transceiver design that is primarily focused on the TT&C of small LEO satellites, based on SDR. The high-level architecture divides the transceiver between analog and digital. While the analog section relies on older technology (parts populating a printed circuit board), the digital section is employed in a FPGA [17].

Lastly, SDR technologies are becoming more prevalent in use with GNSS (GPS, GLONASS, EGNOS, and Galileo) receivers. Software-Defined Radios appear to be an attractive option for these systems due to their ability to adapt to modifications of navigation signal waveforms and their capability to apply effective algorithms for interference and multipath reduction. The Centre Tecnologic de Telecommunications de Catalunya (CTTC) has an open-source project GNSS-SDR. The software is a computer program that executes all the signal processing from navigation satellites signals to the computation of time, position, and velocity [20]. In November 2013, a Galileo-based

position fix was collected using a standard GNSS antenna and a RF front-end connected to a service PC running GNSS-SDR [17].

2.3.6 On-Ground Space Applications

Software defined radio technology is also being used to implement ground user terminals, i.e., satellite earth stations, providing the following benefits for users and manufacturers [17]:

- Flexibility: allowing the progression of modulation and air-interface algorithms and protocols, by simply uploading new software onto the current hardware platform
- Adaptability: ground stations can alter their capability more rapidly as network or traffic operational conditions change
- Simpler integration
- Reduced manufacturing time and quicker time-to-market
- Reduced sustainment costs
- Economics of scale: essentially, a common hardware platform can be used across different applications by loading it with the proper software application.

The Inmarsat Broadband Global Area Network (BGAN) is example of an SDR satellite Earth station, which is a well-established system that provides Internet connections to single users, via geostationary satellites. The figure below shows the

block diagram of a commercial-of-the-shelf (COTS) L-band SDR transceiver for the BGAN service [17].

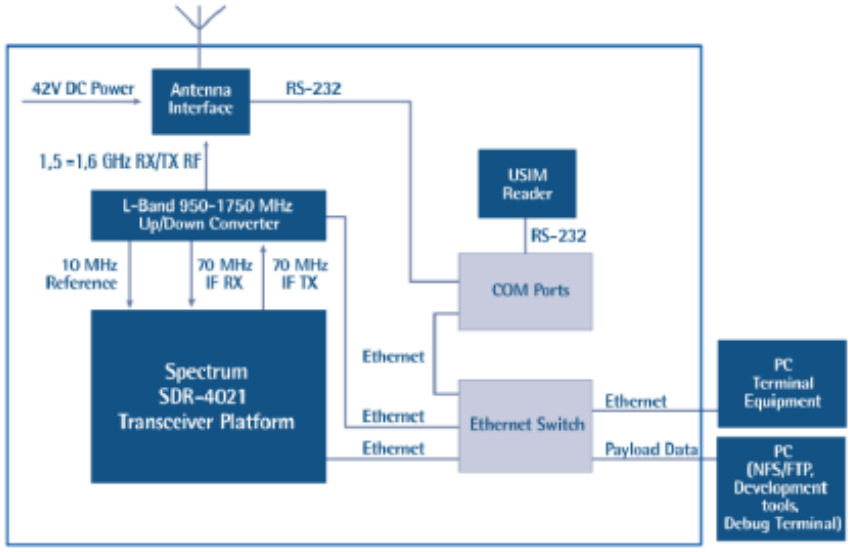

Figure 6. Block Diagram of SDR 4000 [17].

Software defined radio techniques are also being used to implement Earth-to-satellite range measurements. Pseudo Random Noise (PRN) sequences have been used to BPSK modulate, at 2 MChips/sec, the uplink carrier. The SDR processing was carried out on a PC interfaced to an SDR. The downlink signal was received from a "transparent" transponder enabled range measurement with a Root Mean Square (r.m.s) accuracy of 5m [21].

2.4 Typical Software-Defined Radio Modulation Schemes in Small Satellites

As previously mentioned, SDRs are being implemented in on-board transponders for Telemetry, Tracking, and Command (TT&C). Telemetry data is normally a few kilobytes long, but data generated from a satellite payload, can be many megabytes long. That amount of data needs a high data rate, in the order of hundreds of times, therefore, frequency bands where more bandwidth is available are usually used. A recent trend is the utilization of the X-band. This section discusses, from a practical viewpoint, how modulators of X-band space communications can be implemented [22].

The X-band is in the frequency range from 8 to 12 GHz. Higher frequencies such as these tend to have more available bandwidth per channel in comparison with lower frequencies [22].

Even if the transmitted data is digital, its electromagnetic waves are still analog. As shown in the block diagram of Figure 7, the modulator and microprocessor can both be programmed into a FPGA chip applying a hardware description language (HDL). This enhancement gives the possibility to change the modulation scheme, e.g., phase shift keying (PSK), Gaussian Minimum Shift Keying (GMSK), or Orthogonal Frequency Division Multiplexing (OFDM), with minimal to no hardware change [22].

The soft processor, a microprocessor designed via software using a programming language, controls the functions of the modulator. A control software runs by the soft processor core from the FPGA's block memory. More hardware is needed to perform the digital to analog conversion using a high-speed DAC, to filter the transmitted signal and match the impedance in RF circuit. The oscillator frequency inside the RF circuit is

tunable via command sent to the soft processor. This command indicates that the operating frequency of the transmitter is controlled by the software, as well, if it is inside the frequency range of the oscillators. Today, SDRs are purchased at low cost and its frequency range can be several Gigahertz (GHz) [22].

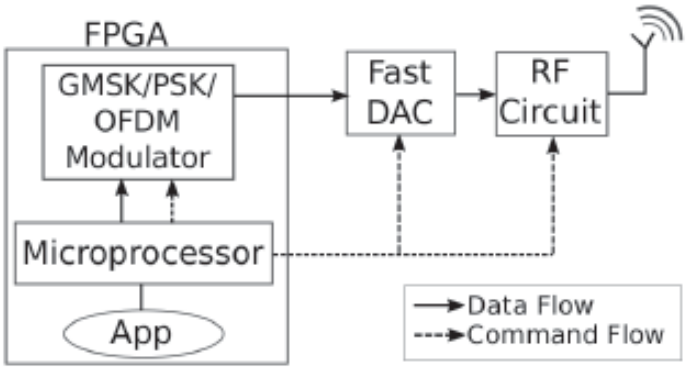

Figure 7. Block Diagram of SDR with GMSK Modulator [22].

For low Signal-to-Noise Ratio (SNR) systems like SATCOM, modulations that are more robust to the noise are mostly used. This characteristic applies to the GMSK that transmits one bit per symbol. But, when the SNR is higher, such as when directional antennas are used, it is possible to increase data rates by using modulations schemes that transmit more than one bit per symbol. In this case, the prescribed modulations for the X-band are the offset quadrature phase shift keying (OQPSK), which transmits 2 bit per symbol and the trellis coded multidimensional 8-PSK (TCM8PSK) [23].

2.4.1 Bandwidth-Efficient Modulation

The main purpose of spectrally efficient modulation techniques is to expand bandwidth efficiency. The ever-growing need for digital transmission channels has led to increased research of spectrally efficient modulation techniques to expand bandwidth efficiency and thus help improve the spectral congestion issue [24].

Some systems require additional modulation requirements besides spectral efficiency. Satellite systems with highly nonlinear transponders require constant envelope modulation. This is because nonlinear transponders generate unnecessary sidebands when transmitting a signal with amplitude fluctuations (due to a process call AM-to-PM conversion). These spurious sidebands steal away a portion of transponder power and can also produce (adjacent channel or co-channel interference). Two examples of constant envelope modulation schemes are Quadrature Phase Shift Keying (QPSK) and OQPSK both are attractive options for systems using nonlinear transponders [25].

2.4.2 PSK Signaling

Phase-shift keying (PSK) is a digital modulation process that sends data by changing or modulating the phase of constant frequency reference signal (the carrier wave). The modulation is achieved by changing the sine and cosine inputs at a precise time. This type of modulation is widely used for wireless LANs, Radio Frequency Identification (RFID), and Bluetooth communication [26].

A simple technique to characterize PSK schemes is with a constellation diagram. A constellation diagram shows the points in the complex plan where the real and imaginary axes are termed the in-phase (I) and quadrature (Q) axes, respectively due to their 90º separation. Constellation points are used to position with uniform angular spacing around a circle. This spacing provides maximum phase-separation between adjacent points thus giving the best protection to corruption. The circles are arranged in a specific manner so they can all be transmitted with the same energy. Two well-known examples are "binary phase shift keying" (BPSK) which use two phase angles and QPSK which uses four phase angles [26].

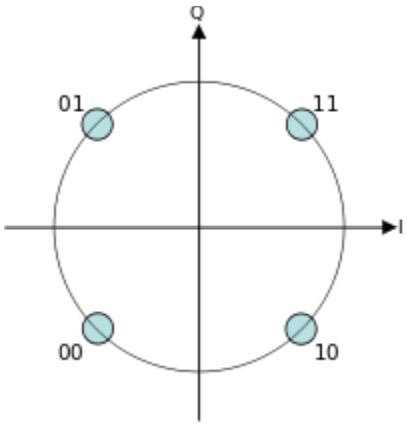

Figure 8. Constellation Diagram of QPSK Signal [27].

2.4.2.1 Quadrature phase-shift keying (QPSK)

The partitioning of a typical pulse stream for QPSK modulation can be described by

$d_I(t) = d_0, d_2, d_4 \ldots$ (even bits), (1)

and

$d_Q(t) = d_1, d_3, d_5 \ldots$ (odd bits). (2)

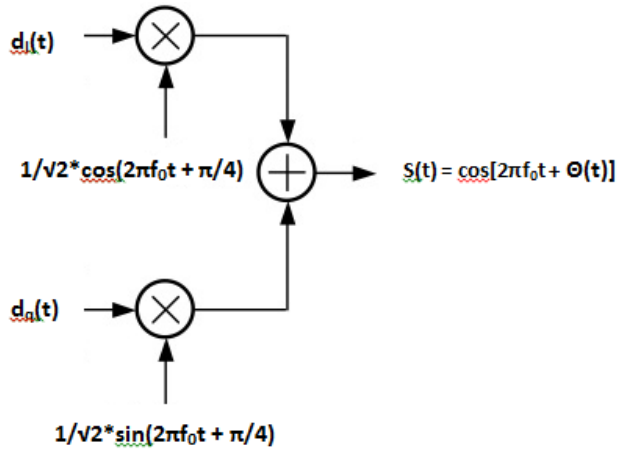

Figure 9. Simple QPSK Modulator [22]

A convenient orthogonal realization of a QPSK waveform, *s(t)*, is achieved by amplitude modulating the quadrature data streams and in-phase onto the sine and cosine functions of a carrier wave, as shown by [25]

$$s(t)= \frac{1}{\sqrt{2}} d_I(t) \cos(2\pi f_0 + \frac{\pi}{4}) + \frac{1}{\sqrt{2}} d_Q(t) \sin\left(2\pi f_0 t + \frac{\pi}{4}\right), \quad (3)$$

after applying the trigonometric identities, (3) can also be written as

$$s(t) = \cos[2\pi f_0 t + \theta(t)]. \quad (4)$$

The QPSK modulator in Figure 9 uses the sum of sine and cosine terms. The pulse stream $d_I(t)$ amplitude-modulates the cosine function with an amplitude of +1 or -1. This modulation is equivalent to shifting the phase of the cosine function by 0 or π; as

a result, it generates a BPSK waveform orthogonal to the cosine function. The summation of these two orthogonal components of the carrier signal produces the QPSK waveform. The value of $\theta(t)$ will correspond to one of four possible combinations of $d_I(t)$ and $d_Q(t)$ in (3): $\theta(t) = 0°$, ± 90º, or 180º, the output signal vectors are seen in the signal space illustrated in Figure 15. Since $\cos[2\pi f_0 t + \pi/4]$ and $\sin[2\pi f_0 t + \pi/4]$ are orthogonal, the two BPSK signals can be detected separately [25].

2.4.2.2 Software Defined Radio Using QPSK Modulation

The QPSK block diagram is shown in Figure 10. First, a non-return to zero (NRZ) signal is parallelized from serial to a 2-bit output bus. The signal is then filtered by a square root raised cosine (SRRC) filter at the in-phase (I) branch. The same process occurs at the quadrature (Q) branch. Then, the signals generated from the I and Q branches are multiplied by a sine and cosine, respectively, both with frequency *f*. Lastly, the signals are summed together and sent to the radio frequency circuit. The output modulation signal can be described by

$$s(t) = y_i(t)\sin(2\pi f t + \phi) + y_q(t)\cos(2\pi f t + \phi), \quad (5)$$

where phi is the phase of the sine and cosine waves [22].

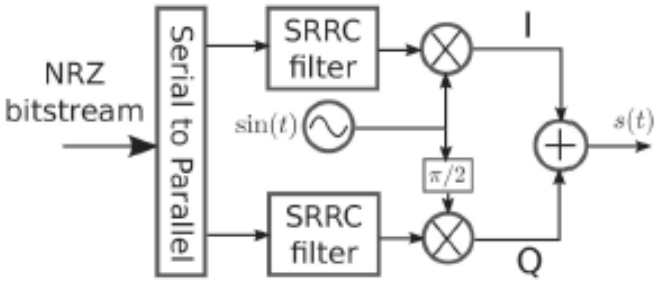

Figure 10. Block Diagram of QPSK Modulator [22].

Figure 11 displays the symbol transition diagram. (Note: the transitions between all symbols means that the symbols transit through the origin, and not just around the origin).

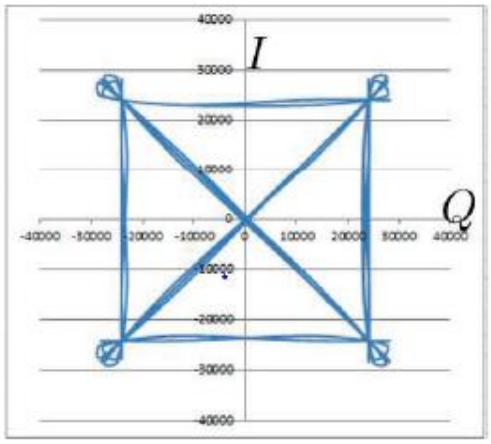

Figure 11. QPSK Symbol Transitions [22].

2.4.2.3 OQPSK Signaling

Offset QPSK (OQPSK) signaling can also be represented by (3) and (4); the main difference between the two modulation schemes, QPSK and OQPSK, is only in the alignment of the two baseband waveforms. The duration of each pulse is $2T$. For OQPSK, there is the same data stream partitioning and orthogonal transmission; the difference is that the timing of the pulse stream $d_I(t)$ and $d_Q(t)$ is shifted so that the alignment of the two streams is offset by T [25]. Figure 12 depicts this offset.

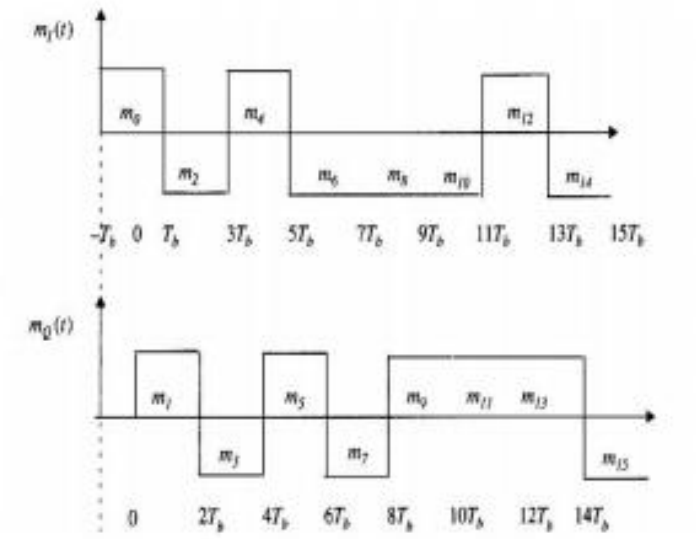

Figure 12. Offset QPSK Data Streams [25].

With a standard QPSK, because of the coincident alignment of $d_I(t)$ and $d_Q(t)$, the carrier phase can change only once every *2T*. The carrier phase during any *2T*

interval can be any one of the four phases shown in Figure 13, depending on the values of $d_I(t)$ and $d_Q(t)$ during that interval. During the next *2T* interval, if neither pulse stream changes sign, the carrier phase remains the same. If only one of the pulse streams change sign, a phase shift of ± 90º occurs. A phase shift of 180º occurs if both streams change sign [25]. Figure 14 shows a standard QPSK waveform compared to an OQPSK waveform.

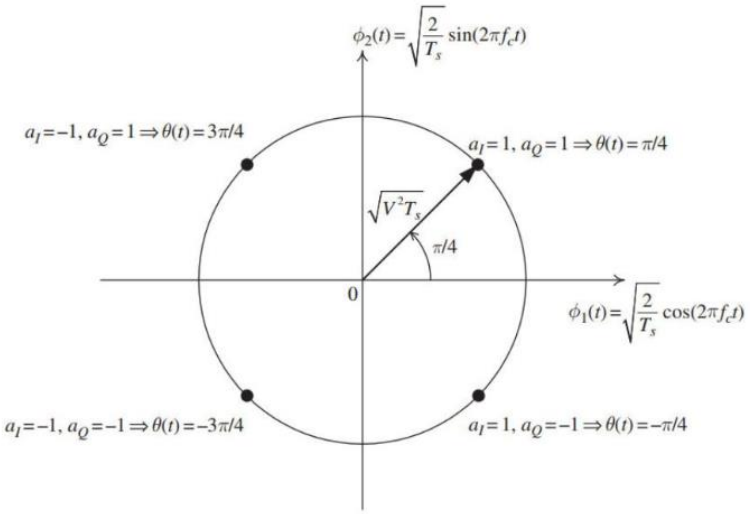

Figure 13. Signal Space for QPSK and OQPSK [25].

When a QPSK modulated signal is filtered to reduce the spectral sidelobes, the resulting waveform will no longer have a constant envelope, also an intermittent 180º phase shift can cause the envelope to go to zero momentarily. When these signals are used in satellite channels utilizing highly nonlinear amplifiers, the constant envelope will usually be restored. However, simultaneously, all the unwanted frequency side-lobes,

which can cause interference for nearby channels and other communication systems, are also restored [25].

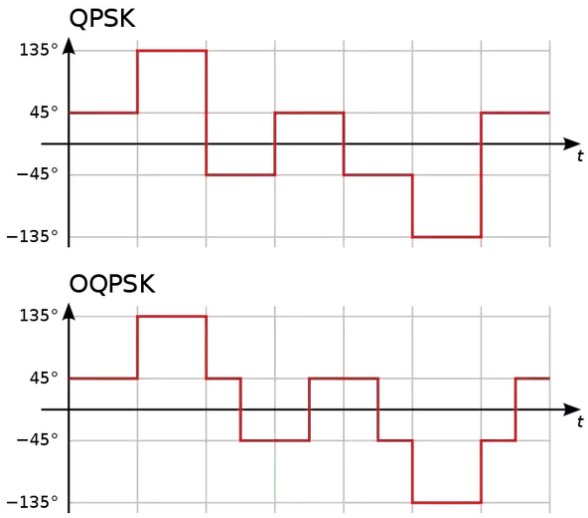

Figure 14. QPSK and OQPSK Waveforms [27].

In OQPSK, the pulse streams $d_I(t)$ and $d_Q(t)$ are staggered and therefore do not change states simultaneously. The probability of the carrier changing phase by 180º is eliminated, since only one component can transition at one time. Changes are limited to 0º and ± 90º every *T* seconds. When an OQPSK signal goes through band limiting, the resulting inter-symbol interference causes the envelope to slightly sag in the region of the ± 90º phase transition. However, since the phase transitions of 180º have been avoided in OQPSK, the envelope will not go to zero as it does with QPSK. When the bandlimited OQPSK passes through a nonlinear transponder, the envelope droop is eliminated; however, the high-frequency components associated with the collapse of the envelope are not reinforced. Therefore out-of-band interference is avoided [25].

2.4.2.4 Improved Software Defined Radio Using OQPSK Modulation

To compete with the power variation mentioned previously, in this section, a variation of the QPSK signal containing a phase offset, the OQPSK, is shown. The block diagram of an OQPSK modulator is shown in Figure 15. The same process from Figure 10 occurs, except that an offset of half of a sample period $T_s/2$ is introduced. Thus, the OQPSK transmitted signal is given by [22]

$$s(t) = y_i(t)\sin(2\pi ft + \phi) + y_q\left(t - \frac{T_s}{2}\right)\cos(2\pi ft + \phi). \quad (6)$$

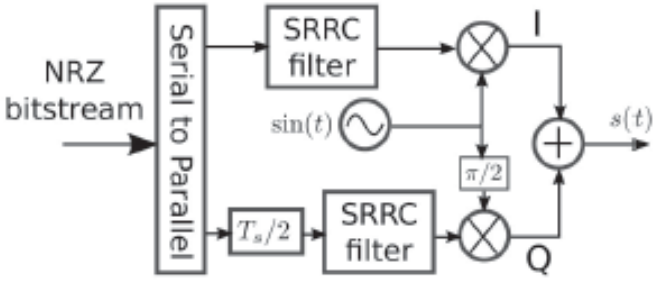

Figure 15. Block Diagram of OQPSK Modulator [22].

This offset only causes one-bit change per transition, which results in the symbol transitions shown in Figure 16. As a result, the transitions no longer crossing the origin and power variations are less severe.

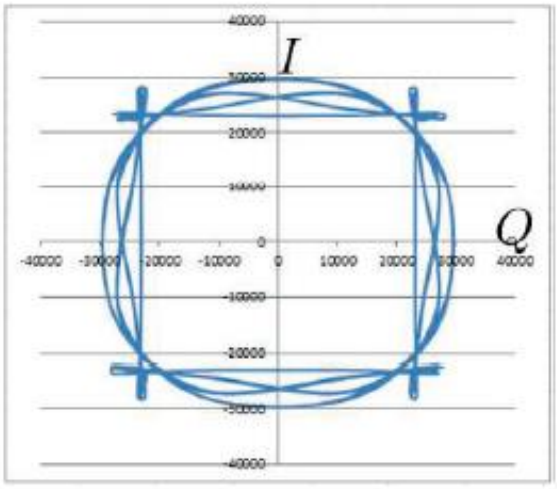

Figure 16. OQPSK Symbol Transitions [22].

Furthermore, the normalized power of an OQPSK signal is shown in Figure 17, where the power has a variation of approximately 0.6, while in Figure 18, the power variation is 1 for a QPSK signal. The power variation in Fig. 17 results in a strong linear restriction to the amplifier. Power variation simplifies the amplifier design so that linearity is only needed within this section. Linear amplifiers are understood to be inefficient and draw inert current. On the other hand, using non-linear amplifiers results in an overall improved efficiency of the OQPSK transmitted compared to the QPSK transmitter [28].

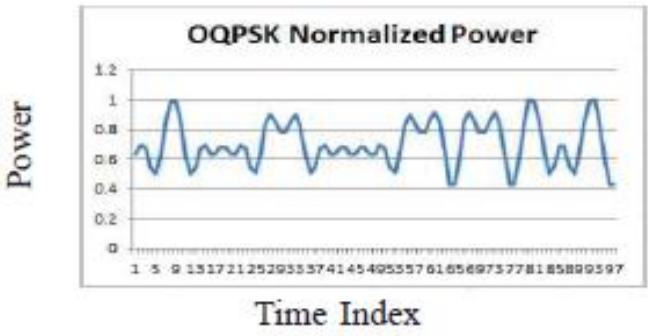

Figure 17. Normalized Power of OQPSK Signal [22].

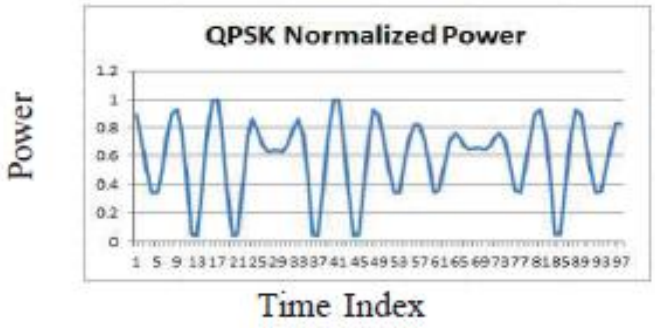

Figure 18. Normalized Power of QPSK Signal [22].

Finally, a modulator is designed so that it generates 4 sampler per symbol and the constellation is shown in Figure 19. The generated samples are displayed as red dots, and the blues lines show the paths from one sample to another. Since the original signal is filtered by the Square Raised Root Cosine (SRRC) filter, the signal is spread and samples are generated at the edges of the constellation, and at its transitions [22].

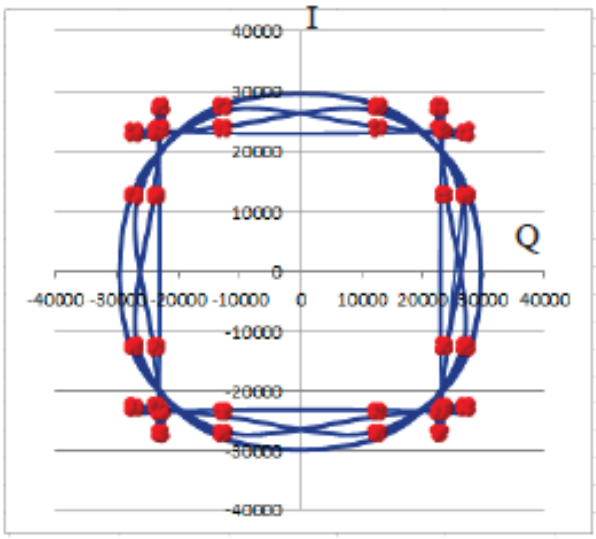

Figure 19. Constellation of QPSK Signal [22]

2.4.3 Error Performance of QPSK and OQPSK

It is known that QPSK and BPSK have the same bit-error probability because QPSK is configured as two BPSK signals modulating orthogonal components of the carrier signal. Because staggering the bit streams does not affect the orthogonality of the carrier signals, OQPSK has the same theoretical bit error performance BPSK and QPSK [25]. Although OQPSK is slightly more complex between the transmitter and receiver, the high-speed capabilities of communication systems absolve this issue. Therefore, OQPSK is a more efficient phase shift keying modulation.

2.5 Bandwidth Expansion

There has been recent research conducted at AFIT on the topic of SDRs and using them for bandwidth expansion. Previous research experimented with a technique to mesh multiple simultaneous SDR collections to realize instantaneous bandwidth expansion through MATLAB simulations and hardware testing. The research proposed having each SDR collect a particular portion, or sub-band of the transmitted signal which will then be combined with other the sub-bands to create a high bandwidth. Because each SDR is collecting different sub-band simultaneously, it is not required to know the priori signal to help with the bandwidth meshing. For his research effort, the use of Commercial Off the Shelf (COTS) SDRs was employed. Two factors were considered to achieve instantaneous bandwidth expansion [29].

First, the SDR clocks must be synchronized so the SDRs are sampling at the same time but over different frequency bands. A GPS Disciplined Oscillator was used to ensure synchronization. If an individual SDR is sampling at a faintly different frequency than another radio, the data will cover a different time span and possibly not support the non-Wide Sense Stationary (WSS) characteristics of the preferred collection. Secondly, the phase offset between the received signals must be corrected, which will require some overlap of the frequencies between the SDRs [29]. Further explanation of phase offset correction will be described in Chapter 3 (Methodology).

To prove the suggested technique, several different simulations and hardware tests were designed to demonstrate bandwidth expansion for several types of waveforms. The simulations were limited to two SDR collection scenarios; one SDR collected a lower

frequency portion while the other SDR collected a higher frequency portion. A Gaussian pulse with a predetermined bandwidth was utilized for the first simulation. The second simulation utilized a QPSK signal that was randomly generated. The randomly generated QPSK signal allowed use of bit error checking to find out if the expansion technique influenced the data contained [29].

When the phase offset was corrected, an alignment of the frequencies was necessary to ensure accurate results were collected. Each SDR is equipped with an internal master oscillator that tunes the radio. The result of this self-tuning function is that small frequency mismatch will occur between the SDRs. If the mismatch is not corrected, it will result in poor bandwidth expansion or a higher bit error rate (BER).

Three signal types (low, high, and combined) passed through a QPSK demodulation routine in MATLAB for comparison. The demodulation failed when utilizing only the low or high receiver collections. However, when the combined signal was sent through the demodulator it resulted in a 0.02% BER. The reason for this low bit error rate is because the QPSK signal contains 2 MHz bandwidth and the combined signal contains only 1.98 MHz bandwidth. As for the unsuccessful trials when only using the high or low receiver collections, the demodulation routine underperformed due to the large tuning offset. When a similar test was performed with a 1 MHz bandwidth collected at transmit center frequency, the demodulation resulted in a BER of 20%. With this outcome, it was proven that by using SDRs to retrieve specified sub-bands, correct for the phase offset between sub-bands, correct for the frequency alignment or mismatch between sub-bands then combine the sub-bands to form a wider bandwidth signal, would be successful at instantaneous bandwidth expansion [29].

2.6 Summary

Chapter II provided some background information on software defined radios, their use in space applications, and a review on past bandwidth expansion research. Section 2.1 provided an introduction into the bandwidth demands of the defense and commercial sectors; Section 2.2 detailed satellite communications and proposed solutions for increasing bandwidth in SATCOM; Section 2.3 provided a description software defined radio technology and their current application in space operations; Section 2.4 provided theoretical background on bandwidth expansion and QPSK modulation; finally, Section 2.5 detailed previous bandwidth expansion research. Chapter III details the simulation and hardware tests that will expand upon the past research conducted at AFIT and lend more study to the bandwidth demand topic.

III. Methodology

The overall objective of this research effort is to transmit a signal and have multiple SDRs simultaneously collect a specified portion of the received signal which will then be meshed back together with other sub-bands to produce a high bandwidth signal. For this proposed solution to be successful using Commercial Off the Shelf (COTS) SDRs, two factors need to be considered. First, frequency offset and phase errors must be accounted to ensure accurate signal recovery. Secondly, an algorithm must be developed to locate the beginning of the transmit signal, since the resultant received signal is unknown.

The following sections describe the research methodology. Section 3.1 describes Device under Test; Section 3.2 describes the Experimental Hardware Setup; Section 3.3 discusses bandwidth expansion models; Section 3.4 details the communication system model and signal generation; Section 3.5 details frequency offset; Finally, Section 3.6 discusses single and dual SDR simulation and hardware tests.

3.1 Device Under Test

The Device Under Test (DUT) for this research is a Universal Software Radio Peripheral (USRP) B205 mini software defined radio (SDR) from Ettus Research and is pictured in Figure 20. The USRP B205 mini is a next-generation SDR with a frequency range of 70 MHz to 6 GHz and 2x2 MIMO (Multiple Input Multiple Output) streaming. The USRP B205 mini is compatible with several software programs to include:

GNUradio, GQRX, SDR-Radio, and SDR# which can all be operated with Windows, Linux and macOS [30].

The RF front end uses the Analog Devices AD9364 RFIC transceiver with 56 MHz of instantaneous bandwidth.

Figure 20. USRP B205 mini [30].

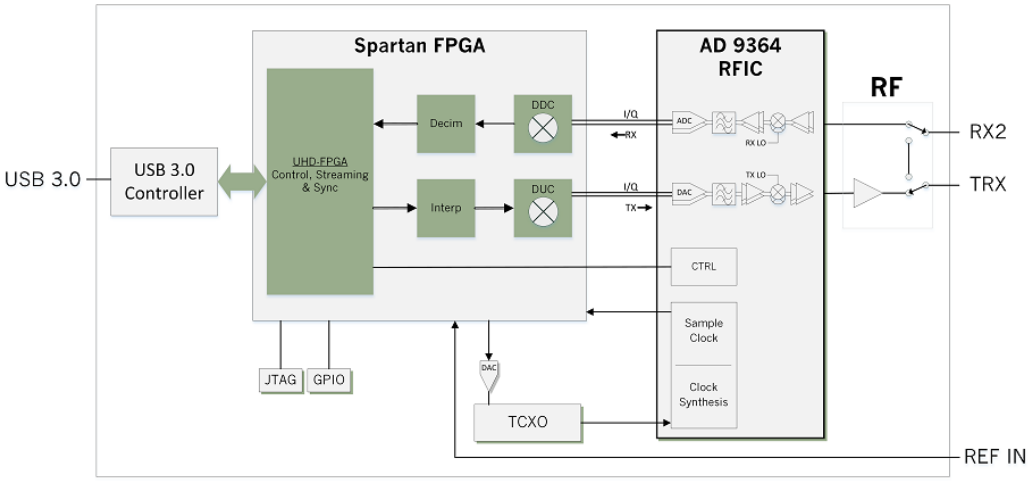

Figure 21. Block Diagram of USRP B205 mini [30].

3.2 Experimental Hardware Setup

This section covers the equipment, hardware, and hardware settings used in signal collection for the USRP B205 mini.

3.2.1 Dell Latitude 7550

The USRP B205 mini radios were connected to a Dell Precision 7550 laptop. The Dell Precision 7550 has an Intel Core i5-10400H CPU, 100GB of random-access memory (RAM), and 256GB of storage. For more streamlined operation, the laptop was configured with Linux operating system by downloading the Ubuntu open-source software. MATLAB® R2020b was also downloaded for signal processing and data analysis.

3.2.2 50-Ohm Coaxial Cable & Attenuator

A 50-Ohm coaxial cable was used to connect the USRP B205 mini radios during hardware testing. To prevent burnout of the radios on the receiver end, an attenuator was connected between the coaxial cables and receiver radios. An attenuator is an electronic device the reduces the power of a signal without significantly distorting its waveform. An attenuator is essentially the opposite of an amplifier. Whereas, an amplifier provides dB gain, an attenuator provides dB loss [31]. Further details on the experimental hardware setup will be discussed in Section 3.6.

3.3 Signal Reconstruction Models and Techniques

The signal reconstruction solution for this research effort involves having multiple SDRs receive a transmitted signal which will then be combined back together to form a high bandwidth signal. For this technique to work, the SDRs need to be synchronized with the transmitter clock to ensure they are sampling at the same interval. It is critical to control the sampling interval to ensure the signals are meshed properly. Each radio must sample at the same frequency, if one radio is sampling at a slightly different frequency than another radio, the received signal will not cover the same time span. Also, the frequency and phase offset between the received signals must be corrected. Even a slight phase offset will result in the sub-bands having an inaccurate phase relationship, which will result in a distorted waveform when combined [29]. Frequency and phase offsets will be discussed further in Section 3.5.

3.3.1 Autocorrelation

The research effort's bandwidth expansion technique relies upon autocorrelation. Autocorrelation is the correlation of a signal with a delayed copy of itself, which is a function of delay [32]. Autocorrelation is an analysis tool used for finding repeating patterns, such as a periodic signal shrouded by noise. In this instance, autocorrelation can be used for recovery of the delayed version of the transmitted signal [33]. For discrete-time processes, the autocorrelation function is,

$$R_{XX}[n_1, n_2] = E[X[n_1]X[n_2]] \tag{7}$$

which, in the case of our transmitted signal expands to,

$$R_{XX}[m] = \sum_{-\infty}^{\infty}\{x[n]x^*[n+m]\}, \qquad -\infty < m < \infty, \tag{8}$$

where $x[n]$ represents a stored replica of the transmit signal, $x^*[n+m]$ is the complex conjugate of the delayed transmit signal. The lag index is represented by $[m]$, while * indicates the complex conjugate [33].

The Fourier time-shift property can be used to exploit the lag index $[m]$. The Fourier time-shift property is,

$$x(t - t_0) \leftrightarrow e^{-j\omega t_o}X(\omega) \tag{9}$$

when substituted for the delayed transmit signal, it is

$$x[n-m] \leftrightarrow X[k]e^{-j\frac{2\pi k[n-m]}{N}}, \tag{10}$$

where $j = \sqrt{-1}$ is the complex operator and k is the frequency index having a period of N [33]. In the time-shift property, the lag index $[m]$ is used to align the receive signal sub-band with the equivalent segment of the transmit signal [5]. For this research effort, the Fourier time-shift property is used to align the received signal sub-bands with the corresponding portion of the transmitted signal. The Fourier time-shift property basically

detects the sub-band start times for the receive signal. When the start time of the transmit signal is known, it allows us to find the stop time of the receive signal. In Section 3.5.4 we discuss how autocorrelation is used to correct for phase offset.

3.4 Communications Signal System and Signal Generation

In this section we discuss how the transmission signal was generated.

3.4.1 Communications System Development

The communications system used for this research effort was developed using Matrix Laboratory (MATLAB®) 2020b. Coded scripts were used in the initial development until a final process was agreed upon. The following sub-sections provide an overview of the communications system model to include its transmitter and receiver. Bandwidth expansion simulations and hardware tests are discussed in later sub-sections.

3.4.1.1 The Transmit Signal Generation

Signal generation and transmission process began by using MATLAB® 2020b to generate 2000 uniformly distributed randomly sampled bits to form a complex symbol. A function in MATLAB was created that used a specific number of bits and bursts to generate a transmit signal. Two preambles, named *'preambleLow'* and *'preambleHigh'*, were generated to locate the beginning of the transmit signal. *'PreambleLow'* is shifted to the left by the sampling frequency, while *'preambleHigh'* shifted to the right by sampling frequency. After the preambles are shifted, they are summed together to form the transmit signal preamble. The transmit signal preamble is then combined with the

transmit signal. The transmit signal preamble is then autocorrelated to find the frequency offset of the transmit signal which must be known for successful transmission and reception. Figures 22, 23, 24 and 25, show the power spectral density (PSD) plots for *'preambleLow'*, *'preambleHigh'*, the transmit signal preamble, and the transmit signal.

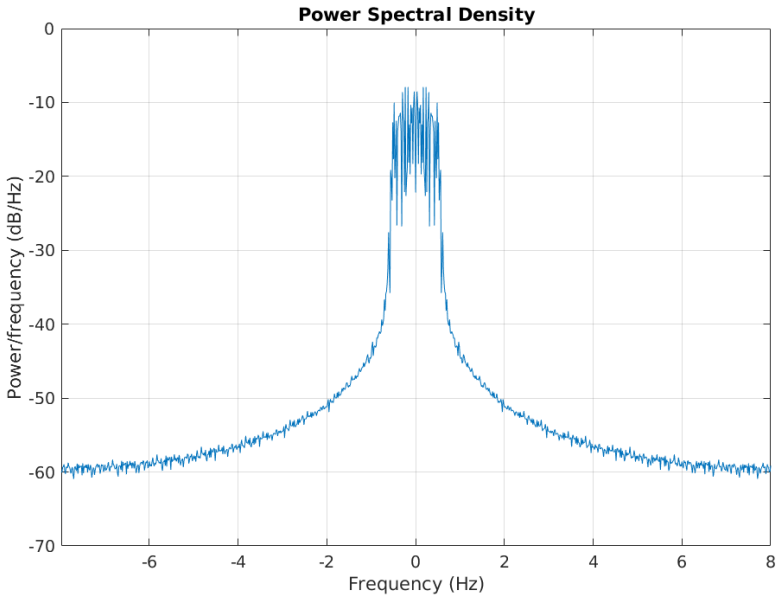

Figure 22. Power Spectral Density (PSD) plot of lower frequency transmit signal preamble.

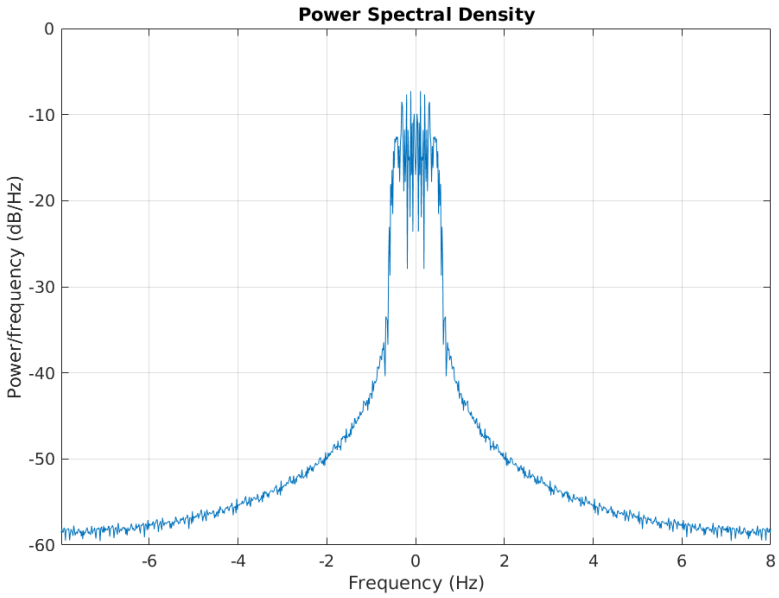

Figure 23. Power Spectral Density (PSD) plot of higher frequency transmit signal preamble.

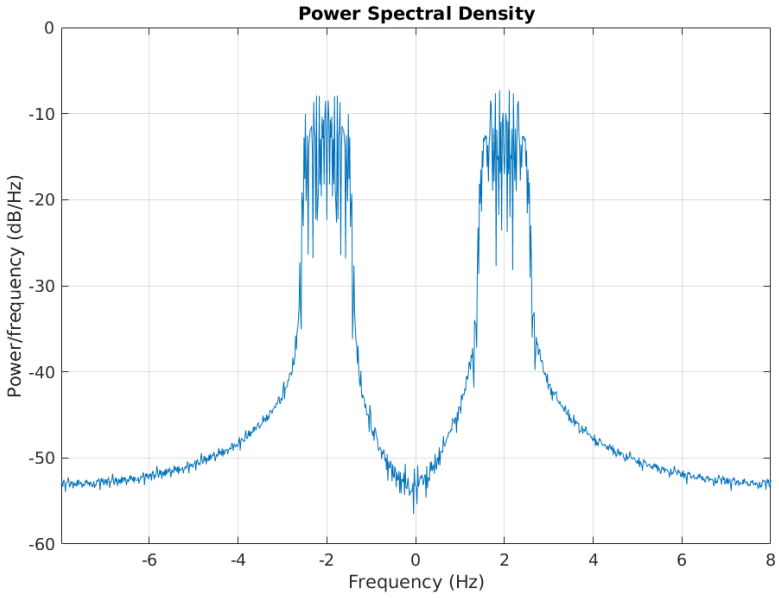

Figure 24. Power Spectral Density (PSD) plot of the transmit signal preamble.

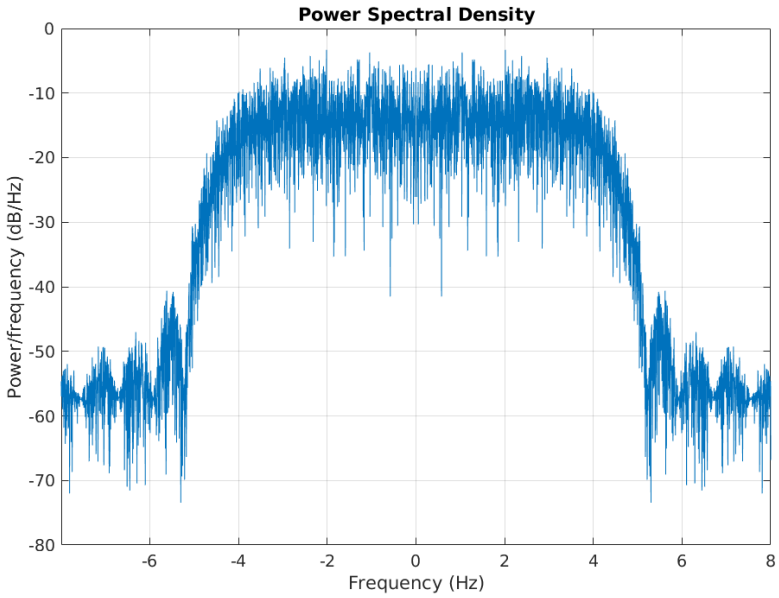

Figure 25. Power Spectral Density (PSD) plot of transmit signal.

3.4.1.2 The Received Signal Generation

To generate the simulated received signal, Additive White Gaussian Noise (AWGN) was added to the transmit signal via the comm.AWGNChannel toolbox in MATLAB. During propagation while the transmit signal is being sent, it is assumed that the signal is exposed to Additive White Gaussian Noise (AWGN). The USRP B205 mini SDRs receive the AWGN-afflicted transmit signal after propagation delay. The propagation delay is relative to the propagation distance between the transmitter and receivers. In the simulation, the received signal should have the same number of bits as the transmit signal. Hence, during the two-receiver test, the propagation delay values with differ slightly for each SDR [4].

3.4.1.3 Communication System Validation

To validate this communication system, a QPSK simulation was generated. The goal is to confirm whether the transmit signal can be sent to multiple receivers and spliced back together with equal or greater bandwidth. The QPSK simulation will input a 10 MHz bandwidth signal into the communication system process model [5].

To avoid symbol interference, a frequency shift of 10 kHz was applied to the QPSK signal, one positive and one negative. A function was generated in MATLAB to implement the frequency shift. By avoiding symbol interference, a unique bit sequence can be demodulated. For bandwidth expansion purposes, frequency shifting allows adjacent sub-bands to be aligned.

3.4.2 QPSK Signal

The theory behind the QPSK was discussed in Section 2.4.2.1. The QPSK modulation is used in various applications in modern digital communication systems, to include satellite communications. It provides twice the bandwidth performance of Binary Phase Shift Keying (BPSK) with the same bit error rate.

3.4.3 Bit Error Rate Measurement

To analyze the communication system's output, the probability of error for a given noise degradation must be measured. After it is computed, the probability of error is then compared to simulated and theoretical values of QPSK signals. This measurement

determines the consequence from using the bandwidth expansion technique in terms of the required signal and symbol power [5].

Probability of error versus Signal-to-Noise Ratio (SNR) is applicable to analog and power signals. However, in this case we are measuring digital signals. Digital signals are energy signals, and their probability of error is normally presented as Bit Error Rate (BER) and SNR is presented as Energy per Bit to Noise Power Spectral Density (E_b/N_0) [25]. In terms of a complex input signal, the relationship between E_b/N_0 and SNR is,

$$\frac{E_b}{N_0} = \frac{S}{N}\left(\frac{W}{R}\right), \tag{11}$$

where E_b is the bit energy, N_0 is the noise power spectral density, $\frac{S}{N}$ is the SNR, W is the bandwidth and R is the bit rate [25].

The basis of digital messages is information bits. The measurement of E_b/N_0 allows two communication systems to be compared at the information bit level [25]. Therefore, E_b/N_0 is the most logical figure of merit for digital communication systems. The Bit Error Rate (BER) is produced by counting the bit errors in the presence of noise once the new signal is generated.

3.4.4 Signal Transmission and Reception

To produce a transmission and reception between two USRP B205 minis, the radios were configured using a downloaded application for Linux, GNU radio, that sets parameters for the radios in the command terminal. In the GNU radio interface, the

radios connected to the desktop are detected, once their parameters (i.e., frequency, sample rate, bandwidth, and gains) are set their roles must be established. One radio is set as the master/transmitter and the second is set as the slave/receiver. Once each is armed with its role, the signal is ready to be transmitted.

On a high level, the transmission and reception process are split into three sections. First, a signal is transmitted from a USRP B205 SDR. The signal is then received by the two receivers where the signal bandwidths will then be combined. Finally, the signal goes through phase, timing, and gain correction to generate the high bandwidth signal [34].

For successful signal combination, the two signals must be summed together in the frequency domain. There cannot be any magnitude or phase distortion when summing the received signals. Any distortion from the phase or magnitude will corrupt the digital data and invalidate the combing process. A Nyquist filter is used to shape the bandwidths of the signal. Having a Nyquist filter allows the two received signals to be summed together without any magnitude distortions. For phase distortion correction, the signals must pass through a phase error control loop. This loop is comprised of a phase error detector, a loop filter, and multiplication by a complex exponential. The complex exponential shifts one received signal up in frequency and shifts the other received signal down in frequency. These shifts arrange the signals in their proper places to be summed together and combined [34].

After being multiplied by the complex exponential, the received signals enter the phase error detector. Once the signals enter the phase error detector, each one passes through a low-pass filter. This filter isolates the part of the two signals that overlap one

another. An error signal is created from the overlapping sections of the signal. The error signal is then passed through a low pass filter to prepare for the loop filter. The loop filter then outputs a signal with no phase distortions when summed together [34].

3.5 Carrier Frequency Offset

Once the signal is collected, its offsets must be corrected. The transmitting and receiving nodes are typically two distinct and spatially separate units. Because of this, relative frequency offsets will exist between their local oscillators (LOs) due to natural effects such as temperature differences, electrical noise, and impurities. Since these offsets are generally dynamic the LOs will drift electrical noise and temperature differences with respect to one another. Generally, these offsets will have phase mismatches, frequency drift, random noise, and of course frequency offset [35]. When exploring options for commercial oscillators, it must be noted that the frequency offset is given in parts per million (PPM), which translates into the highest carrier offset for any given frequency. The maximum carrier offset Δf can be related to the operating frequency f_c by

$$f_{o,\max} = f_c \; x \; PPM/10^6 \tag{12}$$

A distorted source signal at baseband *s(k)* can be calculated with a carrier frequency offset of $2\pi f_o$ *(*or ω_o) as

$$r(k) = s(k)e^{j(2\pi f_o kT+\theta)} + n(k) = s(k)e^{j(\omega_o kT+\theta)} + n(k), \quad (13)$$

where $n(k)$ is a zero-mean Gaussian random process, T is the symbol period, θ is the carrier phase, and ω_o the angular frequency [35].

Normally for a source signal with frequency and phase offsets, the frequency offset is corrected. Once the frequency offset is corrected, the signal bits are stabilized and positioned in a fixed location. Even though the bits are fixed and no longer moving, they are now out of position and the phase must be corrected. The phase recovery arranges the bits in their original space. Figures 26, 27 and 28 demonstrate this process on a QPSK signal with offsets. For this correction to occur, it is key to understand the relationship between phase and frequency. In terms of phase, angular frequency ω, or equivalently in frequency $2\pi f$, is just a measure of altering phase θ over time [35]

$$\omega = \frac{d\theta}{dt} = 2\pi f. \quad (14)$$

Therefore, recovering the phase of the signal is essentially recovering its frequency. Due to this relationship, the frequency of the signal can be estimated more easily since it cannot be directly measured, unlike phase. This method can be displayed by writing a MATLAB script that produces a basic continuous wave (CW) tone at a given frequency, measures the instantaneous phase of the signal, and takes the difference of those

measurements as the frequency estimate. The instantaneous phase θ for any complex signal can found using

$$\theta = tan^{-1}(\frac{Im(x(k))}{Re(x(k))}), \tag{15}$$

where Re and Im represent the real and imaginary components of the signal, respectively [35].

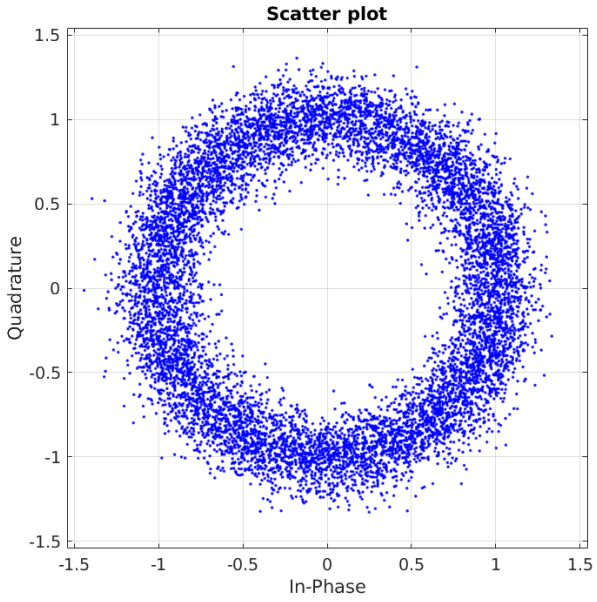

Figure 26. QPSK signal uncorrected for frequency and phase offsets.

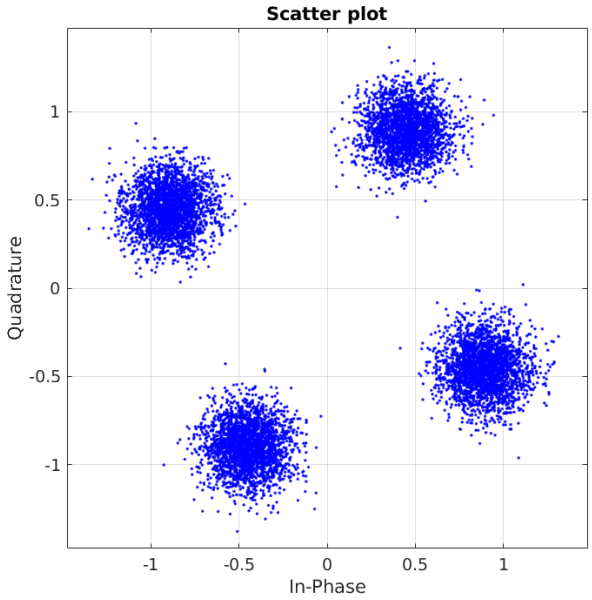

Figure 27. QPSK signal corrected for frequency offset only.

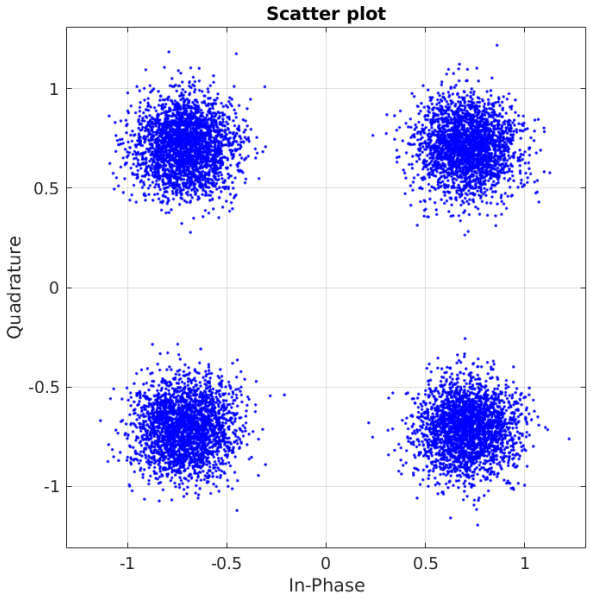

Figure 28. QPSK signal corrected for frequency and phase offsets.

3.5.1 Effects of Carrier Frequency Offset

Carrier Frequency Offset (CFO) is a non-ideal condition that may affect a baseband receiver. When considering the design parameters for a baseband receiver, the designer must notice the degradation invoked by noise and the non-ideal channel. The designer must also account for RF and analog parts as well. A few other non-idealities include phase noise, IQ imbalance, power amplifier, and sampling clock offset to go along with CFO [36].

Carrier frequency offset typically arises when the LO signal for down-conversion does not match with the carrier signal contained in the received signal. This mismatch is usually due to two critical reasons: the Doppler effect as the transmitter or the receiver is moving; and a frequency mismatch in the transmitter and receiver oscillators. As a result, the received signal is shifted in frequency. A mismatch in carrier frequency usually results in inter-carrier interference (ICI). Over time, the oscillators for both the transmitter and receiver will not oscillate at same frequency. Therefore, carrier frequency offset always exists even if there is no Doppler effect [36].

3.5.2 CFO Estimation

Carrier frequency estimation has two primary categories: data-aided (DA) and blind correction. Data-aided techniques typically use correlation type structures that depend on understanding of the received signal, usually a preamble, to estimate the carrier offset f_o. Even though DA methods usually give accurate estimates, their performance is limited by

the length of preambles [37], so as the preamble length is increased the system output is decreased [35].

On the other hand, blind or nondata-aided (NDA) techniques operate over the entire duration of the signal [35]. For the purposes of this research, an DA FFT-based technique was used for frequency estimation and correction. The preambles are overhead and are used for frequency estimation. Figure 29 shows two frequency domain signals with and without frequency offset.

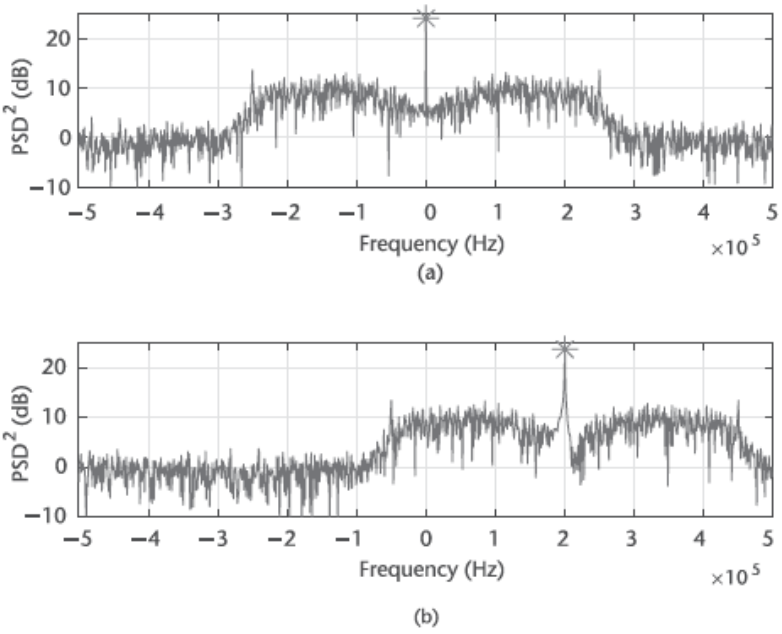

Figure 29. Comparison of domain signals with (a) and without offset (b) [35].

These figures give a rough estimate on the frequency offsets, but since the signal is not symmetrical in frequency, taking the peak from the FFT will not be fully accurate.

To correct this issue, the modulation components must be removed from the received signal by raising the signal to its modulation order M. From Fig. 29 the following is observed by

$$r^M(k) = s^M(k)e^{j(2\pi f_o kT+\theta)M}. \quad (16)$$

From this modulation the offset shifts to M times from its initial position and makes $s(t)$ completely real or completely complex. Hence, the $s^M(t)$ term can be ignored and just the remaining exponential or tone remains. To approximate the location of this tone, the FFT of $r^M(t)$ is calculated and correlated to the bin with the highest energy to the position of this tone. The frequency plot of $r^M(t)$ in Fig. 29 is a BPSK signal with an offset of 10 kHz. It is easy to see the peak is visible at two times the expected frequency. The frequency estimation is calculated by

$$f_o = \frac{1}{2TK} arg\left|\sum_{k=0}^{K-1} r^M(k)e^{-j2\pi kT/K}\right|, \quad (17)$$

where *K* is the FFT length [38]. The resulting f_o is one of the *K* values generated by the FFT. However, the accuracy of this estimation can be extended across a fixed number of FFT bins over multiple estimates if required. The frequency resolution of each FFT bin for the signal is defined as

$$f_r = \frac{1}{MTK}. \quad (18)$$

the performance of the estimator can be increased by increasing the size of the FFT or by decreasing system's sample rate. However, the sample rate of the system should not be decreased below the bandwidth of the signal [35].

3.5.3 Phase Offset Correction

The phase offset must be corrected after the signal's frequency offset has been estimated. Phase offset leads to undesirable phase rotation as well as ICI. Correcting for phase offset will require some overlap of the frequencies when using multiple SDRs. The synchronization of the clock should enable a more accurate calculation of the phase offset which is required for bandwidth expansion. If there is a slight phase offset between SDRs, the sub-bands will have an incorrect phase relationship [29].

Previous research was able to determine the phase offset between the sub-bands by utilizing a slight overlap in the collected bandwidth. The overlap enabled the collected signals to be cross correlated in the time domain, which gave a lead or lag between the two signal collections. The lead/lag was then applied to one of the signals in the time domain. A time shift is then applied to correct for the phase offset in the frequency domain through the Fourier time-delay property:

$$x(t - t_d) \leftrightarrow e^{j\omega t_d} X(j\omega) \tag{19}$$

Once the phase offset is eliminated, the consecutive bandwidths can be combined in the frequency domain to create a wider bandwidth [29].

3.5.4 Phase Correction – Auto-correlation

Auto-correlation plays a significant role in phase correction. In summary, the proposed phase correction technique calculates the angular difference between the inner product of two complex vectors. It assumes that the angular difference is restricted to values between 0 and 2π radians. When the angular difference exceeds 2π, the signal must be repetitive. This method assumes the vectors are the same length (*M*). The angle (θ_m) is,

$$\theta_m = \angle \sum_{i=1}^{M} \{x_i[n] x_i^*[n+m]\}, \tag{20}$$

where $\angle$ is the angle operator, $x[n]$ is a stored replica of the transmit signal, and $x^*[n+m]$ is the complex conjugate of the delayed transmit signal, or the lag index $[m]$. * denotes the complex conjugate and subscript *i* indexes the *ith* vector value [33].

The replica transmit signal $x[n]$ and its delayed version $x^*[n+m]$ are represented by complex vectors. Angle (θ_m) returns a value in radians. To phase-align the complex vectors, θ_m is substituted into the following equation. The phase-corrected vector is given by

$$x[n] = x[n+m] e^{j\theta_m}, \tag{21}$$

where θ_m is replaced with opposite sign [33]. The phase correction is achieved in the phasor domain. Also, the resultant shift time-aligns the signal in the time domain [5].

3.6 Simulation and Hardware Testing

The Department of Defense (DoD) has always used simulation and modeling in many ways. These applications have covered the full range of available techniques: continuous, discrete, combined continuous-discrete, and Monte Carlo. Despite being technically challenging, simulation and modeling allows the DoD to examine many strategic and tactical options in a "near-laboratory" environment [39]. In Sections 3.6.1 and 3.6.2, the methodology for the single receiver and dual receiver simulations will be described. The results of these tests will serve as a baseline for the hardware tests that aim to achieve bandwidth expansion.

3.6.1 Single Receiver Single Channel Simulation

The aim of the single receiver, single-channel simulation is to recover a 10 MHz transmit bandwidth using a single SDR. This simulation is initially performed with no frequency offset and phase error. Since this simulation is not coded with the frequency offset or phase error, alignment between the recovered signals (channels) and the transmit signal is not required. It is expected that the autocorrelation plots will identify zero lag, thus requiring no subsequent correction [5].

For the 10 MHz QPSK signal, the sample rate is set to 10 MS/s. A simulated 10 MHz bandwidth transmit channel was generated in MATLAB using the same parameters discussed Section 3.4.1.1. The transmit signal has a sample rate of 10 MS/s [5]. The symbol rate 2.5×10^6 symbols per second. The results from the single-receiver, single-channel simulations are detailed in Section 4.2.1.

3.6.2 Dual Receiver Simulations

The set-up for the dual SDR receiver simulations is like that of the single SDR receiver simulations. The QPSK dual simulation retains the 10 MS/s sample rate. To simulate SDR local oscillator variation, random frequency offsets and phase errors were applied to the signal. Based on preliminary observation of typical USRP B205 mini SDR performance, a frequency offset of -2.5 MHz to 2.5 MHz and phase error of 0 to 2π were applied. The offset correction method described in Section 3.4.3 is used to correct the simulated frequency offsets. The autocorrelation phase correction method from Section 3.4.4 is used to correct the simulated phase errors. The results for the dual receiver simulations are detailed in Section 4.2.2.

3.6.3 Single Receiver Hardware Tests

The methodology for the single receiver hardware test is discussed below. Section 3.6.3.1 utilizes a single channel method. The purpose of this test is to compare the Bit Error Rate (BER) versus E_b/N_0 to the dual receiver hardware test BER versus E_b/N_0.

3.6.3.1 Single Receiver Single Channel Tests

The aim of the single-receiver, single-channel test is to serve as a baseline for the dual receiver hardware test. The wide-band signal is assumed to be time-invariant, thus requiring simultaneous collection of the transmitted signal bandwidth. It is also assumed that the bandwidth of the wide-band signal exceeds the instantaneous receiver bandwidth [5].

For this test, a QPSK signal bandwidth is transmitted. The signal is transmitted at a center frequency of 2.4 GHz and collected using a file name '*Rx1.bin'*. The '*Rx1.bin'* is a binary file that must be read into a MATLAB function that converts binary files to a column vector of complex numbers. In MATLAB, the sampling rates are set to 10 MS/s.

The USRP B205 mini SDRs are connected to the laptop via a USB 3.0 cable which is the default USB cable that comes supplied with the radios. The SDRs are connected through their respective transmission and reception channels by a coaxial cable. This hardware setup remains the same throughout. An illustration of the single SDR receiver hardware tests setup is depicted in Figure 30. Autocorrelating the transmit bits with the received bits will confirm symbol recovery. The results for the single receiver hardware test are detailed in Section 4.3.1

3.6.4 Dual Receiver Hardware Tests

The aim of the dual receiver hardware test is to recover a 10 MHz QPSK transmit signal using two SDR receivers, with each receiver recovering half of the transmit signal. The purpose of this test is to confirm whether the signal reconstruction technique can generate a high bandwidth signal using multiple SDRs. The frequency offset and phase errors arising from the additional receiver must be accounted and corrected [5].

The two USRP B205 mini SDR receivers, labeled as '*RxLow'* and '*RxHigh'*, were prepared for this hardware test. The first receiver, *RxLow*, has a center frequency of 2.3995 GHz, while the second receiver, *RxHigh*, has a center frequency of 2.4005 GHz. The sampling rate in MATLAB is set to 10 MS/s and the sampling rate for the two receivers is set to 5 MS/s. The USRP B205 mini software shifts the transmit signal

frequency from baseband to 2.4 GHz during transmission. The purpose of the frequency shift is to ensure the signal is transmitted at frequencies that are supported by the USRP B205 minis. The two receivers are responsible for the capturing each end of the transmit bandwidth; '*RxLow'* captures the lower half of signal, while '*RxHigh'* captures the upper half. A bandwidth overlap is provided by the frequency offset settings. It is assumed the frequency offset settings provide a contiguous bandwidth recovery [35]. Again, the received signals are passed via the USB 3.0 cable to the laptop for signal processing in MATLAB. Figure 31. depicts the hardware test circuit for this test. The results for the dual receiver hardware test are detailed in Section 4.3.2.

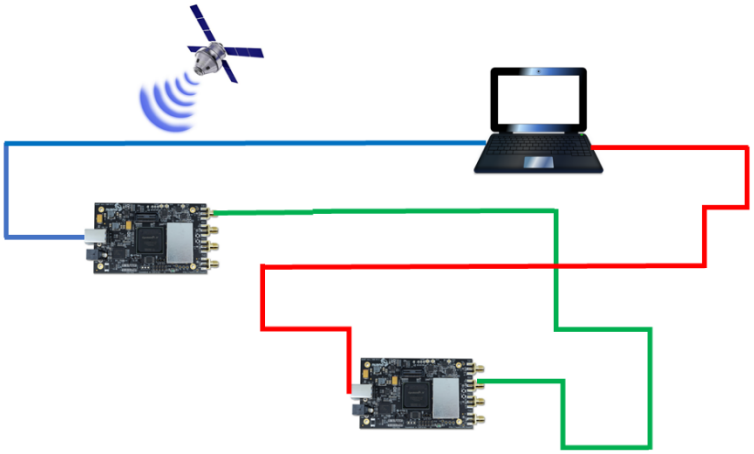

Figure 30. Hardware circuit setup for single SDR receiver test. Connection legend: TX USB 3.0 cable (Blue); RX USB 3.0 cable (Red); Coaxial Cable (Green).

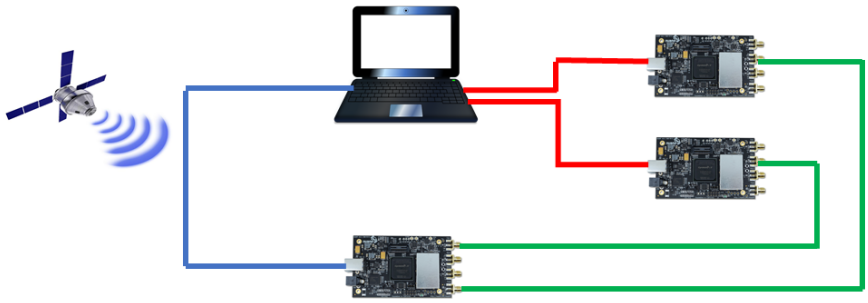

Figure 31. Hardware circuit setup for dual SDR receiver test. Connection legend: TX USB 3.0 cable (Blue); RX USB 3.0 cable (Red); Coaxial Cable (Green).

3.7 Summary

Chapter III described the methodology for this research effort. Device under test (DUT), equipment, bandwidth expansion techniques, communication system and signal generation, QPSK, symbol recovery metrics, and frequency offset were described. Signal preparation details were provided and the bandwidth expansion simulation and hardware tests for single and dual SDRs were outlined. Chapter IV will provide results and analysis of the simulations and hardware tests.

IV. Results and Analysis

The following sections provide results and analysis for the research effort. Section 4.1 provides results and analysis for the communication system process model validation. Section 4.2 provides simulation results and analysis for single and dual receiver cases. Finally, Section 4.3 provides test results and analysis for the single and dual receiver hardware tests.

4.1 Communication System Process Model Validation Results

To validate the effectiveness of the proposed communication system, a Monte Carlo QSPK signal simulation was conducted. A Monte Carlo simulation is a model that is used to predict the probability of different outcomes when the inclusion of random variables is present. A Monte Carlo simulation helps explain the impact of risk and uncertainty in prediction and forecasting models [40]. The Bit Error Rate (BER) versus E_b/N_0 for 1 to 10 dB is averaged over 100 simulations/runs. The simulation is performed with no phase errors or frequency offsets. The QPSK signal simulation BER versus E_b/N_0 is expected to be equivalent to theoretical QPSK results. Figures 32 and 33 show there are no frequency or phase offsets, respectively. Figure 34 displays the BER versus E_b/N_0 for the QPSK signal simulation. An BER of approximately 8 x10^{-4} at an E_b/N_0 of 10 dB is shown for the QPSK simulation. This result is equivalent with the theoretical QPSK BER versus E_b/N_0. The concave plot, along with the simulation

results being consistent to the theoretical QPSK BER versus E_b/N_0, indicate that accurate bit recovery is feasible using this communication system process model.

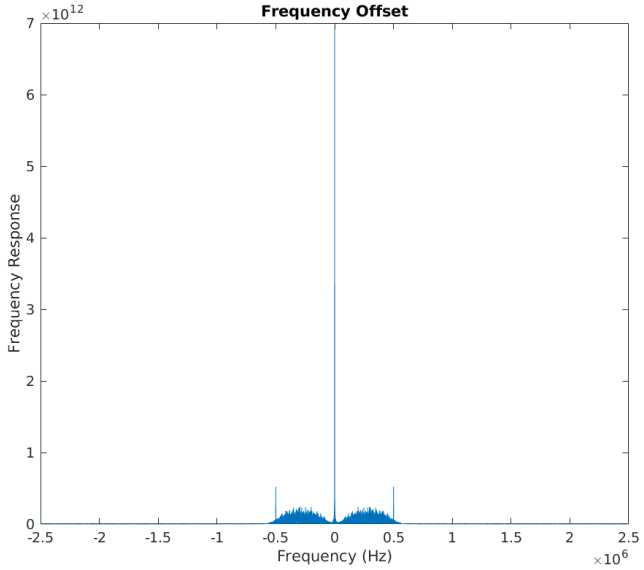

Figure 32. The frequency offset plot shows there is no frequency offset for the communication system simulation.

The carrier frequency of the M-PSK signal can be estimated by raising the sampled M-PSK signal to the M power to remove the modulation. A significant tone is generated at M times when the signal is raised to the M power, revealing the suppressed carrier frequency. In this specific case of a QPSK signal, the tone at four times the carrier frequency is apparent by [41],

$$R^4(n) = S^4a^4(n)\exp(j8\pi f_c t) + 4S^3a^3(n)\exp(j6\pi f_c t)\,\omega(n) + 6S^2a^2(n)\exp(j4\pi f_c t)\,\omega^2(n) + 4Sa(n)\exp(j2\pi f_c t)\,\omega^3(n) + \omega^4(n)\,. \quad (22)$$

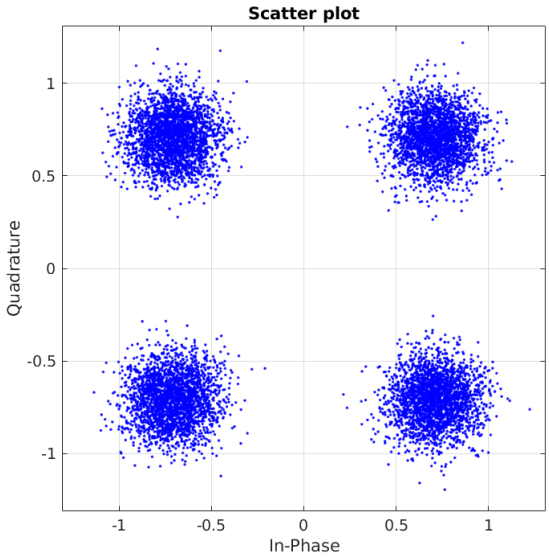

Figure 33. The phase offset plot shows there is no offset for the communication system simulation.

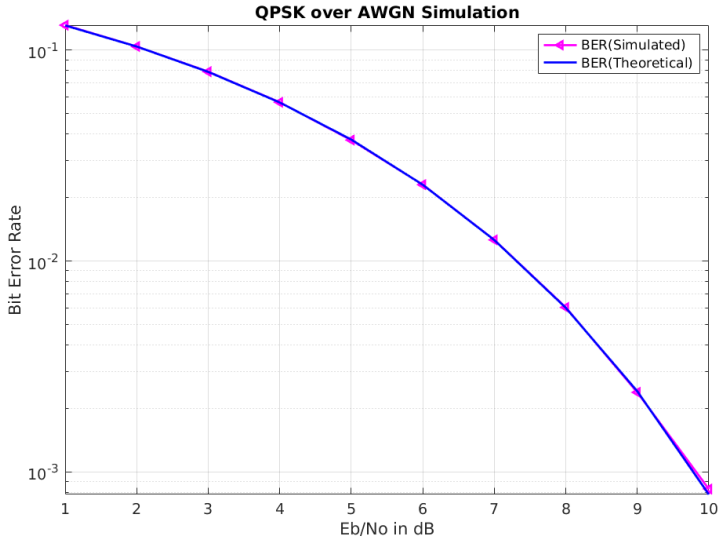

Figure 34. Probability of Bit of Error (BER) vs. E_b/N_0 for simulated QPSK communication system versus the theoretical QPSK BER.

4.2 Simulation Results

This section shows the results for the single and dual receiver simulations. The result for the single-receiver, single channel is used as a comparative measure for the Bit Error Rate (BER) plots for the dual receiver simulations. The single receiver, single-channel simulation results are discussed in Section 4.2.1. The dual receiver simulation results are discussed in Section 4.2.2.

4.2.1 Single-Receiver Single-Channel Simulations

The aim of the single-receiver, single-channel simulation is to generate comparative plots for the bandwidth expansion technique to compare with the dual receiver simulations. Figure 35 shows the PSD plot of the simulated 10 MHz QPSK transmit signal bandwidth for the single SDR collection, which will be the same transmit signal used for each scenario. Figure 36 shows the PSD plot of the simulated 10 MHz received signal bandwidth with Additive Gaussian White Noise (AWGN).

The Monte Carlo simulated BER versus E_b/N_0 plot for the 10 MHz QPSK transmit signal bandwidth is provided in Figure 37. The Monte Carlo simulation was again conducted over 100 runs/simulations. The single-receiver, single-channel QPSK simulation BER was compared with the theoretical QPSK signal BER. A BER of $3\text{x}10^{-5}$ at an E_b/N_0 of 9 dB is indicated. As expected, there is no dB loss between the simulated single-receiver collection and the theoretical QPSK signal BER.

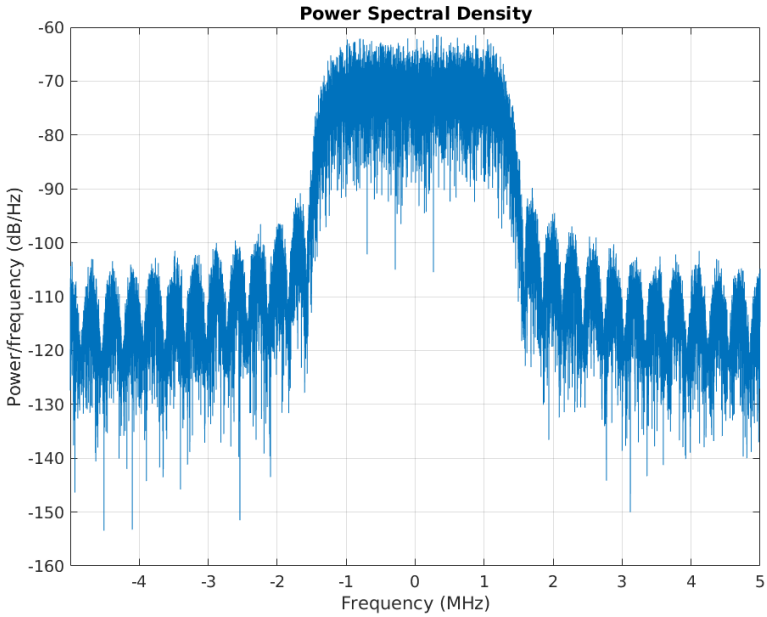

Figure 35. Simulated 10 MHz QPSK transmit signal PSD.

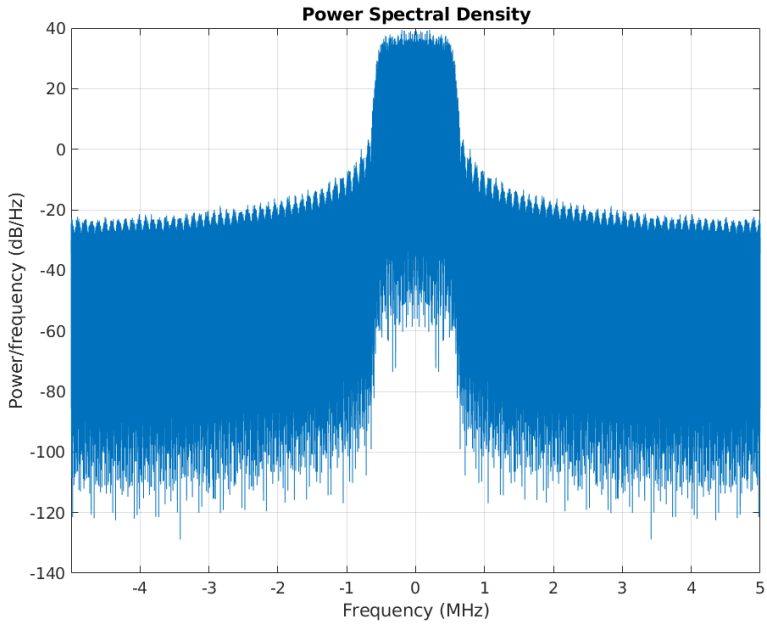

Figure 36. Simulated 10 MHz QPSK received signal PSD. Additive White Gaussian Noise (AWGN) was added to the transmit signal to simulate real-life interference.

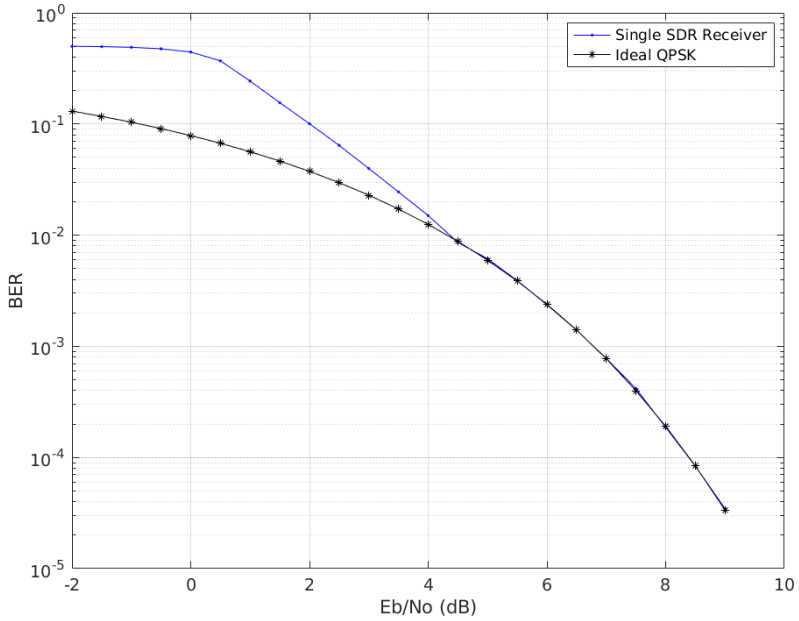

Figure 37. Shows the BER vs. E_b/N_0 of the received signal. The low SNR at the beginning of the signal shows the simulation is trying to locate the frequency offset.

4.2.2 Dual Receiver Simulations

Frequency offset and phase error values are introduced into the dual receiver simulation to replicate the expected local oscillator (LO) drift in the hardware tests. It was not necessary to include these variations in the single receiver, single-channel simulations. Power Spectral Density (PSD) plots of the QPSK dual receiver simulation show typical responses spanning the defined bandwidth. A PSD plot for the QPSK signal bandwidths was produced. Figure 38 shows the simulated 10 MHz QPSK transmit signal bandwidth for a simulated dual receiver collection. Figures 39 and 40 show the simulated 5 MHz QPSK received signal bandwidths for each receiver. Figure 41 shows

the simulated 10 MHz QPSK received signal bandwidth. Figure 42 shows the resultant simulated 10 MHz QPSK received signal bandwidth summation.

Figure 43 shows the Monte Carlo simulation BER versus E_b/N_0 plot for the dual receiver QPSK simulation. This simulation was again conducted using 100 runs. The simulated received QPSK signal BER is compared with the theoretical BER for a QPSK signal. A BER of 4×10^{-5} at an E_b/N_0 of 12 dB is indicated for the simulated dual SDR collection. There is a 3 dB loss between the simulated dual SDR collection and the theoretical QPSK signal BER.

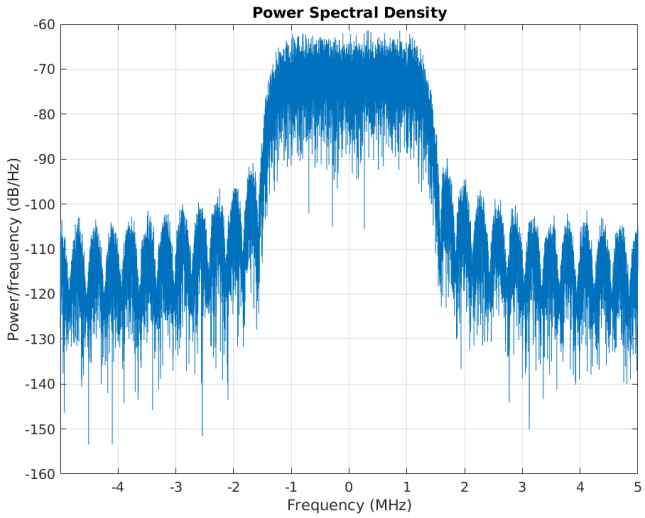

Figure 38. Simulated 10 MHz QPSK transmit signal PSD.

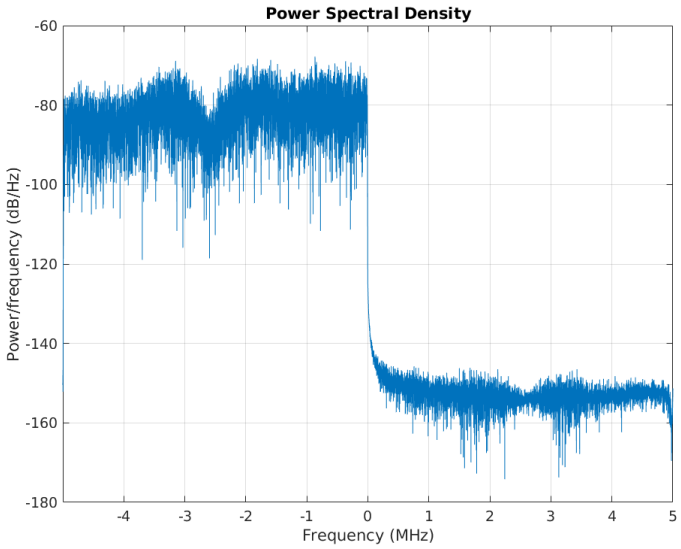

Figure 39. Simulated 5 MHz QPSK signal showing the lower frequency receiver ('RxLow') SDR collection.

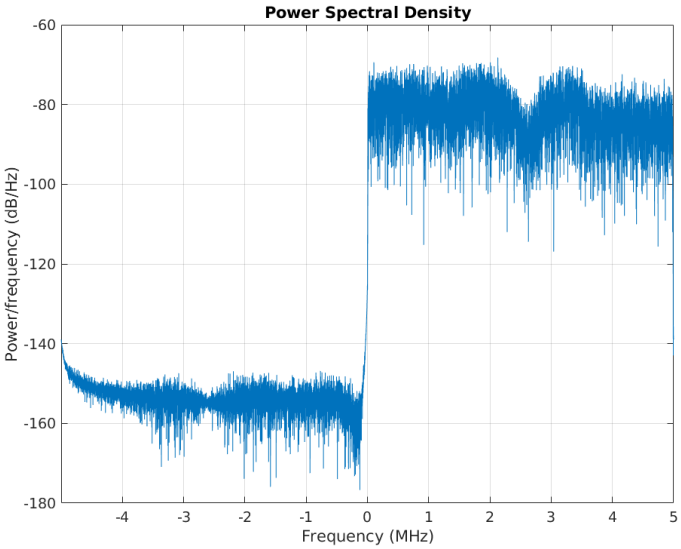

Figure 40. Simulated 5 MHz QPSK signal showing the lower frequency receiver ('RxHigh') SDR collection.

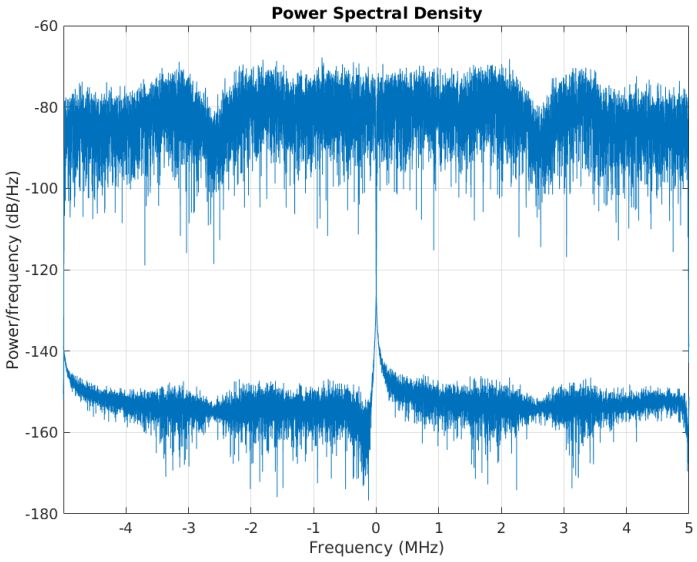

Figure 41. Simulated 10 MHz QPSK overlay signal PSD showing the two 5 MHz SDR collections. Each simulated SDR 5 MHz sub-band has undergone frequency and phase correction before being summed to recover the 10 MHz signal.

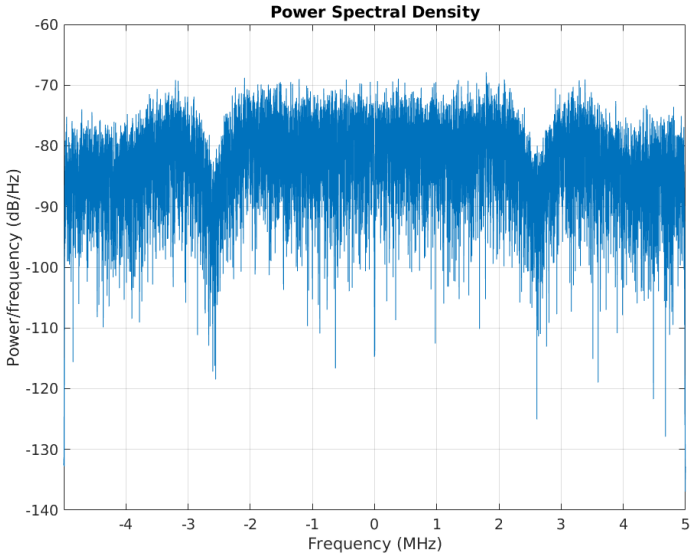

Figure 42. Simulated 10 MHz QPSK recombined signal PSD for the dual SDR collection. Each simulated SDR collected a 5 MHz sub-band which undergoes frequency and phase correction before the sub-bands are summed to recover the 10 MHz signal.

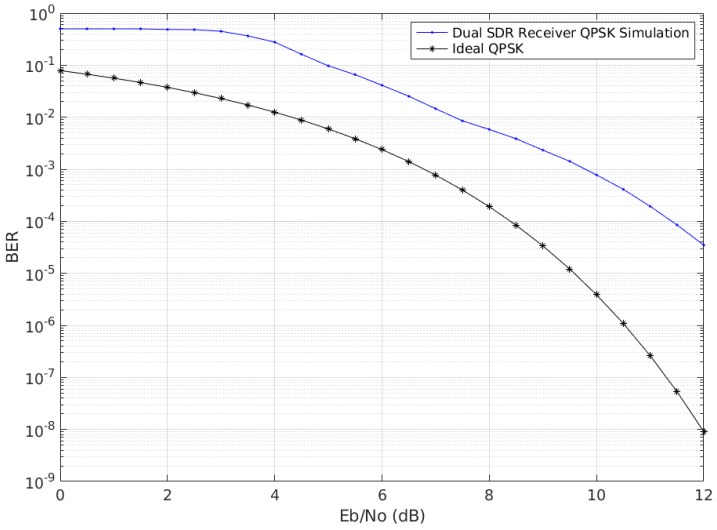

Figure 43. Simulation test of 10 MHz QPSK BER versus E_b/N_0 for 0 to 12 dB, for a dual SDR collection. The receivers do not fully synchronize until approximately 6 dB SNR. There is approximately a 3 dB penalty.

4.3 Hardware Test Results

This section shows the results for the single and dual receiver hardware tests. Section 4.3.1 shows the results for the single receiver hardware tests. Section 4.3.2 shows the results for the dual receiver hardware tests.

4.3.1 Single Receiver Single Channel Hardware Test Results

Section 4.3.1 provides results for the single-receiver, single-channel hardware test. These results confirm the previous expectation that bit loss is due to occur when the transmitted signal bandwidth exceeds the instantaneous bandwidth of the receiver. Also,

the single-receiver, single-channel results show the frequency change assumption for signals of shorter duration needs to be reassessed.

4.3.1.1 Single Receiver Single Channel Hardware Test Results

The aim of the single-receiver, single-channel test is to demonstrate the bandwidth limitations of a single SDR receiver that is required to collect a wide-band signal. A single SDR is unable to collect a wide-band signal with a bandwidth that exceeds the instantaneous bandwidth of the SDR. Figure 44 shows the result for the single receiver, single-channel hardware test QPSK transmit signal bandwidth. There is a 1 dB loss between the theoretical QPSK signal and hardware test single SDR receiver QPSK signal.

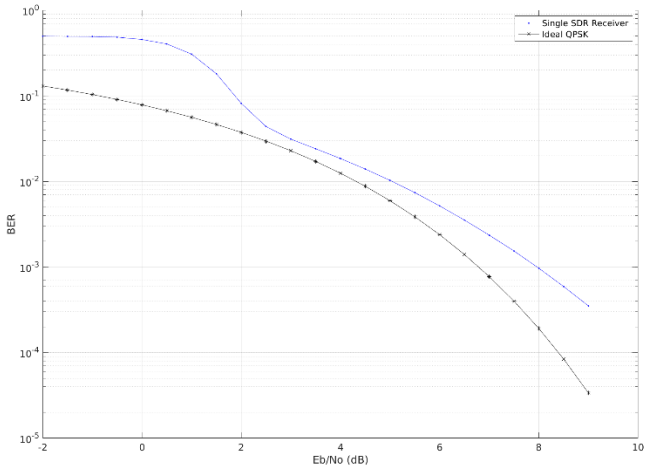

Figure 44. Shows the BER vs. E_b/N_0 of the hardware test 10 MHz QPSK single SDR receiver.

4.3.2 Dual Receiver Hardware Test Results

Since this test utilizes two separate receivers, there are frequency and phase differences due to the local oscillator (LO) drift. Power Spectral Density (PSD) plots were generated for the transmit and receive QPSK signal bandwidths. The transmit signal bandwidth is the same from the previous experiments. Figure 45 shows the hardware test 10 MHz QPSK BER versus E_b/N_0 for a dual SDR collection. An BER of $8x10^{-4}$ at an E_b/N_0 of 11 dB is shown. There is a 4 dB loss between the dual receiver SDR collection BER and the theoretical QPSK signal BER. There is a 1 dB loss between the dual receiver SDR hardware test and the dual receiver SDR simulation.

The hardware test of the 10 MHz QPSK signal dual SDR collection is the first successful demonstration of the auto-correlation bandwidth technique for SDRs transmitting and receiving random bits. In previous research efforts, the outcome of the received signal was already pre-determined.

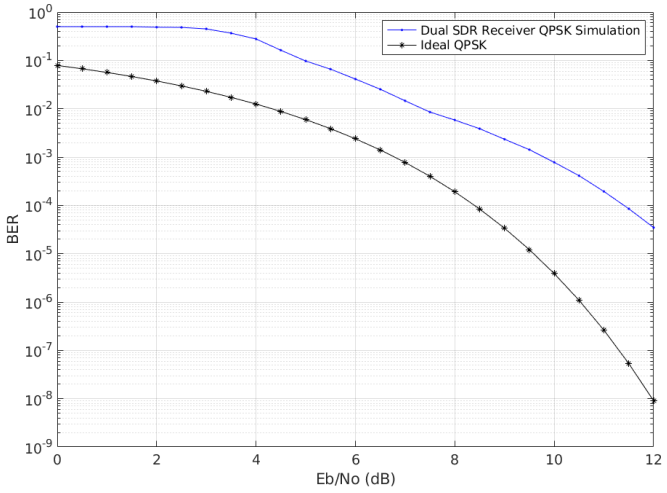

Figure 45. Hardware test of 10 MHz QPSK BER versus E_b/N_0 for 0 to 12 dB, for a dual SDR collection. As with the simulation, the receivers do not fully synchronize until approximately 4 dB SNR. There is approximately a 4 dB loss between the dual SDR collection and the theoretical QPSK signal.

4.4 Summary

Chapter IV discussed the result and analysis of the research effort. The results show that a QPSK transmit signal composed of random bits, can be collected simultaneously and then reconstructed using two SDRs to generate a higher bandwidth signal. The autocorrelation technique from Chapter III was used for signal reconstruction. No overlap of the collected sub-bands was required for this autocorrelation technique. The PSD plots for the simulation tests demonstrate this technique achieves the desired results. Bit recovery for the simulation and hardware tests were also accurate at the E_b/N_0 value prescribed for both SDRs. Chapter V will provide research effort conclusions and future work recommendations.

V. Conclusion

The aim of this research effort was to demonstrate a wide-band signal transmission and collection simultaneously spanning multiple USRP B205 mini–Software Defined Radios (SDRs) using a QPSK signal. The resultant received signal was unknown, which distinguished this effort from previous research on this manner. The collections were then meshed to produce a wide band received signal. The Bit Error Rate (BER) of the received signal was then compared to the theoretical QPSK BER versus E_b/N_0 to validate accurate symbol recovery.

The initial attempts to generate an effective code in MATLAB for the simulations and hardware test were met with some difficulty. This limited the amount of testing that was performed. The initial aim was to see if more than two USRP B205 minis could simultaneously receive and recombine a transmitted signal, but time constraints limited the research to two SDRs. Despite these limitations, there were some significant outcomes that came from this research that can possibly be used towards future research in this area. These outcomes were:

- The successful simulation and hardware testing of an autocorrelation bandwidth expansion using two USRP B205 mini SDRs to collect a transmitted signal and generate a high bandwidth signal.
- The successful hardware testing of SDRs to collect, restore, and expand a signal without the use of a Universal Software Radio Peripheral (USRP) X310 device. These devices are known to limit bandwidth expansion. It is recommended that

future work increases the bandwidth expansion to 20 MHz or 100 MHz without USRP device to validate if this technique is usable on the USRP B205 minis at high bandwidths.

5.1 Future Work

Due to the difficulties faced with developing a working simulation and autocorrelation technique, only two SDRs receivers could be tested. Also, to ensure viable results would be attained, the bandwidth for the QPSK was constrained at 10 MHz just to verify the USRP B205 minis could generate a high bandwidth signal.

5.1.1 Multiple SDR Receivers

For future research, it is suggested that more than two USRP B205 minis are used for this bandwidth expansion technique with a higher transmit signal bandwidth. Previous research was able to verify this bandwidth expansion technique with other SDRs. Testing up to four USRP B205 minis simultaneously would be a suitable continuation of this research.

5.1.2 BladeRF SDRs

Initially, the BladeRF software defined radio was the intended device under test (DUT). The BladeRF 2.0 micro is a next-generation SDR with a frequency range of 47 MHz to 6 GHz, 61.44 MHz sampling rate, and 2x2 MIMO (Multiple Input Multiple

Output) streaming. Through the libbladeRF software, the BladeRF 2.0 micro is compatible with several software programs to include: GNUradio, GQRX, SDR-Radio, and SDR# which can all be operated with Windows, Linux and macOS [42].

A radio frequency (RF) shield cap protects sensitive RF parts from Electromagnetic Interference (EMI) and provides additional thermal dissipation, allowing the radio to operate in challenging environments like space. The RF SubMiniature (SMA) ports can provide power over bias-tee circuitry to wideband amplifiers and pre-amps. The radio also has an advanced clocking architecture that enables it to collect and provide a 38.4 MHz fundamental clock from and to other devices [42].

For the next phase of this research venture, it should be determined whether the auto-correlation bandwidth expansion technique developed in this effort can be used on an SDR like the BladeRF. The BladeRF is an SDR that is ideal for satellite operations due to its hardware features.

5.1.3 RF-DNA Tests

Radio frequency distinct native attribute (RF-DNA) fingerprint processing is a method that extracts features received from a RF Signal. Every electronic device emits signals with unique characteristics, or features, that can distinguish that device from other similar devices. This notion is like human fingerprinting, where a person's fingerprints can distinguish them from another person. Radio frequency distinct native attribute (RF-DNA) has been in development at AFIT since 2006 and has demonstrated passive RF emission feature extraction from many different devices [43]. An ideal test would be to

demonstrate whether the BladeRF SDRs can be distinguished using RF-DNA while transmitting under the autocorrelation bandwidth expansion technique.

5.1.4 CubeSat Application

To tie-in the SATCOM portion of this research effort, the SDRs should be tested in conjunction with the CubeSat. A CubeSat is miniaturized satellite used for space research. CubeSats are generally no bigger than 1.33 kg or 3 lbs. Demonstrating the viability of SDRs with a smaller satellite would be a major step in validating the usefulness of SDRs in space applications.

Bibliography

[1] M. R. Maheshwarappa and C. Bridges, "Software Defined Radios for Small Satellites," in *NASA/ESA Conference on Adaptive Hardware and Systems (AHS)*, 2014.

[2] M. Dillinger, K. Madani and N. Alonistioti, Software Defined Radio: Architectures, Systems, and Functions, Wiley and Sons, 2003.

[3] L. M. Reyneri, "Software Defined Transceivers Design in Nano and Picosatellites," in *International Astronautical Conference*, Torino, Italy, 2012.

[4] B. Stewart, K. Barlee, D. Atkinson and L. Crockett, Software Defined Radios Using MATLAB and Simulink and RTL-SDR, Glasgow, United Kingdom: Strathclyde Academic Media Ltd., 2015.

[5] N. O'Brien, "A Non-Destructive Evaluation Application Using Software Defined Radios and Bandwidth Expansion," Air Force Institute of Technology, Wright-Patterson Air Force Base, 2020.

[6] D. Bosworth and T. Wyatt, "Bandwidth Demands Place New Strains on Satellite Communications," Analog Devices, Norwood, MA, 2018.

[7] R. Ludwig and P. Bretchko, RF Circuit Design, Theory and Applications, New Jersey: Prentice Hall, 2000.

[8] "IEEE Standard Letter Designations for Radar Frequency Bands," IEEE, 8 January 2003. [Online]. Available: ieeexplore.iee.org. [Accessed August 2020].

[9] S. G. a. W. G. 4, "IADC Space Debris Mitigation Guidelines," in *Inter-Agency Space Debris Coordination Committee*, 2007.

[10] A. Campbell, "SCaN Glossary," 10 July 2015. [Online]. Available: https://www.nasa.gov/directorates/heo/scan/definitions/glossary/index.html. [Accessed August 2020].

[11] N. R. Council, Autonomous Vehicles in Support of Naval Operations, Washington D.C.: The National Academies Press, 2005.

[12] G. Kaur and V. Raj, "Multirate Digital Signal Processing for Software Defined Radio(SDR) Technology," in *First International Conference on Emerging Trends in Engineering and Technology*, Mumbai, 2008.

[13] A. Haghighat, "A Review on Essential and Technical Challenges of Software Defined Radio," *IEEE Journals*, 2002.

[14] G. B. Giannakis, "Highlights of Signal Processing for Communications," *IEEE Signal Processing Magazine,* 1999.

[15] N. Hammada, "Digital Signal Processing Progress Over the Last Decade and the Challenges Ahead," *IEICE Transcation Fundamental,* Vols. 84-A, no. 1, 2001.

[16] D. M. Pearson, "SDR (System Defined Radio): How Do We Get There From Here," *IEEE Journals,* 2001.

[17] P. Angeletti, M. Lisi and P. Tognolatti, "Software Defined Radio: A Key Technology for Flexibility and Reconfigurability in Space Applications," pp. 399-403, 2014.

[18] B. Pallassa and C. Morlet, "Flexible Satellites: Software Radio in the Sky," *IEEE*, 2003.

[19] G. Roeper, R. Goldsmith, C. Hatziathanasiou and P. Mclaren, "Miniature TT&C Module for Small Satellites in Low Earth Orbits," in *European Space Agency*, Noordwijk, Netherlands, 2010.

[20] C. Fernandez-Prades, J. Arribas and P. Closas, "GNSS-SDR: An Open Source Tool for Researchers and Developers," in *ION GNSS Conference*, Portland, Oregon, 2011.

[21] A. Alilla, D. Carlofelice, M. Faccio, I. Lucresi and P. Tognolatti, "Software-Defined Satellite Ranging Measurements Using Laboratory Signal Analyzer," in *IEEE International Workshop on Metrology for Aerospace*, Benevento, Italy, 2014.

[22] R. K. Miranda, "Implementation of Improved Software Defined Radio Modulation Scheme and Command and Telemetry Software Interface for Small Satellites in 5G Systems," in *19th International Conference on OFDM and Frequency Domain Techniques*, Essen, Germany, 2016.

[23] CCSDS, "Bandwidth-Efficient Modulations, Summary of Definition, Implementation, and Performance Information," Green Book, 2006.

[24] L. C. Tillotson, "Efficient Use of the Radio Spectrum and Bandwidth Expansion," *IEEE*, vol. 61, no. 4, 1973.

[25] B. Sklar, Digital Communications: Fundamentals and Applications, Pearson Education, 2009.

[26] T. Nelson, E. Perrins and M. Rice, "Common Detectors for Shaped Offset QPSK (SOQPSK) and Feher-patented QPSK (FQPSK)," in *IEEE Global Telecommunications Conference*, 2005.

[27] H. Stern and S. Mahmoud, Communication System, Pearson Prentice Hall, 2004.

[28] S. C. Scripps, RF Power Amplifiers Wireless Communications, Second Edition, Artech House, 2006.

[29] N. Everett and P. Collins, "Instantaneous Bandwidth Expansion Using Software Defined Radios," in *2019 International Radar Conference*, Wright-Patterson AFB, OH, 2019.

[30] "ettus.com," Ettus Research, [Online]. Available: ettus.com/wp-content/uploads/2019/01/USRP_B200mini_Data_Sheet.pdf. [Accessed 25 January 2021].

[31] "Attenuators, Fixed," [Online]. Available: www.keysight.com. [Accessed 14 September 2020].

[32] J. A. Gubner, Probability and Random Process for Electrical and Computer Engineers, Cambridge University Press, 2006.

[33] B. P. Lathi and Z. Ding, Modern Digital and Analog Communication Systems Fourth Edition, New York, NY: Oxford University Press, 2009.

[34] D. Sorensen, "Exponent: High Speed CubeSat Radio," Utah State University, Logan, UT, 2018.

[35] T. F. Collins, R. Getz, D. Pu and A. M. Wyglinski, Software Defined Radios for Engineers, Artech House, 2018.

[36] G. L. Stuber, "Broadband MIMO-OFDM Wireless Communications," *IEEE*, vol. 92, pp. 271-293, 2004.

[37] M. Morelli and U. Mengali, "Feedforward Frequency Estimation for PSK: A Tutorial Review," *European Transactions on Telecommunications*, vol. 9, no. 2, pp. 103-116, 1998.

[38] Y. Wang, K. Shi and E. Serpedin, "Non-Data-Aided Feedforward Carrier Frequency Offset Estimators for QAM Constellations: A Nonlinear Least Squares Approach," *EURASIP Journal on Advances in Signal Processing*, 2004.

[39] T. F. Schuppe, "Modeling and Simulation: A Department of Defense Critical Technology," in *Winter Simulation Conference*, Wright-Patterson Air Force Base, OH, 1991.

[40] D. P. Kroese, T. Brereton, T. Taimre and Z. Botev, "Why the Monte Carlo Method is So Important," *WIREs Computer Statistics*, vol. 6, no. 6, pp. 386-392, 2014.

[41] J. A. Betances, "Physical Layer Defenses Against Primary User Emulation Attacks," Air Force Institute of Technology, Wright-Patterson AFB, OH, 2016.

[42] "NUAND," [Online]. Available: http://www.nuand.com. [Accessed 14 September 2020].

[43] M. W. Lukacs, A. J. Zeqolari, P. J. Collins and M. A. Temple, ""RF-DNA" Fingerprinting for Antenna Classification," *IEEE Antennas and Wireless Propagation Letters,* vol. 14, pp. 1455-1458, 2015.

[44] D. Barnhart, T. Vladimirova and M. N. Sweeting, "Very Small Satellite Design for Distributed Space Missions," *Journal of SpaceCraft and Rockets,* pp. 1294-1306, 2007.

[45] "Minotaur I Launch," October 2013. [Online]. Available: http://www.cubesat.org/index.php/missions/upcoming-launches/135-ors3-launch-alert. [Accessed October 2013].

[46] "Dnepr-19 Launch," 2013. [Online]. Available: http://space.skyrocket.de/doc_lau_det/dnepr-1.htm. [Accessed October 2013].

[47] B. Klofas, "Upcoming CubeSat Launches: The Flood has Arrived," in *AMSAT-NA Symoposium*, Houston, TX, 2013.

[48] M. Ribero, "Software Defined Radio over CUDA," UNIVERSITAT POLITECNICA DE CATALUNYA, Barcelona, Spain, 2015.

[49] R. G. Lyons, Understanding Digital Signal Processing, Second Edition, Upper Saddle River, NJ: 2004, 2004.

[50] "its.bldroc.gov," [Online]. Available: https://www.its.bldrdoc.gov/fs-1037/dir-029/_4230.htm#:~:text=pull%2Din%20frequency%20range%3A%20The,local%20oscillator%20can%20be%20locked.. [Accessed 10 February 2021].

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 074-0188

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From – To)*
25-03-2021	Master's Thesis	Sept 2019 – March 2021

TITLE AND SUBTITLE

Utilizing Software Defined Radios for Increased Bandwidth in Satellite Communications

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Brian M. Wright, Captain, USAF

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way, Building 640
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT-ENG-MS-21-M-096

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Intentionally Left Blank

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

DISTRUBTION STATEMENT A. APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This research explores the background and future possibilities of using cooperative satellites equipped with Software Defined Radios (SDRs) to combine their bandwidth and increase their signal reliability. Software Defined Radios are a potential solution to realize various software applications that support a reconfigurable and adaptive communication system without altering any hardware devices or features. This benefit, along with others that are offered by SDRs and the ongoing improvements in commercial digital electronics have sparked an interest in developing small satellites for advanced communications. This research effort sets out to prove if a high bandwidth signal can be generated by using low-cost SDRs. The SDR receivers will each receive different sub-bands of the transmitted signal with the goal of the meshing the received signals to form a high bandwidth signal.

15. SUBJECT TERMS

Software Defined Radio, Satellite Communications, Bandwidth Expansion

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			Kenneth M. Hopkinson, AFIT/ENG
U	**U**	**U**	**UU**	104	**19b. TELEPHONE NUMBER** *(include area code)* (937) 255-3636, x4579 Kenneth.hopkinson@afit.edu

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. Z39-18