FPGA-BASED GNSS RECEIVER DESIGN FOR REFLECTOMETRY APPLICATIONS

SURABHI GURUPRASAD

A DISSERTATION SUBMITTED TO THE FACULTY OF GRADUATE STUDIES
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

GRADUATE PROGRAMME IN EARTH AND SPACE SCIENCE
YORK UNIVERSITY
TORONTO, ONTARIO

OCTOBER 2022

ABSTRACT

Research has shown that Global Navigation Satellite System (GNSS) signals reflecting off the Earth's surface can be detected by receivers in low Earth orbit (LEO). The weak reflected signal properties are analyzed to characterize geophysical properties such as soil moisture, sea surface height, ocean surface wind speed and sea ice detection. This method of remote sensing is known as GNSS reflectometry (GNSS-R).

Commercial GNSS receivers have historically been designed to only detect and process direct GNSS signals and cannot be repurposed to collect relevant data for reflectometry. Data collected by some orbital receivers have been made public; however, due to the high volume of data, data are truncated and are insufficient for science applications. Therefore, to be able to develop and test new algorithms, a custom GNSS-R receiver is designed and implemented.

The developed GNSS-R receiver is implemented using field programmable gate array (FPGA) technology. The GNSS-R receiver prototype uses 1-bit signal resolution resulting in a compact design requiring minimal FPGA resources. Several field results show that the receiver prototype can successfully track direct and reflected GNSS signals in real-time. The observations indicate that the carrier-to-noise density ratio of signals reflecting from the surface of water was on average approximately 7 dB higher than the C/N_0 recorded when tracking land reflections. The difference in C/N_0 between water and

land reflections is significant enough to conclude that a GNSS-R receiver using a one-bit quantization GNSS signal can be used for reflectometry applications.

To increase the sensitivity of the receiver to weak reflected signals (-140 dBm), the FPGA-based signal processing module is enhanced using the alternate half-bit method. The receiver sensitivity improved from -35 dB to -46 dB (signal-to-noise ratio).

Another challenge for GNSS-R receivers is generation of high-resolution delay Doppler maps (DDMs). DDMs provide insight into the reflecting surface characteristics based on signal scattering. The developed GNSS-R receiver demonstrates that high resolution DDMs can indeed be generated in real-time using 1-bit GNSS signal resolution. Data collected by the CYGNSS spacecraft are used to test and validate the receiver implementation. Results show that distinct DDMs are generated for land and water reflections and correlates (in terms of SNR) with research conducted using 2-bit signal resolution.

.

DEDICATIONS

Dedicated to Siddharth, Amma, Appa and Sampath for being the constant source of inspiration and motivation.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude towards my supervisor Dr. Sunil Bisnath, for providing me the opportunity to be a member of his research team and learn under his highly appreciated guidance. I would also like to thank him for the continuous support, motivation and expertise he provided throughout this dissertation. I also extend my sincere thanks to Professor Regina Lee, for providing valuable suggestions and insights during the study for this research.

I also had the opportunity to work and collaborate with some brilliant lab members. I am grateful to have met and forged some lifelong friendships with Sudha Vana, Garrett Seepersad, John Aggrey, Maninder Gill and Junchan Lee.

The GNSS lab is also where I met my husband, Siddharth Dave, without whom the writing of this dissertation would not have been possible. Thank you for the constant love, support and encouragement.

Last but not the least, I am immensely grateful towards my *amma*, *appa* and brother for the endless inspiration, support and encouragement they have provided me all these years. This work has been possible only because of them.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS

ADC	Analog-to-Digital Converter
ASIC	Application Specific Integrated Circuits
C/A	Coarse Acquisition
COTS	Commercial off-the-shelf
CWF	Complex Waveform Product
CYGNSS	Cyclone Global Navigation Satellite System
DDM	Delay Doppler Map
DMA	Digital Memory Access
EMI	Electromagnetic Interference
FMC	FPGA Mezzanine Card
FPGA	Field Programmable Gate Array
GNSS	Global Navigation Satellite System
GNSS-R	GNSS Receiver for Reflectometry
GPS	Global Positioning System
HDL	Hardware Description Language
HLS	High Level Synthesis
IF	Intermediate Frequency
IP	Intellectual Property
LEO	Low Earth Orbit

LHCP	Left Hand Circular Polarisation
NAVIC	Navigation with Indian Constellation
NBRCS	Normalised Bistatic Radar Cross Section
NF	Noise Figure
PLL	Phase Lock Loop
PRN	Psuedo-random Noise
PVT	Position Velocity and Timing Information
QZSS	Quasi-Zenith Satellite System
RAM	Random Access Memory
RF	Radio Frequency
RHCP	Right Hand Circular Polarisation
SDR	Software-defined Radio
SDR	Software-defined Radio
SEL	Single Event Latch
SEU	Single Event Upset
SNR	Signal-to-noise Ratio
SoC	System-on-Chip
TDS	Tech Demo Sat
TMR	Triple Modular Redundancy
TTFA	Time to First Acquire
uP	Microprocessor

Chapter 1 INTRODUCTION

Global Navigation Satellite Systems (GNSS) have established themselves as an essential component for a myriad of terrestrial applications, such as surveying, remote sensing, object or animal tracking and aviation (Egea-Roca et al. 2022). To process distinct signals transmitted from various GNSS satellites, dedicated receivers have been built and today, the receivers are advanced and economical enough to fit into miniature chips on cellphones and other tracking devices. In this chapter, a general overview of the concepts related to the dissertation is provided.

1.1 Present state of Global Navigation Satellite Systems

GNSS encompasses various constellations: Global Positioning Systems (GPS) developed by the US, GLONASS by Russia, the European Galileo system and BeiDou developed by China. There are also regional constellations: the Indian NAVIC and Japan's Quasi-Zenith Satellite System (QZSS). The GNSS constellations transmit signals at various frequencies, ranging from 1 to 2 GHz. Navigation signals also differ from each other in terms of access methods, due to which constellations have overlapping transmitting frequencies.

New signals and constellations are continually introduced to meet evolving industry demands. The signal of interest in this dissertation is the GPS L1 Coarse/Acquistion-code (C/A-code). The C/A-code is a sequence of 1023 elements or

chips that are transmitted by each GPS satellite. The sequence, also known as pseudo-random noise (PRN), repeats itself every millisecond and hence every chip length corresponds to 1 microsecond or 300 metres (Borre et al. 2007).

Each satellite encodes a unique PRN code sequence. The correlation properties of the PRN code are such that two identical sequences will be highly correlated if they have no phase difference between them. In all other cases, the correlation value is extremely low. Also, two PRN sequences from different satellites have very low correlation for all the phases. Understanding the correlation properties helps identify the unique satellite embedded in the GPS signals.

1.2 Reconfigurable GNSS receivers

The main function of a GNSS receiver is to collect and process all available GNSS signals to compute position, velocity and timing information. A GNSS receiver can comprise several hardware and software processing components and due to various existing processing technologies, a wide spectrum of GNSS receivers exist.

GNSS receivers can be broadly categorised into hardware receivers and software receivers. Traditional receivers use hardware platforms known as Application Specific Integrated Circuits (ASIC). ASIC-based receivers are known to have a high processing speed, low power consumption and are compact in size. However, with advancing software processors, ASIC-based operations can now be handled by microprocessors and microcontrollers. The primary reason for migration from hardware to software is straightforward: flexibility. Software receivers offer developers and users immense

flexibility in terms of design or algorithm alteration. Since software receivers no longer require re-fabrication of circuit boards, significant reduction in development costs and time is observed.

However, a major disadvantage of using a software-only receiver is the limited processing speed when compared to an ASIC. GNSS signal processing is computationally heavy and software receivers are often much slower than hardware receivers due to the lack of parallel processing abilities. For example, research presented by Guruprasad (2015) shows that when certain software components were replaced with hardware equivalent, the processing speed improved by 20%. The mounting advancement of Field Programmable Gate Array (FPGA) technology aids software receivers' viability.

FPGA is a type of reconfigurable hardware that has the capability and flexibility to be easily programmed without the need for re-fabrication of hardware components. Typically, numerous correlators can be programmed in an FPGA and they function and perform as a hardware entity. Hence, FPGAs can be used to implement various parallel tasks. Use of FPGAs in conjunction with fast software processors helps establish modern reconfigurable GNSS receivers (Fridman and Semenov 2013).

1.3 GNSS receiver for space-based reflectometry

Reflectometry is a type of remote sensing using Earth-reflected GNSS signals. As seen in Figure 1-1, the GNSS receiver onboard a Low Earth Orbit (LEO) satellite receives both direct and reflected GNSS signals. The reflected signals scatter depending on the type of reflective surface and by studying the properties of the reflected signal, this technology

has the potential to provide scientists environmental data of a wide coverage area at a relatively low-cost, as the GNSS signals are global and freely available.

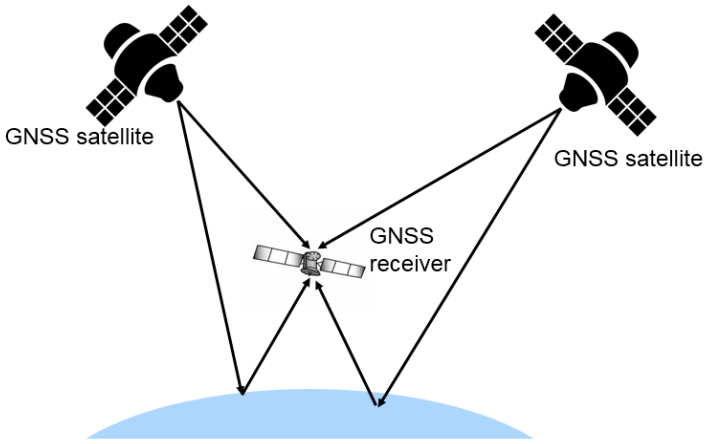


Figure 1-1: Direct and reflected GNSS signals arriving at a LEO receiver

Some instances of geophysical parameters that can be estimated using reflectometry are ocean surface roughness, wind speed, sea ice detection, sea ice altimetry, hurricane monitoring, vegetation and surface soil moisture (Nogués-Correig et al. 2007; Clarizia et al. 2014; Foti et al. 2015; Chew and Small 2018; Zribi et al. 2019; Rodriguez-Alvarez et al. 2019). To conduct research in the field of space-based reflectometry, certain receiver specifications must be met that are not commercially available. Due to the lack of a commercial-off-the-shelf (COTS) receiver that supports reflectometry, this dissertation presents a low-cost reconfigurable receiver design that can be used for remote sensing purposes.

A high-level block diagram of a GNSS-R receiver is presented in Figure 1-2. Direct GNSS signals are known to be right hand circularly polarised (RHCP) and upon

reflection, the signal changes to left hand circularly polarised (LHCP). Therefore, two antennas are used for reflectometry, the zenith RHCP antenna captures direct signals and the nadir LHCP antenna senses weak reflected GNSS signals. The radio frequency (RF) front end digitizes the signals and the GNSS-R receiver further processes received signals.

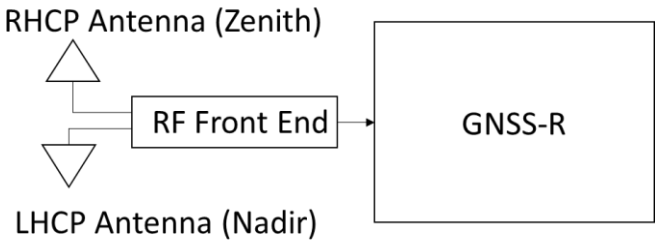


Figure 1-2: GNSS-R block diagram

The reflected GNSS signals are extremely weak and can be in the range of -140 dBm or lower. In comparison, direct GNSS signals have a nominal signal strength of approximately -128 dBm (the unit dBm represents the power of the signal in milliwatts). Hence, a GNSS-R receiver must be designed to be highly sensitive to weak signals and capable of tracking these signals efficiently. Reflected GNSS signals differ from direct signals in terms of signal power, time delay, Doppler frequency shift and polarisation (Gamba et al. 2015). Figure 1-3 demonstrates the arrival and detection of direct GNSS signals, as well as that of reflected signals. High correlation power indicates a strong signal whereas low correlation power indicates a weak signal. Reflection of the signal and its scattering (correlation peak shape) is largely dependent on the type of surface. A

GNSS-R receiver in LEO can sense the scattered signals and classifies the reflected signals based on two main parameters: frequency and chip delay. These parameters help determine the specular point (the surface point of reflection) and characteristics of the surface estimated.

In Figure 1-4, delay Doppler maps (DDMs) are presented. The horizontal axis represents the Doppler frequency of the received signal (reflected) with respect to the specular point and the vertical axis represents the chip delay relative to the same specular point. DDMs are explained further in Section 2.2. These DDMs were shared and published by researchers at Surrey Satellite Technology Limited (SSTL), where they have access to data from a technology demonstration mission: TechDemoSat-1 (King et al. 2021). The DDMs indicate that land reflections and ocean reflections can be distinguished from a LEO GNSS-R receiver. Analysis of DDMs and its correlation with the geophysical parameters (such as soil moisture, ocean surface wind speed etc.) is a scientific field that has recently garnered significant amount of interest.

Even though DDMs are computationally demanding, it is preferred that the raw IF data from the RF front end is not recorded and is discarded after DDMs are computed in orbit. DDMs are logged and downlinked to ground stations as the volume of data is compressed and more manageable when compared to transmission of raw IF data.

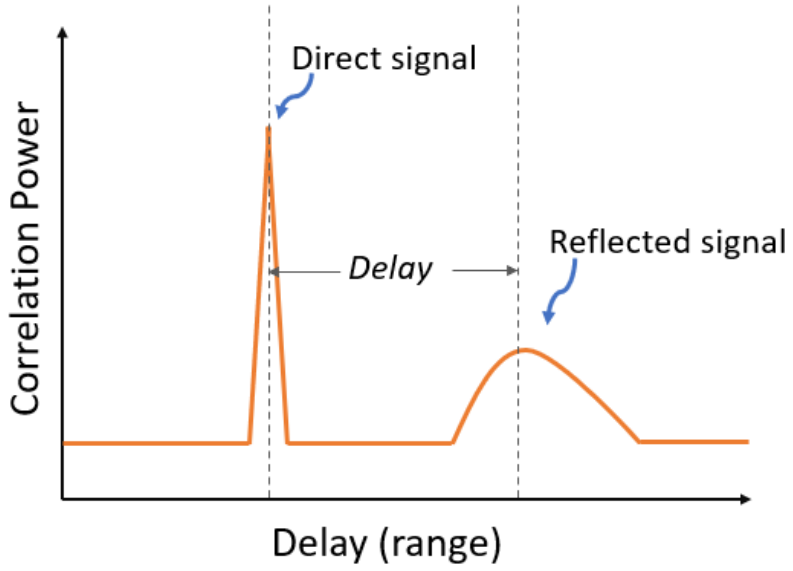


Figure 1-3 Representation of the delay between direct signal and reflected signal (Masters 2004)

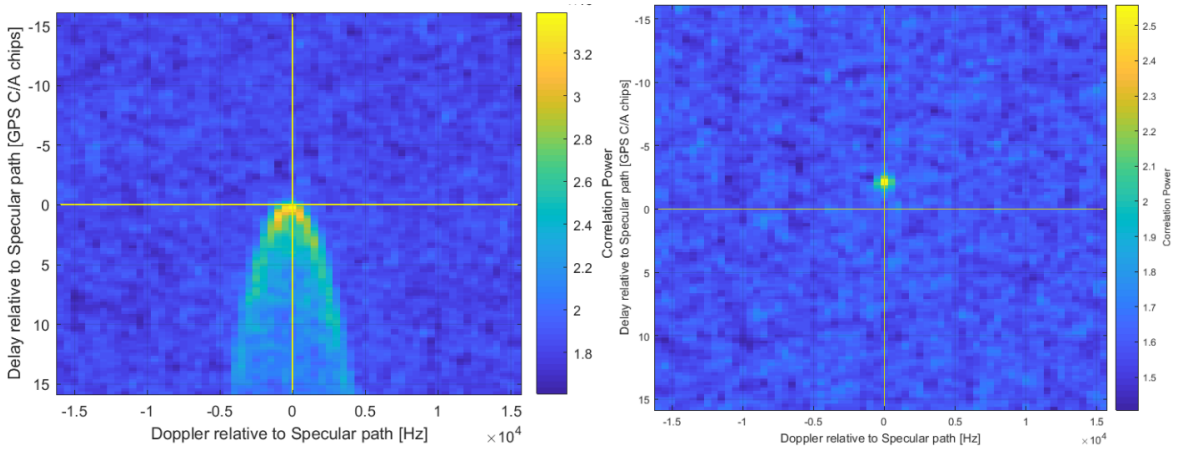


Figure 1-4 Delay Doppler Maps (DDM) (King et al. 2021): (a) A standard DDM formed due to ocean reflection. (b) Characteristic DDM feature observed from land reflections.

1.4 Motivation for the development of a custom FPGA-based GNSS-R receiver

GNSS receivers do not inherently have the capability to perform computations for reflectometry. To use an existing, commercially available GNSS receiver for reflectometry is nearly impossible for the following reasons:

1. GNSS receivers are designed to process direct signals only. Any signal component that is not RHCP is considered as multipath signals and hence attempted to be mitigated. In contrast, reflected (LHCP) signals are essential for reflectometry.
2. COTS GNSS receivers provide position, velocity and timing information (PVT). Most receivers are also capable of outputting various range measurements and satellite ephemeris. However, for reflectometry, raw data such as Doppler frequency of the signal and the phase of the code are necessities. Such low-level information can be provided only by custom-built GNSS receivers and hence it is imperative to have a custom receiver for reflectometry applications.

Additionally, during this research it was discovered that there is a lack of open-source real-time FPGA-based GNSS receivers that can be assembled and built using COTS hardware (FPGA platform, RF front end, antennas) resulting in a low-cost system. For an instrument such as a GNSS receiver, especially in academic research, there is a need for an iterative design approach, where modularity and reconfigurability is the key receiver feature. Novel research contributions can be developed and tested using reconfigurable hardware ensuing low development time. Furthermore, performance of GNSS-R

receivers using 1-bit signal resolution is unknown. The significance and impact of 1-bit quantization is addressed in detail in Section 2.4.5.

Commendable contributions have been made by several research groups with reference to reflectometry from space, including SSTL's TDS-1 and NASA's CYGNSS. Data gathered from these missions are available to the public; however, the data are limited and defined by the instrument specifications. As an example, the majority of CYGNSS datasets offer truncated DDMs (low resolution) since raw data requires increased storage and the data downlink bandwidth is limited. However, for many science applications, higher resolution DDMs are necessary (Musko et al. 2020). Although CYGNSS does occasionally publish raw data and high resolution DDMs, data available are for short durations and for limited scenarios only. Therefore, researchers must largely continue to use low resolution DDMs due to the lack of other sources.

In addition, there are several hardware components that are custom built such as the nadir antenna that existing GNSS-R receivers use. There has been a lack of demonstration of detection of weak reflected signals using a COTS antenna. This dissertation investigates the use of a COTS antenna for detection of a reflected signal. If a COTS antenna is not adequate for the weak reflected signals, can any signal processing technique enhance the signal power enough to be detected by the receiver? Can the receiver's sensitivity be increased with novel processing implementation? The ability to use a COTS antenna for reflectometry would be considered noteworthy given that not all research groups have access to experts in the field of antennas. Furthermore, custom

antennas may cost more in the context of limited research funds and will also lead to increase in development time.

Furthermore, generic GNSS receivers cannot generate DDMs. Even though DDM generation is computationally intensive and is known to have a large volumetric amount of data, they are still significantly smaller in file size compared to raw IF data generated at the front end. Hence, for a flight campaign or a space mission, it is logical to either store or downlink DDMs rather than the raw IF data. Currently, amongst the scientific research community, either raw IF data are processed on the ground or truncated DDMs downlinked from space are studied (Musko et al. 2020).

Lastly, no COTS GNSS-R receiver is available for researchers to customise based on various scientific applications. All these limitations have driven the objectives of this research as discussed in the following section.

1.5 Research novelty, objectives and significance

Novel contributions made in this dissertation are highlighted in each of the points listed below along with the research objectives and its significance:

1. **Development of a low-cost, real-time FPGA-based GNSS-R receiver using COTS hardware and 1-bit signal resolution**: This research focuses on development of a GNSS-R receiver that can detect and track direct and reflected GNSS signals (GPS L1 C/A) of 1-bit resolution.

   Significance:

- 1-bit quantization of the GPS signal results in a compact design
- COTS hardware provides a path to fast and efficient instrument development.
- Ability to track direct and reflected GNSS signals. The real-time SNR information can be further characterised to estimate geophysical parameters.
- Reconfigurability leads to low development costs.
- Real-time processing capability implies that the receiver can support field/flight campaigns for scientific research.

2. **GNSS-R receiver enhancements**: A COTS LHCP antenna with a nominal gain is procured to detect weak reflected signals. Due to its inadequacy in detecting extremely weak reflected signals, the sensitivity of the GNSS-R receiver is assessed and improved. The advanced weak signal acquisition implementation incorporates a known algorithm: alternate half-bit method introduced by (Lin and Tsui 2001) in the FPGA. With the enhanced receiver sensitivity, it is demonstrated that reflected GNSS signals as weak as -140 dBm can be detected by the receiver.

   Significance: Enhances the receiver's ability to acquire and track extremely weak reflected signals (-140 dBm or lower) as is the case in LEO or at higher altitudes.

3. **High resolution, real-time DDM generation using 1-bit signal resolution resulting in a compact design**: Missions such as TDS-1 (Camps et al. 2018) and CYGNSS (Ruf et al. 2012) and the recently launched DoT-1 (Unwin et al. 2021a) have the capability to generate DDMs on board the spacecraft; however, their resolution is limited to 17 Doppler bins by 11 code delay bins. This dissertation demonstrates the real-time generation of DDMs with a higher resolution of 20 Doppler bins by 40 code delay bins.

   Significance:

   - There is a need for real-time higher resolution DDMs, since existing sources of DDMs provide truncated information that are insufficient for science applications.

1.6 Dissertation organisation

Chapter 2 delves into the background of GNSS receiver technology and its application for reflectometry. Current state of research is discussed along with the challenges that are to be pursued to make significant progress in using a GNSS-R receiver for remote sensing purposes. This chapter also provides the context for research in this dissertation.

In Chapter 3, a novel reconfigurable GNSS-R receiver implementation is presented. Technical specifications such as the hardware components used, data rate, hardware/software partitioning, processing stages and final implementation is discussed.

The developed receiver is initially tested using simulated GNSS signals to characterize the system. The GNSS-R receiver is also tested in the field and the results presented.

Chapter 4 discusses the enhancements required for the GNSS-R receiver to be used for reflectometry applications. The receiver is also characterised in terms of its sensitivity to weak reflected signals. Design considerations, additional hardware required and algorithms to improve sensitivity of the receiver is presented. Results from tests performed in various scenarios is shown.

In Chapter 5, design and development of an FPGA-based DDM generator is discussed. CYGNSS datasets are used to validate the DDM generator and the field test results for varying surfaces such as land and water are presented.

Chapter 6 concludes this dissertation by addressing the research questions answered in this dissertation. Recommendations for future work are also provided.

Chapter 2 GNSS RECEIVER AND ITS ROLE IN REFLECTOMETRY

The emphasis of this chapter is to introduce the concept of reflectometry and the role GNSS receivers play to successfully retrieve geophysical parameters. Some recent GNSS-R missions and their relevance to this dissertation are presented. Current state of GNSS-R instruments is discussed including instruments with recent space heritage. This chapter also provides details on GNSS receiver architecture and addresses its inadequacies in the context of reflectometry.

2.1 GNSS receivers

GNSS has played a pivotal role in revolutionising the concept of navigation and its application in all spheres of technology. GPS, first conceived in the 1970s by the US, has been a crucial source for technological advancement (Kaplan and Hegarty 2013). As devices and digital electronics evolved, so have receivers. New digital signal processing (DSP) technology was used along with other microprocessors and, with that, software receivers were created. In this section, traditional hardware receiver designs and their limitations will be discussed, followed by the concept of software receivers and their positive impact on several fields of application.

2.1.1 Generic GNSS receiver architecture

The architecture of a generic receiver is presented in Figure 2-1. The received satellite signal goes through various processing stages before the navigation solution is computed. An overview of each of the processing blocks is provided along with the associated equations representing the state of signals.

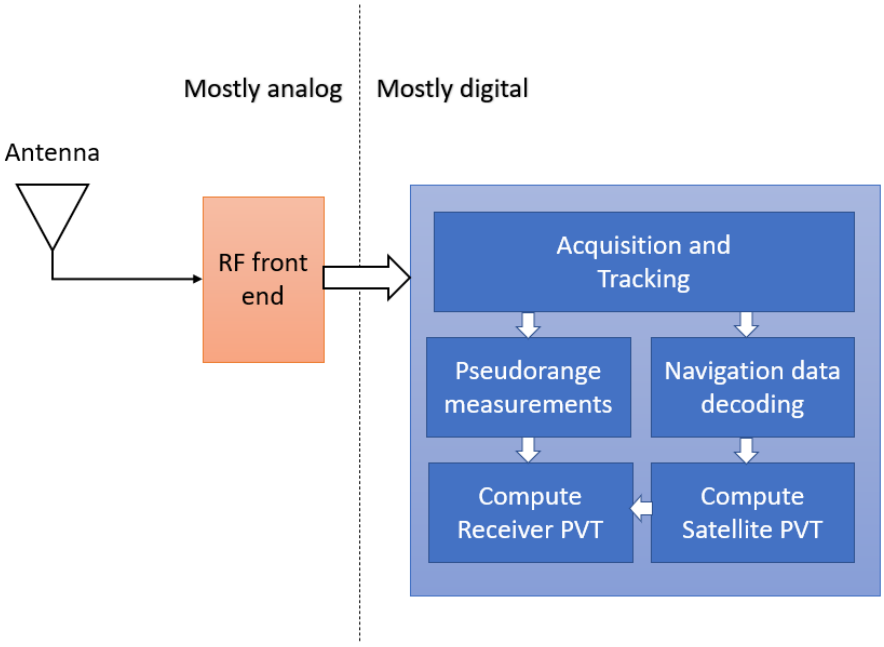


Figure 2-1 Block diagram of a conventional GNSS receiver

The antenna converts the sensed electromagnetic fields into electrical voltages. The received signal $s^k(t)$ for GPS L1 C/A-code is represented by Equation 2-1 (Borre et al. 2007).

$$s^k(t) = \sqrt{2P}D^k(t)C^k(t)\cos(2\pi f_{L1}t) \qquad \text{Equation 2-1}$$

where P is the received signal power, D^k is the data modulation, C^k is the spreading code modulation and the cosine function represents the carrier signal at frequency L1. The spreading code C^k is a sequence of 1023 chips and has a frequency of 1.023 MHz. The spreading code unique to every GPS satellite is also referred to as PRN. Analog signals from the antenna are further conditioned with the aid of components such as bandpass filters, amplifiers, mixers and analog-to-digital converters (ADCs). The conditioned signal at the output of the RF front end has an intermediate frequency (IF) usually much lower than the L1 frequency. The process of signal mixing is also known as down-conversion and is represented by Equation 2-2 (Borre et al. 2007).

$$s^k(t) = \sqrt{2P}D^k(t)C^k(t)\cos(\omega_{IF}t) \qquad \text{Equation 2-2}$$

where ω_{IF} is the intermediate frequency (IF) that the front end has downconverted the carrier signal to. The final product of the RF front end is expected to be a digitized product that is then further processed by digital processor channels. The signal, after the ADC conversion, is represented by Equation 2-3 (Borre et al. 2007).

$$s^k(n) = D^k(n)C^k(n)\cos(\omega_{IF}n) + e(n) \qquad \text{Equation 2-3}$$

where n represents the signal in discrete time. These digital signals are still suppressed under the noise floor (refer to Table 4-1 for link budget); however, the signals are further processed to detect all "visible" GNSS satellite signals in a step known as acquisition. During acquisition, the input signal is mixed with carrier and code replicas. The result of carrier replica mixing is represented in Equation 2-4 (Borre et al. 2007):

$$s^k(n)\cos(\omega_{IF}n) = D^k(n)C^k(n)\cos(\omega_{IF}n)\cos(\omega_{IF}n) \quad \text{Equation 2-4}$$

$$= -\frac{1}{2}C^k(n)D^k(\mathrm{n}) - \frac{1}{2}\cos(2\omega_{IF}n)C^k(n)D^k(n) \quad \text{Equation 2-5}$$

The second part of the signal is a high frequency component and is removed by the low pass filter effect of integration that occurs during processing. The core filtered signal is therefore only the first component. The next step entails mixing of code replica and the resulting signal is represented by the following equation:

$$\sum_{n=0}^{N-1} D^k(n)C^k(n)C^k(n) = ND^k(n) \quad \text{Equation 2-6}$$

where N is the number of data points in the signal. As seen in Equation 2-6 (Borre et al. 2007), only the navigation data D^k is retained. Although the process remains identical for acquisition and tracking, each of the processing modules have dedicated purposes.

While the objective of acquisition is to do a sweep through the frequency-code search space, tracking is responsible for continuously tracking satellite signals to extract the embedded navigation data from each signal. Satellite ephemerides are decoded and used to compute satellite positions. Based on the time it took for the signal to travel from the satellite to the receiver, pseudoranges are computed. Pseudoranges are biased estimates of the epoch distance between the satellite and the receiver. With known estimates of satellite positions and pseudoranges, the receiver's position can be estimated (Kaplan and Hegarty 2013).

2.1.2 Hardware receivers

For the typical user, a precise GNSS receiver is a black box which delivers specified measurements with very limited configuration possibilities (Overbeck et al. 2015). Conventional GNSS receivers rely heavily on application specific integrated circuits (ASIC)-based operations that are typically controlled by a central microprocessor. ASIC-based receiver design is compact, consumes low amounts of power and can process signals at a high data rate (Gleason and Gebre-Egziabher 2009). However, ASIC-based receivers are not suitable in all application scenarios. As an example, during research and development of a receiver, one would be required to fabricate a new circuit board for every design modification. With every board that is manufactured, testing and verification is also conducted. Therefore, development time and costs are typically higher for such an approach, and, in such situations, it is beneficial to not use rigid hardware platforms.

Today ASIC-based receiver designs have evolved to an extent that hardware receivers are “almost” irreplaceable, given that even low-cost ASIC receivers in mobile phones provide high quality output that can be used for high accuracy algorithms such as Real-Time Kinematic (RTK) and Precise Point Positioning (PPP) (Egea-Roca et al. 2022). However, accuracy is subject to the quality of the antenna used and is clearly the differentiating factor between a geodetic receiver and a mass market chip.

The cost of hardware receivers can range from a few dollars to tens of thousands of dollars. The accuracy and precision of the navigation solution computed typically

dictates the cost of the receiver. Antennas also play a significant role in determining the quality of the navigation solution. Mobile phones most commonly use inverted-F antennas to sense GNSS signals. However, more recently studies have shown centimetre-level accuracy with receiver chips embedded in smartphones. However, solutions are only under ideal conditions of open sky (Purfürst 2022). High-end geodetic-grade receivers are also available in the market offering mm-level accuracy. Geodetic grade receivers often are used with choke ring antennas that reject large amounts of multipath. Therefore, depending on the application, there is a wide range of receivers and antennas available to choose from.

There are certain applications such as reflectometry that rely upon data derived from the intermediate processing stages of the GNSS receiver. Most commercial receivers listed earlier only provide products that are processed to an extent or in the form of a navigation solution. The user does not have access to any raw data from the front end and, traditionally, there was no need for such a feature. In addition, hardware receivers are not designed to process LHCP signals and compute DDMs (essential for reflectometry). Nevertheless, with modern concepts and technology, especially in the realm of remote sensing, it is inevitable and imperative for researchers to design and build custom GNSS receivers. Hence, the innovative and ever-growing field of reflectometry presents a unique opportunity for this dissertation to address and offer robust solutions.

2.1.3 Software receivers

As of today, several open-source software receivers are available to anyone who would be interested in processing GNSS signal datasets. The software GNSS receiver community is large and active with several contributors developing extensive features for the receivers (Bernabeu et al. 2022).

Software receivers can be implemented on microprocessors or computers. Experienced programmers can conveniently make design modifications to change how a receiver processes data or determine what the final product of the receiver should be. Software receivers offer design flexibility that hardware receivers do not. The disadvantage of using a software receiver is slower processing speed compared to hardware receivers. The major reason for this slower performance is that microprocessors execute programs sequentially. Even if a multi-thread approach is adopted, at the operating system kernel level, the threads are time-scheduled denoting that the concept of parallel threads or tasks on a processor is just artificial. In case of a multi-core processor, the multi-thread approach is more time efficient however the increased power consumption can be unsuitable for a satellite application. If the best of both worlds (software and hardware receivers) is used in conjunction, that would definitely enhance the processing platform abilities. Prior to discussing the software and hardware configuration, the concept of FPGAs must be introduced.

2.1.4 Field Programmable Gate Arrays (FPGAs)

FPGAs are a type of reconfigurable hardware. The advantages of using FPGA alongside a microprocessor is well-known and has gained popularity both in industry and academia. The most significant benefit of using an FPGA is its ability to perform tasks in parallel and in real-time. Parallel tasks can function independently without the intervention of any other logic block (Hein et al. 2006). This powerful feature is ideal for a GNSS receiver, since a receiver typically consists of several tracking channels that do not depend on each other.

Figure 2-2 shows the evolution of FPGA architecture. The left portion of the diagram shows the initial simple architecture containing logic blocks including look-up tables (LUTs) and flip flops (FFs). FPGAs are constructed using programmable interconnects and can be programmed to perform any combinational or sequential operation. A modern FPGA, shown on the right half of Figure 2-2, contains several complex blocks such as dedicated digital signal processors (DSPs), block random access memory (RAM), processor subsystems and external memory controllers. Each FPGA chip has a fixed number of resources and systems employing FPGAs must ensure that sufficient resources are available for a successful implementation. Amongst these blocks, the instantiated block RAMs are the most susceptible to radiation effects (Nidhin et.al., 2017). As discussed in later sections, the utilization of 1-bit signal resolution ensures that no BRAM is required to be instantiated hence reducing the risk due to radiation effects.

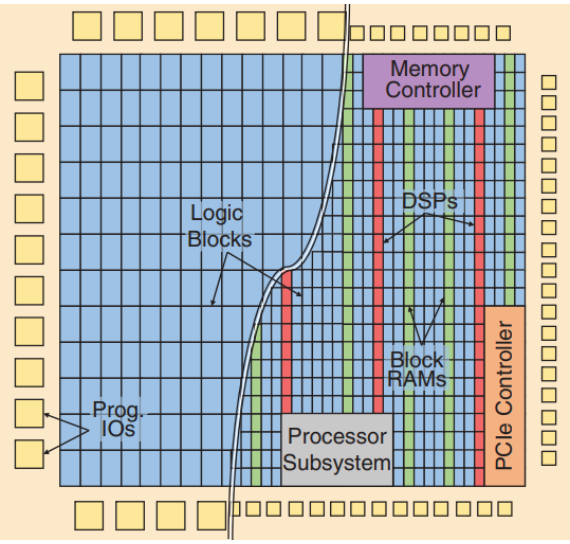


Figure 2-2 Evolution of FPGA architecture: Early architecture was more simplistic compared to modern FPGAs comprising complex components such as DSPs, memory controller and processor subsystems (Boutros and Betz 2021).

2.1.5 System-on-Chip

Manufacturers have realized the popularity of using microprocessors alongside FPGAs and hence also offer embedded softcore processor options (eg., Nios II, Microblaze, etc.). Such embedded processors can be used to control and monitor the heavy signal processing that the FPGA performs. Developers may want to use more powerful and efficient embedded processors given that certain transcendental functions such as arctangent, square root, etc., are only required at a low rate, hence software based iterative approaches such as the CORDIC algorithm is more suited to run in the CPU rather than consuming a significant amount of hardware resources. For such possibilities, FPGA platforms are also built with embedded hardcore processors (such as ARM or

PowerPC) that are 3-4 times faster than softcore processors (Weber and Chin 2006). This aspect can be especially beneficial for GNSS signal processing. A System-on-Chip (SoC) is the integration of both software and hardware processing elements on the same silicon chip. Using SoC provides huge benefits as they have lower power requirements, are more efficient, occupy lesser board area and facilitates easier integration (Ben Salem et al. 2008).

2.1.6 Comparison of processing platforms

The performance and flexibility of various processing platforms is compared in Figure 2-3. As discussed earlier, ASICs have the best performance, but are rigid and do not accommodate post-fabrication hardware changes. Microprocessors allow easy design changes but are slow compared to ASICs. There are also digital signal processors (DSPs), that have dedicated correlators which are faster than microprocessors, but still not as fast as ASICs. FPGAs are reconfigurable hardware and, along with a microprocessor, make a powerful processing platform offering the best-case scenario of performance and flexibility. Graphical processing units (GPUs) are powerful platforms in terms of parallel processing ability and are also deemed to be flexible making them a suitable candidate for GNSS receivers. However, GPUs have very high power requirements that would be unsuitable for a satellite platform to support (Mota et al. 2022). Since the GNSS receiver designed in this research is within the scope of the space domain, GPUs are not considered as potential candidates. SoCs (combination of FPGA and microprocessor) best satisfy the requirements of a GNSS receiver prototype for reflectometry applications

for the following reasons: i) low-cost platform, ii) reconfigurable, iii) hosts a combination of software and hardware elements, and iv) real-time signal processing capabilities.

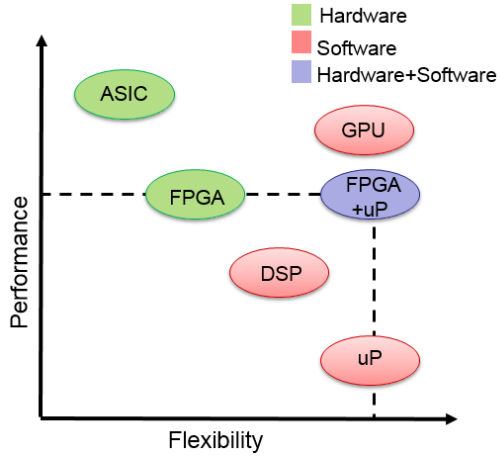


Figure 2-3 A relative comparison of performance versus flexibility of various processing platforms

2.2 GNSS reflectometry

GNSS reflectometry (GNSS-R) is a method of Earth's surface characterization using reflected GNSS signals. GNSS-R receivers detect direct and reflected signals and compare the two signals in terms of signal strength, to estimate geophysical information of the reflecting surface (Unwin et al. 2016). The reflected GNSS signals vary from direct signals in terms of time delay, Doppler shift, signal power and polarization (Gamba et al. 2015). Properties of reflected signals depend on various geophysical parameters such as surface roughness, moisture and vegetation (Motte et al. 2016a; Nogués-Correig et al. 2007).

Reflected signals are conventionally considered as multipath signals or as a contributor to the error vector when computing a navigation solution. The concept was first introduced as multistatic reflectometry by (Hall and Cordey 1988). The emerging remote sensing technology using GPS signals was first proposed by Martin-Neira et al. (2001). Since then, several such scientific proposals followed and experimental data was collected and proven (Emery and Camps 2017). Figure 2-4 illustrates the arrival of several reflected GPS signals at the receiver. The receiver in a low Earth orbit also detects direct GPS signals. Each of the signal paths shown are unique and hence for every satellite signal, an exclusive reflection point is observed and assessed.

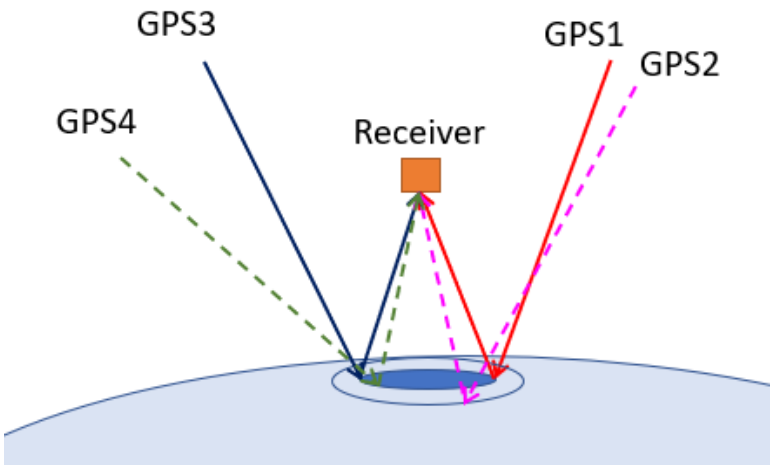


Figure 2-4 Reflected GNSS signals from multiple satellites (Kaplan and Hegarty 2013)

2.2.1 Signal path

Consider the path of a signal that is transmitted by a GNSS satellite. The signal travels to Earth and gets reflected back to space and is received by the LEO receiver. The point of

reflection on the Earth's surface, for a signal with the shortest path between the transmitter-Earth-receiver, is known as the specular point. The incident angle and the reflected angle of the signal are equal in this scenario. In Figure 2-5, the specular point is highlighted for the signal with path 1. The elliptical rings around the specular point are known as iso-delay rings. Each of the rings represent a delay relative to the specular point. As an example, referring to Figure 2-5, if a signal were to take path 2 before arriving at the receiver, it would be delayed with respect to the signal on path 1 since it is a longer route. In fact, all the surface points on the ellipse indicate the same code delay with reference to the specular point. Similarly, depending on factors such as receiver-satellite position and velocity, varying Doppler shifts in reflected signals are also observed (Gleason and Gebre-Egziabher 2009; Jales 2012). The varying frequency can be represented by using iso-Doppler hyperbolas (determined by the relative motion between the Earth and the receiver) as indicated in Figure 2-5 and all the signals reflecting off the surface points on a hyperbola exhibit identical Doppler shift.

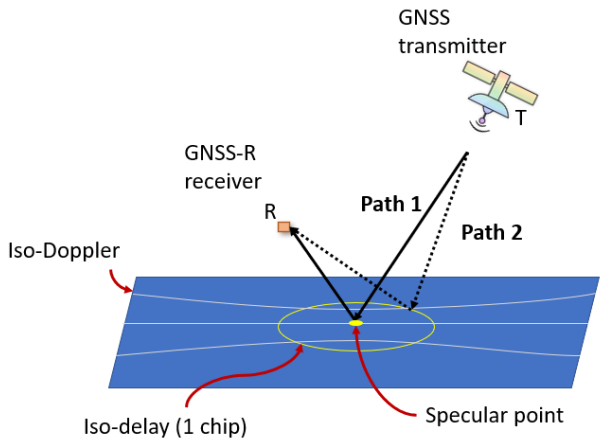


Figure 2-5 Iso-delay ellipses and iso-Doppler hyperbolas represented relative to the specular point.

2.2.2 Properties of reflected GNSS signals

Unlike the traditional technique of remote sensing using radar, where the back scattered signals are analysed, remote sensing using reflected GNSS signals relies on scattering that occurs in a forward direction. Since the transmitter and receiver are not co-located, the setup has a bistatic configuration (Gleason and Gebre-Egziabher 2009). Therefore, the reflected signal can be modelled by the bistatic radar equation and the total received power can be represented using Equation 2-7 below.

$$\langle |Y(\tau, f; t)|^2 \rangle = \frac{\lambda^2 P_t G_t}{(4\pi)^3} \iint \frac{G_r \chi(f, \tau)^2}{R_1^2 R_2^2} \gamma \sigma_{pq}^0 dS \qquad \text{Equation 2-7}$$

where $P_t G_t$ represents the effective isotropic radiated power (EIRP) of the antenna, G_t and G_r are the transmitter and receiver antenna gain, respectively, R_1 is the distance between the transmitter and surface point, R_2 is the distance between the surface point

and the receiver, $\chi(\cdot)$ is the Woodward Ambiguity Function (WAF), γ is the attenuation factor due to vegetation canopy, σ_{pq}^{0} is the total polarization-dependent normalised bistatic radar cross section (NBRCS), with p and q the incident and scattered polarisations respectively (Loria et al. 2020). The WAF of the signal is the result of the matched filtering for the delay τ and Doppler frequency f of the reflected signal. The NBRCS factor accounts for the physical properties of the reflecting surface (Jales 2012).

In Section 2.1.1, the direct GPS L1 C/A-code signal is theoretically discussed, as well as the impact of mixing the signal with locally generated carrier and code replicas is described (Equation 2-1 to Equation 2-6). Similarly, the reflected signal can also be represented as shown in Equation 2-8,

$$Y(\tau, f; t) = \int_{0}^{T_i} a(t + t')u(t + t' + \tau)e^{2\pi i f t'} dt' \qquad \text{Equation 2-8}$$

where T_i is the coherent integration period, $a(t)$ is the PRN code replica, $u(t)$ is the reflected signal, f represents the Doppler shift and t' is the time delay. The following subsections highlight the parameters of interest with respect to processing of direct and reflected signals. Each of the parameters have a significant impact on receiver design and hence are addressed.

2.2.2.1 Reflected Signal Strength

Direct GNSS signals (GPS L1 C/A) are known to arrive on Earth's surface with a minimum strength of -128.5 dBm. They are already relatively weak, as they are at least

18 dB below the noise floor (-110 dBm for a bandwidth of 2 MHz) (when computing the difference between two values with the unit dBm, the resulting value is represented in dB) and cannot be detected without processing using advanced correlation methods (Borre et al. 2007). Upon reflection, it is expected that the signal will be suppressed further down below the noise floor, challenging the sensitivity of the receiver and detection algorithms. Open loop tracking methods must be used to track reflected GNSS signals in space. Using Doppler frequency and code phase measurement derived from direct GNSS signals, the expected reflected signal code delay and Doppler can be estimated. Using these estimation, the delay Doppler map (DDM) generators can be steered to detect and track reflected signals successfully.

2.2.2.2 Polarisation

Direct GNSS signals exhibit right hand circular polarisation (RHCP). Therefore, GNSS receivers typically use zenith pointing RHCP antennas. Upon reflection, the polarity of the signal transitions to left hand circular polarisation (LHCP). The extent of energy conversion from RHCP to LHCP depends on the surface dielectric constant and the satellite elevation angle. The geodetic community often uses high quality choke-ring antennas that exhibit high RHCP gains. These antennas are remarkable at suppressing reflections especially from metal surfaces, where most of the energy is converted to LHCP. They do not perform as well to discard reflections from natural surfaces such as soil or ground (Wan et al. 2015). It is essential to note these details in the context of

reflectometry. Therefore, there is a need for an LHCP antenna, in addition to the RHCP antenna that a standard GNSS receiver usually has. Furthermore, the LHCP antenna must point down to be able to detect the reflected signals. Hence, it is necessary for a GNSS-R receiver to process both RHCP and LHCP signals simultaneously. Details about antenna requirements are further discussed in Chapter 3.

2.2.2.3 Code delay and Doppler shift

In

Figure 2-5, two signal paths are demonstrated. Both the signals arrive at the receiver yet have different code delays and Doppler shifts. Depending on the point of reflection around the specular point, which is used as a reference point, code delay and Doppler shift can be estimated using delay-Doppler maps (DDMs). DDMs are further explained in detail in Section 2.3.

2.2.3 Challenges to detect weak reflected signals in space

The most prevalent challenge lies in detecting and acquiring weak reflected GNSS signals (Gleason and Gebre-Egziabher 2009). Scattering of signals can make it even more difficult for the receivers to differentiate valid signals from noise. GNSS-R receiver requirements such as LHCP antenna with high-gain characteristics, higher Doppler shift tolerance and short integration intervals, are challenging and make it a unique and interesting problem to solve.

2.3 Delay Doppler maps

The essential GNSS-R measurement is the delay-Doppler map (DDM). DDMs represent cross correlation values between the locally generated PRN code sequence, locally generated carrier signal and the reflected GNSS signals. Since the signals are 1-bit resolution, the correlation is simply a logic operation (XOR). The map is formed by correlating over a range of delays and Doppler frequencies (Clarizia et al. 2009). The pattern in the DDM depends on the roughness of the reflecting surface. As the surface becomes rougher, the GNSS signals are scattered over a larger area and in varying directions. This results in spreading of the power from the specular point to the neighbouring bins. Another point to note when mapping the DDM to spatial locations is that only the specular point bin represents a single location on the surface. Other than that, multiple locations are mapped to the same DDM bin (University of Michigan 2016). Further details are provided in Chapter 5.

2.4 Review of GNSS-R instruments

Various research groups and organizations across the world are developing technologies to support remote sensing using GNSS signals. While several GNSS-R software receivers exist, there are a limited but growing number of hardware-based instruments for space applications. Some instruments also have space heritage. This subsection provides a summary of known software GNSS-R instruments. The hardware-based instruments

developed for space are discussed in more detail, given their relevance to this dissertation.

2.4.1 Software-based GNSS-R instruments

Open-source Software GNSS Receiver has been enhanced to also support reflectometry. This software receiver was developed by (Lestarquit et al. 2016a) using the GNU-radio GNSS-SDR developed by Centre Tecnologic de Telecomunicacions de Catalunya. Due to the lack of access to low level observables and limited control over signal processing, the team was motivated to develop a custom GNSS receiver for reflectometry applications.

Software receivers are a great tool for post-processing raw data and can be advantageous during the validation process. Raw data collected from various sources can be processed on a laptop for quick results. New software processing modules can also be added with a short turnaround time. However, the objective of this research is to develop a real-time GNSS-R receiver prototype using a hardware platform.

2.4.2 Hardware-based GNSS-R instruments

This subsection describes some of the GNSS-R receivers built or being developed along with a brief architectural review relevant to this dissertation.

- *GOLD-RTR*: This dedicated GNSS-R receiver has been developed by the Institute of Space Sciences (IEEC, CSIC) (Nogués-Correig et al. 2007). Although it is referred to as a software receiver (Li et al. 2022), the

baseband processor uses hardware correlators to generate products. The developed receiver processes signals from three antennas: one zenith RHCP antenna and two nadir LHCP antennas. The reflectometry receiver works in conjunction with a commercial GPS receiver card that computes the navigation solution. More recently, the ground-based GNSS-R receiver now generates complex waveform products. These publicly available products are used later to verify the DDM generator design in this dissertation.

- *UK-TDS1:* SSTL gathered sufficient data from their previous satellite mission called Disaster Monitoring Constellation (DMC) to establish that reflected GPS signals could indeed be gathered from a LEO satellite. While the instrument that flew on the DMC satellite was based on an older Space GPS Receiver (SGR) model consisting of ASIC correlators and an ARM processor (Blunt 2007), the new generation SGR-Remote Sensing Instrument (ReSI) is based on a Flash-based non-volatile FPGA (De Vos Van Steenwijk et al. 2010). The Actel ProASIC3 FPGA encompasses the correlators and functions in association with a LEON3 softcore microprocessor. Significant enhancements were made to the instrument, making it a highly flexible receiver capable of processing multi-frequency, multi-constellation signals. SGR-ReSI is also capable of generating near real-time DDMs using the Fast Fourier Transform (FFT) method to

compute data for each delay/Doppler pixel. The FFT search method increases the computational complexity, which is reduced by decimating the signal prior to Fourier transform. Though decimating the signal introduces undesirable features towards the edges of the bands and have to be carefully removed using filters (De Vos Van Steenwijk et al. 2010). In addition, the processing time is directly proportional to the number of pixels for every channel and for SGR-ReSI, a DDM of dimensions of 128 delay by 20 Doppler shift is generated. Relevance of the DDM dimensions is discussed further in Chapter 5.

- *CYGNSS:* The Cyclone Global Navigation Satellite Systems (CYGNSS) was launched in 2016 as an eight-satellite constellation for the purpose of measuring ocean surface wind speed. Near-surface wind speed monitoring over oceans offers insight into the dynamics of tropical cyclones especially near the inner core of the cyclone itself (Ruf et al. 2016). The reflectometry instrument is the same as that flown on TDS1: SGR-ReSI. The products generated by the instrument are similar to the TDS1 mission, where truncated DDMs are the main product. DDMs are downlinked in a truncated format due to the limited data bandwidth availability. On some occasions however, to support scientific research, raw data and full DDMs are also recorded and downlinked. The products are made available to the public via the PODAAC portal. Using CYGNSS data, various applications

have been demonstrated: soil moisture measurement (Chew and Small 2018), and inland water and flooding detection (Gerlein-Safdi and Ruf 2019).

- *STRATOS:* Spire Global launched the first of many GNSS-R Cubesats in 2019. The satellite carries an advanced GNSS-R receiver based on the SDR concept and utilisation of FPGA for processing acceleration. GNSS-R products are now commercially available that estimates geophysical parameters such as ocean wind, soil moisture, sea ice, wetlands and flood inundation. Limited information is available about the receiver design itself; however it is known that the receiver can process signals from multiple constellations (Li et al. 2022).
- *BuFeng-1:* The first GNSS-R satellite mission that was launched by China in 2019 is BuFeng-1 (BF-1). These twin satellites were launched to observe and measure ocean wind speeds during typhoons and hurricanes. The GNSS-R instrument on board the satellite is capable of processing GPS and Beidou signals. Similar to the SGR-ReSI (used in TDS1 and CYGNSS), BF-1 can process four DDM channels simultaneously and can generate DDMs (128 delay x 21 Doppler bins) at the rate of 1 Hz (Jing et al. 2019).
- *FSSCat:* The GNSS-R payload is one of the two 6-unit CubeSats launched in 2020 as part of the FSSCat mission. The GNSS-R instrument is based

on GPS reflectometer instrument for passive advanced unit concept (griPAU) (Valencia et al. 2010; Munoz-Martin et al. 2020). The instrument generates DDMs at a fairly high rate of 25 Hz, resulting in increased spatial resolution. The adverse effect of short integration periods is loss of lock on tracked signal as suggested by initial validation results (Munoz-Martin et al. 2020). The DDM dimensions are 40 delay by 5 Doppler frequency bins and is mostly limited by the FPGA resources.

- *FengYun-3E:* This satellite launched by China in 2021 is mainly designed and developed for the purposes of GNSS-radio occultation (RO). This satellite can also function as a GNSS-R receiver and contribute to tropical cyclone monitoring. The payload weighs about 25 kilograms and consumes about 55 W of power. The GNSS-R instrument has eight channels dedicated for processing reflected signals. The DDMs, generated at the rate of 1 Hz, have a dimension of 122 delay bins by 20 Doppler bins, but the delay bins are not uniformly spaced. The delay bin spacing varies between 0.25 chips and 0.125 chips (Yang et al. 2022).

2.4.3 Future missions

Some of the upcoming missions with GNSS-R receiver payloads are described below:

- *HydroGNSS:* HydroGNSS, designed and developed by SSTL, is a mission set to launch in 2024 with the objective of monitoring key essential climate variables such as soil moisture, wetlands, freeze/thaw state and above ground biomass.

Some new innovations that have not previously been available are coherent signal processing channels, LHCP and RHCP reflected signals processing, GALILEO signals, and L1 and L5 signal tracking ability. The GNSS-R instrument to be developed is based on an advanced architecture of the SGR-ReSI (Unwin et al. 2021b).

- *Triton:* The satellite Triton is an experimental technology demonstration mission to be launched in 2022 carrying a space-grade GPS receiver (GPSR) developed by the National Space Organisation (NSPO, Taiwan). The GPSR has been modified to process reflected GPS L1 and QZSS signals using four dedicated channels. The DDM dimension is set to have 128 delay and 40 Doppler frequency bins (Tsai et al. 2021).

2.4.4 Prevailing challenges

From the listed missions, it can be inferred that since the technology of reflectometry was first introduced and proven, there has been significant interest from both industry and academia to pursue further advancements. There are challenges that remain to be addressed and are reviewed in this reasearch:

- Although data from a few GNSS-R instruments are public, the users of these datasets are limited by the specifications of the instrument. For example, when using CYGNSS data, based on the DDM dimensions, the spatial resolution is restricted to ~25 km depending on the satellite elevation angle (Chew et al. 2016). Validation of data derived from the instrument versus in-situ measurement is

challenging considering that the specular point could be located anywhere in the 25 x 25 km region and that the geophysical parameters could vary significantly in that area.

- Considering the high data volume of DDMs, the dimensions of the DDM is defined by the architecture of the receiver and the resources available in the FPGA. An additional challenge is the capability of the satellite's bandwidth to downlink the DDMs. Ideally, with enough maturity of the scientific models developed, onboard algorithms can provide real-time science data - instead of recording full DDMs.
- GNSS-R receivers rely on high-gain, LHCP antennas and since this is not a readily available commercial component, custom antennas must be built.
- For potential new applications GNSS-R applications such as estimating soil moisture for agriculture analysis, weather forecast and flood prediction, a customizable GNSS-R receiver is best suited, permitting easy validation of the developed scientific models. However, commercially available GNSS receivers cannot be repurposed to support reflectometry. Research groups targeting specific GNSS-R applications are left to build their own GNSS-R receiver.

2.4.5 Is 1-bit quantization the answer?

Given the current challenges in procuring a customizable GNSS-R receiver, this dissertation demonstrates the design and development of a reconfigurable GNSS-R

receiver. In this section, the benefit of using a 1-bit signal resolution for GNSS-R receivers is presented. GNSS receivers using 1-bit signal resolution for correlation is not new. However, the performance of a 1-bit GNSS receiver for reflectometry applications is unknown. This section intends to provide context and insight into the relationship between quantization width (in bits) and memory requirements (FPGA resources).

2.4.5.1 Quantization loss

Several studies (Sauriol et al. 2007; Hegarty 2011; Principe et al. 2011; Kaplan and Hegarty 2013) have compared the impact of various quantization bit widths on the overall performance of the GNSS receiver. The ADC converter in the RF front end introduces a quantization loss where the magnitude of loss depends on the number of bits used to represent the input analog signal (in this case GPS L1 C/A-code). A 1-bit quantization results in a 1.96 dB loss. The loss is reduced to 0.54 dB when the quantization is 2-bit. For 3-bit and 4-bit quantization the loss is 0.17 dB and 0.05 dB, respectively (Kaplan and Hegarty 2013).

Even though increased bit resolution results in higher SNR (due to the lower quantization losses), for the extent of multiplier complexity introduced by a 3 or 4-bit signal resolution, the returns are minimal (when compared to 2-bit resolution) (Kaplan and Hegarty 2013). Also, (Egea-Roca et al. 2022), observed that most GNSS receivers use 1 or 2-bit signals to reduce memory requirements, as well as power consumption. The biggest difference is observed when going from 1-bit to 2-bit quantization and therefore

most GNSS receivers use either 1-bit or 2-bit signal resolution even though they introduce higher implementation losses.

2.4.5.2 Importance of a compact design

The design and memory requirements of devices in space are of significant relevance as the harsh operating environment introduces various technical challenges, such as limited power availability, damage risk by radiation and other thermal concerns. In recent times, instead of investing in expensive radiation hardened hardware, commercially available low-cost components are gaining popularity for LEO missions.

Commercial FPGAs adopt certain design strategies to mitigate risks triggered by radiation. The three types of failures that can occur due to radiation are total ionization dose (TID), single event upset (SEU) and single event latch-up (SEL). While the FPGA can be recovered from SEU and SEL, TID causes an irreversible damage to the logic gates over time. Below is a brief list of design approaches (for FPGA) that are highly recommended (Roosta 2004; Parkinson et al. 2011) to avoid failures caused by radiation:

1. Avoid the use of block random access memory (BRAM) in the FPGA.
2. When integrating hardware subsystems, differential signals must be used.
3. Use of triple modular redundancy (TMR) for most, if not for all logic.

Even though the developed GNSS-R receiver in this research will require additional features before it can be flown to space, the design in this dissertation considers the above listed points for the baseline design.

The choice of 1-bit signal resolution aligns very well with the risk mitigation techniques listed above. With the use of 1-bit quantization, there is no longer a need to use BRAM given that all the local signals are generated in real-time (as opposed to being stored in an BRAM) therefore reducing the resource requirements. The RF front end used in this dissertation generates differential signals at the output. With a compact design, there is plenty FPGA area available for implementing TMR (creates three copies of the same logic for redundancy). An additional benefit to a compact design is that it can be appended to a satellite as a secondary payload instead of having an expensive dedicated mission (further discussed in Section 3.2.3.).

Table 2-1 lists all the receivers and missions previously described in section 2.4.2. Each of the satellites are classified and compared in terms of real-time signal processing capability and the signal resolution used. Some payloads, such as CYGNSS, were flown as primary missions, whereas missions such as FengYun-3E were one of many other payloads on the satellite. With technical demonstration of a compact GNSS-R receiver, the prototype would be a suitable secondary payload for a nanosatellite. With a constellation of nanosatellites, complete coverage of the Earth's surface is possible.

Table 2-1: Comparison of satellite missions for reflectometry applications

Receiver/Mission	For space application	Real-time	1-bit quantization	Payload	Satellite classification
GOLD RTR	No	No	Yes	No	-
UK-TDS1	Yes	Yes	No	1/8 payloads	Small satellite
CYGNSS	Yes	Yes	No	Primary	Microsatellite
SPIRE-LEMUR	Yes	Yes	No	1/3 payloads	Nanosatellite

BuFeng-1	Yes	Yes	No	Primary	Small satellite
FSSCat	Yes	Yes	No	Primary	Cubesat
FengYun-3E	Yes	Yes	No	1/11 payloads	Large satellite
Developed GNSS-R	Yes	Yes	Yes	Secondary*	Nanosatellite*

*The developed receiver is not scheduled for any flight/mission; this is the category that the GNSS-R receiver is aiming for.

2.5 Proposed GNSS-R instrument

Given all the challenges listed above, this research provides a solution to an extent where a low-cost, COTS-based GNSS-R receiver is designed and implemented. Table 2-2 lists features of the developed receiver introduced in each of the subsequent chapters.

Table 2-2 Features of the developed GNSS-R receiver prototype

Chapter 3	• A *reconfigurable* real-time GNSS-R receiver. • *Simultaneous processing capability* of direct and reflected GPS L1 C/A signals. • Locally generates C/A-code sequence and carrier signals to avoid usage of BRAM (FPGA resource) contributing to a *compact design*. • *Real-time carrier-to-noise density ratio* information for every signal is generated
Chapter 4	• Considering that reflected signals are weak, with novel enhanced algorithm implementation, the *sensitivity of the developed receiver is increased* to detect GPS L1 signals that are as weak as -140 dBm.
Chapter 5	• *Real-time DDMs* are generated at the rate of 2 Hz and relies on open loop tracking method. • *High-resolution DDMs* using 1-bit signal resolution. • The *DDM dimensions are reconfigurable* and user defined.

While, in the list above, each of the chapter describes the features of the developed GNSS-R receiver, the Table 2-3 shown below also highlights the research questions this dissertation addresses. At the end of each chapter, these questions are re-visited and answered.

Table 2-3: Research questions addressed in this dissertation

Chapter 3	*Question 3A*	Can a compact one-bit quantized GNSS-R receiver be used for reflectometry applications?
	Question 3B	How well does it perform? How distinct are the reflections from land versus water?
Chapter 4	*Question 4*	Can the sensitivity of the developed GNSS-R receiver be enhanced further to be able to detect extremely weak reflected signals?
Chapter 5	*Question 5A*	How feasible is the implementation of a high resolution (40 x 20) real-time DDM using one-bit quantized GNSS signal? Is there any room for data optimization?
	Question 5B	How does the FPGA-based DDM generator perform with real, reflected GNSS signals received in space?

Chapter 3 GNSS-R RECEIVER DESIGN, DEVELOPMENT AND IMPLEMENTATION

This chapter discusses the GNSS receiver architecture that has been developed within the context of reflectometry in space. The precursor to the developed receiver is a C-based software receiver (Guruprasad 2015), which has been tested and validated against other highly regarded software receivers such as the one developed by Borre et. al. (2007). The software receiver has been ported over to the processing platform, where the software functions are translated to custom HDL modules. The hardware accelerators introduced in the receiver results in a processing speed improvement by 20% (Guruprasad et. al. 2015). During this migration, several implementation modifications were made to the algorithm leading to a novel approach. While this chapter addresses the GNSS-R receiver prototype, the next chapter provides insight into additional complex features that are added to the base prototype to improve the sensitivity of the GNSS-R receiver.

3.1 Overview of GNSS-R receiver prototype

The complete receiver prototype is made of three major components: antennas, RF front end and a GNSS signal processing unit. A high-level diagram is presented in Figure 3-1. The hardware entities are described in the next few subsections, along with the selection rationale.

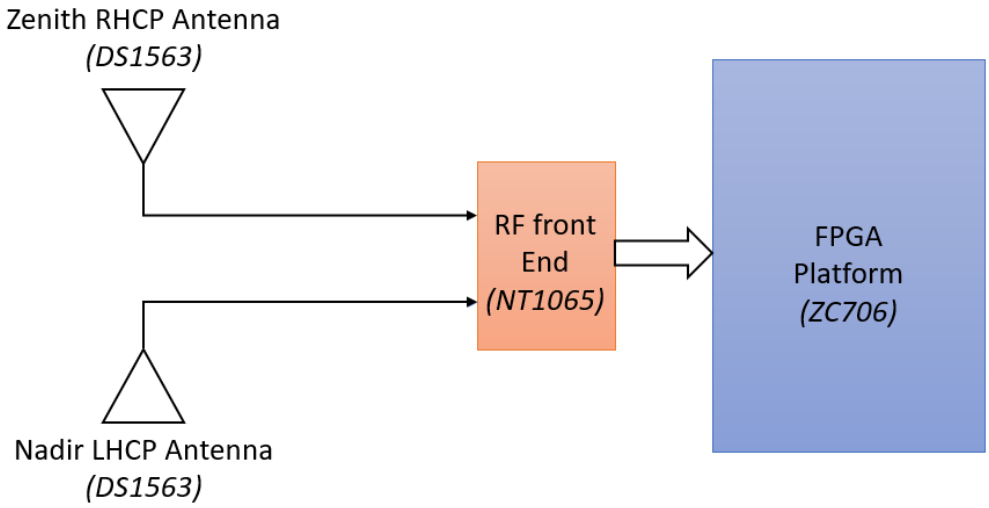


Figure 3-1 High level diagram presenting the main components of the GNSS-R receiver such as the antennas (DS1563), the front end (NT1065) and the processing platform (ZC706).

3.1.1 Hardware components

One research objective is to use COTS hardware components to construct the receiver prototype and demonstrate its feasibility. The reasoning being to reduce receiver development time and focus on the niche research area of reflectometry and weak signal processing.

3.1.1.1 Antennas

Antennas form the first point of contact between the transmitted GNSS signals and the receiver. The antennas hugely influence the quality of the signal as both received GNSS

signals and noise are converted from electromagnetic waves in space to electrical voltages (Kaplan and Hegarty 2013). Other than the cost and form factor of the antenna being the major considering factor, bandwidth and polarisation drive antenna selection. The required characteristics of the nadir antenna is more stringent than the zenith antenna.

Antennas built for GNSS applications are traditionally designed to reject LHCP (reflected) signals given that multipath impedes the accuracy of the receiver's navigation solution. However, for reflectometry, the reflected signals are the key data points. Ideally, the nadir antenna must have a high gain and also be able to collect weak reflected LHCP signals. Custom designed antennas have been used by several researchers due to these rigid criteria (Manandhar and Shibasaki 2007; Unwin et al. 2013). The options for a commercially available LHCP antenna are very limited. At the time of development of the receiver only Cobham manufactured COTS antennas capable of detecting LHCP in the GNSS signal frequency band. The two options available were DS1265 and DS1563 (both passive) (Figure 3-2), both manufactured by Cobham (2018).

Referring to Table 3-1, DS1265 has a higher forward gain (in dBiC) and would have been a good candidate for the GNSS-R receiver prototype. However, DS1563 was selected to form a part of the final design for several reasons. The first reason for not choosing DS1265 is that the high gain antenna is both much larger (380 mm x 380 mm) and heavier and would be unsuitable for future flight campaigns. The second reason for choosing DS1563 is that it is a dual polarised antenna and can capture both LHCP and

RHCP signals. The dual polarised antenna is a bonus as both zenith and nadir antennas can now be identical. In addition, the nadir antenna provides RHCP and LHCP signals, which is again significant data for reflectometry as examining reflected signals from both polarisations can assist in distinguishing effects of soil moisture from soil roughness (Unwin et al. 2021b). The dual polarised feature of the antenna was considered as an added benefit and hence was chosen to be part of the prototype, despite having a slightly lower gain relative to DS1265.

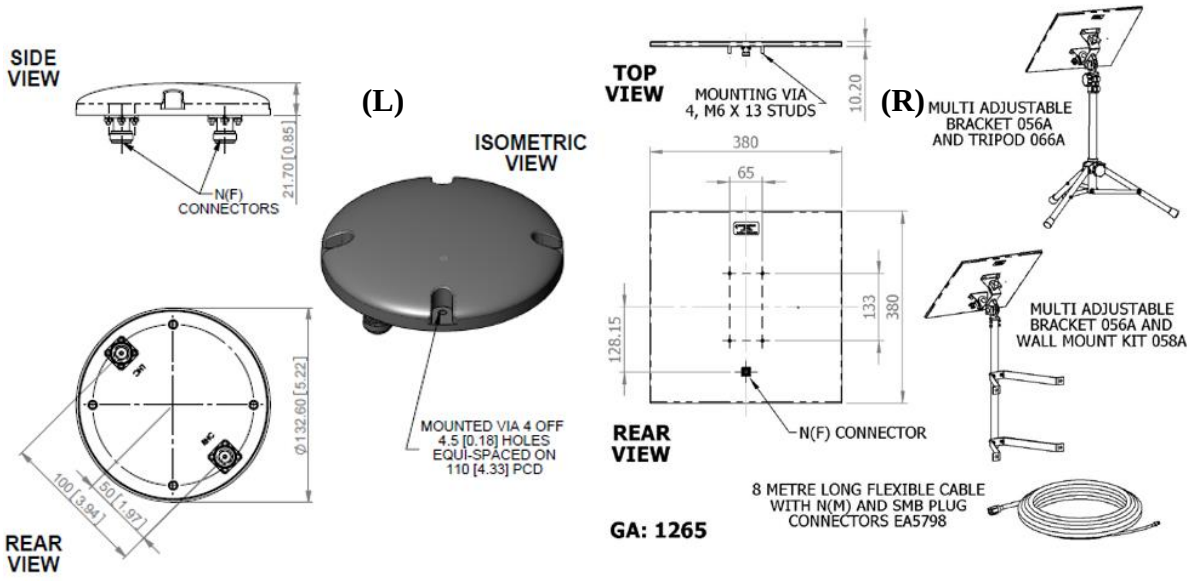


Figure 3-2: DS1563 (L) and DS1265 (R) antennas manufactured by Cobham (2018)

Table 3-1: Comparison between DS1265 and DS1563

Parameter	**DS1265**	**DS1563**
Frequency	1.5250 – 1.6605 GHz	1.5250 – 1.6605 GHz
Peak Gain	14.0 dBiC Peak	7.25 dBiC Nom.
Polarisation	LHC	LHC and RHC

Beamwidth	~27° (Az) x 27° (El)	~67° (Az) x 65° (El)
Mass	1000 g	190 g
Dimension	380 mm x 380 mm	132 mm diameter

3.1.1.2 RF front end

The primary objective of the RF front is to amplify the signal received by the antenna (Kaplan and Hegarty 2013). Certain parameters such as Noise Figure (NF) also come into play when the receiver sensitivity and gain factors are measured. NF measures the overall noise that is introduced due to the analog signal conditioning in the front end and is discussed in depth in the next subsection. Various RF front ends exist that are suited to operate for L-band signals and in many cases designed specifically to support GNSS signals. A popular chip named MAX2769 is often used by many researchers; as well as incorporated into various industrial products (Weber and Bremer 2008; Tsai et al. 2021; Huang et al. 2021). Maxim (a chip manufacturer) also offers RF development kits, though they were not considered for the receiver prototype developed in this dissertation, as there were assembly and integration concerns of the RF front end with the FPGA, where the reference clock signals were prone to EMI/EMC.

A new and upcoming company at the time of hardware procurement was NTLab. The chip NT1065 developed by them can support up to four antenna inputs. The FPGA Mezzanine Card (FMC) compatible development kit was purchased and integrated with the FPGA platform, ensuring high signal integrity between the two platforms. The differential signals and clocks are also an added advantage (protected from EMI/EMC

effects). NTLab also provides an extensive GUI (see Figure 3-3) that lets the user control the reference clock frequency, sampling frequency, gain and many other essential parameters.

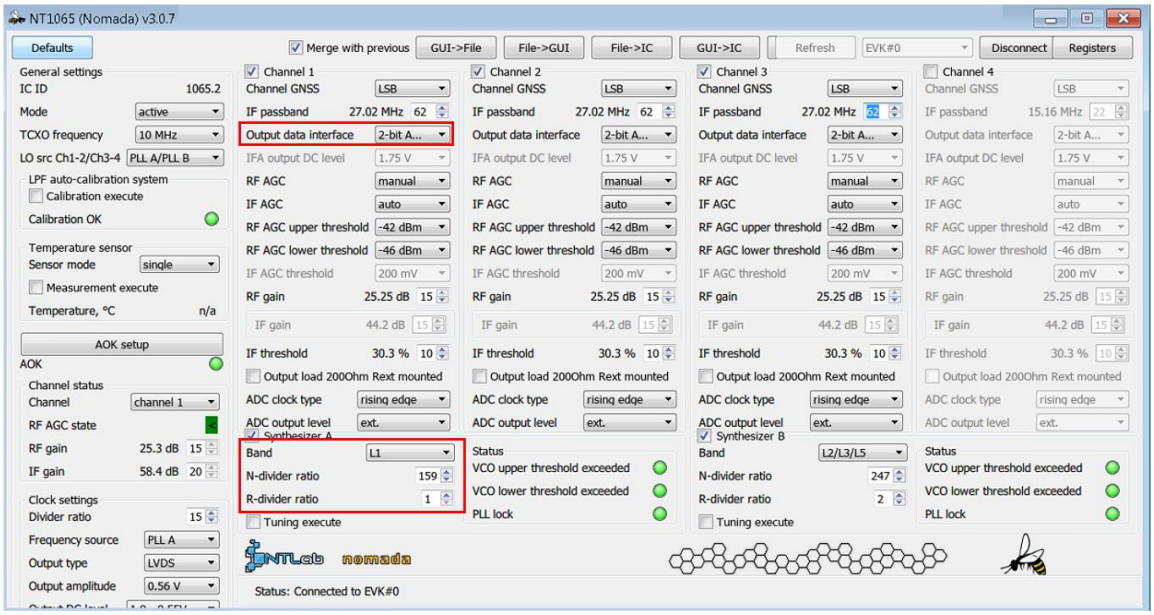


Figure 3-3 NTLab: Front end configuration GUI

3.1.1.3 Noise figure of RF front end

An RF front end consists of components such as amplifiers, filters, attenuators, and mixers, where each of them are associated with fundamental parameters of gain, NF and third order intercept. These parameters and the order the components are arranged in determine the gain, sensitivity, and the dynamic range of the receiver. Equation 3-1 represents the overall gain of the front end.

$$G_T = G_1 G_2 \dots G_N \qquad \text{Equation 3-1}$$

where G_T is the overall gain and G_1, G_2 and so on are the gains of each of the components in the front end.

In Equation 3-2, F_T is the overall NFof the front end and F_i represents the noise figure of each component (Borre et al. 2007). Quantifying the noise added during the process of signal conditioning determines the SNR of the resulting signal. It should be noted that the contribution of the first component to the overall NF of the system is the highest and hence this device must be cautiously selected to have ideally a very low NF.

$$F_T = F_1 + \frac{F_2 - 1}{G_1} + \frac{F_3 - 1}{G_1 G_2} + \frac{F_N - 1}{G_1 G_2 \dots G_{N-1}} \qquad \text{Equation 3-2}$$

In Equation 3-3, Q_{3i} is the third order intermodulation product of each individual component and the overall product is represented as Q_{3T}. (Dennis et al. 1996). A compromise between parameters NF and Q_{3T} must be found since a low NF and a high intermodulation factor is desired. According to NTLab's datasheet for the NT1065 evaluation kit (NTLab), the overall NF of the front end is 3.8 dB and has a total gain of 90 dB.

$$Q_{3T} = \frac{G_T}{\frac{G_1}{Q_{31}} + \frac{G_1 G_2}{Q_{32}} + \dots + \frac{G_1 G_1 \dots G_{N-1}}{Q_{3N}}} \qquad \text{Equation 3-3}$$

3.1.1.4 FPGA processing platform

Various vendors, such as Xilinx/AMD, Intel (formerly Altera), and Microchip/Actel, manufacture FPGA platforms. Amongst these manufacturers, Xilinx/AMD has emerged

as one of the leaders in manufacturing and developing advanced FPGA platforms. In addition, they also offer established development tools, such as Vivado, HLS, SDK and Vitis that aids the design process. The depth of resources and extent of support offered by Xilinx is also commendable. Moreover, due to the large user base across the world, there is a strong community in terms of bug fixes, work arounds or open-source reading material. Due to the well-established Xilinx legacy in the GNSS laboratory at York University as well as experience with the development tool and environment, the Xilinx/AMD development kit was chosen as the processing platform.

To determine the type of Xilinx development board that would be used, two essential requirements were considered:

1. The board must support FMC pins. FMC pins allow easy assembly of the RF front end with the FPGA platform without compromising signal integrity.
2. The board must include hard-core processors (SoC). Xilinx offers a series of boards marked under the Zynq series where a pair of ARM processors (hard-core processor) are present in addition to the FPGA fabric. Hard-core processors are versatile and have more features than a soft-core processor such as higher operating frequency, dedicated communication channels and interconnects (Subramaniam and Fayadh 2022).

Given the above criteria, the Xilinx development board ZC706 (Figure 3-4) was deemed to meet the two requirements listed above. It has been well established in the research community that GNSS signal processing demands for a large number of correlators and

multipliers. Hence it is of importance to ensure that enough logic area is available to implement such operations. ZC706 has 350K logic cells, 19.5 Mb block RAM and 900 DSP slices. The development board also has two FMC connectors, SD card interface and 1 GB DDR3 memory. These features are important considering not just the development of the current receiver prototype but also the future versions of the product.

Figure 3-4 ZC706 processing platform with the FPGA chip concealed under the cooling fan.

3.1.2 Signal data rates

As mentioned in the scope definition in Chapter 1, the focus of this dissertation is to develop a GNSS-R receiver using GPS L1-C/A signals only. This subsection lists the key data rates and frequency settings that are used for receiver implementation. The RF front end has several settings for each of the individual input channels. The sampling

frequency and IF frequency selected here are important to note as they will be used during tracking and decoding of navigation data. The parameters listed in Table 3-2 show all the frequency settings for the receiver implementation. The minimum sampling frequency that can be set with NT1065 is 53 MHz. This number is fairly high when compared to other popular RF front ends such as MAX2769. Higher values of sampling frequency translates to increased number of samples per millisecond and this inadvertently leads to larger number of multiplications per millisecond. If the system clock frequency meets the Nyquist criteria (of 106 MHz), there is no impact of higher sampling frequency on the FPGA's processing capability. In this GNSS-R receiver implementation, a system clock frequency of 200 MHz is used. The only negative consequence of a high sampling frequency is when raw IF data needs to be recorded, larger memory allocation is required.

Table 3-2 Selected frequency settings for RF front end

Signal of interest	GPS L1 C/A-code (1575.42 MHz)
Number of antenna inputs	2
Input GPS signal resolution	1 bit
Sampling frequency	53 MHz
Intermediate Frequency (IF)	-14.58 MHz

The IF value used is the default value listed in the configuration file for the front end. The value is noted since the local signal generators implemented in the FPGA operate at the

same IF frequency, i.e., in this case at -14.58 MHz. The IF value is also flexible and can easily be modified using NTLab's provided GUI as mentioned earlier. The default values were used as is, since there were no issues or concerns related to the settings.

3.1.3 Spectrum analysis

Prior to testing the FPGA algorithms, a considerable amount of time and effort was spent on ensuring the integrity of the data flow from the antenna to the FPGA input pins is solid. Confirming that the GNSS signal is correctly being captured and sampled by the RF front end is an important step.

The minimum sampling frequency of the GNSS signal that is possible using one of the preset configuration files provided by the front end manufacturer is 53 MHz. Sampling frequency of 53 MHz is a high data rate (53,000 2-bit words per millisecond) requiring high data throughput rate. An FPGA module was designed and implemented for saving raw GNSS data. Bursts of input data from the front end is passed through a direct memory access (DMA) module, which is then transferred to an external SD card.

The raw signals (L1 band) saved on the SD card are retrieved and analysed in the frequency domain (shown below in Figure 3-5). Strong low frequency signals are observed, but of interest is to verify if a strong signal exists at the expected IF. According to the preset configuration, the IF is set to -14.58 MHz and a peak is observed.

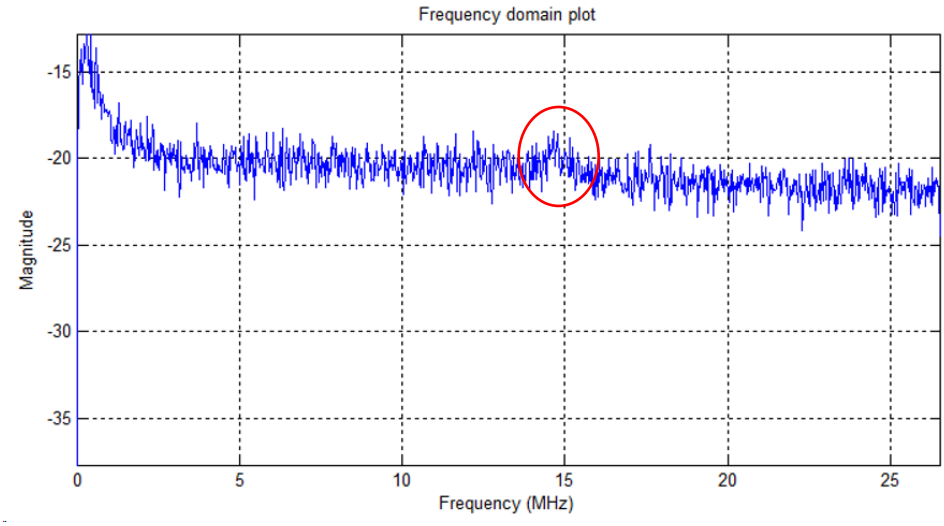


Figure 3-5 Frequency domain plot of raw GNSS signals (dB vs MHz), with -14.58 MHz peak, captured by the RF front end

To verify and validate the integrity of the raw GNSS signals, the recorded data set is processed using two software GNSS receivers (both PC-based). The results were first verified using Kai Borre's software-defined GNSS receiver. The acquisition (detection of visible satellites) results are shown in Figure 3-6 where various satellites such as PRN 1, 14, 17, 20, 22, and 31 are detected.

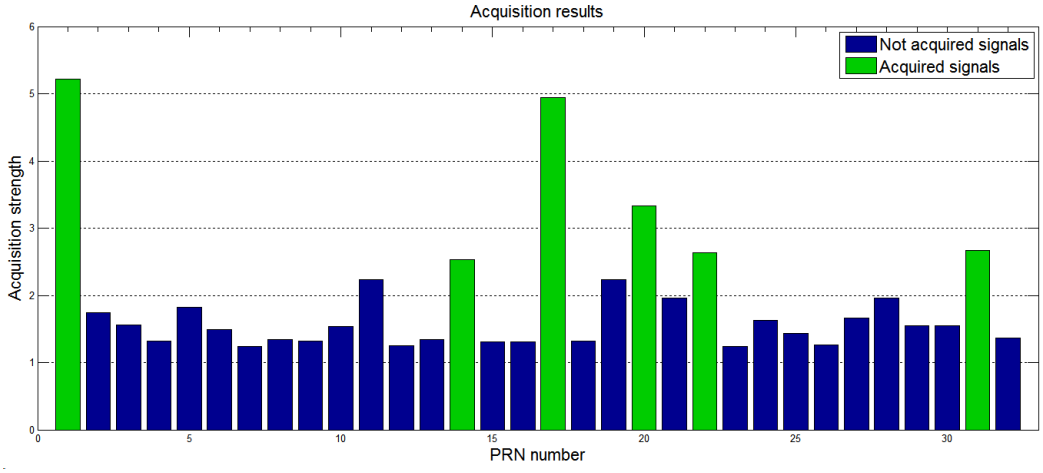


Figure 3-6 Acquisition results using Borre SDR (Borre et al. 2007) for GNSS applications

In Figure 3-7, the tracking results with the same dataset is shown. The top-right plot of Figure 3-7 clearly shows high correlation and bit flips occurring in the embedded navigation message. Located in the top-left is the scatter plot indicating the correlation values for both, the inphase and quadrature phase of the signal. It should be noted that higher correlation values in the inphase arm and lower correlation values in the quadrature arm are ideal characteristics and reflect successful signal tracking (Borre et al. 2007). The other subplots show performance of the various filters used in the delay lock loop (DLL), frequency lock loop (FLL) and phase lock loop (PLL). Smoother discriminator output indicates fine tracking of the signal.

The same dataset is processed using the C-based software GNSS receiver that was developed as a precursor to this research (Guruprasad 2015). The result of this test is shown in Figure 3-8. Only the acquisition strength of the satellites that are detected are

plotted. Although the dataset is very short (~500 milliseconds), it is sufficient to perform acquisition and a few milliseconds of tracking the satellite signal successfully. By comparing both the test results it is concluded that the raw GNSS signals get correctly sampled and recorded by the FPGA and the signal integrity of the data path is not compromised. The variation in the plotting format presented in Figure 3-6 and Figure 3-8 is due to varying scripts used to generate the results. The Figure 3-6 is plotted using Kai Borre's open-source script whereas Figure 3-8 is plotted using a custom script that was developed during this research. Also, the acquisition strengths of the corresponding PRNs in the two figures vary given two different software tools and FFT libraries were used to compute the results.

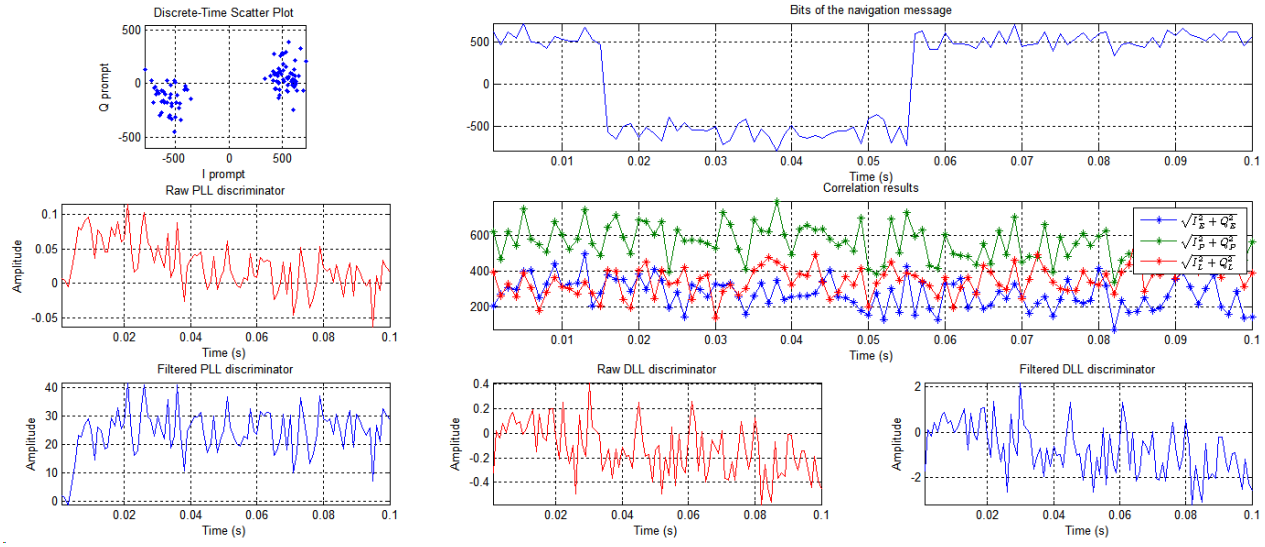


Figure 3-7 Tracking results for PRN 1 using Borre SDR (Borre et al. 2007)

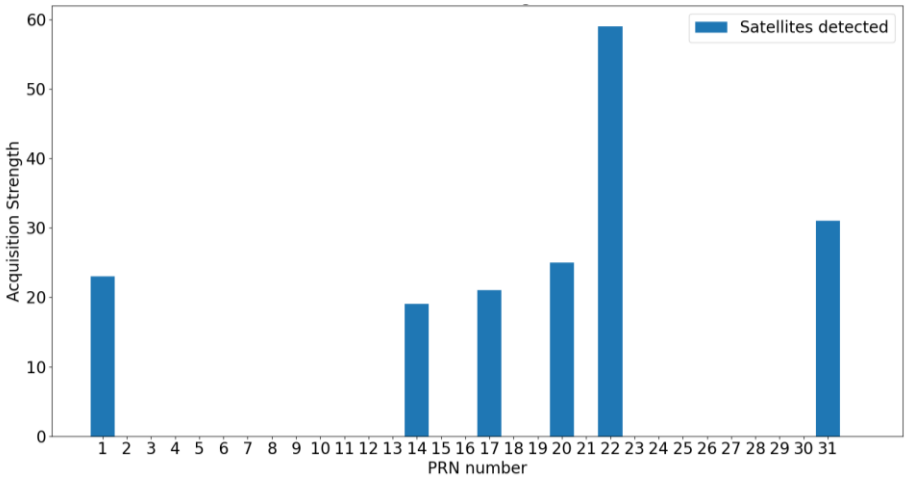


Figure 3-8 Acquisition results using developed software GNSS receiver

3.2 GNSS-R receiver design and implementation

The development of the receiver went through multiple iterations of design, simulations, verification, testing and validation, and this section encapsulates all of the low-level intricacies of the GNSS-R receiver prototype developed followed by the implementation details.

3.2.1 GNSS-R receiver architecture overview

The architecture of the GNSS-R receiver prototype is shown in Figure 3-9. The RF front end receives signals from the zenith and nadir antennas. These signals are then digitized and processed in the FPGA-based tracking channels. The direct and reflected signals are handled independently and are managed by dedicated ARM processors (ZC706 is equipped with dual-core ARM processors).

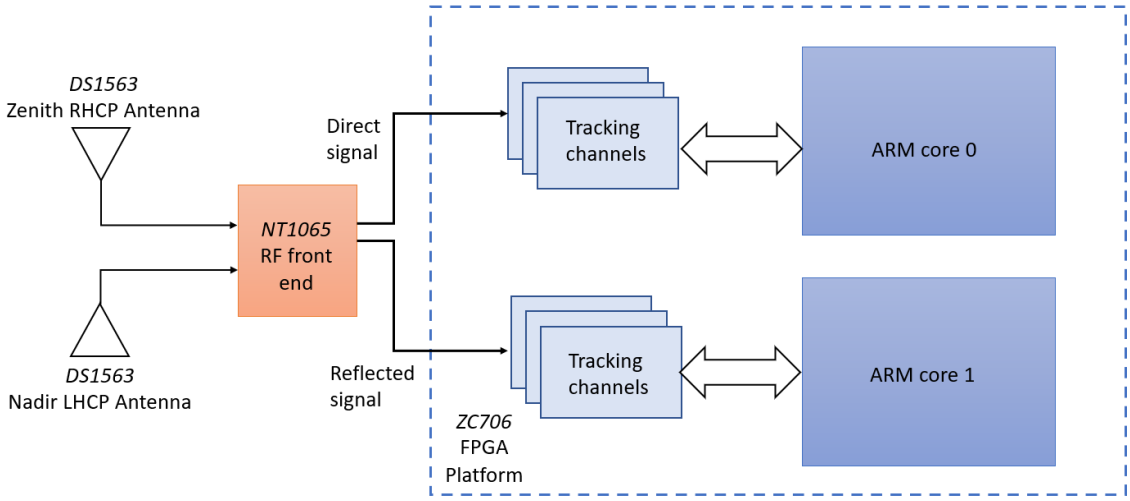


Figure 3-9 GNSS-R receiver architecture and implementation

The first step the receiver performs is acquisition. All the "visible" satellites are detected in this stage, along with essential information such as Doppler frequency, code phase and psuedo-random noise (PRN) number of the incoming signal. Referring to Figure 3-9,

it is worth noticing that the process of acquisition and tracking is combined. For this research, tracking blocks are configured to also perform acquisition, since plenty of common hardware modules are used between the processes. Instead of having multiple hardware modules occupy additional FPGA resources, with the help of configuration registers, at the end of successful acquisition, the same modules transition into tracking mode. The implementation details along with the architecture of the acquisition and tracking channel is discussed in Section 3.2.3.

3.2.2 Hardware/software partitioning strategy

Given the computational load of GNSS-R receivers, ASIC-based receivers are often preferred due their ability to function at higher speed. The main reason being that conventional receivers have numerous and dedicated hardware correlators. Software receivers (based on computers or microprocessors) on the other hand, follow sequential processing, which is a significant limitation for which they are almost never considered for real-time GNSS receiver applications (Kaplan and Hegarty 2013).

Several researchers have taken the hardware and software co-design approach to enhance the software receiver's processing abilities (Fridman and Semenov 2013; Wang 2017; Galley et al. 2018; Unwin et al. 2021b). A combination of FPGA and a microprocessor is a common approach and several FPGA manufacturers market it as a System-on-Chip (SoC), where the hardcore processor is connected to the FPGA fabric with a dedicated communication channel.

The development board used for this research (ZC706), contains two ARM processors. The processors are used merely as a tool to configure the FPGA-based modules at start-up. The control register configures the FPGA IPs based on the user-controlled settings. The local signal generators are also set based on the IF configuration in the front end and to preserve flexibility. These values can be easily set or modified via the processor/software. Figure 3-10 illustrates the distribution of the registers and cores that control the various operations across the FPGA fabric. The advanced extensible

interface (AXI) interconnect is a high-speed bus protocol that forms an essential bridge between the FPGA and the processor cores.

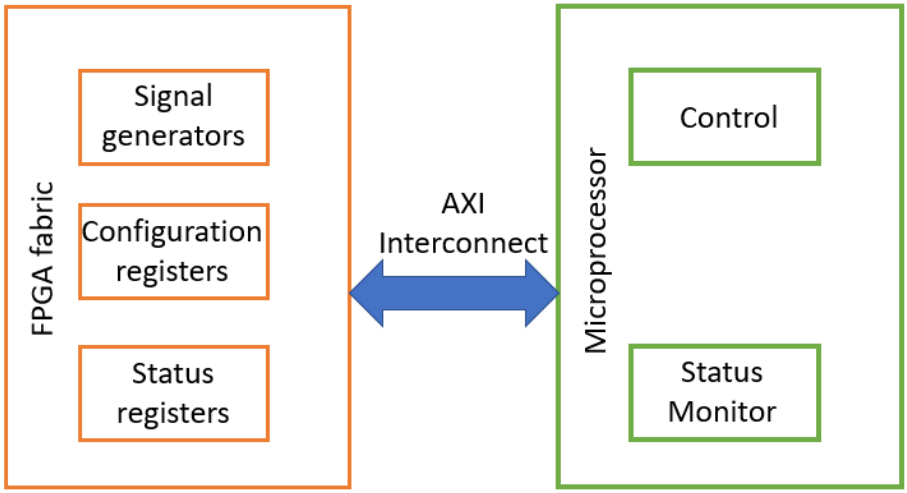


Figure 3-10 Software/hardware partitioning where FPGA is considered as hardware and microprocessor is the software component

3.2.3 1-bit quantization

The input GNSS data is a 2-bit signed signal, but it is simplified to either a 0 or a 1 when sampled in the FPGA. A 2-bit signal represents one of the four levels of [-3, -1, 1, 3] as shown in Figure 3-11. All samples below zero are assigned the value 1 and all samples above 0 are assigned the amplitude of 1 as shown in Figure 3-11. In other words, only the sign bit is retained. Although the motivation to convert a 2-bit signal to 1-bit input signal has been discussed in section 2.4.5, a few additional details are:

1. The conversion process simplifies the multiplication in the FPGA significantly, as multipliers are replaced with an XOR gate.
2. Fewer FPGA resources are required with this technique, e.g., local carrier signal generators do not require any memory blocks or lookup tables such as RAM.
3. The compact design makes it highly scalable.
4. The tracking channels also operate as acquisition channels.

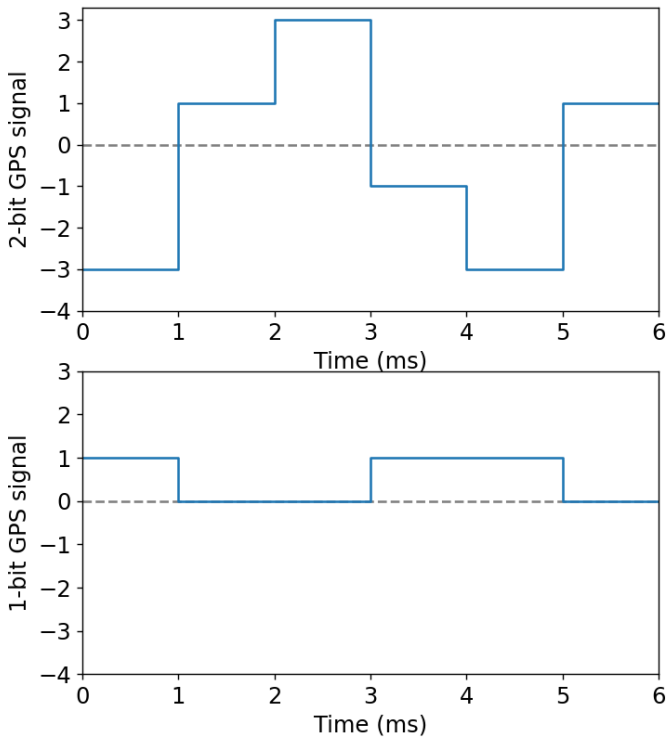


Figure 3-11 Conversion of 2-bit GPS signal to 1-bit GPS signal

As mentioned before, the effect of downsizing the bit width is known as quantization loss. For a signal resolution of 1-bit, the quantization loss is -1.96 dB (Kaplan and Hegarty 2013). The effect of this loss becomes apparent when the overall gain of the

receiver is computed and a degrading impact on the receiver sensitivity is observed (discussed in Subsection 3.2.5).

3.2.4 Acquisition and tracking architecture

In software receivers, the most popular method of acquisition is the parallel code phase search method, where a data sample length as short as 1 millisecond is sufficient to determine the Doppler frequency of the input signal (Borre et al. 2007). As mentioned in Chapter 2, the algorithm needs to step through all possible PRN sequences until it detects a satellite. Although the frequency domain search is relatively faster (in units of time), it must still function sequentially for all code phases and then repeat the process for all possible satellite signals.

In this research, FPGA-based acquisition and tracking is performed in parallel. Since the focus is to use GPS L1-C/A signal, 32 acquisition channels are set up for each of the possible 32 GPS satellites that the receiver can potentially detect. Each dedicated channel functions independently and requires only certain initialisation effort from the processor side. The registers set by the processor are provided in Table 3-3.

Table 3-3 Initialisation of tracking channels

Parameter	**Module**	**Comments**
Carrier phase increment	Carrier signal generator	Determines carrier signal frequency
Code phase increment	PRN sequence generator	Determines code phase sampling frequency
PRN	PRN sequence generator	Channel initialisation

The carrier phase increment and code phase increment are computed using the following equations:

$$\phi_{carr} = \frac{f_{IF}}{f_{sys}} \cdot k \qquad \text{Equation 3-4}$$

$$\phi_{code} = \frac{f_{C/A}}{f_{sys}} \cdot k \qquad \text{Equation 3-5}$$

$$k = \frac{f_{sys}}{f_s} \cdot 2^{32} \qquad \text{Equation 3-6}$$

where ϕ_{carr} is the carrier phase increment value, ϕ_{code} is the code phase increment value, f_{IF} is the intermediate frequency of the signal, f_{sys} is the system clock frequency that is used as the base clock across all the FPGA modules, $f_{C/A}$ is the PRN code frequency, f_s is the sampling frequency and k is the clock factor that must be considered when computing the phase increments. The width of the increment register is 32 bits and hence all the values must be scaled to a value that can be represented by a 32-bit register. In this dissertation, using equations Equation 3-4 and Equation 3-5, $\phi_{carr} = 0x04F0F7EC$ *and* $\phi_{code} = 0x465D52E8$ where $f_{IF} = -14.58\,MHz$, $f_s = 53\,MHz$, $f_{sys} = 200\,MHz$ and $f_{C/A} = 1.023\,MHz$.

At the end of acquisition, the user is informed (relayed via UART) of all the detected satellites, along with the Doppler frequency each was detected at as well as the code phase of the C/A code. Figure 3-12 represents the functional block diagram of the design.

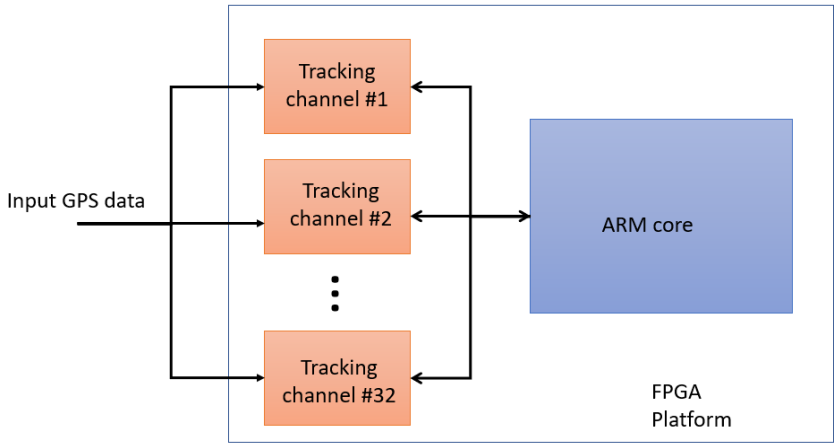


Figure 3-12 Functional block diagram of acquisition/tracking channels in the developed GNSS-R receiver

Each of the tracking channels shown in Figure 3-12 contains dedicated correlators, signal generators and accumulators. Figure 3-13 presents the architecture of a single channel where various low-level modules and components are presented and are discussed in detail in the subsequent subsections.

3.2.4.1 Carrier replica

Referring to Figure 3-13, it can be seen that the local carrier signals with varying frequencies are generated in parallel. The number of carrier replicas generated in parallel depends on the Doppler frequency search range and the total number of Doppler bins required. In Figure 3-13, only three instances of carrier replicas are shown; however, in the current FPGA implementation a total of 20 Doppler bins are instantiated.

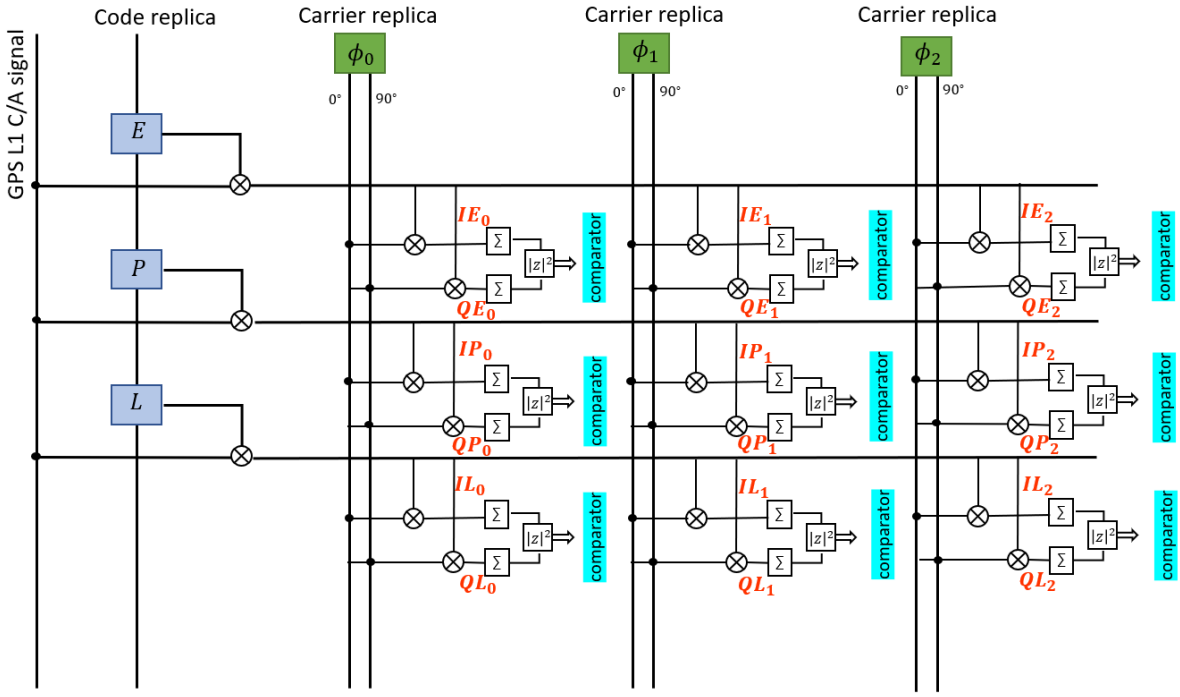


Figure 3-13 Acquisition and tracking architecture

Referring to Equation 3-4, each carrier signal generator is initialised with a carrier phase increment value ϕ_{carr}. Considering that each tracking channel has 20 Doppler bins (or 20 carrier replicas), they are each identified and initialised with a carrier phase of ϕ^0_{carr}, ϕ^1_{carr}, and so on. If a carrier replica initialised with the phase value ϕ^n_{carr} generates a signal of frequency $f^n_{Doppler}$, then the Doppler step size $f_{DopplerStepSize}$ is computed using Equation 3-7.

$$f_{DopplerStepSize} = f^{n+1}_{Doppler} - f^n_{Doppler} \qquad \text{Equation 3-7}$$

The Doppler bin size is determined using a standard estimating equation of $2/3T$ where T is the integration period resulting in 667 Hz (Kaplan and Hegarty 2013). The equation

suggests that for a signal with nominal strength, the Doppler bin size must be a minimum of 667 Hz. In the receiver implementation, a Doppler bin of 500 Hz is chosen to simplify computations. Moreover, several open-source receivers (Borre et al. 2007) also successfully use the standard bin size of 500 Hz for coarse acquisition. In the developed GNSS-R receiver, a Doppler frequency search between the frequency range -5 KHz to 5 KHz is conducted with a step size of 500 Hz resulting in 20 Doppler bins. Therefore, a total of 20 carrier signal generators are instantiated and function simultaneously in parallel.

Although the initialisation and configuration of each of these signal generators seems like a tedious task, it has been grossly simplified. Instead of every carrier signal generator receiving corresponding ϕ_{carr}^{n} information, only the ϕ_{carr}^{0} and $f_{DopplerStepSize}$ are provided by the processor. Configuration logic within each channel initialises the remaining signal generators (ϕ_{carr}^{1} to ϕ_{carr}^{19}) based on Doppler step size and initial carrier phase information.

The carrier signal generator produces only two 1-bit signals, where one is the sine signal and the other is the 180-degree phase shifted version of the same sine signal (cosine). The signal generator uses the concept of a numerically controlled oscillator (NCO). A 32-bit accumulator is used where the value is incremented every system clock pulse, based on the initial phase (ϕ_{carr}) information provided by the processor (shown in Figure 3-14). When the accumulator reaches the halfway point, the most significant bit (MSB) switches from 0 to 1. At the peak, the register overflows, and the MSB value is

reset to the initial value of 0 and the process repeats. Such an implementation generates a 1-bit sine signal. Similarly, cosine signal can also be extracted from this design. Equation 3-8 and Equation 3-9 show the derivation where $accum$ is a 32-bit register with MSB being the 31st bit (index starts at 0). The frequency of the generated signals can be controlled by how quickly the accumulator ramps up to the peak. It should be noted that the NCO requires no assistance from the software except for the initial information regarding the frequency of the signal it must generate.

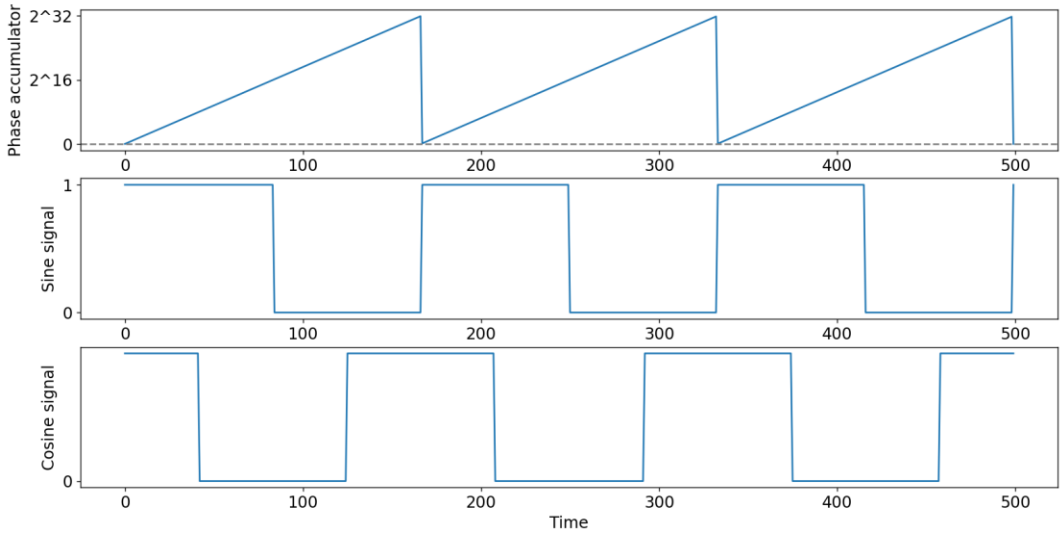


Figure 3-14 Phase accumulator driving the frequency of the sine and cosine signals

$$sinvalue = \sim accum[31] \qquad \text{Equation 3-8}$$

$$cosvalue = accum[31] \sim^{\wedge} accum[30] \qquad \text{Equation 3-9}$$

Although it is straightforward to use a LUT for sine and cosine functions, multiple LUTs would have to be instantiated for every signal generator (20 LUTs in this case). In

addition, if the step size or the range of Doppler frequency search is modified, the processor will have to overwrite LUT contents sequentially to reflect the changes. Also, no BRAMs is required in this method. Hence NCOs are used in the developed GNSS-R receiver, where only Doppler step size and the initial carrier phase is to be defined for initial configuration.

3.2.4.2 PRN signal generator

Each satellite transmits a unique PRN code and this code is repeated every millisecond (Borre et al. 2007). In the GPS signal ICD-200G (Dunn 2012), technical details pertaining to the generation of the C/A-code sequence is provided. Using the information, a PRN signal generator module is implemented in the FPGA.

The signal generator uses a linear feedback shift register (LFSR) to generate the 1023-bit PRN sequence. Certain registers and bits (defined in (Dunn 2012)) become part of a polynomial equation and are fed back into the shift register. After 1023 counts, the sequence repeats itself. As seen in Figure 3-13, the PRN generator produces three versions of the PRN sequence in real-time: Early (E), Late (L) and Prompt (P) with a difference of 0.5 chip from each other. Since the difference is of half a chip, 2046 code phases need to be searched for instead of 1023. At the end of the integration time of 1 millisecond during acquisition, if no satellite is acquired after comparing the correlation result with a threshold, a delay of half a chip is introduced and the correlation process is repeated. Therefore, for the next millisecond of acquisition, correlation will be performed

with the half-chip shifted version of the same PRN. The process of adding a delay to the code phase is repeated until an extremely high correlation result surpassing the threshold is obtained. If all 2046 code phases are checked and no correlation peaks are observed, the channel concludes operation and deems the PRN or corresponding satellite to be absent.

3.2.4.3 Mixers

To extract the navigation data, the input GPS signal is mixed with the local carrier signal and the local PRN sequence. This operation is performed by using a one-dimensional XOR gate. 1-bit multiplication operation in the FPGA is equivalent to using an XOR gate.

3.2.4.4 Accumulators

Six 32-bit accumulators have been included per tracking channel named In-phase-Early (IE), In-phase-Prompt (IP), In-phase-Late (IL), Quadrature-Early (QE), Quadrature-Prompt (QP), Quadrature-Late (QL). They each collect samples accumulatively until the end of the integration period and are reset to zero for the next iteration. At the end of the integration period, the total power in the I and Q arms are examined using Equation 3-10:

$$Total\ correlation\ strength\ (P) = I^2 + Q^2 \qquad \text{Equation 3-10}$$

During the acquisition stage, only the total power in the prompt arm is inspected. The other products (E,L) become essential during tracking.

3.2.4.5 Comparators

At the end of an integration period for a particular code phase delay, the full range of Doppler frequency search is conducted. This search results in many accumulated values that must be compared against a threshold at the end of the integration period - in this case 1 millisecond. Given that the system clock frequency $f_{sys} = 200\,MHz$ the time period of each clock cycle can be computed using Equation 3-11.

$$T_{sys} = \frac{1}{f_{sys}} = 5\,nanoseconds \qquad \text{Equation 3-11}$$

Therefore, to sequentially compare each correlation result against the threshold register, the receiver would require 20 clock cycles or $20 \cdot 5 = 100\,nanoseconds$. Although, $0.1\,\mu seconds$ is a small duration already, it is further optimised to finish the comparison within a few clock cycles. The enhanced peak search is demonstrated in Figure 3-15, where only eight instances are shown as an example. The parameter P indicates the resulting power computed in Equation 3-10 at the end of an integration period. Every clock cycle, comparisons are made and within five clock cycles (25 nanoseconds), the peak is detected. The peak value is then compared against the threshold. If the peak value is greater than the threshold, tracking mode commences. If not, the search for the satellite signal continues.

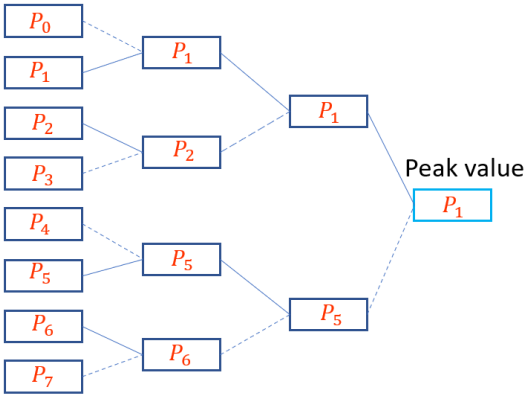


Figure 3-15 Optimised comparator structure to find the peak correlation strength

3.2.5 Acquisition results using simulated signals

Shown in Figure 3-16 is the test setup used during the entire course of development and validation of the GNSS-R receiver. For in-lab testing, a Spirent GSS9000 GNSS simulator is used as RF input to the GNSS receiver. Although the receiver can process two antenna signals (zenith and nadir) simultaneously, the simulator only has one output port and therefore laboratory tests were conducted only for the zenith tracking channels in the receiver.

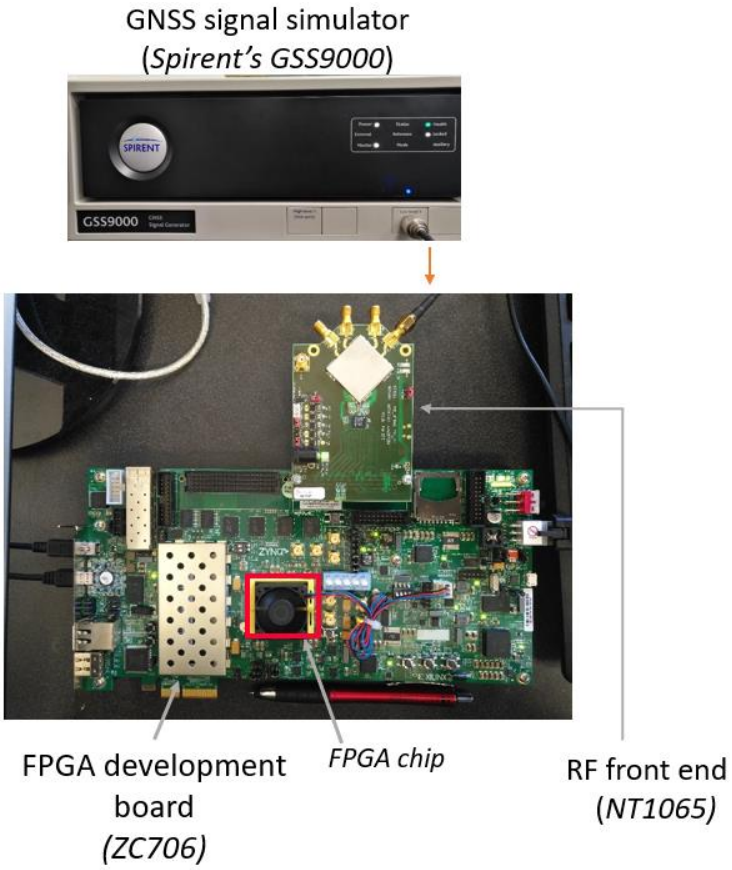


Figure 3-16 GNSS receiver test setup with simulator

The GNSS signal simulator permits the repeated generation of controlled GNSS signals in the lab environment. The output of the GNSS signal simulation is directly fed into one of the input ports of the RF front end. The main objective of the test is to determine if the acquisition module successfully detects satellites. Figure 3-17 demonstrates the results of three different scenarios, where the simulated PRN 1 signal strength is the only varying parameter. For comparison purposes, the results for acquisition of PRN 2 and PRN 3, are also recorded and presented; however, they are both not visible to the receiver and are

deemed to be absent during this step. The real-time acquisition sequence in the FPGA is started and the following parameters are noted down: PRN number detected, maximum correlation strength, Doppler frequency and code phase. The processor reads these registers after the acquisition stage is complete and presents it to the user.

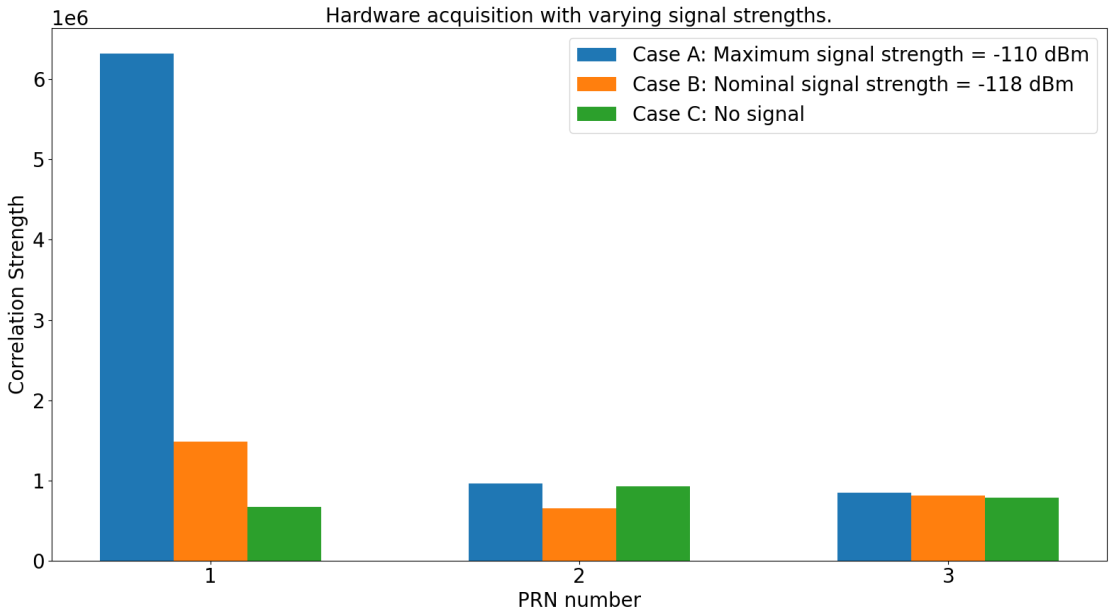


Figure 3-17 FPGA-based acquisition results

Referring to case A, the PRN signal strength is set to the maximum value allowed by the simulator (-110 dBm). The results indicate that the correlation strength of PRN 1 is significantly higher than PRN 2 and 3 indicating the successful detection of PRN 1. For case B, the signal strength is reduced to a nominal strength of -128 dBm. The correlation strength is more than twice the correlation strength of an absent satellite, indicating that the acquisition module correctly identifies the visible satellite. In the last case C, is when PRN 1 is no longer visible to the receiver. As the results indicate, PRN 1 is now no

longer detected in the FPGA-based receiver. The acquisition module results reflect changes in the PRN signal strength as expected and hence the functionality of the acquisition implementation is successfully validated. The validation also indicates proper functioning of the carrier signal and code replica generators.

3.2.5.1 Acquisition time

Time-to-first-fix (TTFF) is the time taken by the GNSS receiver to compute a navigation solution. Since the prototype receiver is devoid of computing the navigation solution (as the research focus is signal acquisition), time-to-first-acquire (TTFA) is calculated. TTFA is the time taken by the receiver to complete the acquisition step. The receiver upon a power cycle (reset), has no access to information related to ephemeris, location, or time. This process is known as a cold start. The receiver must acquire and track all the visible satellite signals to compute position and time. During a cold start, the TTFF can take several seconds or several minutes. For the GNSS-R receiver presented in this research, since no ephemeris is stored, a new set of data are processed every millisecond and hence there is always a cold-start after power cycling.

The maximum time required by the receiver to finish searching all visible GPS satellites is small, only 2.046 seconds. Table 3-4 illustrates the parameters that are used to compute the overall acquisition time. The number of possible C/A-code phases is 1023 with a chip delay of 0.5, therefore a total of 1023 x 2 = 2046 phases are stepped through. The Doppler bins are implemented in parallel and therefore the time to step through the

Doppler-code search space depends only on the number of C/A-code phases that must be checked - in this case 2046 phases. Since the integration period is 1 millisecond, to search a total of 2046 phases, 2.046 seconds would be required per channel.

Table 3-4 Calculation of TTFA

C/A-code			Doppler (Hz)				Per channel
Number of phases	Chip size	**Total number of phases**	Search start	Search end	Search step size	Number of bins	Total time (ms)
1023	0.5	**2046**	-5000	5000	500	20	**2046**

3.2.5.2 Receiver sensitivity

The sensitivity of the receiver can be defined by computing the carrier-to-noise density ratio denoted by C/N_0 and the SNR for the various processing stages in the receiver. The C/N_0 value determines the signal quality that is independent of any receiver operation such as acquisition, correlation or tracking. The noise density N_0 in dBW/Hz is determined using Equation 3-12:

$$N_0 = 10 \log_{10} kT_{eff} \qquad \text{Equation 3-12}$$

where $k = 1.38 \times 10^{-23}\, W/K - \text{Hz}$ (Boltzman's constant) and T_{eff} is the effective system noise temperature computed using the Friis formula as shown in Equation 3-13:

$$T_{eff} = T_A + (F - 1)T \qquad \text{Equation 3-13}$$

where T_A is the antenna temperature (130 K), T is the ambient temperature (290 K) and F is the front end noise figure (in this case 3.8 dB). Using Equation 3-12 and Equation

3-13, for this implementation, $N_0 = -201.31\ dBW/Hz$. Therefore, for a carrier signal strength of $-158\ dBW$, the C/N_0 for the receiver is computed as

$$\frac{C}{N_0} = -158.5 - (-201.31) = 42.81\ dB - Hz \qquad \text{Equation 3-14}$$

The above computed C/N_0 is in the nominal range of a typical receiver (Joseph et al. 2018). The SNR is another metric that must be computed to determine the performance of the receiver. Unlike C/N_0, SNR of a receiver is affected by the various signal processing stages present in the receiver. To compute the SNR, the bandwidth of signal must be considered. Equation 3-15 is the total noise power after the down conversion and signal conditioning performed in the front end.

$$N_0^{Watts} = N_0 \cdot BW \qquad \text{Equation 3-15}$$

The bandwidth denoted by *BW* is known to be 3 MHz, hence the total noise power is $-96.99\ dBm$. For a carrier signal strength of $-158.5\ dBW$ or $-128.5\ dBm$, the SNR after the down conversion in the front end is $-21.95\ dB$. The SNR is expected to be negative as the signal of interest is well below the noise floor.

To analyse the processing gain of the receiver, the actual coherent gain is estimated using Equation 3-16.

$$Actual\ coherent\ gain = Ideal\ coherent\ gain - |\Delta| \qquad \text{Equation 3-16}$$

where $|\Delta|$ are the implementation losses of the receiver. In this case, the *ideal coherent gain* is estimated to be 33.10 dB, whereas the total implementation loss accounting for the quantisation loss, frequency mismatch, IF filtering loss and code alignment loss totals

to –3.26 dB. With these values, the *actual coherent gain* results to 29.83 dB. Given that the SNR of the input signal to the receiver was –21.95 dB, the resulting SNR after the coherent integration is now 7.89 dB. The SNR ratio is 2.48, which indicates that the signal power is 2.48 times the noise power. The resulting SNR is sufficient to detect available GNSS signals.

3.2.5.3 Resource utilization

FPGA resources can be broadly categorised as LUTs, registers, RAM and DSP slices. Each FPGA chip has a pre-defined resource limitation. For the GNSS-R receiver prototype, the resources required for one acquisition channel including the local signal generators is listed in Table 3-5. Considering the compactness of a single acquisition channel, a much smaller and low-cost FPGA chip can indeed be used. For example, the Artix-7 FPGA chip that comes with the evaluation kit Arty (manufactured by Xilinx, costs ~$160) can comfortably fit sixteen tracking channels of which, one half of the channels can be used for direct signals whereas the other half for reflected signal tracking. This evidence solidifies the confidence in building an inexpensive GNSS receiver using COTS hardware.

Table 3-5 Resource utilization in the FPGA

	LUTs	**Registers**	**RAMs**	**DSPs**
Available	218600	437200	545	900

Acquisition Channel	812	1180	0	18
Resource Utilization	0.3%	0.2%	0%	2%

For context, the developed GNSS-R receiver is also compared with another receiver presented in (Sauriol et al. 2007). Although the plot in Figure 3-18 presents similar usage of registers, there is noteworthy difference in the LUT usage of the channel. Another researcher (Tran et al. 2017) presents that each channel implementation uses only 440 LUTs. While these approaches are commendable and represents nearly half of what is used in this research, there is no information on how many RAMs or DSP slices were utilized. Additionally, navigation data decoding is not performed in the FPGA and cannot be fairly compared with the current research (and hence not used for comparison). It should also be noted that there is little to no information available about the reflectometry receiver resource utilization from any of the missions listed in 2.4.2.

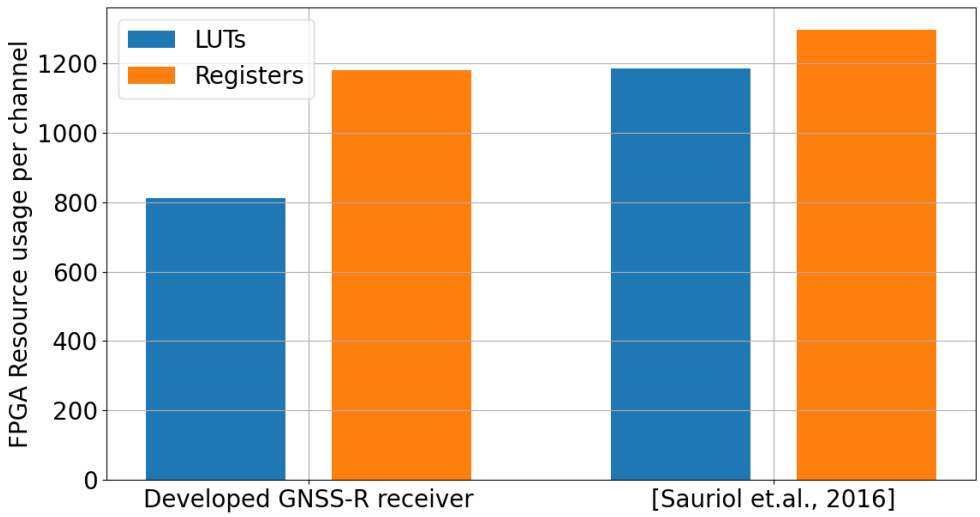


Figure 3-18: Comparison of FPGA resource usage per channel

3.2.6 Tracking loop realization in reconfigurable hardware

When a peak is found during acquisition that exceeds the threshold level, the channels switch to tracking mode automatically (without any external intervention such as a processor or a user). The Doppler frequency and the C/A-code phase is known to the channel and these parameters are locked. Unlike acquisition, during tracking, only one carrier replica per channel is generated. Moreover, three code replicas are generated that are 0.5 chips apart from the adjacent replica. The only other difference is that the accumulated values are fetched by the processor periodically to compute corrections to the carrier and code signal generators. These modifications can be noticed in Figure 3-19, where only the highlighted area is active, while the masked area is dormant. In the FPGA, no design changes take place when tracking mode starts, only certain configuration changes occur.

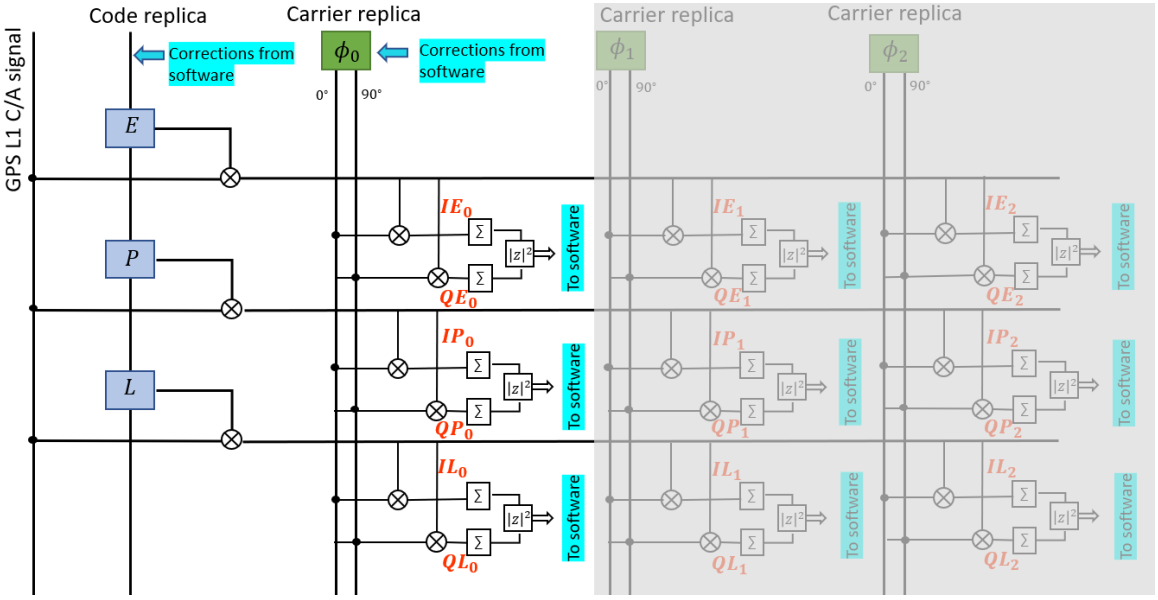


Figure 3-19 Tracking architecture and implementation

During tracking of a signal, the processor plays a significant role in this implementation. The incoming signal must be tracked accurately and any changes in the incoming signal frequency must be reflected in the locally generated signals as well. To continuously track the GNSS signal, a closed feedback loop is placed in the receiver where accumulated values are read by the processor and based on these values, corrections to the local replicas are made. The idea is that with every iteration, the local replicas are better aligned with the input GPS L1 C/A signal.

Since all tracking modules are designed and implemented in the FPGA, it seems obvious to construct the components forming the feedback loop, in hardware as well. However, the feedback loop consists of discriminators and filters that require operations such as division, arctangent and multiplication operations (see Chapter 2). Division and

tan inversion implementation in an FPGA are not straightforward and require significant time and effort to develop, and hence has been marked as a future task.

Consequently, a portion of the feedback loop is implemented in software (processor). The DLL, FLL and PLL loops are based on established designs described in (Gleason and Gebre-Egziabher 2009). The filter bandwidth, amplitude and damping ratio has been determined based on the equation provided in (Groves 2013). The code tracking error is computed using Equation 3-17 as shown below:

$$D_{err} = \frac{E - L}{P + P} \qquad \text{Equation 3-17}$$

where:

$$E = I_E^2 + Q_E^2 \qquad \text{Equation 3-18}$$

$$P = I_P^2 + Q_P^2 \qquad \text{Equation 3-19}$$

$$L = I_L^2 + Q_L^2 \qquad \text{Equation 3-20}$$

The first second of tracking relies on a first-order FLL, where the carrier signal frequency is tracked closely. The frequency error e_ω is computed using Equation 3-21 where I'_p and Q'_p denote the accumulated value from the previous integration interval. The FLL has a high gain (coefficient is 1.15 corresponding to a 50 Hz bandwidth) and is sensitive to noise, but can be used to swiftly converge close to the true carrier signal frequency. The second order PLL kicks in after one second which has a narrow bandwidth can track the signal for the rest of the duration. Finer details such as the bandwidth, damping ratio

ζ, amplitude and summation interval is provided in Table 3-6. The phase error is represented by e_ϕ and is defined in Equation 3-22.

$$e_\omega = arctan2\left(\frac{I'_p \cdot Q_p - Q'_p \cdot I_p}{I_p \cdot I'_p + Q_p \cdot Q'_p}\right) \quad \text{Equation 3-21}$$

$$e_\phi = arctan\left(\frac{Q_p}{I_p}\right) \quad \text{Equation 3-22}$$

Table 3-6 PLL parameters

Bandwidth	15 Hz	Low noise bandwidth
Damping ratio	0.7	Determines how fast the loop reaches settling time
Gain	0.25	Amplitude
Summation interval	0.001 ms	Integration duration

An important aspect to consider in a receiver during tracking is also its signal 'lock' status. When a signal is being tracked, its SNR is monitored. If SNR falls below the threshold, the channel loses lock and attempts to re-acquire the satellite signal. Monitoring SNR also presents an opportunity to compare receiver performance in varying environments (as will be seen in reflectometry).

Navigation solution computation is the last processing stage in the receiver and is also implemented in the microprocessor. The developed C-based software GNSS receiver has several functions related to navigation solution computation that is ported over to the processor effortlessly. Ideally, in the developed GNSS-R receiver, navigation data would

be read by the processor and based on the decoded ephemeris, satellite position would be computed and subsequently, the receiver position is estimated. However, since the multi-tracking channel has not been implemented yet, the navigation solution algorithm has only been tested using the software receiver.

3.2.6.1 Software design to hardware implementation migration challenges

This subsection addresses certain challenges that come with hardware-based GNSS-R receiver. The distribution of the design between hardware and software components gave rise to the following challenges:

1. Communication between the hardware-software modules:

   Considering Figure 3-19, data are continuously passed between the hardware and software modules. The processor can read/write from/to certain registers in the FPGA using AXI. The speed of data transfer depends on AXI clock, which unfavourably drives the receiver performance.

2. Development time and effort:

   To implement the hardware equivalent of a software module requires significant effort leading to a longer development time. The hardware module in this case is designed in Verilog (HDL) that has various development, simulation, verification and testing stages. Additionally, most FPGA development tools have a fixed flow of synthesis, placement and routing, bitstream generation requiring an average of one hour per compilation. Any slight design modification requires a full re-

compilation of the project. Therefore, there's a limited number of compilations one can run per day inhibiting development.

3. Noise, signal integrity, electrostatic discharge:

   It is important to be aware of environmental noise parameters that can propagate into the receiver via the antennas, cables or surrounding equipment. The cable between the antenna and receiver was deliberately chosen to be short (less than 1 metre) to limit the overall noise figure of the receiver. Also, during the development of the receiver, a RF front end with FMC pins is selected (over other GPIO-based front ends such as MAX2769) to avoid propagation or introduction of any potential EMI.

3.2.7 Tracking results with simulated signals

The test setup with the GNSS signal simulator remains the same as what is presented in Figure 3-16. The objective of the test is to observe the developed receiver's ability to acquire and track satellite PRN 1 in real-time in the hardware implementation.

The tracking results are presented in Figure 3-20 to Figure 3-23. The plots are generated from data retrieved from the FPGA for a total of 10 seconds. In Figure 3-20, the accumulated values in the IP arm and QP arm are plotted. In the scenario where the GNSS signal is tracked correctly, all the correlation power is observed in the IP arm and what remains in the QP arm is the residual noise values. High correlation strength is observed in the IP accumulator (as seen in Figure 3-20), whereas a low correlation in the QP accumulator indicating successful lock on the incoming GPS L1-C/A-code signal.

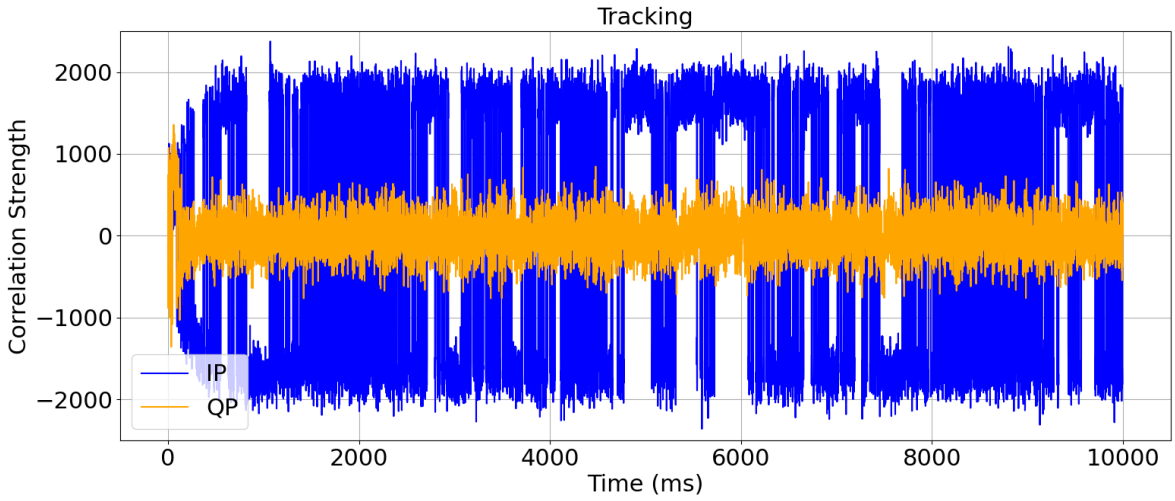


Figure 3-20: IP and QP accumulator values

Figure 3-21 represents the errors that are recorded from the carrier and code tracking loops implemented. The errors are initially high, however, within half a second, the errors converge to a minimum and stay low for the rest of the tracking duration. Similar trends are observed during tracking using an SDR. Figure 3-22 demonstrates a smaller window (approximately 1 second) of the IP arm, where the navigation data bits 0 or 1 can be more clearly observed. Looking at the discrete scatter plot in Figure 3-23, it becomes evident that the pseudorange and carrier phase are locked, since the IP arm consistently has much higher values than the QP arm. Also, as mentioned before, the navigation data from the IP arm is decoded to bits in the FPGA and the parameters are cross-referenced with data transmitted by the GNSS simulator confirming not only correctly functioning tracking channels, but also correct implementation of navigation data decoding logic.

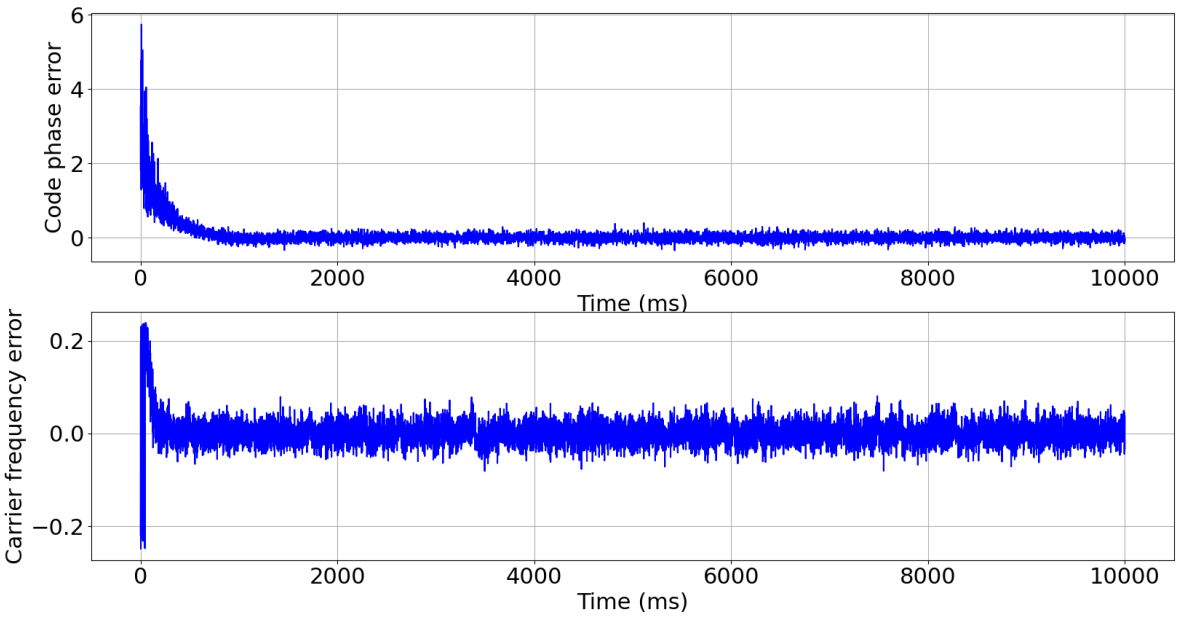


Figure 3-21: Tracking loop errors

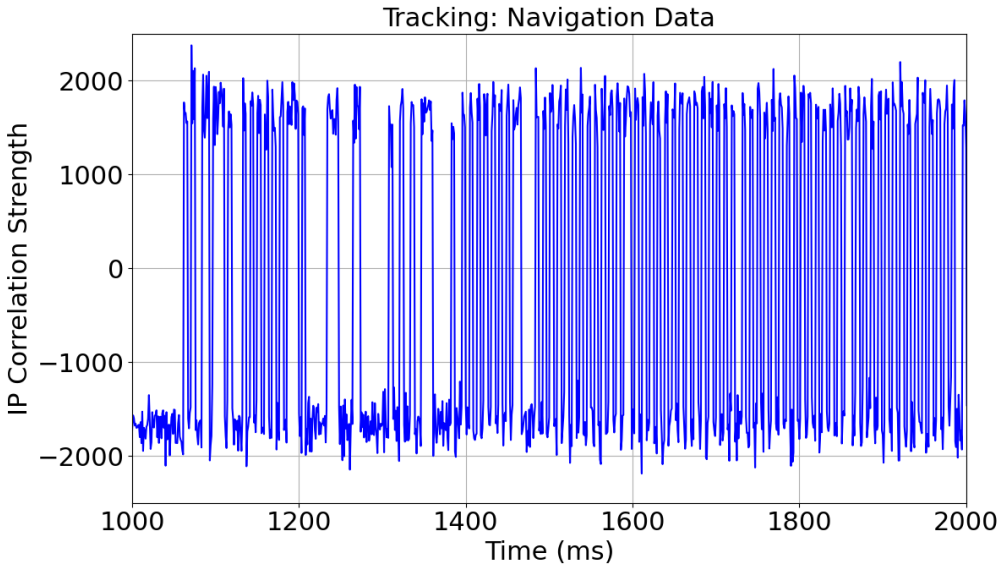


Figure 3-22: Recovered navigation data

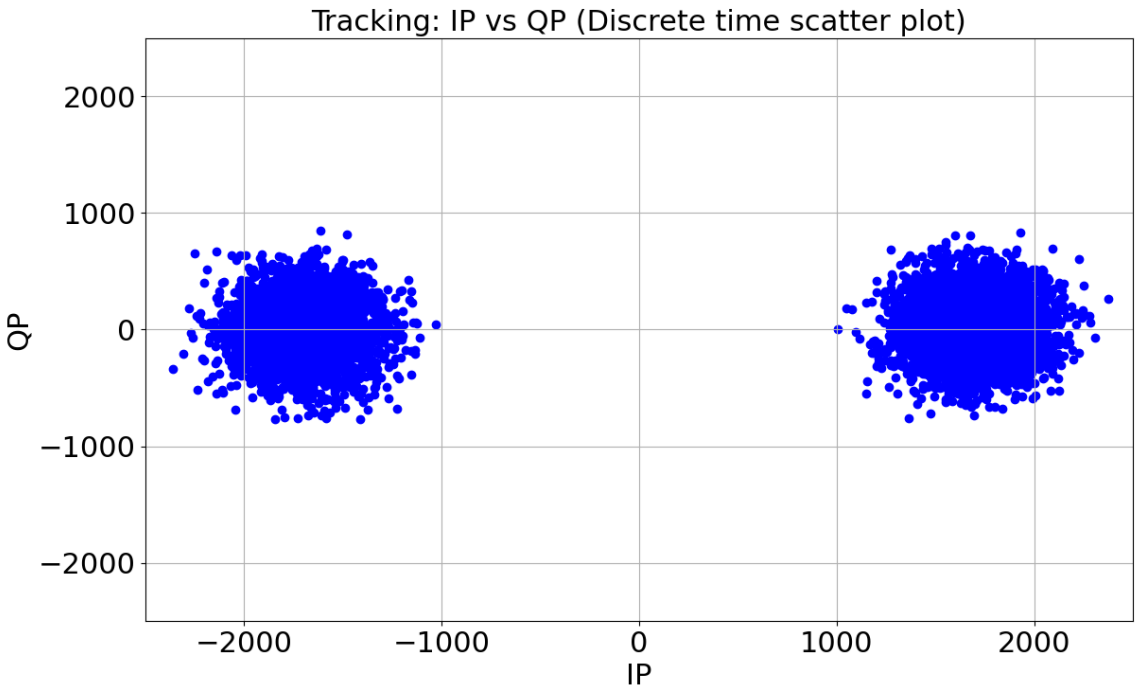


Figure 3-23: IP vs QP scatter plot

3.2.8 Tracking direct and reflected GNSS signals in the field

The GNSS receiver field test setup for reflectometry is presented in Figure 3-24 and Figure 3-25, where the zenith and nadir antennas are mounted on a tripod approximately 1.5 metres tall. Even though it is preferred that the antenna is positioned higher, the tripod's height is sufficient to demonstrate the reflectometry abilities of the receiver.

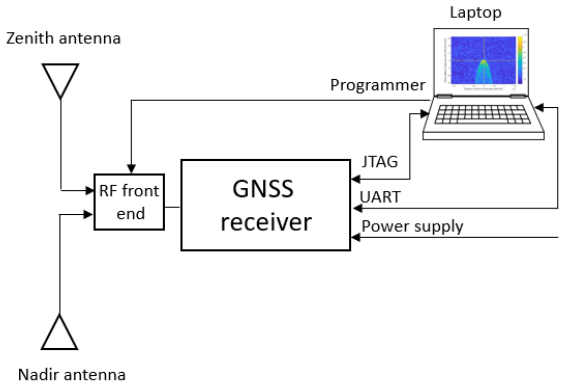


Figure 3-24 GNSS receiver test setup for reflectometry

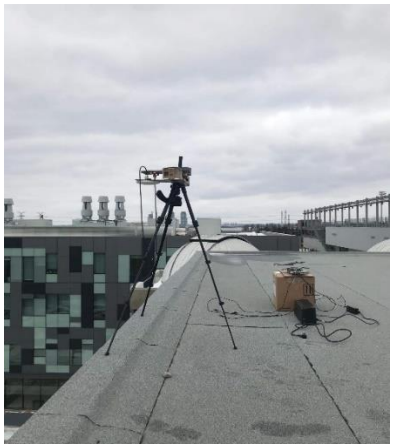


Figure 3-25 Standard GNSS-R receiver test setup with tripod mounted zenith and nadir antennas.

While the previous tests focused on the direct signal acquisition and tracking, this subsection focuses on the GNSS-R receiver operation where both the direct and reflected signals are acquired and tracked simultaneously. The field tests are conducted in two different locations. Referring to the image in Figure 3-26, the first test was conducted in

an open field (location A), whereas the second test (Figure 3-27) was conducted on the edge of a small pond (location B). The two test sites were intentionally chosen as the objective was to assess the receiver's ability to differentiate reflected signals from land versus water.

Figure 3-26 York University locations where tests were conducted. Location A (yellow) indicates where land reflections were collected. Location B (red) indicates where the receiver was stationed to collect water reflected signals.

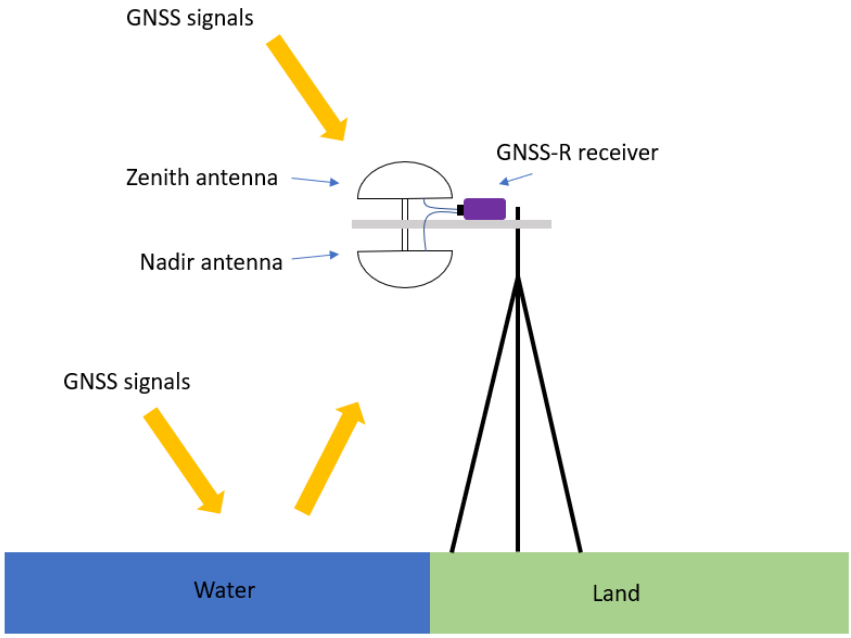


Figure 3-27 Representation of the GNSS-R receiver test setup at location B near the Stong Pond

3.2.8.1 Land surface reflections

Figure 3-28 shows the results obtained by tracking a reflected GPS signal (PRN 7) in real-time, using a nadir looking LHCP antenna. The presented data are collected in location A (open field). The IP correlator values exhibit a distinct trend much different from results obtained from direct signal tracking (Figure 3-20). Firstly, considering the IP versus QP values, the tracking loops still maintain a lock on the reflected GPS signal pseudorange and carrier phase, although it is not as well-defined as the direct signal discrete scatter plot. Secondly, the navigation bits appears to be present in the IP arm;

however, these navigation bits are not decoded since they are of no value. Lastly, the IP arm is observed to be noisier than the IP arm values obtained during direct signal tracking. The navigation data rate and the carrier signal frequency are affected by Doppler shift as well.

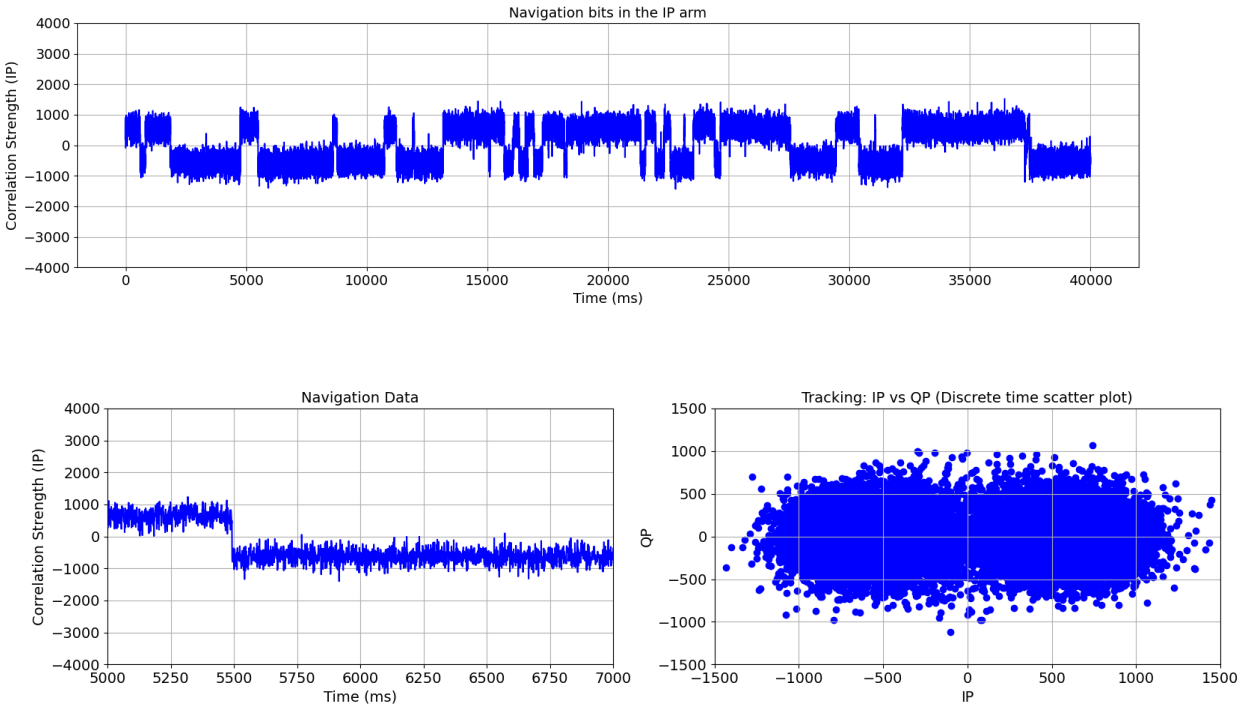


Figure 3-28 Land reflections: Reflected GPS signal tracking

3.2.8.2 Tracking performance: Carrier-to-noise density ratio

Measuring the carrier-to-noise density (C/N_0) ratio is an essential parameter in determining a successful signal lock during tracking (Pini et al. 2008). The C/N_0 can vary minimally amongst various receivers, since each of the instruments can potentially

have unique front ends, digital signal processors and architectures. Based on the satellite elevation angle and the signal strength, typically the C/N_0 of the receiver will be in the range of approximately 37 to 45 dB-Hz (Van Diggelen 2009; Angelo Joseph 2018).

In this research, the receiver performance in varying environments can be characterized using the C/N_0 metric (measured after correlation). In Figure 3-29, the simulated signal has an average of 42.12 dB-Hz for a signal with nominal strength. The GNSS-R receiver was also used to collect direct and reflected signals (open-field; location A) and the C/N_0 of each of these signals is also plotted. As observed in the figure, the direct signal has more noise with an average C/N_0 of 39 dB-Hz. As expected, the reflected signal has relatively even more noise with an average C/N_0 of 35.59 dB-Hz. These trends make sense as the C/N_0 of the reflected signal is expected to be lower than the direct signals. Also, when the direct and reflected signals are compared with the simulated signals, more noise is observed as expected.

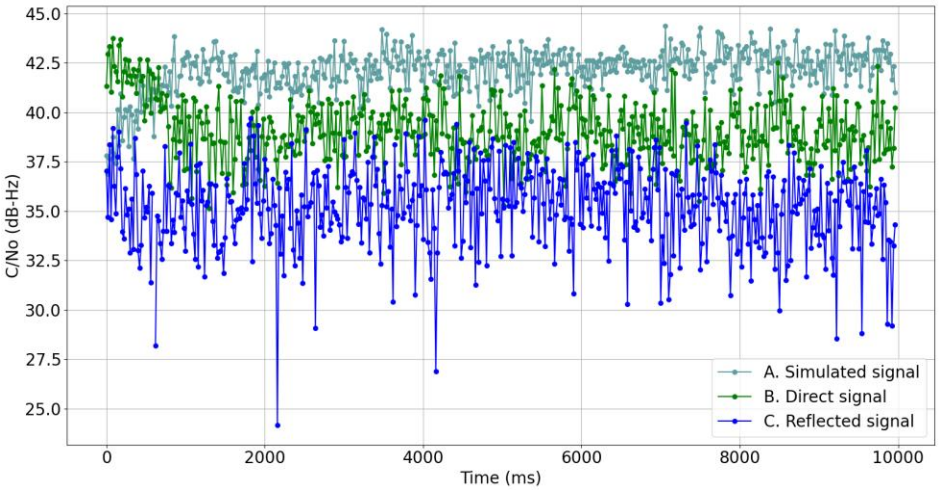


Figure 3-29: Comparison of carrier-to-noise density ratio of simulated signal, direct signal and reflected signal.

3.2.8.3 Water surface reflections

The plots below represent the tracking of PRN 31 GPS signal in real-time. The signal reflects on the surface of the water (location B) and arrives at the LHCP antenna. The receiver detects and tracks the weak reflected signal successfully. From initial testing, it has been observed that the signal does not suffer as significant a Doppler Shift or distortion as the signals reflecting off land. It should also be noted that the IP versus QP scatter plot (Figure 3-30) is much distinct and less noisy than the plots shown in Figure 3-28.

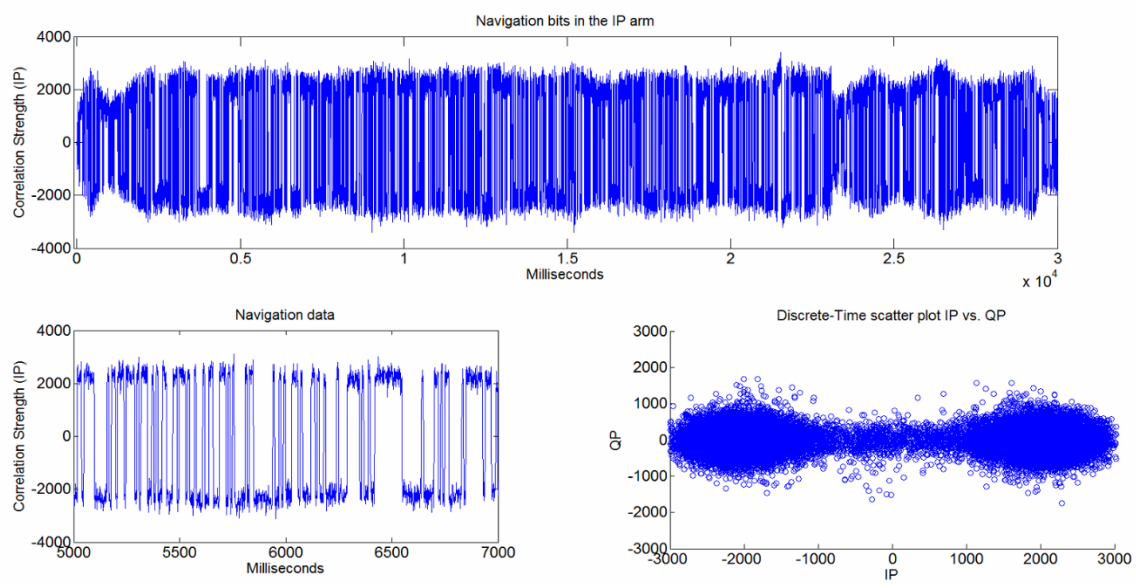


Figure 3-30 Water surface reflections: GPS signal tracked are stronger indicated by distinct IP and QP feature in the scatter plot.

3.2.8.4 Comparison between land and water reflected signals

From the tracking plots in Figure 3-28 and Figure 3-30, the immediate distinction one can make is the difference in correlation strength in the IP arm. In Figure 3-31, a comparison of the tracking performance of the reflected signals is presented. The green markers represent the signal reflecting off land (PRN 1), whereas the blue markers represent the signals reflecting off water (PRN 31). The average C/N_0 of land reflections is 34.64 dB-Hz. The average C/N_0 of the water reflections is significantly higher measuring 41.40 dB-Hz. The clear difference in measured C/N_0 indicates that the receiver can distinguish between land and water reflections even without the need for DDM generation. However,

it should be noted that this distinction is only relevant when the antenna is a few metres higher than ground level (test scenario).

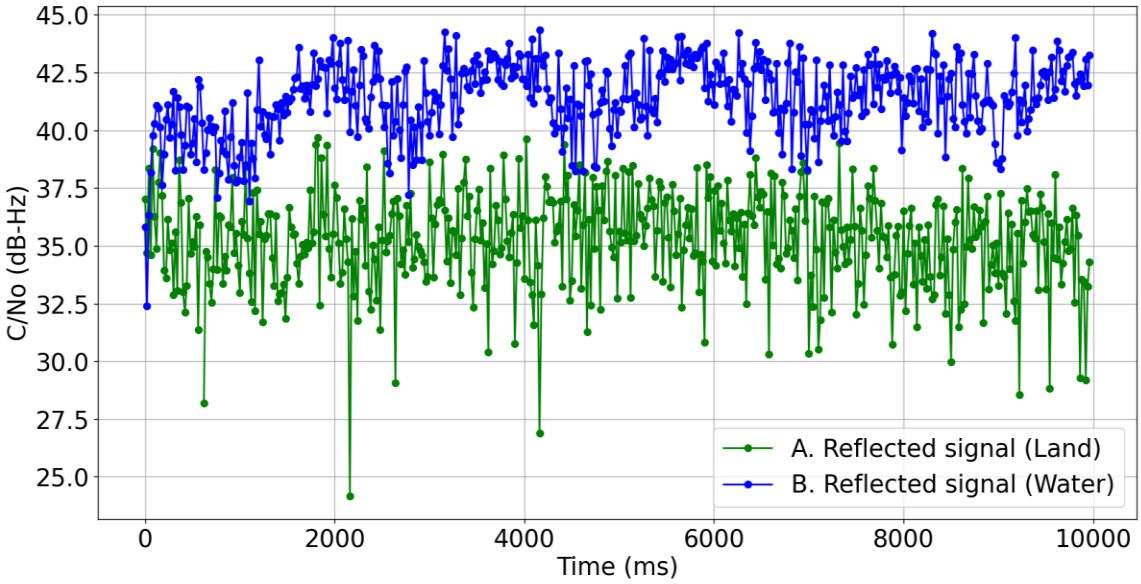


Figure 3-31: C/N_0 of land versus water reflected signals

The correlation strength of the reflected signals in the IP arm are almost as strong as correlation strength during direct signal tracking. The observations made in this research align well with results presented by Chew et. al. (2016), where higher SNR was detected in the instrument when flying over rivers. This result is a clear indication that the developed GNSS receiver can distinguish between land and water.

3.2.9 Design of navigation data recovery module in hardware

During successful tracking, the inphase-prompt accumulator is monitored to recover the navigation data. A new navigation data bit occurs every 20 milliseconds and is either a 1

or a 0. The recovered bits are then decoded following the scheme provided in the ICD-200G.

Navigation data has a much lower frequency (50 Hz) than the IF and the C/A-code. Due to this reason, microprocessors can comfortably process the data without slowing down the receiver operation. All GNSS-R receivers described in section 2.4.2 perform data recovery in software.

However, in this implementation, the complete navigation data recovery module is implemented in hardware. As a result, the software computational load is further reduced. As indicated in Figure 3-32, the process of navigation data decoding involves several more stages of correlation and computations before the satellite ephemeris is extracted. Since the developed GNSS-R receiver requires less than 0.5% of the FPGA resources per channel (refer Table 3-5), there is ample available processing capacity room to also perform navigation data decoding in the FPGA. When determining the maximum number of acquisition and tracking channels that can be implemented, other than considering the limited FPGA resources, the microprocessor capability is also studied. Fewer software computational tasks results in a GNSS-R receiver capable of incorporating a higher number of independent tracking channels. Given the ever-increasing number of GNSS signals and constellations, higher number of channels in the FPGA-based GNSS-R receiver is beneficial.

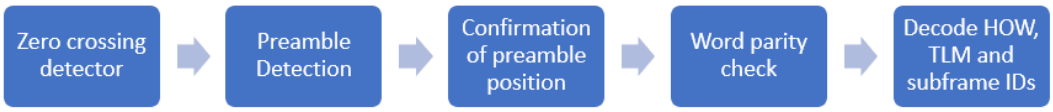


Figure 3-32 Navigation data decoding sequence implemented in the FPGA

The navigation data decoding module follows the format described in the GPS ICD. The module is designed to monitor the inphase-prompt (IP) accumulator value. A zero-edge detector keeps a note of the current bit value which could be a zero or a one. The bit value can change only once every 20 milliseconds (navigation data rate defined by ICD). Detecting the edge of the bit transition successfully at intervals of 20 milliseconds is known as bit synchronisation. However, in the case of consecutive bits of value 0 or 1, the receiver does not detect any bit transition.

The next step in the process of data decoding is preamble detection. A known preamble of the pattern 10001011 or 01110100 is transmitted at the start of every subframe embedded in the GPS signal. Successful detection of preamble would mean that the start of a subframe has been found. A simple correlator is implemented that checks for the preamble pattern as shown in Figure 3-33. Upon detection of a preamble, it becomes a potential preamble candidate and goes through thorough checks to confirm the validity of the preamble. The location of preamble is confirmed by detecting the same sequence at the next 300 millisecond mark. A subframe occurs every 300 millisecond duration therefore a preamble must occur at the same interval time.

After the preamble is successfully located, the receiver can start decoding the consequent bits of data. Since the navigation data format is known, according to the subframe ID embedded in the signal, various other parameters are decoded and saved. This navigation data decoding is performed in real-time. During testing (discussed in later sections), decoded time-of-week (TOW) and other satellite parameters are saved in FPGA registers.

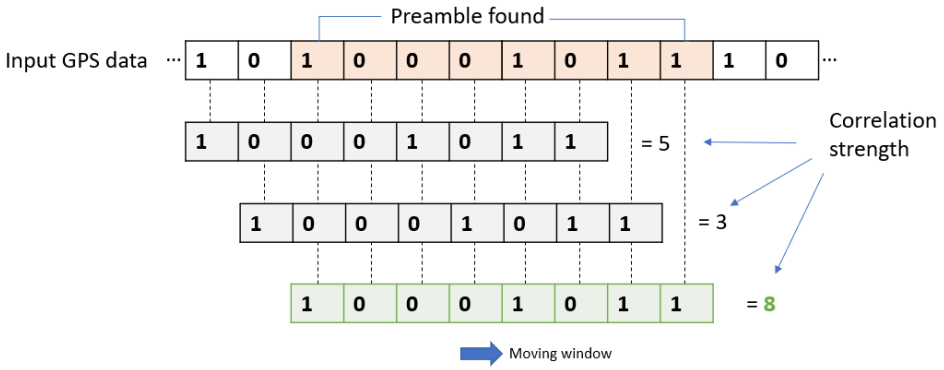


Figure 3-33 Preamble detection with a moving window

3.2.9.1 Verification of recovered navigation data using simulator

During tracking, the decoded data bits are sequentially forwarded to the next stage of processing such as navigation bit edge detection and synchronization, preamble detection, parity check and recording subframe data (currently only subframe 1-3 is decoded). The GNSS signal simulator is used to verify correct navigation data retrieval in the FPGA. The simulator software GUI (SimGEN) displays the TOW that is embedded in the GNSS signal. The initial TOW recovered by the GNSS receiver and the consecutive TOW

values matched the values transmitted by the simulator (indicated by the log) indicating successful navigation data decoding.

3.3 Summary

As it has been mentioned before, to have full control over GNSS signal processing and to be able to invent and test new algorithms, custom built hardware and software is inevitable (Lestarquit et al. 2016b). This chapter described the design and implementation details of an FPGA-based GNSS-R receiver followed by its assessment in terms of remote sensing capabilities. It is shown that the design decision of building a GNSS-R receiver with a one-bit quantization signal results in an extremely compact design requiring only 812 LUTs (0.3% of available resources, refer Table 3-5). The following points significantly contribute to the reduced design area in the FPGA:

1. Local signal generators do not require any RAM (unlike FFT implementations).
2. Multipliers are reduced to simple XOR gates.

It is also demonstrated that the GNSS-R receiver is capable of simultaneously processing the direct and reflected signals in real-time. Additionally, during tracking, the GNSS-R receiver records the C/N_0 for the processed signals. C/N_0 is an important dataset for generating additional products for geophysical parameter estimation such as soil moisture measurement, ocean surface wind speed measurement and other similar parameters.

The GNSS-R receiver was tested in a controlled environment with simulated signals. The receiver was also tested in static outdoor scenarios during which direct and

reflected signals were successfully acquired and tracked. Two such scenarios highlighted in the chapter are where signals are received from land and water reflections. The accumulated register values in each of these distinct tests are then compared in terms of C/N_0. The observations indicate that the C/N_0 of signals reflecting from the surface of water was on average approximately ~7 dB higher than the C/N_0 recorded when tracking land reflections. The difference in C/N_0 between water and land reflections is significant enough to conclude that a GNSS-R receiver using a one-bit quantization GNSS signal can be used for reflectometry applications where the nadir antenna is at a height of approximately 2 metres above ground level.

Chapter 4 GNSS-R RECEIVER SENSITIVITY IMPROVEMENTS

Research related to reflectometry using GNSS signals has demonstrated the capability of estimating several geophysical parameters from space. Due to the lack of a low-cost commercially available GNSS-R receiver for researchers, this dissertation emphasises on the necessary modifications made to standard GNSS receiver operation for reflectometry. Chapter 3 laid the foundation in describing the architecture and construction of a reconfigurable GNSS-R receiver capable of tracking direct and reflected GNSS signals. This chapter focuses on the technical features and algorithms that are designed and implemented to enhance the developed GNSS-R receiver further. The enhanced modules increase the sensitivity of the GNSS-R receiver to extremely weak reflected GNSS signals.

Construction of a GNSS-R receiver comes with its own set of challenges, especially in that the developed GNSS-R receiver is expected to detect extremely weak reflected signals. In this dissertation, weak signals are defined to have a signal strength of -140 dBm or less. Nominal GPS L1 C/A signals detected by the zenith antenna on the Earth's surface typically have a signal strength of -128.5 dBm (Dunn 2012). Since the nadir antenna detects weak reflected signals, the selection of antenna plays a critical role. Additionally, the receiver processes weak signals and tracks them using specialised algorithms. The processing gain of the algorithms implemented also plays a key role in

the overall SNR of the resulting signal. The following sections will describe how the GNSS receiver prototype developed in Chapter 3 is further enhanced to suit reflectometry applications. The description leads with requirements of a highly sensitive GNSS-R instrument. This section is followed by estimation of the receiver sensitivity and link budget calculation. The final two sections introduce a signal processing module to increase the sensitivity of the GNSS-R receiver and the results from the tests are presented.

4.1 GNSS-R receiver requirements

There are three critical requirements that the GNSS-R demands. Firstly, the receiver must be highly sensitive to weak signals. Secondly, the nadir antenna must have a high gain and lastly, the overall noise figure of the receiver including the antenna and RF front end must be low.

4.1.1 High sensitivity receiver

The sensitivity of a receiver can be improved in several ways. One way to increase sensitivity is to use a band pass filter and a low noise amplifier (LNA) placed in series between the antenna and the front end. While this approach is standard, additional hardware is required and will only be considered if all other methods of improvement are exhausted. Reconfigurable techniques are also considered for implementation in the FPGA resulting in a higher processing gain during acquisition and tracking. The

enhanced signal processing technique must improve the overall SNR of the signal leading to a highly sensitive receiver.

4.1.2 High gain nadir antennas

Weak reflected signals are LHCP in nature and hence require an LHCP antenna with a reasonably high gain (usually higher than the zenith antenna gain). Considering the lack of availability of LHCP high gain antennas in the GPS L1 frequency domain, several research groups have successfully designed custom antennas to suit their applications (Manandhar and Shibasaki 2007; SSTL 2016). Since designing antennas is not the focus of this research, antennas that are best suited for reflectometry are considered.

Only two LHCP antennas manufactured by Cobham that fit the criteria are noted (Figure 3-2). A brief description of the antenna has been provided in section 3.1.1. As mentioned earlier, two dual-polarised antennas are used for this research. Each antenna has two outputs, RHCP and LHCP. For the zenith antenna, only the RHCP component is processed, whereas for the nadir antenna only the LHCP component is considered. In the future, the RHCP components detected by the nadir antennas will also be processed.

4.1.3 Low noise figure RF front end

There are several losses that are introduced at every stage of signal processing starting at the point where the electromagnetic waves are converted to electrical voltages in the antenna. The concept of noise figure (NF) has been discussed in 3.1.1. It has also been mentioned that the NF of the selected RF front end is 3.8 dB. Ideally, lower NF is better,

but, since the front end is best suited in terms of reconfigurability, offers multiple antenna inputs and can support a wide range of GNSS signals (L1, L2 and L5). Each of these contributing losses are detailed in the link budget explained in the next section.

4.2 Link budget calculation

An overview of the link budget has already been calculated and elaborated in Section 3.2.5, where the sensitivity of the developed GNSS-R receiver is determined. In this section, the topic of receiver sensitivity is revisited since the antenna is replaced with the GNSS signal simulator (Spirent's GSS9000) for this segment of the research. The receiver sensitivity is theoretically computed and later compared against results obtained from hardware testing.

4.2.1 Simulated versus real signals

As seen in Figure 4-1, the GNSS signal simulator is connected to the RF front end. The antenna is illustrated in the same figure to highlight the differences in the ambient temperature when compared to the simulator. The ambient temperature is the major point of difference when receiver sensitivity is estimated. The same equations used in Section 3.2.5 (Equation 3-12 to Equation 3-14) are used except the T_A is now also the ambient temperature (290 K) (Van Diggelen 2009).

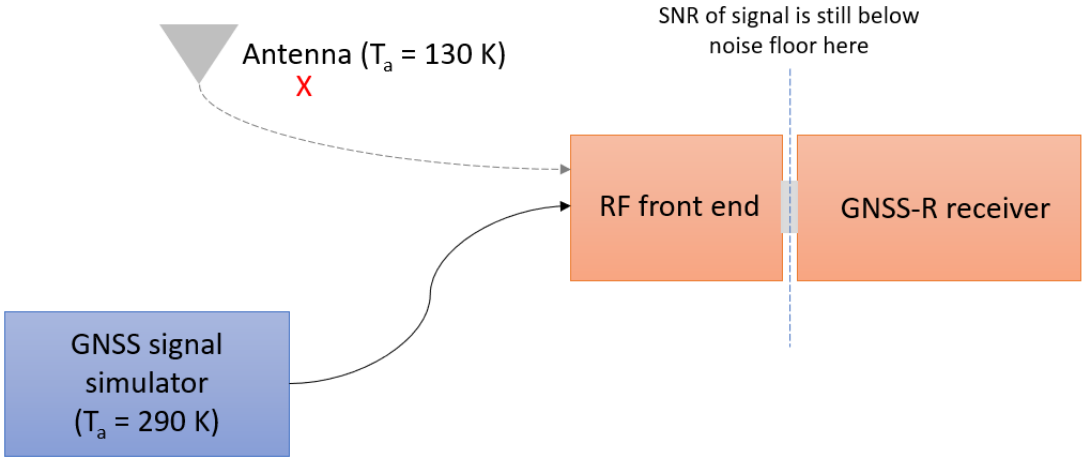


Figure 4-1: GNSS-R receiver setup with the simulator during receiver sensitivity tests

Table 4-1 lists all the parameters used to compute the resulting SNR of a signal. In this scenario, a nominal signal of -128.5 dBm (or -158.5 dBW) is presented. Since the antenna is replaced by the simulator and is an environment of ambient temperature, the antenna temperature is enumerated as 290 K.

Table 4-1: Sensitivity calculation for a nominal GNSS signal strength of -128.5 dBm or -158.5 dBW

Description	**Symbol**		**Units**
Boltzmann's Constant	k	1.38E-23	W/K-Hz
Front end noise figure	F	3.8	dB
Ambient Temperature	T	290	K
Antenna Temperature	Ta	290	K
Effective temperature	Teff	695.7	K
Noise Power	No	-200.2	dBW/Hz
IF Bandwidth	BW	3	MHz
Input Signal Strength		-128.5	dBm

		-158.5	dBW
Carrier-to-noise density ratio	**C/No**	**41.68**	**dB-Hz**
Signal-to-noise ratio	**SNR**	**-23.1**	**dB**

With the known parameters, the SNR for two specific scenarios are computed. The first scenario uses an antenna whereas the second scenario uses simulated signals generated in the laboratory. The comparison of the two resulting SNRs show that the SNR at the output of the front end will always be worse when a simulator is connected to the receiver, instead of an antenna pointing to the sky. As seen in Figure 4-2, for the same GPS L1-C/A signal strength, the SNR for the simulated signal has a constant offset of -1.14 dB from the expected SNR of real signals. This difference indicates that when using the simulator to accurately replicate the real GPS signals, the difference must be accounted for during calibration.

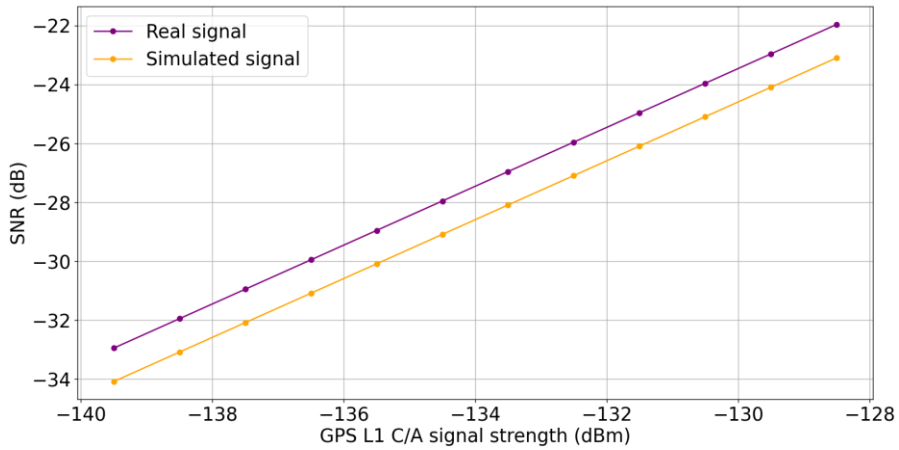


Figure 4-2: The SNR at the output of the front end for a real signal versus a simulated signal. Theoretically, the SNR of the simulated signal is always lower by 1.14 dB than the real signals received from the antenna.

4.3 Enhanced reflected signal processing module

As shown in Figure 3-9 in Chapter 3, the reflected LHCP signal is digitized by the RF front end just as the direct RHCP signal. The output of the RF front end is a 1-bit signal for the reflected signals and is also the input to the signal processing module in the FPGA. An enhanced independent IP core is designed to process weak reflected signals (explained in Section 4.3.3.) and this is where the processing algorithms for the direct and reflected signals diverge in similarity.

There are several reasons for using two separate hardcore processors for each of the direct and reflected signals. Firstly, the direct and reflected signals are in a true sense independent from each other and can take advantage of the multiple sophisticated ARM microprocessors available. Secondly, reflected signals require additional processing along with frequent processor intervention (as discussed in Section 3.2.6.1). Lastly, a separate core means the design is better organised, modular and portable. The modularity of the design also eases the verification and simulation of the module. In the following subsections, the integration methods implemented in the algorithm are introduced followed by the processing gain computation of the algorithm itself.

4.3.1 Concept of coherent and noncoherent integration

As mentioned in section 4.2, the digital output signal of the RF front end has an SNR of -23 dB indicating that the signal of interest is still well below the noise floor. The correlation and integration process offers significant processing gain to the signal

resulting in a positive SNR. There are two methods of integration used in this dissertation: coherent and noncoherent.

4.3.1.1 Coherent integration

It is well known that the navigation data occurs every 20 millisecond and at every bit edge occurrence, the phase of the signal changes (Borre et al. 2007). Coherent integration takes into consideration the phase of the GNSS signal. If the integration time is 1 millisecond, the I and Q values are accumulated over the integration period of 1 millisecond. For a weaker signal, the integration period can be increased to get a better signal-to-noise ratio (Van Diggelen 2009). However, it is not recommended to exceed the integration time beyond 20 millisecond, as the navigation data bit duration is 20 millisecond and the transition bit edge (1 to 0 or vice versa) might occur during the integration period. If a transition does occur during the integration period, the overall accumulated value gets lowered and might lead to incorrect results. Due to this limitation, coherent integration can detect signals of only certain strengths and higher.

4.3.1.2 Noncoherent integration

Noncoherent integration does not account for the phases of the correlated values. If the integration period is 1 millisecond, the I and Q accumulated values are squared and combined before being compared with the threshold (Van Diggelen 2009). For a noncoherent integration of period 10 milliseconds, each of the squared values at the end

of every 1 millisecond up to the 10th millisecond is accumulated and compared with a threshold to determine the presence of a satellite. If a phase shift occurs, only that 1 millisecond of data will be affected and it will have a minimum impact on the overall accumulated value. Noncoherent integration increases the overall SNR by $\sqrt{M}$, where M is the length of the integration period and hence this method can have an integration period of at least a few seconds. On the other hand, noncoherent integration also squares the noise and suffers a squaring loss (Xiaohui et.al. 2006). This squaring loss is taken into account when the overall SNR of the design is computed.

4.3.1.3 Limitations of coherent and noncoherent integration

Weak signal acquisition often requires long coherent and noncoherent durations. Coherent integration improves sensitivity but is limited by the navigation bit transition. Noncoherent integration is not bound by the same limitations as coherent integration, however it is less sensitive (Ziedan, 2006). Hence, the integration periods must be cautiously chosen based on the operation scenario and application. In the developed GNSS-R receiver, the integration periods can be dynamically controlled and reconfigured during operation based on the input signal strength.

4.3.2 Receiver sensitivity estimation

Determining the receiver sensitivity is essential when the developed receiver is expected to process weak reflected signals. The processing gain of the receiver determines if the

receiver is capable of correctly detecting a visible satellite signal. The focus of this subsection is to estimate the processing gain offered by the existing correlation and integration implementation (in the FPGA-based acquisition module). An effort to enhance the processing gain further is also discussed.

4.3.2.1 Baseline processing gain

The FPGA-based acquisition algorithm and the integration period have a significant influence on the resultant SNR. The concept of coherent gain has already been discussed in Section 3.2.5 and presented in Equation 3-16. Table 4-2 presents the various factors that influence the gain computation. Each row is briefly described:

1. Even though the PRN frequency is 1.023 MHz, in this implementation, 2 samples represent each chip and hence the frequency is 2.046 MHz.
2. While the integration interval can be varied, in the current implementation, it is set to 2 milliseconds.
3. The number of points (Mc) is the product of values listed in row 1 and 2.
4. The resulting ideal coherent gain is computed using equation:

$$Ideal\ coherent\ gain = 10 * log_{10}(Mc)$$

5. Losses introduced by the front-end during filtering and bandlimiting are known as IF losses.
6. Quantization loss has been mentioned in Section 2.4.5. In this case, due to 1-bit quantization, the loss is expected to be -1.96 dB.

7. Frequency mismatch loss is introduced when the local replica is not an exact match of the input signal.
8. Code misalignment loss is introduced to the receiver when the input signal code and local code replica do not perfectly correlate.
9. The total implementation losses are an accumulation of all the system losses.

The actual coherent gain takes into consideration all the losses and for a GNSS signal with the strength of -128.5 dBm, the SNR ratio is approximately 2.93. As seen in the discussion above, the coherent gain can be further increased by modifying certain parameters presented in Table 4-2

Table 4-2: Processing gain estimation for coherent integration implementation.

	Coherent integration		Unit
1.	PRN sampling frequency	2.046	MHz
2.	Coherent integration Interval	2	ms
3.	Number of points, Mc	4092	
4.	Ideal Coherent gain	36.11935625	dB
5.	IF loss	0	dB
6.	Quantization Loss	-1.96	dB
7.	Frequency mismatch loss	-0.5	dB
8.	Code alignment loss	-1.2	dB
9.	Total Implementation loss	-3.66	dB
10.	Actual coherent gain	32.5	dB
11.	SNR	9.4	dB
12.	**SNR Ratio**	**2.9**	

4.3.3 Adapted algorithm to improve receiver sensitivity

Weak signal acquisition often requires long coherent and noncoherent durations. Coherent integration improves sensitivity, but is limited by the navigation bit duration

and transition. Noncoherent integration is not bound by the same limitations as coherent integration, but it is less sensitive (Ziedan, 2006). A combination of both techniques with a slight modification is known as the alternate half-bit method. The alternate half-bit method was introduced by Lin and Tsui (2001) and has been compared extensively with other methods (Li et al. 2012). However, available information regarding the algorithm's performance in a hardware implementation is limited.

The hardware equivalent of the alternate half-bit method is designed and incorporated into the prototype GNSS-R receiver and is expected to improve in acquisition sensitivity of weak signals (as demonstrated by the software receivers described in the literature). Figure 4-3 presents the alternate half-bit implementation method. Two consecutive blocks of 10 milliseconds of GPS data (block 1 and 2) are selected for acquisition. Each of these two blocks are processed coherently (see Section 4.3.1.1. for coherent integration). Since navigation bit transition occurs every 20 milliseconds, the phase change will occur in either block 1 or block 2, but not both.

Referring to Figure 4-3, the bit transition occurs in every odd-numbered block starting with block 1. If coherent integration is performed for 10 milliseconds and every alternate block is added noncoherently (for say 20 milliseconds) then at the end there would be two noncoherently accumulated results: one that suffered a bit transition (*accum1*), while the other that did not (*accum2*). By assessing the two results, the correct SNR and satellite detection is possible.

A point to note here is that as the SNR drops, the acquisition time increases (Xiaohui Ba et al. 2006). According to the results of Psiaki (2001), signal acquisition using the alternate half-bit method as low as 20 dB-Hz can take up to 14 hours. Choice of integration duration then becomes a critical parameter if being considered for real-time GNSS-R receiver operation.

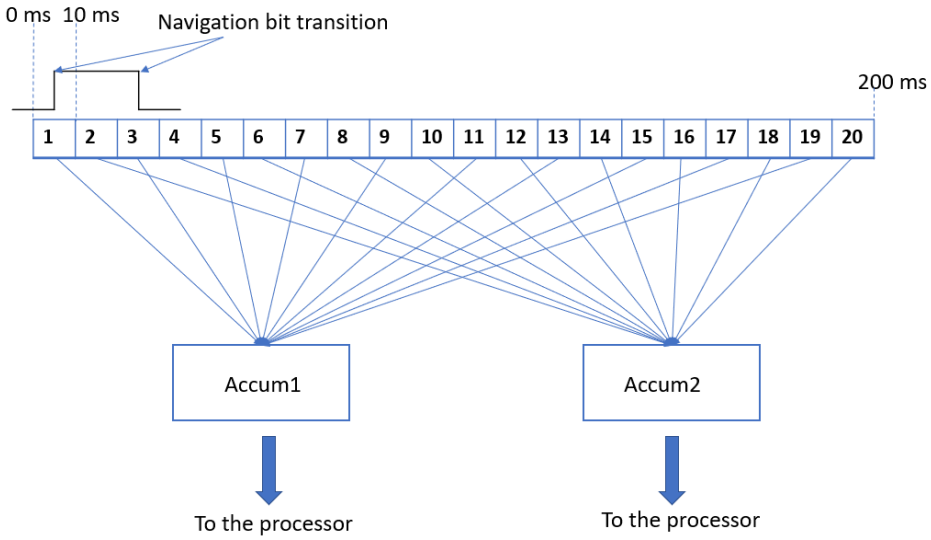


Figure 4-3 Alternate half-bit method implementation

The estimated improvement in sensitivity is computed for the alternate half-bit method and presented in Table 4-3 below. As highlighted, the resulting SNR ratio is 5.97 and is approximately 2 times higher than the baseline processing gain (Table 4-2). Using logarithmic scale to compare, an improvement of 65% in coherent gain is observed.

Table 4-3: SNR ratio for the alternate half-bit method implemented in the FPGA.

Enhanced integration algorithm		Unit
Squaring loss	-3.84	dB

Non-coherent Interval	20	ms
Coherent interval	10	ms
NHC gain	10	
Final SNR	15.5	dB
SNR ratio	**5.97**	

4.4 Performance of enhanced weak signal acquisition module

The alternate half-bit method described in the previous subsection is tested in hardware using a controlled environment and simulated signals. The processing gain estimation showed that the enhanced acquisition module doubles the SNR ratio of the input signal when compared to the baseline processing module. In this subsection, various test scenarios are described and hardware test results are presented.

4.4.1 Acquisition results for varying integration periods

One method of assessing acquisition performance is by running Monte-Carlo simulations to determine the sensitivity of the developed GNSS-R receiver. A total of 100 iterations were performed for varying integration times of 1, 2, 4 and 8 millisecond, as well as varying signal strengths generated by the simulator (Guruprasad 2019). The tests were limited to 100 iterations as a higher number (such as 1000) is not feasible (requires more testing infrastructure and automation) with the current hardware implementation and GNSS simulator, but will be considered in the future.

Figure 4-4 illustrates the minimum SNR required to have a probability of detection greater than 95% at each of the integration duration. According to the graph, to

correctly detect a satellite 95 out of 100 times, with an integration time of 1 millisecond, the SNR must be greater than -30 dB measured at the output of the RF front end. This value corresponds to the nominal value of direct GPS signals. As the signal strength decreases, the required integration time to successfully detect a satellite increases. With an integration time of 8 millisecond, signals in the range of -35 dB are detected. Integration durations higher than 8 millisecond are not considered as there is a risk of a navigation data transition occurring in the sequence leading to satellite misdetection. This test is important in assessing the performance of the GNSS-R receiver prototype as a baseline. It is confirmed that the receiver performs well in environments where the zenith antenna has a clear view of the sky. In case of misdetection of a satellite, the tracking channel will discover the absence of a satellite signal and will drop tracking the signal and make itself available for the next acquired satellite.

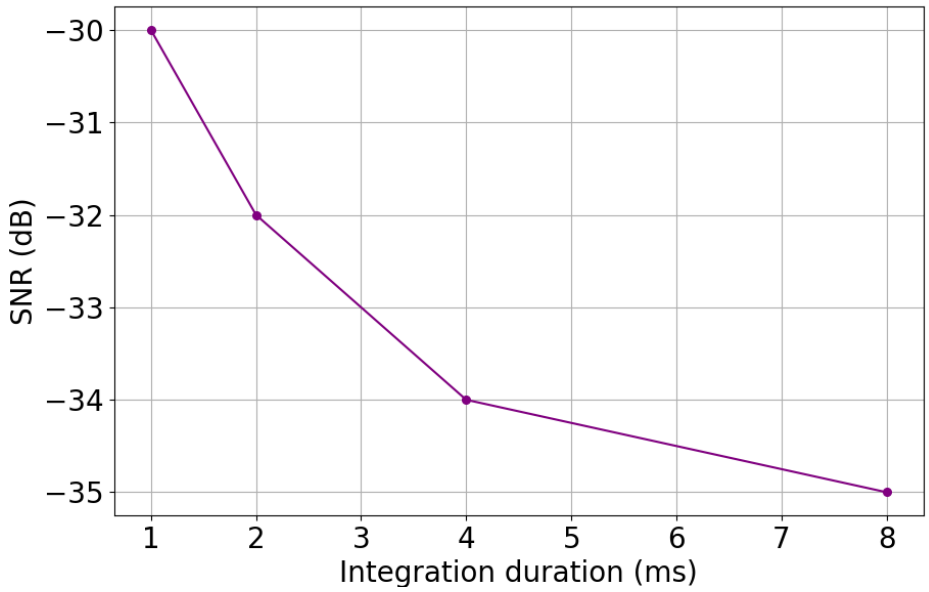


Figure 4-4: Minimum SNR required for 95% probability of detection (simulated signals)

Figure 4-5 and Figure 4-6 present the increasing computational time and complexity with increase in coherent integration time. With increasing integration time, the size of the Doppler bins reduces according to the $3/2T_I$ rule, where T_I represents the integration period. This study was conducted to demonstrate that even though the acquisition search operates in real-time, the receiver performs a serial search for every combination of Doppler shift and code delay and can take an average of 30 minutes (for 8 millisecond integration period) to complete the search for a single satellite signal.

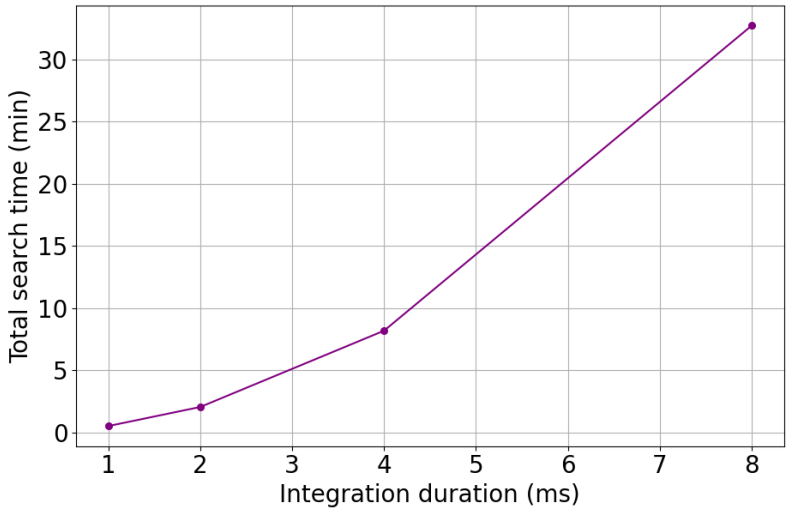


Figure 4-5 Total search time per satellite (min) vs integration time (ms)

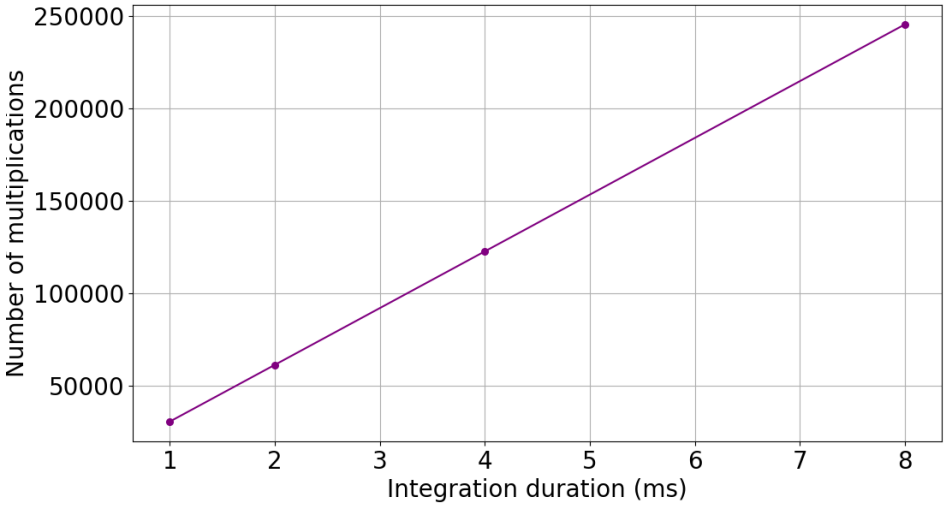


Figure 4-6 Number of multiplications vs integration time (ms)

4.4.2 Impact of coherent and noncoherent integration periods on acquisition sensitivity

Weak signal acquisition is a significant concern as reflected signals in space are incredibly difficult to detect. Additionally, as mentioned in the previous section, signals

with low SNR can be detected by using long integration periods. The alternate half-bit method was also discussed in detail where two accumulators (accum1 and accum2) gather values from alternate integration periods (Figure 4-3). With the aid of the GNSS simulator, weak signals were generated, and the hardware implementation of the alternate half-bit method was tested.

The results presented in Figure 4-7 is for an acquisition implementation that performs 10 millisecond of coherent integration, along with 10 millisecond of noncoherent integration using the alternate half-bit method. Three test cases were conducted starting with SNR -35 dB (corresponding to signal strength -140 dBm), -40 dB and finally -45 dB. The green bars indicate the noncoherently accumulated values in accum1 whereas the alternately occurring blocks were accumulated in accum2. As expected, with the decrease in SNR values, the magnitude of accumulated values in accum1, and accum2 also decrease. Also, accum1 and accum2 differ significantly for SNR -35 dB, indicating that the navigation bit edge occurred in accum1 reinforcing the argument that accum2 does not suffer from any phase shifts due to navigation bits. The difference is not as obvious in the other two test cases, as the SNR is much weaker. Results in Figure 4-7 compared with Figure 4-4 indicate that acquisition sensitivity of the receiver has improved as the receiver can now securely detect signals with SNR values lower than -35 dB (or GNSS signal strength -140 dBm).

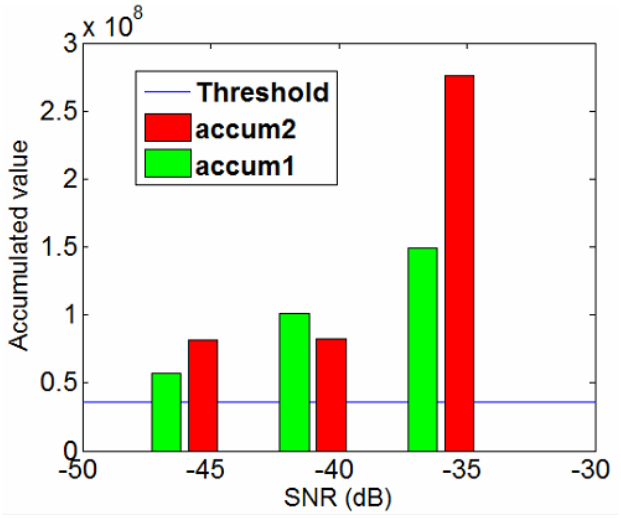


Figure 4-7 Acquisition result of hardware implementation of alternate half-bit method representing high correlation strengths obtained for weak GPS signals that were previously not detected.

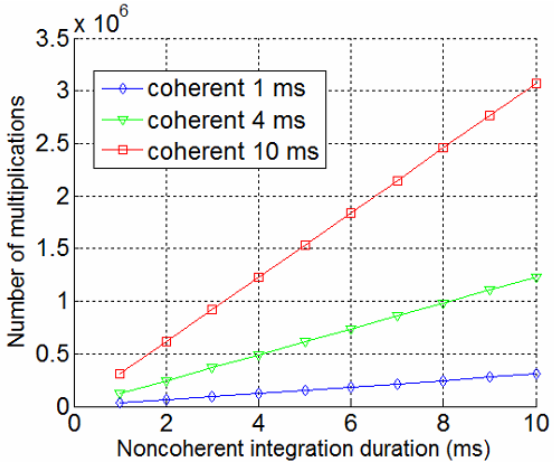


Figure 4-8 Increase in computational complexity with increase in integration period for alternate half-bit method.

Ideally, to assess the performance of an acquisition method, numerous acquisition iterations must be performed to determine the probability of detection of the

implementation. However, for a serial acquisition method, a long coherent/noncoherent integration duration can take minutes or even hours for completion of one iteration. The test cases mentioned above each took approximately two hours for one satellite search. A long integration time is hence not recommended for acquisition of weak signals from space, since the dynamic environment would make acquisition even more challenging. Estimating the search time from the graph shown in Figure 4-8, a 10 millisecond coherent integration and noncoherent integration period is selected and any computation requiring longer integration period is not recommended for real-time applications. Even though the computation complexity of the alternate half-bit method is high, it did improve the sensitivity of the receiver with signals as low as -45 dB crossing the threshold compared to Figure 4-4, where to detect a signal with -35 dB SNR and less, coherent integration period longer than 8 millisecond is required.

4.5 Summary

In this chapter, the significance of receiver sensitivity was emphasized upon and the sensitivity of the developed GNSS-R receiver is estimated. The receiver is sensitive to signals with SNR -31 dB or higher (nominal for GPS signals). A hardware implementation of alternate half-bit method is proposed and implemented to improve the sensitivity of the receiver. The alternate half-bit method is a combination of coherent and noncoherent integration techniques. The hardware implementation is tested using the Monte Carlo method, where the same test was performed one hundred times using the same simulated environment and signals. The results show that the sensitivity of the

receiver to weak signals increases by 65%, and the prototype receiver can successfully detect signals as weak as -45 dB. This chapter demonstrated that the developed GNSS-R receiver prototype has an improved sensitivity using COTS hardware allowing reflected signals of low strengths to be detected and tracked using closed tracking method.

Chapter 5 DESIGN AND IMPLEMENTATION OF A REAL-TIME DDM GENERATOR

The essential GNSS-R measurement is the Delay-Doppler map (DDM). DDMs represent cross correlation values between the received signal and the locally generated PRN code sequence and carrier signals. This chapter describes the design of a DDM generator with a resolution of 40 code delay bins and 20 Doppler bins. The implementation is for a 1-bit signal resolution, leading to a compact DDM generator, capable of providing real-time SNR information. The design description is followed by detailing the FPGA implementation methodology. Open-access satellite data from the CYGNSS mission is accessed and used to test the DDM generator.

5.1 GNSS-R missions

In this section, GNSS-R missions relevant to this dissertation are listed. The data collected by some of these spacecrafts are used to validate the developed GNSS-R receiver. Even though the bistatic radar cross section (BRCS) method of reflectometry using signals of opportunity such as GNSS was known since the 1990s, it was only much later that this technology was demonstrated from space and reported by Lowe et al. (2002). Although some of the missions have been briefly described in Section 2.4, the following subsections provide additional details pertaining to DDM generation and related terminology. Here is a list of all the past and current missions that have

successfully demonstrated GNSS-R conceptually from space and have published raw data for the scientific community to use.

5.1.1 Disaster Monitoring Constellation

In late 2003, the UK launched the disaster monitoring constellation (DMC) to monitor floods using remote sensing. Scott Gleason and his team were one of the very first to successfully process these data to demonstrate ocean surface monitoring and sea ice characterization using reflected GNSS signals (Gleason et al. 2005; Gleason 2006, 2010; Gleason and Gebre-Egziabher 2009). The GNSS-R receiver onboard the satellite was developed by SSTL predominantly for the DMC mission. The navigational receiver was additionally equipped with the ability to record bursts of raw IF data from the zenith and the nadir antennas (20 seconds duration) that was downlinked and further processed using ground-based software receivers (Jales 2012). The satellite was retired in 2011.

5.1.2 TechDemoSat-1

Following the successful demonstration of the concept of GNSS reflectometry, SSTL built and launched TechDemoSat-1 (TDS-1) in 2014. The satellite carries a more advanced GNSS-R receiver named Space GNSS receiver Remote Sensing Instrument (SGR-ReSI). The receiver is implemented using a LEON3 soft-core processor and works in conjunction with an Actel ProASIC3 FPGA. Even though the receiver processes GPS signals in real-time, the DDMs are generated in what is considered as near-real-time (De Vos Van Steenwijk et al. 2010). The GNSS-R receiver can generate a DDM with up to

128 delay and 20 Doppler pixels, but it depends on the application (SSTL 2016). In addition, relevant to the dissertation is the method of DDM generation. SGR-ReSI employs the parallel search phase method for its low computational complexity in comparison with the serial search phase method. An approach called the Zoom Transform Correlator (ZTC) has been implemented and there are four implementations for each of the four channels that are dedicated for reflected signal processing (Jales 2012). The ZTC also compresses the DDMs hence reducing the size and dimension of the DDM. Compressed or truncated DDMs are more manageable onboard the satellite and also ensures the downlink bandwidth to the ground station is not saturated. Therefore, the standard operation of the GNSS-R instrument generates truncated DDMs of 17 delay bins by 11 Doppler bins, however on some occasions upon special operation conditions, full DDMs of the resolution 128 delay bins by 20 Doppler bins is also generated and transmitted to the ground station.

5.1.3 CYGNSS

NASA launched eight satellites equipped with GNSS-R receivers in 2016 with each carrying a DDM instrument (DDMI). The DDMI is a version of the SGR-ReSI that flew on the TDS-1 satellite. This instrument consists of a low-gain zenith antenna and two high-gain nadir antennas. The nadir antennas are on the port and the starboard side of each satellite. The main objective of the mission is to measure sea surface wind speed in areas of tropical cyclones (Ruf et al. 2017). All datasets derived from this ongoing mission have been made available online by the CYGNSS Science Operation Center

(SOC). Datasets have been classified into various levels as summarised in Table 5-1 (University of Michigan 2016).

Table 5-1 Various data level published by the CYGNSS SOC

Data Level	**Contents**	**Format**
Level 0	Raw IF data	Binary
Level 1	Calibrated DDM (in Watts)	netCDF4
Level 2	Wind speed product	netCDF4
Level 3 and 4	Surface wind speed product for data assimilation	netCDF4

The DDMI instrument has four DDM channels and each of these channels generate four DDMs every second. The main products are DDMs with the spatial resolution of 17 delay by 11 Doppler bins. An example is shown in Figure 5-1, where there are 17 delay bins each spaced by half of one chip and 11 Doppler bins each spaced by 500 Hz. The characteristic "horseshoe" pattern in the plot is the result of the power reflecting from a region on the surface of a water body centered around the specular point.

Raw IF data are usually discarded given the high data rate and other subsequent challenges of storing and downlinking large data volumes. The same challenges persist for the standard or "full" DDMs generated with a resolution of 128 delay bins by 20 Doppler bins. Therefore, only truncated DDMs are available with the exception of certain regular instances, where raw IF data is recorded for short duration (~60 seconds) and

downlinked for further processing. Similarly, full DDMs are also occasionally transmitted to the ground station.

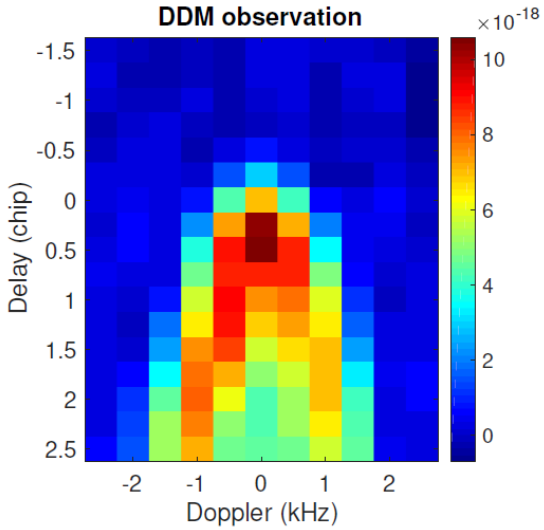


Figure 5-1 Typical Level 1 DDM generated by the DDMI with a resolution of 17 delay by 11 Doppler bins (units in Watts).

5.1.4 Other missions

Other than these three missions, there are several more that have been launched since 2019 such as SPIRE GNSS-R, BuFeng-1, FSSCat, DoT-1, FengYun-3E (Li et al. 2022). Another future mission worth noting is the proposed HydroGNSS mission designed and developed by SSTL where the key objective is to demonstrate the ability of small satellites to support scientific research. This Earth observation constellation will have an advanced version of their flagship receiver known as SGR-ReSI-Z. The main mission

objective is to observe and monitor soil moisture, freeze/thaw states, ground biomass and wetlands (Unwin et al. 2021b).

5.2 DDM design and implementation

A DDM is created by cross correlating the received signal with locally generated signals for varying code delays and Doppler shifts. The new DDM module is built to work in conjunction with a receiver that also processes direct GNSS signals. As shown in Figure 5-2, the design of a single DDM generator channel consists of digitized signals from the RF front end, followed by several bit-level operations. Multiple code replica and carrier replica generators are instantiated. The exact number of replicas are driven by the desired dimension of the DDM. In the implementation discussed in this research, 40 modules of code replica are instantiated with each having a difference of 0.5 chip from the subsequent module. Similarly, the carrier replica also generates a range of carrier signals. In this implementation, a total of 20 replicas are generated with a step size of 500 Hz.

5.2.1 Open loop tracking

Unlike the direct signal acquisition or tracking method, where the local signal generators are frequently corrected using discriminators, PLLs and DLLs, in the open loop tracking method, the initialisation and correction of the replicas are driven by the direct signal tracking loops and transmitter-specular-point-receiver geometry (Gleason 2019). To determine the corrections in real-time, geometrical calculations must be made in real-time in the processor as well. Figure 5-3 describes the logic where position of the receiver is

determined based on navigation data decoded during direct signal tracking. At this point, the position of the transmitter and the receiver is known. With this knowledge, the specular point is estimated.

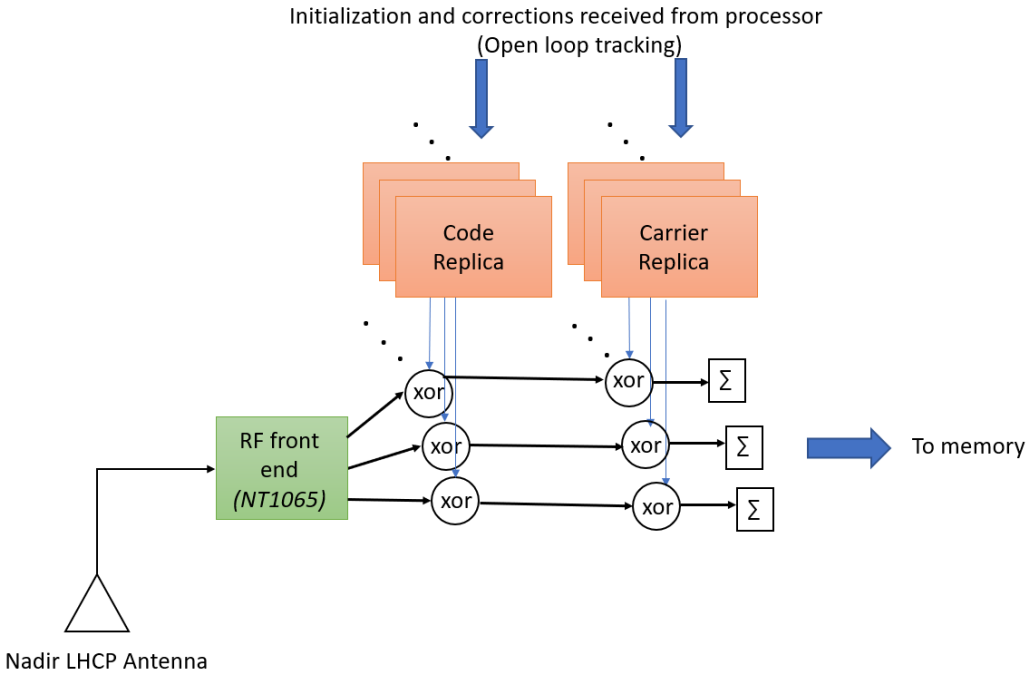


Figure 5-2 Single channel DDM generator design

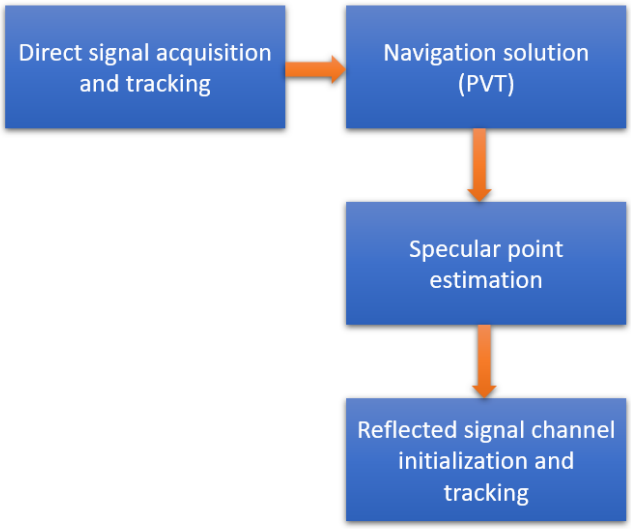


Figure 5-3 Ideal scenario: DDM channel initialization

In Figure 5-4 below, an overview of the geometry is shown. The time taken by the direct signal (t_D) is shown in Equation 5-1 and the time taken by the reflected signal (t_R) is shown in Equation 5-2.

$$t_D = t_{TR} \quad \text{Equation 5-1}$$

$$t_R = t_{TS} + t_{SR} \quad \text{Equation 5-2}$$

After computing the estimates of the specular point, the total range delay between the direct signal and reflected signal is computed using Equation 5-3:

$$\delta\rho_{delay}(t_M) = |\boldsymbol{R}(t_M) - \boldsymbol{S}(t_M)| + |\boldsymbol{T}(t_M - t_R) - \boldsymbol{S}(t_M)| - |\boldsymbol{T}(t_M - t_D) - \boldsymbol{R}(t_M)| \quad \text{Equation 5-3}$$

where $\delta\rho_{delay}(t_M)$ is range delay difference between the direct and reflected signals, $\boldsymbol{R}(t_M)$ is the receiver position, $\boldsymbol{T}(t_M)$ is the transmitter position and $\boldsymbol{S}(t_M)$ is the specular point position (Gleason 2019; Li et al. 2022). Subsequently, the code phase delay, Doppler frequency and carrier phase can be estimated using the following equations, respectively:

$$\phi^r_{code}(t) = \phi^d_{code}(t) - \frac{\delta\rho_{delay}(t)}{c \cdot \alpha_c} \quad \text{Equation 5-4}$$

$$\varphi^r_{carr}(t) = \phi^d_{carr}(t) - \frac{\delta\rho_{delay}(t)}{c \cdot \alpha_c} \cdot f_{RF} \quad \text{Equation 5-5}$$

$$f^r_{carr}(t) = f^d(t) - \frac{\partial}{\partial t}\left(\frac{\delta\rho_{delay}(t)}{c \cdot \alpha_c}\right) \cdot f_{RF} \quad \text{Equation 5-6}$$

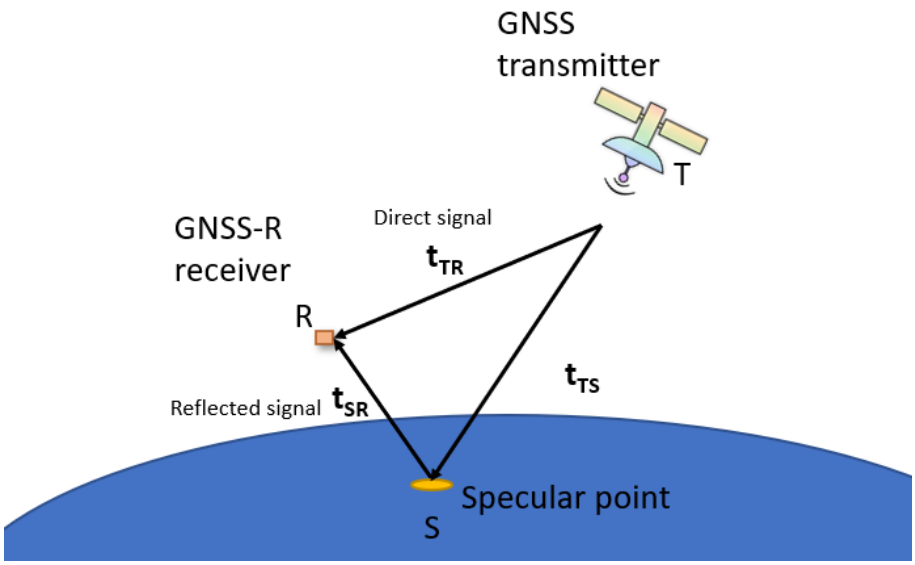


Figure 5-4 Specular point calculation

where ϕ^r_{code} is the code phase, φ^r_{carr} is the carrier phase and f^r_{carr} is the estimated carrier frequency. Speed of light is denoted by c and α_c is the range in metres per code chip (L1-C/A).

Although the DDM channels are initialised with the aid of the receiver's navigation solution (PVT) and recognising that the GNSS-R receiver discussed in this research does not have this capability yet, alternative methods to initialise and test the developed DDM generator were adopted.

5.2.2 Using complex waveform products

Given the limited abilities of the implemented GNSS-R receiver (Chapter 3 and 4) to yet compute a receiver position, alternative methods to test the DDM channels are assessed. As this GNSS-R receiver is meant to operate as a spaceborne LEO receiver. It is only

fitting that the DDM modules be tested with real data published by the missions CYGNSS and TDS-1. CYGNSS has hundreds of raw datasets (~60 seconds in duration) available online for anyone to download and use. On the other hand, Merrbys (TDS-1) offers only three limited raw datasets of about 20 seconds each (and access must be authorised).

Given that the DDM testing and validation will use CYGNSS and TDS-1 datasets, complex waveform products periodically released by Institute of Space Sciences (ICE, CSIC) in collaboration with Institut d'Estudis Espacials de Catalunya (IEEC) are utilised. The institute processes raw IF data released by multiple establishments and processes it using a ground-based software receiver (GOLD-RTR server) (Li et al. 2022).

The data products are released in netCDF4 format and a breakdown of the structure is shown in Figure 5-5. The boxes marked are the products that are used in the developed GNSS-R receiver for DDM module initialisation and validation. As an example, the bistatic delay products present in the cWF dataset are used to predict reflected code phase, carrier phase and carrier signal frequency (Equation 5-4 to Equation 5-6). With these predicted parameters, the DDM code and carrier replicas are initialised and tested. A list of the files that were selected for testing is provided in the next section.

5.2.3 FPGA implementation

In the FPGA, modular code and carrier replica generators are designed and implemented. Depending on the preferred DDM resolution, the number of code delay bins (N_c) and number of Doppler bins (N_d) must be defined prior to the synthesis stage in the FPGA

development cycle. All other configuration parameters such as the difference between each bin and the total range of DDMs can be determined via software after programming the FPGA. These parameters can also be modified during an active DDM generation and the changes will take effect in the next iteration.

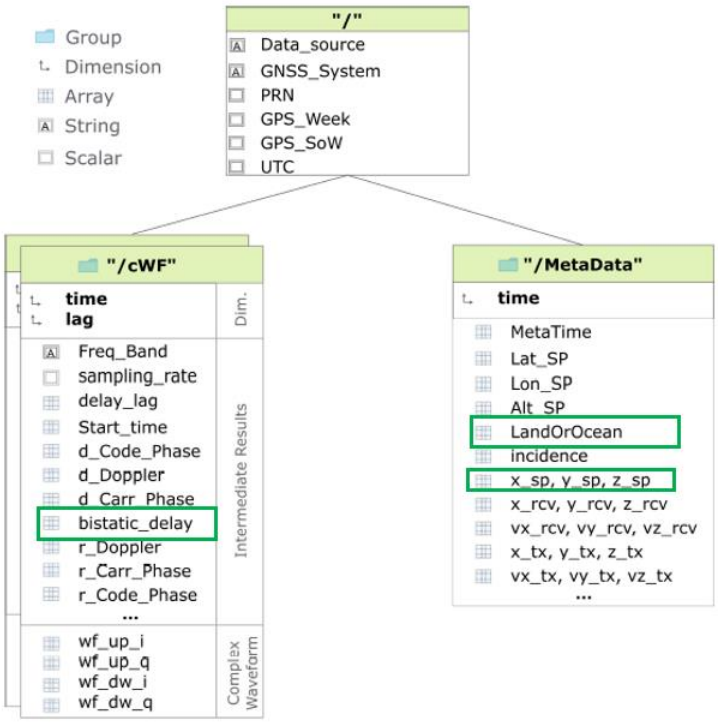


Figure 5-5 Structure of the netCDF4 format cWF products released by ICE-CSIC and IEEC (Li et al. 2022).

Figure 5-6 demonstrates the FPGA implementation of a 3x3 DDM. Three code replicas of varying delays and three carrier replicas of varying Doppler shifts are generated and correlated with the input GPS L1-C/A signal. The resulting data are coherently integrated over 1 millisecond. Each of these “frames” of data are then noncoherently integrated over

between100 to 500 milliseconds, depending on the SNR of the reflected signal. At the end of the noncoherent integration period, the data are transferred to a local memory block. Similar to the DDM in Figure 5-6, the GNSS-R receiver implementation has a DDM dimension of 40 delay x 20 Doppler bins.

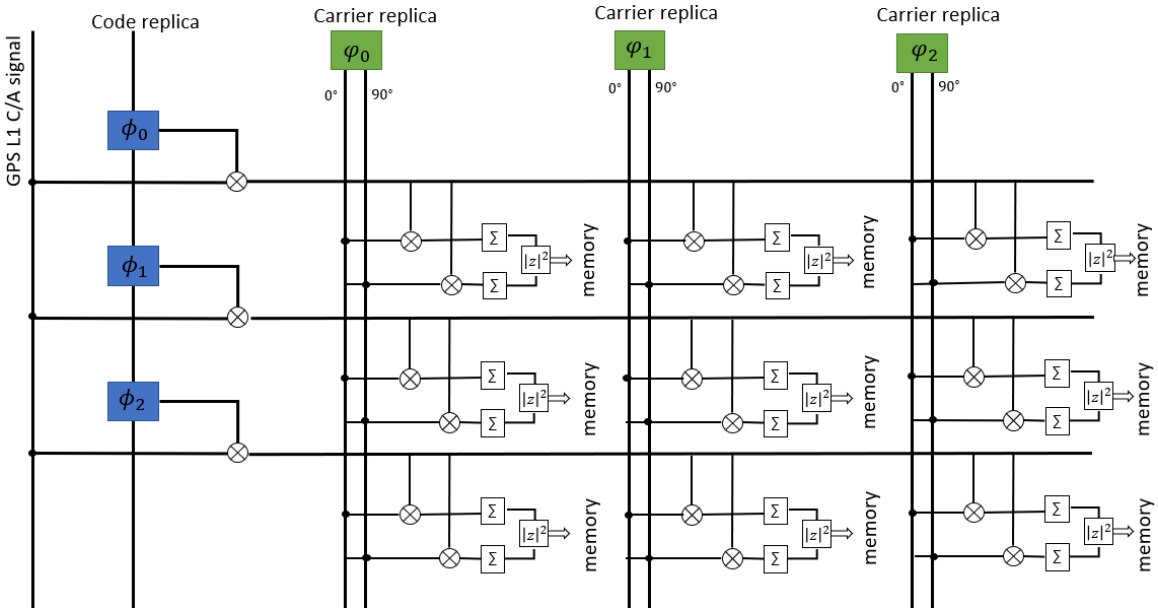


Figure 5-6 A 3x3 DDM pixel representation

5.2.4 Real-time operation

In this research, a DDM resolution of 40 delay bins and 20 Doppler bins is selected ($N_c = 40$ and $N_d = 20$). The 40 x 20 resolution is considered high resolution in comparison with the DDMs delivered by CYGNSS (17 x 11). Similar to the data throughput challenges encountered by SGR-ReSI developers (SSTL), this implementation also faces the same ordeals and requires innovative methods to compress the DDMs before either being stored or transmitted.

In the developed DDM generator, the data computed for each pixel (32-bit) is stowed in a block RAM that is later accessed by the processor. The processor organizes the datasets into files which are saved to an SD card. These files can be retrieved and analysed later.

Considering that the GNSS-R receiver discussed in this dissertation is built within the context of space applications, data volume is a significant attribute of the receiver that must be characterised. Afterall, the primary reason SGR-ReSI compresses DDM images is to reduce the amount of data that needs to be downlinked. The DDM compression method used by the instrument retains only a limited number of delay bins around the specular point and truncates information from the outer delay bins. The truncated DDMs are generated at the frequency of 2 Hz, given that the noncoherent integration period is approximately 500 milliseconds. The dimensions of the truncated DDMs regularly downlinked are insufficient for many scientific applications (Musko et al. 2020). The CYGNSS science operation center (SOC) team, collects both raw IF data and full DDMs over hurricane events with the expectation that better wind estimates can be made.

In the current implementation, the dimension of the DDM is 40 x 20 resulting in a total of 800 datapoints or “pixels”. Each pixel is represented using a 32-bit (4 byte) register. The total number of bytes to represent a single DDM is therefore 800 x 4 = 3200 bytes. Since the frequency of DDM generation is 2 Hz, at the end of the noncoherent integration period, data associated with every pixel is serially written to a local block RAM. There is sufficient time for the processor to transfer it onto the SD card before the next DDM data is ready.

5.3 Results and analysis

In this section, the implemented DDM is tested using datasets published by NASA's CYGNSS. The test objectives are:

1) To demonstrate that the DDM generator can detect weak reflected GPS L1 C/A signals using 1-bit signal resolution.
2) To verify the ability of the implementation to generate high resolution DDMs in real-time.
3) To validate successful open loop tracking to steer the DDMs in the predicted delay/Doppler range.
4) To compute and record real-time SNR upon the generation of a DDM.
5) To predict if specular point is located on land or water by analysing DDMs.

5.3.1 Metadata

To test the FPGA-based DDM generator, two datasets distributed by CYGNSS is utilised. Each of these datasets contain raw IF data in the form of binary files. Each of these files are also tagged with metadata containing information regarding the ID of the spacecraft, the RF front ends contributing to the data file and the associated sampling and IF frequency. Prior to processing the datasets, with some software code, the metadata information of each file is extracted. A summary of metadata from each of the files is shown in Table 5-2. Data collected on 15th January 2022 is referred to as *dataset A* whereas data collected on 25th August 2018 is referred to as *dataset B.*

Table 5-2 Extracted metadata for each of the datasets of interest

	Dataset
Mission	CYGNSS
Number of front ends	3
Sampling frequency	16.0362 MHz
IF	3.8724 MHz
Data bit resolution	2

5.3.2 Validation of DDM generator with direct signal processing

Without adding the complexity of reflected signal processing and other estimates such as code delay and Doppler shift, a simple test to verify the DDM generator module is to use it to process direct GPS L1-C/A signals. Referring to Table 5-2, direct signals from dataset A is used.

In Figure 5-7, the DDM of PRN 15 is shown. A clear peak occurring in pixel (11, 36) can be seen. Figure 5-8 also shows the correlation peak but from a different perspective indicating that the DDM generator can detect direct signals and hence validating the logic of the module. As a side note, the peak is not centered in this scenario and it is not a concern as the tracking loops shift the delay and Doppler to pull the peak in the more central region of the map.

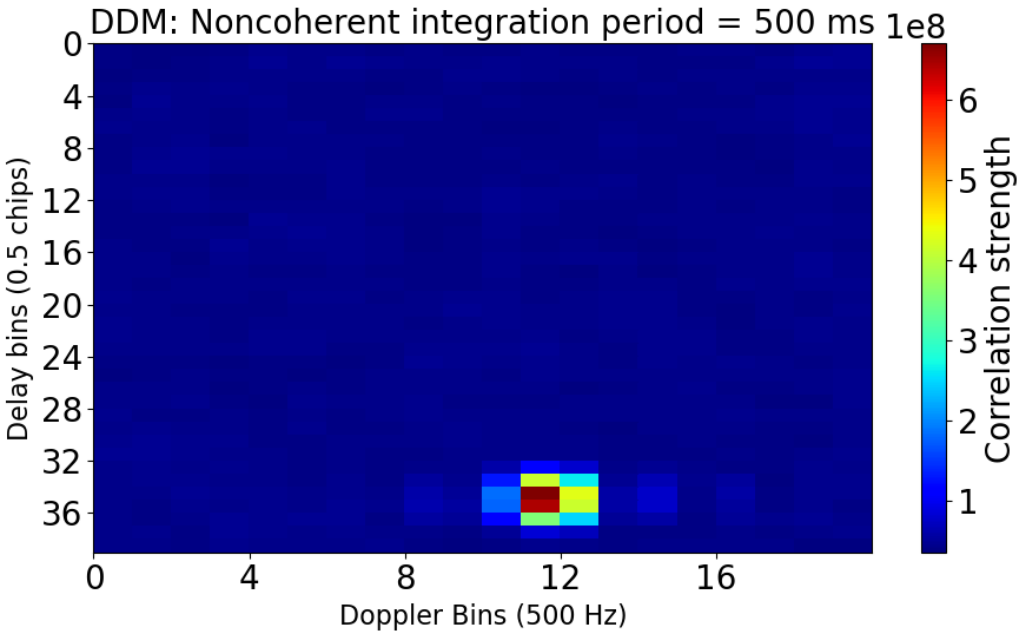


Figure 5-7: Direct signal processing using DDM generator (PRN 15 in Dataset A)

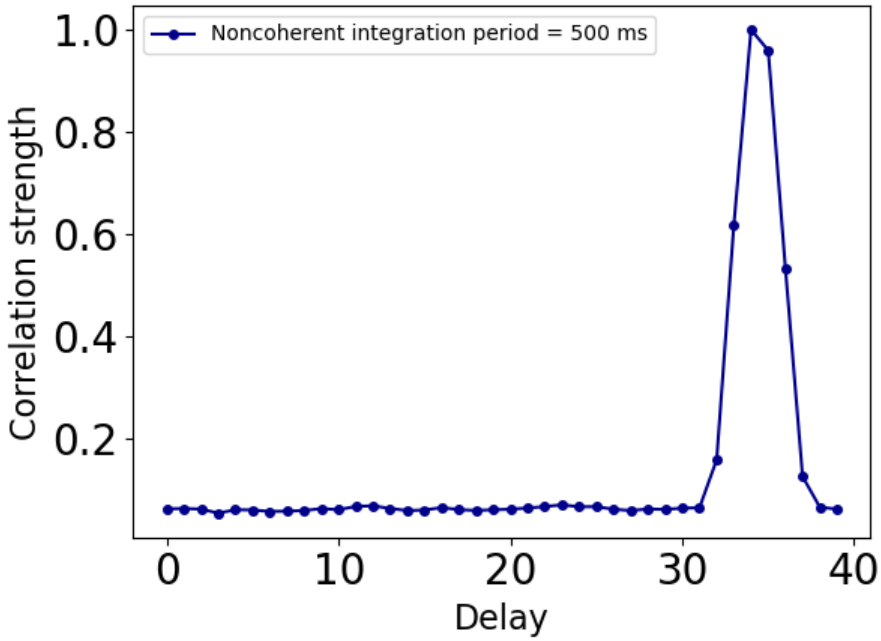


Figure 5-8: Correlation peak for direct signal from PRN 15 (Dataset A)

5.3.2.1 Results from Dataset A (CYGNSS)

The DDM generator upon validation with direct signals is now tested with reflected signals from each of the datasets A and B. In Figure 5-9, the raw IF tracks of the predicted specular point are highlighted in red. An inset of the map is also shown where every second is represented with one marker or dot. The dataset duration is about 60 seconds and as seen in the map, the specular point over the course of time, transitions from land to water and vice-versa several times. The land-water-land transition is precisely the reason this dataset is selected. This dataset can demonstrate the ability of the DDM generator to differentiate between the signals received from land or water reflection. The DDM generator also examines the power spread in the reflected signals and tags the DDM map as reflection from either land or water.

In Figure 5-9, two specular points are marked SP_1 and SP_2. The DDM captured at each of these points are shown in Figure 5-10 and Figure 5-11, respectively. The DDM generated for the land reflection was accumulated noncoherently for 100 milliseconds. The peak when compared against the code delay (extracted from the DDM in Figure 5-10) is plotted in Figure 5-11. From both figures, it can be seen that the reflections occurred on land since there is very limited scattering. It becomes more evident when the trailing edge (after the peak) in Figure 5-11 is observed as the correlation power drops sharply. Upon further modelling and data analysis, other geophysical parameters can be derived from the DDMs; however, these implementations are out of scope for this dissertation and is proposed future work.

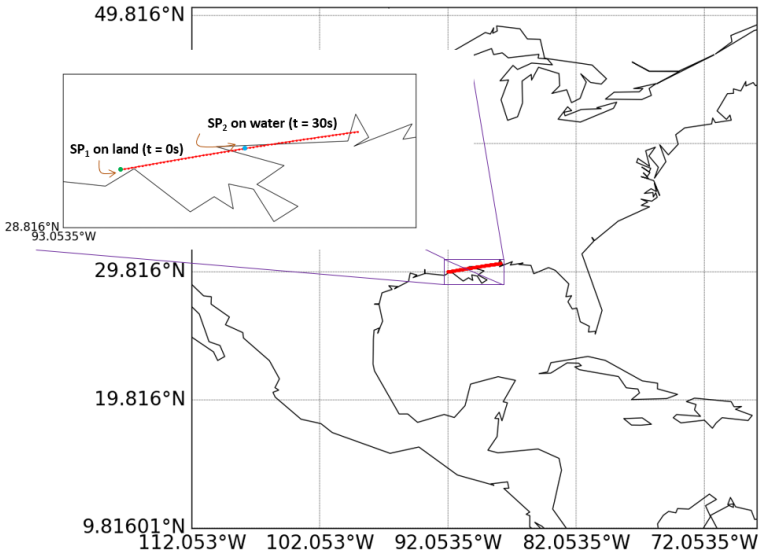


Figure 5-9 Raw IF tracking of predicted specular points (PRN 15 - dataset A)

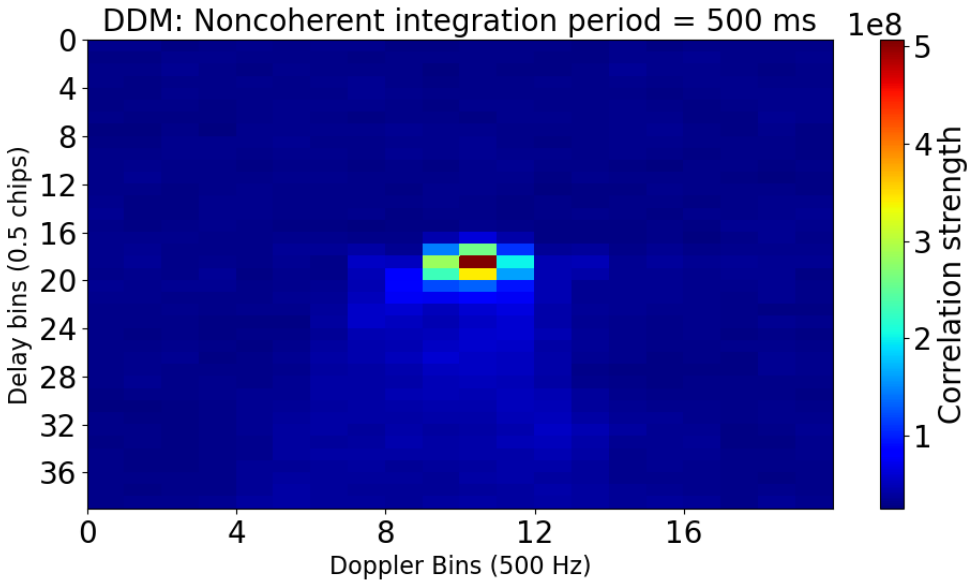


Figure 5-10 Reflected signal DDM generated for SP1 (PRN 15 - dataset A)

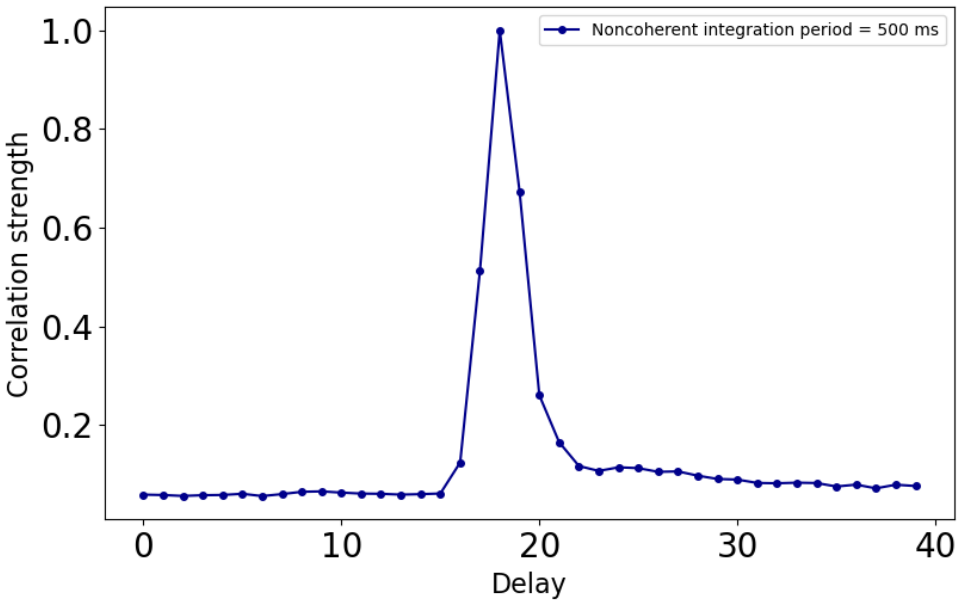


Figure 5-11 Reflected signal correlation peak (PRN 15 - dataset A - SP1)

For the point SP2 highlighted in the track (Figure 5-9), the DDM and reflected signal peak plot is presented in Figure 5-12 and Figure 5-13, respectively. In Figure 5-12, the well-known “horseshoe” pattern can be seen, indicating that the reflection took place on the surface of water. The reflected power is scattered across various delays and the highest correlation peak can be seen in pixel (10, 17).

In Figure 5-13, data from the DDM at the central Doppler bin is extracted and presented. The trailing edge of the peak has a very slow descent and is one of the most evident features of water-based reflections. Again, the characteristic of the trailing edge is of great interest to the scientific community as various geophysical properties can be derived from this attribute.

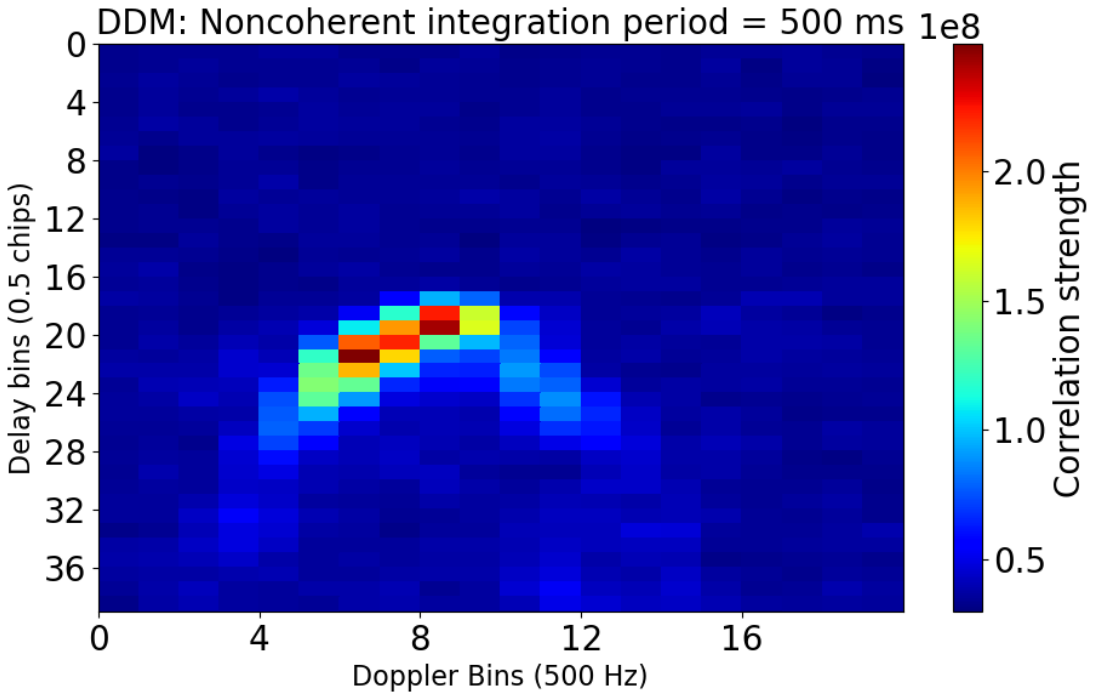


Figure 5-12 Reflected signal DDM at t = 30s (PRN 15 - dataset A at SP2)

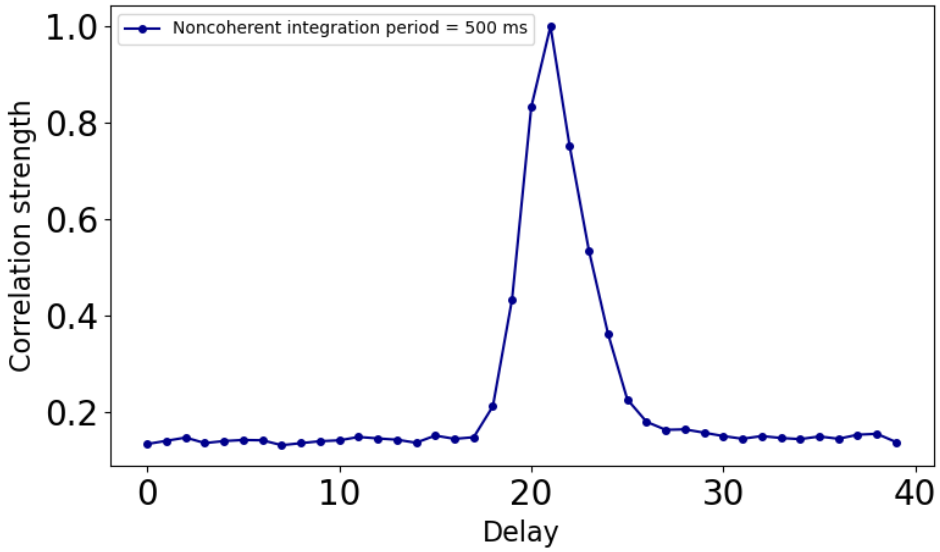


Figure 5-13 Reflected signal plotted against code delay (PRN 15 - dataset A at SP2)

5.3.2.2 Results from Dataset B (CYGNSS)

The raw IF data collected by CYGNSS on August 25, 2017 has been of great interest in the reflectometry community. Hurricane Harvey was approaching mainland and making a landfall on this day and several raw IF datasets have specular points located over the hurricane itself. Once such raw IF track is dataset B. Similar to the test in the case of dataset A, the DDM generator processes the dataset B as well. The location of the raw IF tracks (specular point) is shown in Figure 5-14.

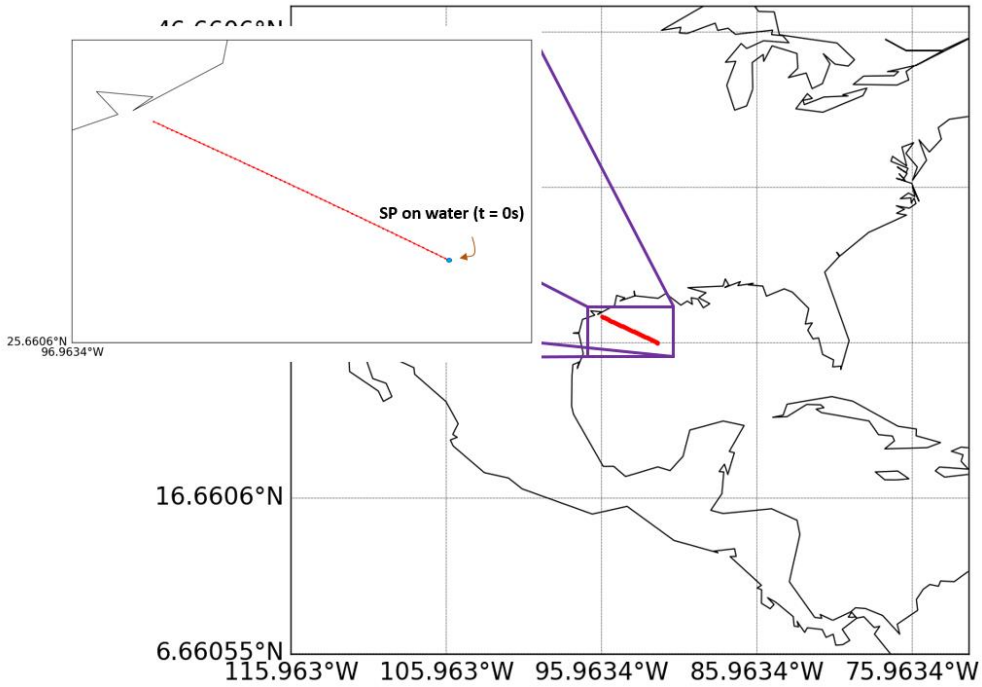


Figure 5-14 raw IF tracks of specular point in dataset B (PRN 28)

The DDM of one of the specular points (SP) is shown in Figure 5-14 and Figure 5-15 highlights the trailing edge of the correlation peak. In this case, the noncoherent

integration period had to be extended to 500 milliseconds to suppress the noise and obtain a better signal to noise (SNR) ratio, which is discussed in detail in the next section.

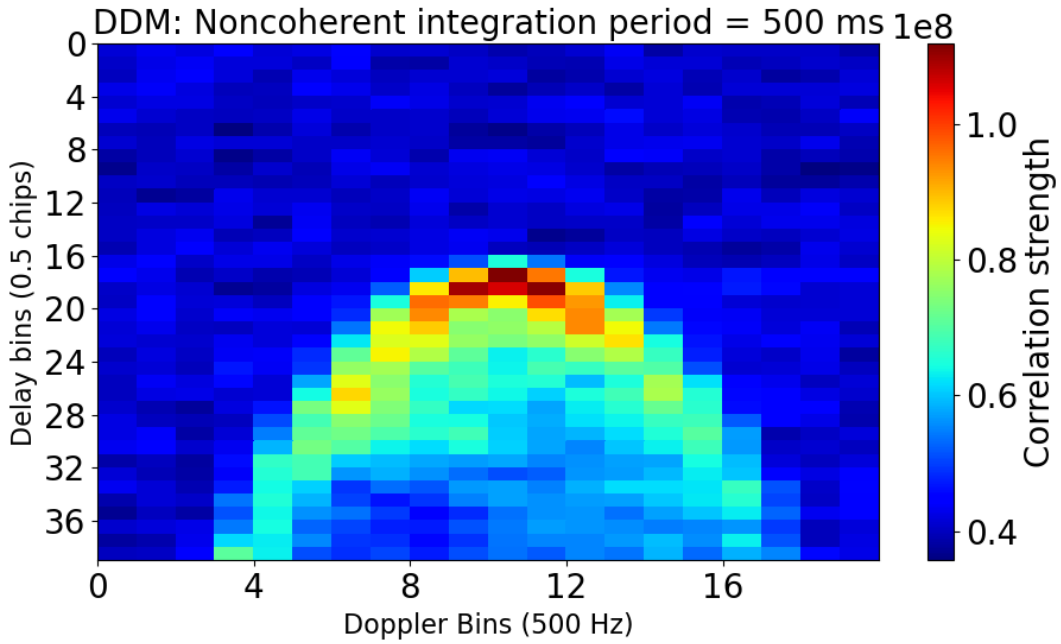


Figure 5-15 DDM showing power scattered across various bins (PRN 28 - dataset B)

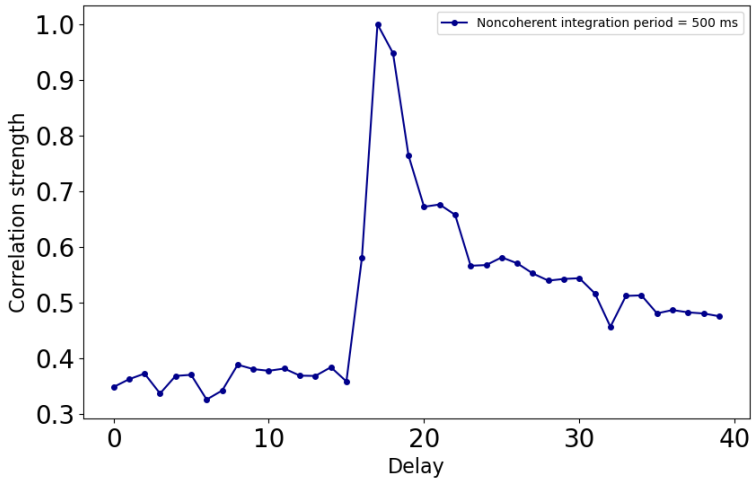


Figure 5-16 Trailing edge of peak indicates water reflections. (PRN 28 – dataset B).

5.3.3 Signal-to-noise ratios

The power of the reflected signal is affected by various factors such as transmitted power, receiver antenna gain, surface bistatic radar cross section (BRCS) and autocorrelation properties of the signal itself (Nan et al. 2021). In this case, within the context of DDM and noncoherent integration, to compute the SNR of the DDM two main components are considered, average noise and peak value.

To compute the average noise, the average value of the "noise box" is computed. The noise box is highlighted in Figure 5-17. Also shown is the peak "box", where all the values around the peak are averaged.

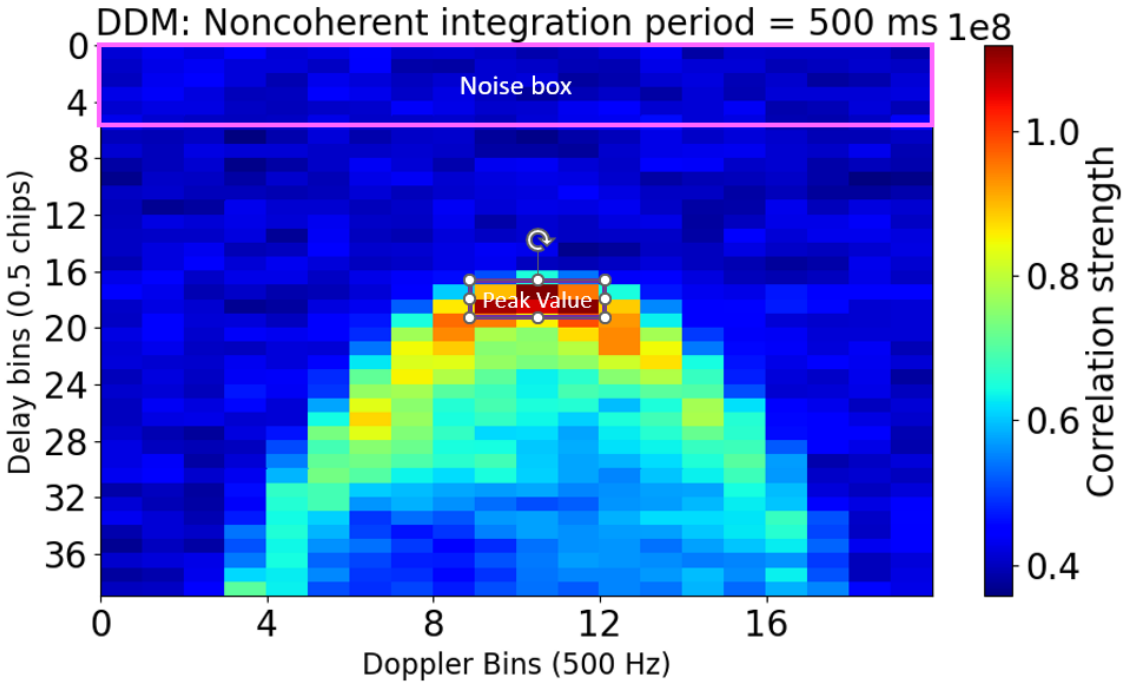


Figure 5-17 DDM with noise box

It is also interesting is to see a comparison of DDM SNR collected from land and water reflections as seen in Figure 5-18. The SNR is extracted from dataset A where SP1 and SP2 are known land and water reflections, respectively. It is observed that the reflections arriving from water have a lower SNR in comparison to land. This difference could be attributed to scattering of the power across a wider glistening zone. Further calibration of the receiver is required to further predict and correlate SNR with the geophysical parameters.

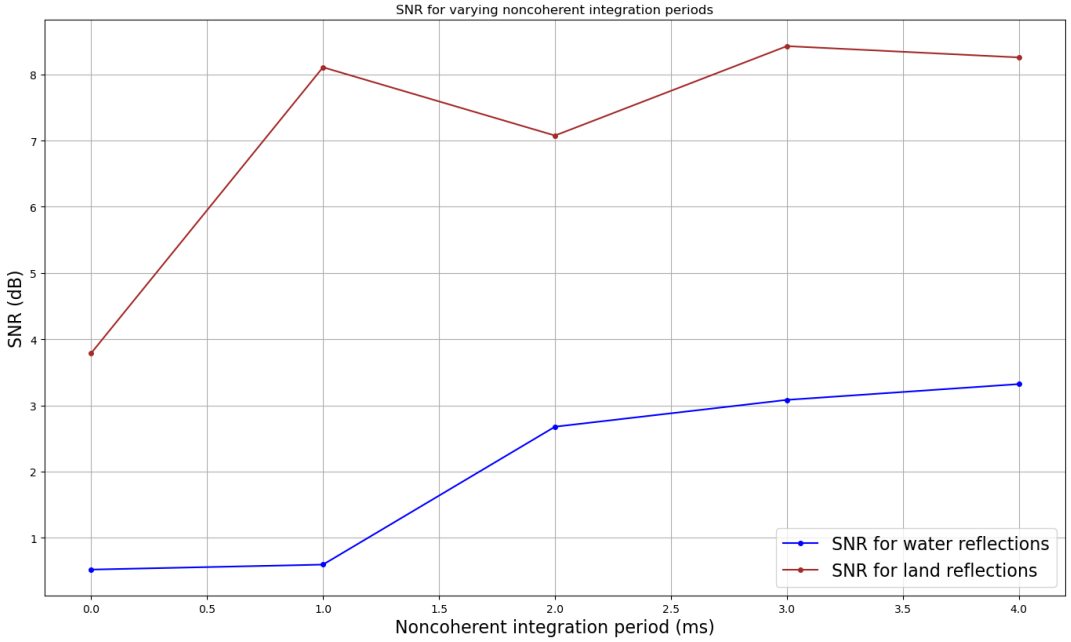
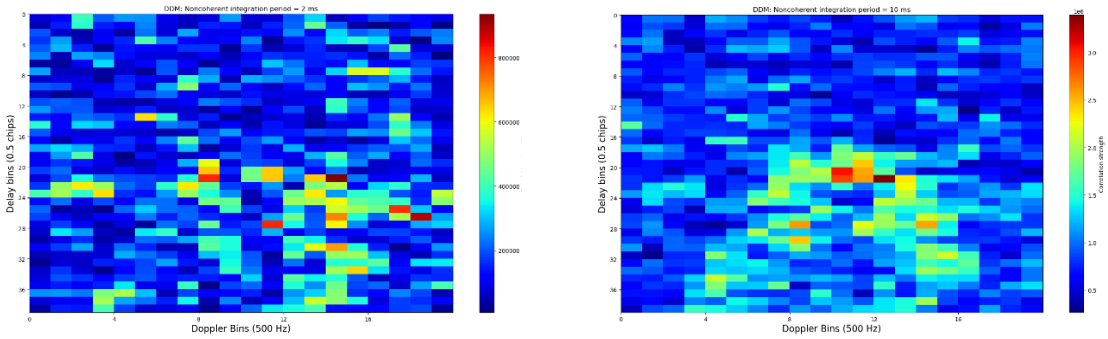


Figure 5-18 Comparison of SNR between land and water reflections for varying integration times.

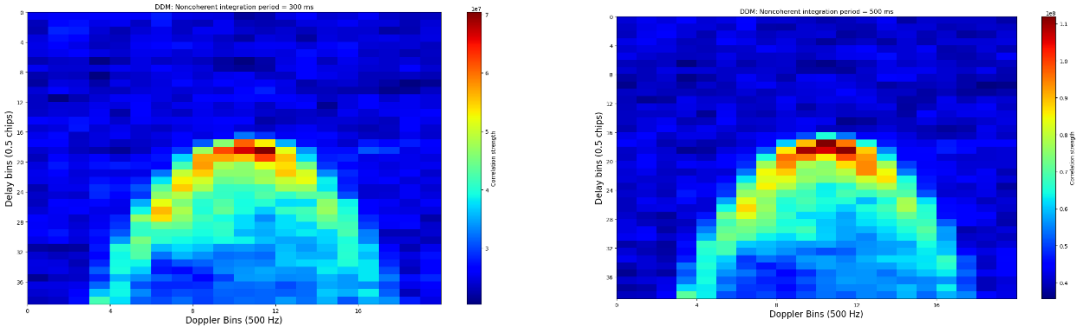
Also demonstrated in a series of figures (Figure 5-19) is the expected increase in SNR with increase in integration time as the noise gets further suppressed. Set of DDMs

shown in Figure 5-19 validate that the “horseshoe” pattern becomes more evident with longer integration time.



(a) Integration period = 2 ms

(b) Integration period = 10 ms



(c) Integration period = 300 ms

(d) Integration period = 500 ms

Figure 5-19 (a to d) Demonstrating noise suppression and SNR improvement with longer integration periods

5.3.4 Resource utilization

The FPGA-based DDM generator surprisingly requires negligible amount of FPGA resources to generate DDMs in real-time. The main reason for the design compactness is that all the operations are performed as 1-bit wide signals given that the input GPS L1-C/A signal is quantised from 2 bits to 1 bit. This low usage is an important factor when the generator is used for any scientific calculations as a quantisation loss of -1.96 dB is to be accounted for.

Table 5-3 FPGA resource utilization of DDM generator

Module name	Slice LUTs	Slice Registers	RAM	DSPs
DDM_box	116	230	0	0
Resource Utilization	0.05%	0.05%	0%	0%

5.3.5 Limitations

Although the DDM module is compact, a trade off must be made in terms of the number of parallel DDM channels versus the resolution of each of the DDMs. If the resolution of the DDM is increased, the processor will be busy transferring and recording data for longer and hence might not be able to service other parallel channels in time or might have to drop frames. In this dissertation, the dimension of 40 delay and 20 Doppler bins is sufficient for the processor to keep up with the data transfer. Data transfer strategies will have to be adopted if the bins are to be extended.

5.4 Summary

In this chapter, design and implementation of a real-time FPGA-based DDM generator is discussed. A summary of all the recent spaceborne GNSS-R missions is provided along with their objectives, findings and receiver limitations.

For the FPGA implementation, open loop tracking method is adopted for reflected GPS L1-C/A signals. Using geometrical models, specular point can be estimated and code delays and Doppler frequencies are predicted. The DDM generator is eventually steered to the estimated parameters and after several accumulations, DDMs are generated.

The DDM module is validated using datasets published by CYGNSS available for public use. Two datasets are processed and the results are presented where distinct features between land and water reflections are noted. These features are also compared with cWF products released by ICE, CSEC for validation. The trailing edge of the correlation peak also uncovers the type of surface reflection that occurred. SNR of land and water reflections are compared, where higher SNR in land reflections have been observed. Lower SNR in water reflections can be attributed to scattering effect.

Furthermore, the resource utilization of the developed DDM module is examined, where it is concluded that given the compactness of the design, several DDM channels can be operated in parallel given an efficient data handling strategy is implemented.

To answer the initial research question, despite the 1-bit quantization of the input signal, the FPGA-based DDM implementation is successful in generating a high-resolution DDM at the rate of 2 Hz or higher.

Chapter 6 CONCLUSIONS AND RECOMMENDATIONS

Remote sensing using Earth-reflected GNSS signals is a promising technique of estimating geophysical parameters such as soil moisture estimation, ocean surface wind speed, sea ice level thickness etc. In this dissertation, design and development of a GNSS-R receiver capable of remote sensing is discussed in depth. The GNSS-R receiver is reconfigurable and highly dynamic, making it easier to upgrade or expand (in terms of functionality). This chapter lists the key conclusions made during this research and also offers recommendations for future work.

6.1 Conclusions

Chapter 3, 4, and 5 have each addressed research questions that have no clear answer in the literature. In this chapter, a summary is provided for all the research conducted in this dissertation.

6.1.1 Real-time FPGA-based GNSS receiver for reflectometry

As mentioned earlier, there is no commercially available instrument that can be purchased and customized to gather raw or partially processed GNSS data suitable for reflectometry. Raw data refers to digital GNSS signals captured by the RF front end (using zenith and nadir antennas) and partially processed data refers to the accumulated values in low-level registers. These datasets are essential to conduct further scientific

research in the field of reflectometry. Also, COTS GNSS receivers cannot be repurposed to work as an instrument for reflectometry. Hence, Chapter 3 of this dissertation emphasizes the following:

1) Design and development of a low-cost, real-time reconfigurable GNSS-R receiver that can detect and track direct and reflected GNSS signals (GPS L1-C/A). The results from the field test of direct and reflected signals validate the real-time performance of the processing units such as multipliers and signal generators.
2) The developed GNSS-R receiver uses COTS hardware and hence no additional specialised equipment is required. With a reasonable antenna gain and COTS RF front end and FPGA platform, the receiver can actively detect and track reflected signals.
3) Moreover, the receiver uses 1-bit data resolution throughout the processing and hence the implementation barely requires any resources in the FPGA. As described in Table 3-5, the resource utilization of a single acquisition channel is 0.3% of the FPGA (LUTs). A total of 50 channels can be implemented for a 100% utilization of the available DSPs.
4) Finally, in this chapter it was noted that the C/N_0 of signals are different for land and water reflections. This result indicates that the developed GNSS-R receiver can be used for reflectometry purposes even though it uses only 1-bit signal resolution.

6.1.2 GNSS-R receiver enhancements

The GNSS-R receiver developed is further improved by modifying the acquisition algorithm to have a higher processing gain. The enhancement allows the receiver to successfully detect weaker signals (as low as -140 dBm). The initial receiver sensitivity assessment revealed nominal performance. However, after the modified algorithm of alternate half-bit method was implemented, the sensitivity of the receiver significantly improved:

1) An active worksheet was established to characterize the receiver performance. Receiver sensitivity was assessed using theoretical and simulated methods and was established to be approximately -30 dB (SNR).
2) The hardware implementation of the alternate half-bit method yielded impressive results. The sensitivity of the receiver to weak reflected signals improved from -30 dB to signals being detected as low as -46 dB. The sensitivity of the receiver enables closed loop tracking of weak reflected signals and hence aids DDM generation without having to rely on information from tracking channels (for direct signals).

6.1.3 High resolution DDM generation

The DDM generator implemented in the FPGA exhibits a 40 by 20 resolution, where there are 40 delay bins and 20 Doppler bins. Each code delay bin width is 0.5 chip whereas each Doppler bin is 500 Hz wide. This output is considered as high resolution when compared with other DDM databases such as CYGNSS (17 x 11).

Below are some key conclusions that are derived from the research conducted and presented in Chapter 5.

1) Chapter 5 demonstrated that DDMs can successfully be generated with 1-bit signal resolution. This result leads to an extremely compact and highly customizable DDM generator module in the FPGA. Table 5-3 in Chapter 5 shows that a single DDM channel only occupies a miniscule 0.05% of the available FPGA resources.
2) The developed DDM generator was tested using raw datasets published by CYGNSS (NASA). The results show that the SNR of the signals reflecting off of water is scattered and weaker in comparison to signals reflecting from land. Noticeable differences in land versus water reflections were also observed. As expected, the DDMs for water reflections had a distinct 'horseshoe' shape indicating the scattering of the signal. The results indicate that using a 1-bit signal resolution for DDM generation has substantial potential.

Scientists researching remotely sensed geophysical parameters are currently using low resolution DDMs available on open-source platforms. Even though CYGNSS provides full DDMs (128x20) occasionally, they do not always operate in that mode due to data downlink limits. Ideally, scientists and researchers, should be equipped with a customizable GNSS-R receiver, which is the over-arching objective of this dissertation.

6.2 Recommendations

There are several features that can be further added to the developed GNSS-R receiver to further enhance its capabilities.

Implementation of additional tracking channels: With the implementation of a minimum of four tracking channels, navigation data and ephemeris can be extracted to compute navigation solution. Since all the channels would be operating in real-time, the solution available will also be updated in real-time. With this upgrade, open-loop tracking can also be implemented for reflectometry and the developed unit will be a self-sufficient receiver.

Nadir RHCP signal processing: Studies (Morris et al. 2022) have shown that processing and analysis of reflected RHCP signals can reveal unique parameters, such as soil roughness and can also help separate biomass from ground reflections (Unwin et al. 2021b).

Instrument calibration: In this research, the SNR measured in DDMs are currently uncalibrated. To calibrate the data, following parameters must be taken into consideration: instrument noise figure, receiver gain, antenna pattern. For a flight/space mission additional factors such as attitude information, LNA temperature and measure direct signal power is necessary.

Flight campaign: In this research, the GNSS-R receiver prototype has been established. The next iteration of the receiver will consider flight campaign as proof of concept. Several other research groups have performed similar flights; however, it will be

interesting to see how a real-time GNSS-R receiver with a 1-bit resolution performs at heights greater than a few hundred metres.

Lossless DDM compression technique: one of the many challenges for spacecrafts today is the inability to downlink full DDMs. As a work around, DDMs are compressed on-board the spacecraft. During compression, the resolution of data is lost and has a degrading effect on subsequent research. If DDMs are considered as images with pixel information, these images can be compressed using video compression tools. There are numerous video compression IP cores (such as JPEG) that can perform lossless compression at a high rate. DDM compression will play a big part in the future receiver implementations even though it is the most overlooked area of research today.

Multi-constellation receiver: Capability of the receiver to process more reflected signals from various constellations is beneficial as more data is generated for further scientific research.

Addition of navigation solution to receiver: Since the navigation data from the tracking channel is extracted, the future work entails implementation of the navigation solution algorithm. Moreover, this capability is essential for the development of a standalone GNSS-R receiver.

Custom nadir antenna design: Design of custom high gain nadir antenna dedicated for reflectometry application would be a significant improvement to the current GNSS-R receiver's sensitivity and performance.

Further receiver testing: The developed GNSS-R receiver must be subject to further field tests and controlled environment tests for further validation.

References

Akos, D. M., & Tsui, J. B. (1996). Design and implementation of a direct digitization GPS receiver front end. *IEEE Transactions on Microwave Theory and Techniques*, *44*(12), 2334-2339.

Ba, X., Yang, Y., Li, J., & Chen, J. (2006, November). Differentially coherent acquisition algorithm for indoor GPS. In *2006 IET International Conference on Wireless, Mobile and Multimedia Networks* (pp. 1-4). IET.

Bernabeu J, Palafox F, Li Y, Akos DM (2022) A Collection of SDRs for Global Navigation Satellite Systems (GNSS). Proceedings of the 2022 International Technical Meeting of The Institute of Navigation, pp. 906–919.

Blunt, P. (2007). *Advanced global navigation satellite system receiver design*. University of Surrey (United Kingdom).

Borre, K., Akos, D. M., Bertelsen, N., Rinder, P., & Jensen, S. H. (2007). *A software-defined GPS and Galileo receiver: a single-frequency approach*. Springer Science & Business Media.

Boutros A, Betz V (2021) FPGA Architecture: Principles and Progression. *IEEE Circuits and Systems Magazine* 21:4–29.

Camps, A., Vall· llossera, M., Park, H., Portal, G., & Rossato, L. (2018). Sensitivity of TDS-1 GNSS-R reflectivity to soil moisture: Global and regional differences and impact of different spatial scales. *Remote Sensing*, *10*(11), 1856.

Chew, C. C., & Small, E. E. (2018). Soil moisture sensing using spaceborne GNSS reflections: Comparison of CYGNSS reflectivity to SMAP soil

moisture. *Geophysical Research Letters*, *45*(9), 4049-4057.

Chew, C., Shah, R., Zuffada, C., Hajj, G., Masters, D., & Mannucci, A. J. (2016). Demonstrating soil moisture remote sensing with observations from the UK TechDemoSat-1 satellite mission. *Geophysical Research Letters*, *43*(7), 3317-3324.

Clarizia, M. P., Gommenginger, C. P., Gleason, S. T., Srokosz, M. A., Galdi, C., & Di Bisceglie, M. (2009). Analysis of GNSS-R delay-Doppler maps from the UK-DMC satellite over the ocean. *Geophysical Research Letters*, *36*(2).

Clarizia, M. P., Ruf, C. S., Jales, P., & Gommenginger, C. (2014). Spaceborne GNSS-R minimum variance wind speed estimator. *IEEE transactions on geoscience and remote sensing*, *52*(11), 6829-6843.

Di Martino, G., Di Simone, A., Franceschetti, G., Iodice, A., Riccio, D., & Ruello, G. (2021). Link Budget Analysis for the Modeling of GNSS-R Sea Surface Returns in Far-from-Specular Acquisition Geometries. In *2021 XXXIVth General Assembly and Scientific Symposium of the International Union of Radio Science (URSI GASS)* (pp. 1-4).

Dunn, M. J. (2012). Global Positioning System Directorate Systems Engineering & Integration Interface Specification IS-GPS-200.

Egea-Roca, D., Arizabaleta-Diez, M., Pany, T., Antreich, F., López-Salcedo, J. A., Paonni, M., & Seco-Granados, G. (2022). GNSS User Technology: State-of-the-Art and Future Trends. *IEEE Access*, *10*, 39939-39968.

Emery, B., & Camps, A. (2017). *Introduction to satellite remote sensing: atmosphere, ocean, land and cryosphere applications*. Elsevier.

Foti, G., Gommenginger, C., Jales, P., Unwin, M., Shaw, A., Robertson, C., & Rosello, J. (2015). Spaceborne GNSS reflectometry for ocean winds: First results from the UK TechDemoSat-1 mission. *Geophysical Research Letters*, *42*(13), 5435-5441.

Franklin, G., Esterhuizen, S., Galley, C., Iijima, B., Larsen, K., Lee, M., Young, L. (2018). A GNSS receiver for small-sats enabling precision POD, radio occultations, and reflections. In *CubeSats and NanoSats for Remote Sensing II* (Vol. 10769, pp. 24-34).

Fridman, A., & Semenov, S. (2013). System-on-chip FPGA-based GNSS receiver. In *East-West Design & Test Symposium (EWDTS 2013)* (pp. 1-7). IEEE.

Ganesan, S., & Alenezi, F. (2022). A course on Advanced SOC FPGA in Embedded systems. In *2022 ASEE-North Central Section Conference*.

Gebre-Egziabher, D., & Gleason, S. (2009). *GNSS applications and methods*. Artech House.

Gerlein-Safdi, C., & Ruf, C. S. (2019). A CYGNSS-based algorithm for the detection of inland waterbodies. *Geophysical Research Letters*, *46*(21), 12065-12072.

Gleason, S. (2010). Towards sea ice remote sensing with space detected GPS signals: Demonstration of technical feasibility and initial consistency check using low resolution sea ice information. *Remote Sensing*, *2*(8), 2017-2039.

Gleason, S. (2019). A real-time on-orbit signal tracking algorithm for GNSS surface observations. *Remote Sensing*, *11*(16), 1858.

Gleason, S., Hodgart, S., Sun, Y., Gommenginger, C., Mackin, S., Adjrad, M., & Unwin, M. (2005). Detection and processing of bistatically reflected GPS signals from low earth orbit for the purpose of ocean remote sensing. *IEEE Transactions*

on Geoscience and Remote Sensing, *43*(6), 1229-1241.

Groves P (2013) Principles of GNSS, inertial, and multisensor integrated navigation systems. Artech house.

Guruprasad S (2015) FPGA-based software GNSS receiver design for satellite applications, *MSc thesis*, York University, 41-50.

Guruprasad, S., Bisnath, S., Lee, R., & Kozinski, J. (2016). Design and implementation of a low-cost SoC-based software GNSS receiver. *IEEE Aerospace and Electronic Systems Magazine*, 31(4), 14-19.

Guruprasad S (2019) Design and development of an improved sensitivity reconfigurable GNSS receiver for space-based reflectometry, *Proceedings of the 32nd International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2019)*, Miami, Florida, September 2019, pp. 3437-3450

Hall, C. D., & Cordey, R. A. (1988, September). Multistatic scatterometry. In *International Geoscience and Remote Sensing Symposium,'Remote Sensing: Moving Toward the 21st Century'*. (Vol. 1, pp. 561-562). IEEE.

Hein, G., Pany, T., Walner, S., & Won, J. H. (2006). Platforms for a future GNSS receiver. *Inside GNSS*, *1*(2), 56-62.

Hegarty, C.J., (2011). Analytical model for GNSS receiver implementation losses. Navigation, 58(1), pp.29-44.

Huang, K. Y., Juang, J. C., Tsai, Y. F., & Lin, C. T. (2021). Efficient FPGA Implementation of a Dual-Frequency GNSS Receiver with Robust Inter-Frequency Aiding. *Sensors*, *21*(14), 4634.

Jales, P. (2012). Spaceborne receiver design for scatterometric GNSS reflectometry, *PhD dissertation*, University of Surrey (United Kingdom), pp. 95-135.

Jing, C., Niu, X., Duan, C., Lu, F., Di, G., & Yang, X. (2019). Sea surface wind speed retrieval from the first Chinese GNSS-R mission: Technique and preliminary results. *Remote Sensing*, *11*(24), 3013.

Joseph, A., & Petovello, M. (2010). Measuring GNSS signal strength. *Inside GNSS*, pp. 20-25. Accessed 28 Apr 2022

Juang, J. C., Ma, S. H., & Lin, C. T. (2016). Study of GNSS-R techniques for FORMOSAT mission. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 9(10), 4582-4592.

Kaplan ED, Hegarty CJ (2013) Understanding GPS: principles and applications. Artech house

King, L., Unwin, M., Rawlinson, J., Guida, R., & Underwood, C. (2021). Towards a Topographically-Accurate Reflection Point Prediction Algorithm for Operational Spaceborne GNSS Reflectometry—Development and Verification. *Remote Sensing*, *13*(5), 1031.

Lestarquit, L., Peyrezabes, M., Darrozes, J., Motte, E., Roussel, N., Wautelet, G. & Zribi, M. (2016). Reflectometry with an open-source software GNSS receiver: Use case with carrier phase altimetry. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *9*(10), 4843-4853.

Li, H., Li, Y., Peng, W., & Wen, B. (2012). A novel algorithm for the weak GPS signals acquisition. In *2012 International Conference on Computer Application and System Modeling* (pp. 738-741). Atlantis Press.

Li, W., Cardellach, E., Ribó, S., Oliveras, S., & Rius, A. (2022). Exploration of Multi-Mission Spaceborne GNSS-R Raw IF Data Sets: Processing, Data Products and Potential Applications. *Remote Sensing*, *14*(6), 1344.

Lin, D. M., & Tsui, J. B. (2001, May). A software GPS receiver for weak signals. In *2001 IEEE MTT-S International Microwave Symposium Digest (Cat. No. 01CH37157)* (Vol. 3, pp. 2139-2142).

Loria, E., O'Brien, A., Zavorotny, V., Downs, B., & Zuffada, C. (2020). Analysis of scattering characteristics from inland bodies of water observed by CYGNSS. *Remote Sensing of Environment*, *245*, 111825.

Lowe, S. T., LaBrecque, J. L., Zuffada, C., Romans, L. J., Young, L. E., & Hajj, G. A. (2002). First spaceborne observation of an Earth-reflected GPS signal. *Radio Science*, *37*(1), 1-28.

Manandhar, D., Shibasaki, R., & Torimoto, H. (2006). GPS reflected signal analysis using software receiver. *Journal of Global Positioning Systems*, *5*(1-2), 29-34.

Martín-Neira, M., Caparrini, M., Font-Rossello, J., Lannelongue, S., & Vallmitjana, C. S. (2001). The PARIS concept: An experimental demonstration of sea surface altimetry using GPS reflected signals. *IEEE transactions on geoscience and remote sensing*, *39*(1), 142-150.

Masters, D. S. (2004). *Surface remote sensing applications of GNSS bistatic radar: Soil moisture and aircraft altimetry,* PhD dissertation, University of Colorado at Boulder, pp. 56-57.

Mota, D., Cruz, H., Miranda, P. R., Duarte, R. P., de Sousa, J. T., Neto, H. C., & Véstias, M. P. (2022). Onboard Processing of Synthetic Aperture Radar Backprojection Algorithm in FPGA. *IEEE Journal of Selected Topics in Applied*

Earth Observations and Remote Sensing, *15*, 3600-3611.

Motte, E., Zribi, M., Fanise, P., Egido, A., Darrozes, J., Al-Yaari, A. & Wigneron, J. P. (2016). GLORI: A GNSS-R dual polarization airborne instrument for land surface monitoring. *sensors*, *16*(5), 732.

Munoz-Martin, J. F., Fernandez, L., Perez, A., Ruiz-de-Azua, J. A., Park, H., Camps, A. & Pastena, M. (2020). In-orbit validation of the FMPL-2 instrument—The GNSS-R and L-band microwave radiometer payload of the FSSCat mission. *Remote Sensing*, *13*(1), 121.

Musko S, Mckague D, Um C, Gleason S (2020) CYGNSS DDM Calibration: General Level 1A and 1B Data Products. Algorithm Theoretical Basis Documents, University of Michigan, pp. 15-17

Nan, Y., Ye, S., Liu, J., Guo, B., Zhang, S., & Li, W. (2021). Signal-to-Noise Ratio Analyses of Spaceborne GNSS-Reflectometry from Galileo and BeiDou Satellites. *Remote Sensing*, *14*(1), 35.

Nogués-Correig, O., Galí, E. C., Campderrós, J. S., & Rius, A. (2006). A GPS-reflections receiver that computes Doppler/delay maps in real time. *IEEE Transactions on Geoscience and Remote sensing*, *45*(1), 156-174.

Nidhin, T. S., Bhattacharyya, A., Behera, R. P., Jayanthi, T., & Velusamy, K. (2017). *Understanding radiation effects in SRAM-based field programmable gate arrays for implementing instrumentation and control systems of nuclear power plants. Nuclear Engineering and Technology*, 49(8), 1589-1599.

NTLab (2022) NT1065 Evaluation Kit Datasheet, https://ntlab.lt/wp-content/uploads/2019/07/Brochure-for-NT1065_FMC.pdf, accessed on April,

2022.

Overbeck, M., Garzia, F., Popugaev, A., Kurz, O., Förster, F., Felber, W., & Eissfeller, B. (2015). GOOSE–GNSS Receiver with an Open Software Interface. In *Proceedings of the 28th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2015)* (pp. 3662-3670).

Parkinson, K. J., Mumford, P. J., Glennon, E. P., Shivaramaiah, N. C., Dempster, A. G., & Rizos, C. (2011, November). A low cost Namuru V3 receiver for spacecraft operations. In *IGNSS Symposium* (pp. 15-17).

Pini, M., Falletti, E., & Fantino, M. (2008, August). Performance evaluation of C/N0 estimators using a real time GNSS software receiver. In *2008 IEEE 10th International Symposium on Spread Spectrum Techniques and Applications* (pp. 32-36).

Principe, F., Bacci, G., Giannetti, F., & Luise, M. (2011). Software-defined radio technologies for GNSS receivers: A tutorial approach to a simple design and implementation. *International Journal of Navigation and Observation*, *2011*.

Psiaki, M. L. (2001, September). Block acquisition of weak GPS signals in a software receiver. In *Proceedings of the 14th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS 2001)* (pp. 2838-2850).

Purfürst, T. (2022). Evaluation of Static Autonomous GNSS Positioning Accuracy Using Single-, Dual-, and Tri-Frequency Smartphones in Forest Canopy Environments. *Sensors*, *22*(3), 1289.

Rodriguez-Alvarez, N., Misra, S., Podest, E., Morris, M., & Bosch-Lluis, X. (2019). The use of SMAP-reflectometry in science applications: Calibration and

capabilities. *Remote Sensing*, *11*(20), 2442.

Roosta, R. (2004). A comparison of radiation-hard and radiation-tolerant FPGAs for space applications. *NASA Electronic Parts and Packaging (NEPP) Program JPL D-31228*.

Ruf, C. S., Atlas, R., Chang, P. S., Clarizia, M. P., Garrison, J. L., Gleason, S., ... & Zavorotny, V. U. (2016). New ocean winds satellite mission to probe hurricanes and tropical convection. *Bulletin of the American Meteorological Society*, *97*(3), 385-395.

Ruf, C. S., Gleason, S., Jelenak, Z., Katzberg, S., Ridley, A., Rose, R., ... & Zavorotny, V. (2012, July). The CYGNSS nanosatellite constellation hurricane mission. In *2012 IEEE International Geoscience and Remote Sensing Symposium* (pp. 214-216).

Ruf, C., Gleason, S., Ridley, A., Rose, R., & Scherrer, J. (2017, July). The nasa cygnss mission: Overview and status update. In *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)* (pp. 2641-2643).

Sauriol, B., & Landry, R. (2007, January). FPGA-based architecture for high throughput, flexible and compact real-time GNSS software defined receiver. In *Proceedings of the 2007 National Technical Meeting of the Institute of Navigation* (pp. 708-717).

SSTL (2016) Product manual TDS-1 MERRBys, http://merrbys.co.uk/wp-content/uploads/2017/07/MERRByS-Product-Manual-V2.pdf, accessed on April 2022.

Tran, V. T., Shivaramaiah, N. C., & Dempster, A. G. (2017). Feasibility analysis of

baseband architectures for multi-GNSS receivers. *GPS solutions*, *21*(1), 1-11.

Troglia Gamba, M., Marucco, G., Pini, M., Ugazio, S., Falletti, E., & Lo Presti, L. (2015). Prototyping a GNSS-based passive radar for UAVs: An instrument to classify the water content feature of lands. *Sensors*, *15*(11), 28287-28313.

Tsai, Y. F., Yeh, W. H., Juang, J. C., Yang, D. S., & Lin, C. T. (2021). From GPS receiver to GNSS reflectometry payload development for the triton satellite mission. *Remote Sensing*, *13*(5), 999.

University of Michigan (2016) CYGNSS Handbook, https:// cygnss.engin.umich.edu, accessed on April 202, pp. 4-5.

Unwin, M. J., Pierdicca, N., Cardellach, E., Rautiainen, K., Foti, G., Blunt, P. & Tossaint, M. (2021). An introduction to the HydroGNSS GNSS reflectometry remote sensing mission. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *14*, 6987-6999.

Unwin, M., Jales, P., Blunt, P., Duncan, S., Brummitt, M., & Ruf, C. (2013, March). The SGR-ReSI and its application for GNSS reflectometry on the NASA EV-2 CYGNSS mission. In *2013 IEEE Aerospace Conference* (pp. 1-6).

Unwin, M., Jales, P., Tye, J., Gommenginger, C., Foti, G., & Rosello, J. (2016). Spaceborne GNSS-reflectometry on TechDemoSat-1: Early mission operations and exploitation. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *9*(10), 4525-4539.

Unwin, M., Rawlinson, J., King, L., Foti, G., Hammond, M., & Burger, T. (2021, July). GNSS-Reflectometry Activities on the DoT-1 Microsatellite in Preparation for the Hydrognss Mission. In *2021 IEEE International Geoscience and Remote Sensing*

Symposium IGARSS (pp. 1288-1290).

Valencia, E., Camps, A., Marchan-Hernandez, J. F., Bosch-Lluis, X., Rodriguez-Alvarez, N., & Ramos-Perez, I. (2010). Advanced architectures for real-time Delay-Doppler Map GNSS-reflectometers: The GPS reflectometer instrument for PAU (griPAU). *Advances in Space Research*, *46*(2), 196-207.

Van Diggelen, F. S. T. (2009). *A-gps: Assisted gps, gnss, and sbas*. Artech house.

Van Steenwijk, R. D. V., Unwin, M., & Jales, P. (2010, December). Introducing the SGR-ReSI: A next generation spaceborne GNSS receiver for navigation and remote-sensing. In *2010 5th ESA Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing (NAVITEC)* (pp. 1-7). IEEE.

Wan, W., Larson, K. M., Small, E. E., Chew, C. C., & Braun, J. J. (2015). Using geodetic GPS receivers to measure vegetation water content. *GPS Solutions*, *19*(2), 237-248.

Wang, S., Shi, S., & Ni, B. (2017, May). GNSS-R software receiver and its preliminary experimental results. In *2017 Forum on Cooperative Positioning and Service (CPGPS)* (pp. 162-166).

Weber, D., & Bremer, R. (2008). Universal GPS receiver lets you use a laptop PC for soft baseband processing, https://pdfserv.maximintegrated.com/en/an/AN4237.pdf, accessed on April, 2022, pp. 1-8.

Yang, G., Bai, W., Wang, J., Hu, X., Zhang, P., Sun, Y., ... & Liu, C. (2022). FY3E GNOS II GNSS Reflectometry: Mission Review and First Results. *Remote*

Sensing, *14*(4), 988.

Ziedan N (2006) GNSS Recievers for Weak Signals. Artech Publishing House, pp 17-35.

Zribi, M., Guyon, D., Motte, E., Dayau, S., Wigneron, J. P., Baghdadi, N., & Pierdicca, N. (2019). Performance of GNSS-R GLORI data for biomass estimation over the Landes forest. *International Journal of Applied Earth Observation and Geoinformation*, *74*, 150-158.

Appendix A: FPGA implementation

In this section, further details regarding the acquisition and tracking implementation are described. As shown in Figure A-1, several Verilog modules are instantiated for every tracking channel implementation. The processor uses the AXI Lite interface to control the channel configuration parameters during initialisation. The primary processing occurs in the module defined in *channel.v*. Each channel implementation further consists of the following modules: *prn_gen, nco_gen, word_parity_checker.*

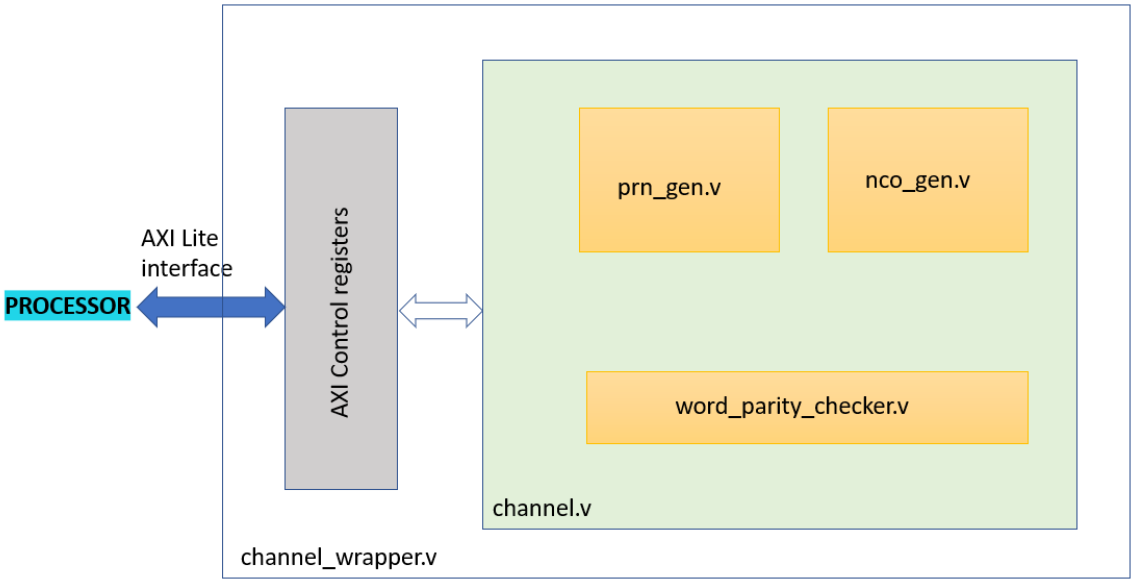


Figure A-1 FPGA architecture demonstrating the various Verilog modules implemented.

As discussed in section 3.2.4., PRN signals are locally generated in the FPGA in real-time. The PRN signals are generated with the same logic implemented on the GPS satellites and described in detail in the GPS ICD published. Since each of these modules

are reconfigurable, the PRN generating module can be configured to generate the PRN sequence of any GPS satellite.

As shown in Figure A-2, each PRN sequence has a total of 1023 chips (*PRN sample count)* spread across one millisecond. Since the PRN sequence phase of the *received signal* is unknown, all 1023 phases of the locally generated PRN sequence is generated sequentially during the search. The parameter *PRN cycle count* keeps a track of all the phases searched for. When the PRN cycle count 1022 is reached with no successful correlation, the channel concludes the search. As seen in the figure, the *PRN phase* is shifted by introducing a delay at the end of every millisecond. Each *channel* monitors the correlation results at the end of every millisecond. If the correlation results exceed the threshold set by the processor, the channel switches to tracking mode where the PRN phase is then locked, that is, no delays are introduced. Since the PRN phase of the received signal and the locally generated signal are aligned, tracking can now commence. For further information related to signal search and acquisition, Kai Borre (2007) is good reference.

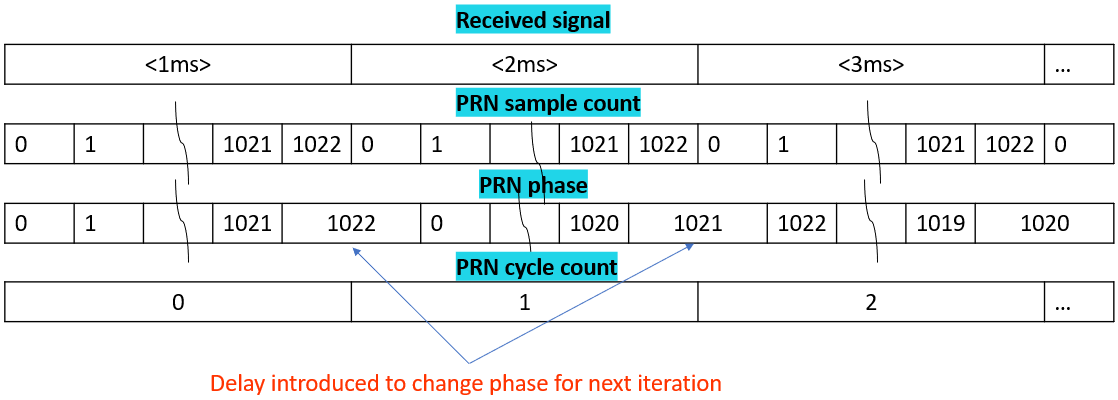


Figure A- 2 PRN generator: Delay is introduced at the end of every millisecond of correlation until either the satellite is successfully acquired or until all 1023 PRN cycle counts are checked for and no satellite was detected.

Appendix B: Link budget analysis for reflected signals

As mentioned earlier, reflected GNSS signals are extremely weak compared to direct signals. In this dissertation, reflected signals with strength lower than –140 dBm are considered to be weak signals that are challenging to detect and track. Several factors affect the signal strength of the reflected GNSS signals and is represented in the form of the following equation (Juang et.al., 2016):

$$P_r = \left(\frac{P_t G_t}{4\pi \|\boldsymbol{\rho}_t - \boldsymbol{\rho}_s\|^2}\right) \cdot \left(\frac{\sigma_0 A}{4\pi \|\boldsymbol{\rho}_s - \boldsymbol{\rho}_r\|^2}\right) \cdot \left(\frac{\lambda^2 G_r}{4\pi}\right) \qquad \text{Equation B-1}$$

The noise power at the output of the receiver is also computed using Equation B-2 (Di Martino et. al., 2021):

$$P_n = k_B(T_a + T_e)B_W \qquad \text{Equation B-2}$$

Using Equation B-1 and Equation B-2, the *SNR* at the output of the correlation step is defined and estimated using Equation B-3:

$$SNR = B_W T_C \sqrt{\frac{T_i P_r}{T_c P_n}} \qquad \text{Equation B-3}$$

The symbols used in the equations above are defined in Table B-2. Typical values used for a spaceborne receiver is also listed. Some symbols have values that can vary depending on the geometry of the transmitter and the receiver such as the incidence angle of the signal will have an impact on the Fresnel zone area.

Table A-1 List of symbols and values used for link budget analysis

Symbol	Parameter	Value
P_t	Transmitted Power	26 W
G_t	Transmitting antenna gain	13 dBi
G_r	Receiving antenna gain	7.25 dBiC
λ	GPS L1 wavelength	0.19 m
A	Scattering area	Varying
$\|\boldsymbol{\rho}_t - \boldsymbol{\rho}_s\|$	Transmitter altitude	20200 km
$\|\boldsymbol{\rho}_s - \boldsymbol{\rho}_r\|$	Receiver altitude	540 km
σ_0	Normalised bistatic cross section coefficient	Varying
T_a	Receiving antenna noise temperature	99.4 K
T_e	Receiver noise temperature	374.35 K
T_i	Incoherent integration time	1 second
T_c	Coherent integration time	1 ms