h e p i a
Haute école du paysage, d'ingénierie
et d'architecture de Genève

Hes·SO GENÈVE
Haute Ecole Spécialisée
de Suisse occidentale

Cost-effective GPS spoofer using Software Defined Radio technology

presented by

Jonas STIRNEMANN

Computing and Communication Systems
Embedded systems

Mars, 2024

Under the guidance of

Fabien Vannel

Mandant

Windshape

Cover illustration legend and source: GPS Block III satellite, https://egnos-user-support.essp-sas.eu/sites/default/files/news/shutterstock_150232019_2.jpg

TABLE OF CONTENTS

h e p i a
Haute école du paysage, d'ingénierie
et d'architecture de Genève

Hes·so GENÈVE
Haute Ecole Spécialisée
de Suisse occidentale

ABSTRACT

In the era of autonomous drones and Unmanned Aerial Vehicle (UAV), there is a pressing need among constructors and major technology firms to establish a robust testing framework that ensures safety, reliability, cost-effectiveness, repeatability, and control, all while providing easily quantifiable metrics. This project endeavors to meet these demands by developing a sophisticated Global Navigation Satellite System (GNSS) spoofer utilizing Software Defined Radio (SDR) technology. The primary objective is to fabricate fake GNSS signals capable of deceiving drones into perceiving forward movement, even when they are stationary in front of an artificial obstacle, such as a fan wall. The overarching goal of this system is to create a controlled testing environment for UAV, facilitating the assessment and quantification of their flying performance metrics. By employing high-frequency position tracking alongside precise control over wind speed and GNSS signals, the system enables UAV to hover in place while experiencing simulated forward motion. This setup provides a comprehensive means of evaluating the UAV's response and performance under dynamic conditions, without the need for costly and potentially hazardous outdoor testing scenarios. The proposed system promises to revolutionize UAV testing procedures by offering a safe, repeatable, and cost-effective solution that can be tailored to specific testing requirements. Moreover, its ability to generate easily quantifiable metrics ensures that developers and engineers can accurately assess the performance of UAV across various flight scenarios. Ultimately, this innovation holds the potential to accelerate advancements in autonomous drone technology by providing a reliable testing platform that accurately reflects real-world challenges.

Candidate:

JONAS STIRNEMANN

Branch : ISC

Professor:

FABIEN VANNEL

In collaboration with: Windshape SA

Thesis subject to an internship agreement: No

Work subject to confidentiality agreement: No

GLOSSARY

AM Amplitude Modulation. 6

BeiDou BeiDou Navigation Satellite System. 5

BPSK Binary Phase Shift Keying. 6, 7

C/A Coarse Acquisition. 7, 8, 16, 20, 23

CDMA Code Division Multiple Access. 7, 8, 24

DGPS Diffential GPS. 20

EGNOS European Geostationary Navigation Overlay Service. 22

FM Frequency Modulation. 6

FMDA Frequency Division Multiple Access. 7

FPGA Field Programmable Gate Array. 9

GAGAN GPS Aided Geostationary Earth Orbit (GEO) Augmented Navigation. 22

Galileo European Global Navigation Satellite System. 5

GEO Geostationary Earth Orbit. vi, 22

GLONASS Globalnaya Navigatsionnaya Sputnikovaya Sistema. 5

GNSS Global Navigation Satellite System. v, 1, 2, 3, 5, 11, 15

GPS Global Positioning System. 1, 5, 7, 8, 11, 12, 14, 15, 18, 19, 20, 22, 26, 28, 43

GSM Global System for Mobile Communications. 20

HEPIA Haute École du Paysage, d'Ingénierie et d'Architecture de Genève. 1

IRNSS Indian Regional Navigation Satellite System. 5

MEO Medium Earth Orbit. 12

MSAS Multi-functional Satellite Augmentation System. 22

PM Phase Modulation. 6

PRN Pseudo-Random Noise. 16

QAM Quadrature Amplitude Modulation. 6

QPSK Quadrature Phase Shift Keying. 6

QZSS Quasi-Zenith Satellite System. 5

R.F. Radio Frequency. 5, 9, 20

RTK Real Time Kinematic. 20

S.V Space Vehicle. 12, 13, 16, 20

SBAS Satellite-Based Augmentation System. 19, 22

SDR Software Defined Radio. v, 1, 5, 9, 27, 28, 32, 43

TDMA Time Division Multiple Access. 7

UAV Unmanned Aerial Vehicle. v, 1

USA United States of America. 16

WAAS Wide Area Augmentation System. 22

LIST OF FIGURES

URL references

- URL01 https://www.gps.gov/systems/gps/space/constellation.jpg
- URL02 https://www.navcen.uscg.gov/sites/default/files/pdf/gps/current.pdf
- URL03 https://www.gps.gov/systems/gps/space/
- URL04 https://www.gps.gov/systems/gps/control/map.png
- URL05 https://commons.wikimedia.org/wiki/File:Phase_modulation_BPSK_GPS.svg
- URL06 https://content.cdntwrk.com/files/aHViPTExODYyNSZjbWQ9aXRlbWVkaXRvcmltYWdlJmZ
- URL07 https://www.researchgate.net/publication/353856800_Survey_and_Performance_Evaluation_of_Multiple_Access_Schemes_for_Next-Generation_Wireless_-Communication_Systems
- URL08 https://commons.wikimedia.org/wiki/File:Gps-atmospheric-efects.png
- URL09 https://syntony-gnss.com/glossary/multipath
- URL10 `https://i0.wp.com/aggps.ca/wp-content/uploads/2021/04/RTK_diagram.jpg?ssl=1`

- URL11 https://globalgpssystems.com/wp-content/uploads/2020/11/inno7-1-600x600-1.jpg
- URL12 https://www.reseau-teria.com/wp-content/uploads/2019/12/SBAS_World_20121212_Extrapolated.png
- URL13 https://i.stack.imgur.com/UOQnn.png
- URL14 https://www.e-education.psu.edu/geog862/sites/www.e-education.psu.edu.geog862/
- URL15 https://www.e-education.psu.edu/geog862/sites/www.e-education.psu.edu.geog862/files/images/Lesson01/NewNavigationMessage.png
- URL16 https://www.e-education.psu.edu/geog862/sites/www.e-education.psu.edu.geog862/files/images/Lesson04/AutonomousGPS2.bmp
- URL17 https://www.ettus.com/wp-content/uploads/2019/01/USRP_B200mini_BD_925x422-1.png
- URL18 https://shop.trenz-electronic.de/media/image/b6/24/a9/32196_0.jpg
- URL19 https://www.mouser.ch/ProductDetail/SparkFun/GPS-18037
- URL20 https://www.swiss-green.ch/1619-large_default/active-gps-antenna.jpg
- URL21 https://www.passion-radio.com/980-thickbox_default/mcx-telescopic.jpg
- URL22 https://www.artisantg.com/itemimages/Mini_Circuits_ZFBT_4R2GW__View2_201818123845.jpg

LIST OF APPENDICES

INTRODUCTION

During our final year of the Bachelor's program in Computing and Communication Systems at Haute École du Paysage, d'Ingénierie et d'Architecture de Genève (HEPIA), we undertake a semester-long project aimed at providing hands-on experience with the tools and technologies relevant to our field. This project serves as a precursor to our Bachelor thesis, allowing us to explore areas of interest and gain practical skills. This project is done in collaboration with Windshape[1], a company that designs, manufactures and operates wind facilities. The company's objective is to offer comprehensive solutions for both aerodynamic research and drone certification. The project detailed in this report focuses on developing a cost-effective, modular and reliable GNSS spoofer using a SDR. This spoofer will play a critical role in a fully controlled testing environment, where we could meticulously assess and certify the performance of UAVs.

The technical challenges of this project are numerous and varied, ranging from understanding the GNSS signal structure to implementing a spoofer capable of generating signals that are indistinguishable from the real ones. This report will only go as far as the proof of concept, but the project will continue in the form of a Bachelor thesis.

The methodology employed in this project began with an initial phase of familiarization with existing GNSS spoofing techniques, alongside the examination of relevant hardware and software tools. Following this, a comprehensive understanding of GNSS systems, with a particular emphasis on the Global Positioning System (GPS) system, was acquired. Initially, authentic GPS signals were captured using both commercial hardware and a software-defined radio (SDR). This was followed by the generation of simulated GPS signals, enabling the spoofing of the same commercial receiver for further investigation and analysis.

The report is structured as follows, the first chapter gives a brief overview of existing systems as a state of the art. The second chapter provides a detailed explanation of the technical concepts that are essential to understanding the rest of the document. The third, fourth and fifth chapters delve into the specifics of the GPS system, exploring its characteristics and limitations. Finally, the sixth chapter presents the different experimentations that were conducted to achieve the project's objectives.

[1] `https://windshape.com/`

CHAPTER 1 : STATE OF THE ART

Some companies are already selling fully configurable GNSS simulators with included software and support for every GNSS technologies. These simulators are very expensive and are not open source, thus hardly customizable. They also are massive and difficult to transport and usually require a dedicated computer with a specific software to control them.

1.1. SPIRENT

Spirent is a company that sells GNSS simulators. They offer a wide range of products from single-frequency to multi-frequency simulators. They also offer a wide range of software to configure and control the simulators. The software is user-friendly and offers a wide range of features. The simulators are very expensive and are not open source.

Here's an example of a Spirent simulator:

FIGURE 1.1: Spirent GSS9000 gnss simulator Source: www.e-education.psu.edu ref: URL14

Here's an idea of the software that comes with the simulator:

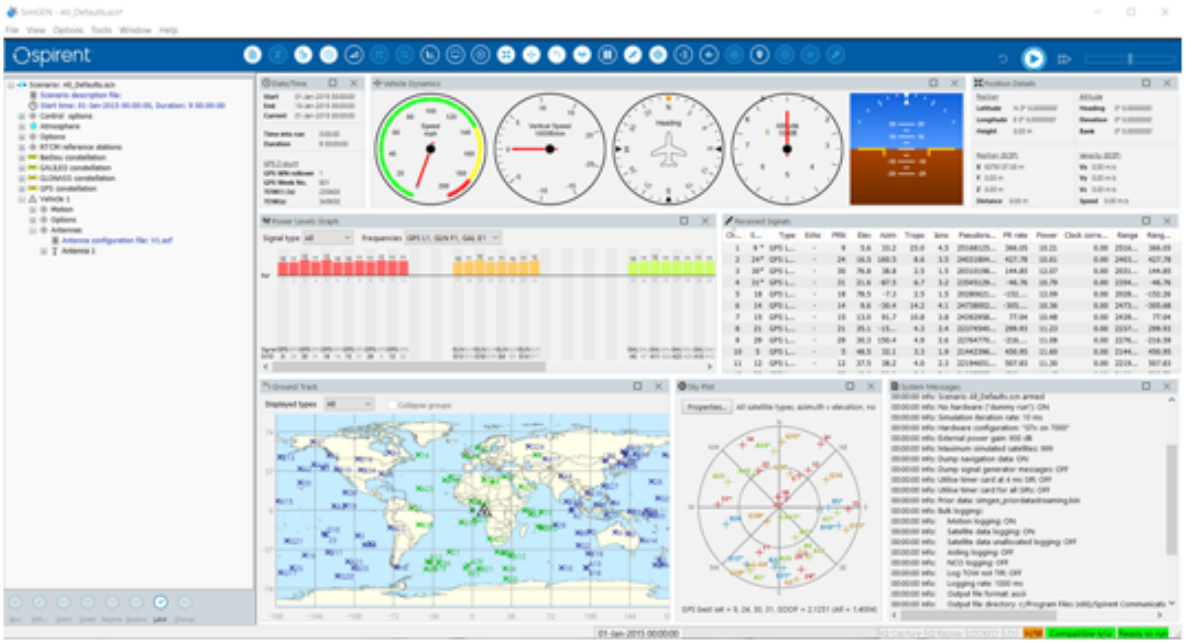


FIGURE 1.2: Spirent existing software Source: www.e-education.psu.edu ref: URL14

1.2. ROHDE N SCHWARZ

Rohde n Schwarz is another company that sells GNSS simulators. They offer a wide range of products from single-frequency to multi-frequency simulators too. Their devices come at around 50'000 $ for a base model ...

Here's an example of a Rohde & Schwarz simulator:

FIGURE 1.3: Rohde n Schwarz Source: www.e-education.psu.edu ref: URL14

CHAPTER 2 : TECHNICAL CONCEPTS

In this chapter, we will brifely explore different technical concepts that are important to understand the rest of the document. We will start by defining what a GNSS is, basic Radio Frequency (R.F.) concepts, trilateration and finally a brief introduction to SDR.

2.1. GNSS

A GNSS is a satellite-based navigation system that provides location and time informations. These systems are used in a massive variety of domains around us and have been adopted by many industries including agriculture, aviation, construction, maritime, mining, public safety, transportation, etc.

Several GNSS have been developed[2] in different countries / regions and are in operation around the world, five of them being worldwide systems, thus available globally anywhere in the world:

- United States's **GPS**
- Russia's **Globalnaya Navigatsionnaya Sputnikovaya Sistema (GLONASS)**
- European Union's **European Global Navigation Satellite System (Galileo)**
- China's **BeiDou Navigation Satellite System (BeiDou)**

Two other systems are regional :

- India's **Indian Regional Navigation Satellite System (IRNSS)**
- Japan's **Quasi-Zenith Satellite System (QZSS)**

We won't go into details about every GNSS but only for GPS which is the most widely used and supported by the majority of devices.

[2] *Satellite navigation*. In: *Wikipedia*. Feb. 23, 2024. URL: https://en.wikipedia.org/w/index.php?title=Satellite_navigation&oldid=1209879496 (visited on 03/16/2024).

2.2. RADIO FREQUENCIES

Radio frequencies are a space within the electromagnetic spectrum associated with radio wave propagation. It is used to transmit information across distances. It's done by modulating the information on a carrier wave. The carrier wave is then transmitted through an antenna and received by another antenna. The information is then demodulated and the original information is recovered.

a. Modulation

Modulation is the process of varying one or more properties of a waveform (Amplitude, Phase, Frequency) in order to encode information or actual bits. The information is then carried by the carrier wave. The process of recovering the original information is called demodulation. We can vary one of the properties at the time or combine them to create more complex modulation schemes. The most common modulation schemes are:

- Amplitude Modulation (AM)
- Frequency Modulation (FM)
- Phase Modulation (PM)
- Binary Phase Shift Keying (BPSK)
- Quadrature Phase Shift Keying (QPSK)
- Quadrature Amplitude Modulation (QAM)

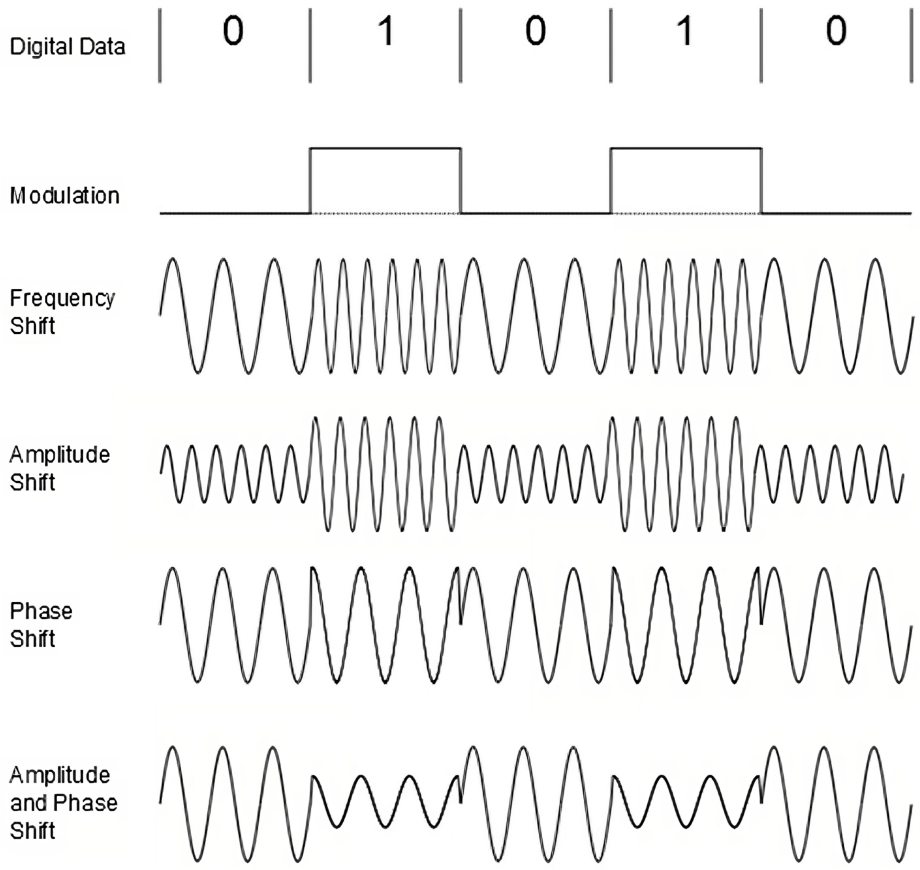


FIGURE 2.1: Example of modulations Source: https://content.cdntwrk.com/ ref: URL06

Note that the GPS system uses BPSK to transmit the Coarse Acquisition (C/A) code.

b. Channel access methods

There exist different methods for multiple devices to communicate on a same transmission medium. Three of the popular methods are Frequency Division Multiple Access (FMDA), Time Division Multiple Access (TDMA) and Code Division Multiple Access (CDMA).

- FMDA divides the available bandwidth into multiple frequency channels, this means each signal sends data on a different frequency.
- TDMA divides the available bandwidth into multiple time channels, this means that each user gets the full bandwidth for a short period of time.
- CDMA uses the exact same frequency at the exact same time, thus multiple transmitters can broadcast data at the same time and on the same frequency. The discrimination

between the different signals is done using a unique code for each transmitter (Cross-Correlation).

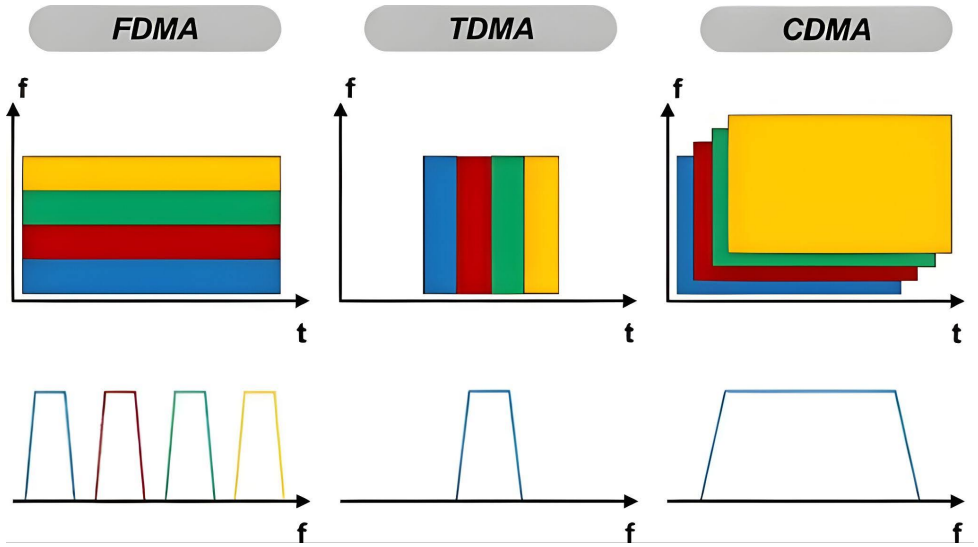


FIGURE 2.2: Channel access methods visual description Source: https://www.researchgate.net/ ref: URL06

The GPS system uses CDMA to transmit the C/A code.

2.3. TRILATERATION

Trilateration is a method used to determine the position of an object using the geometry of the space. Often confused with **triangulation**, trilateration uses distances to determine the position of an object, while triangulation uses angles.

In order to get a position on a 2D plane, we need at least 3 distances from known points. In 3D, we need at least 4 distances from known points.

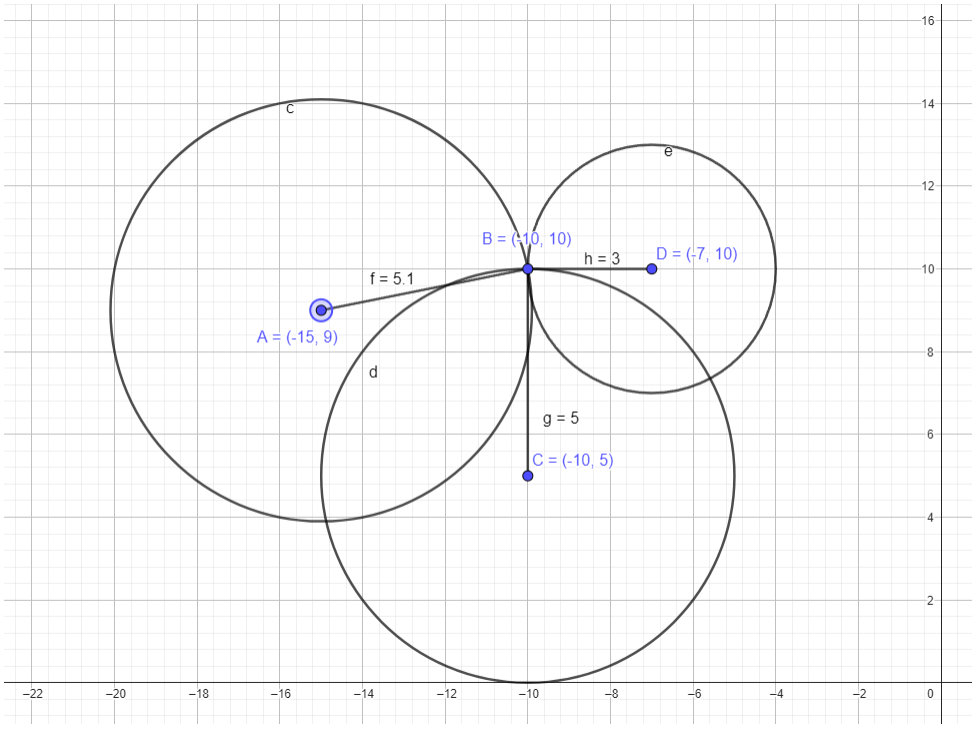


FIGURE 2.3: Trilateration example Source: Made by Jonas Stirnemann

In this example, we have 3 known points A, D and C and we know the distance from the unknown point to each of the known points f, h and g. We can then calculate the position of the unknown point B using the distances and the known positions of the other points.

2.4. SOFTWARE DEFINED RADIO

A SDR is an hardware device that contains a fully fledged R.F. frontend for modulating, creating and transmitting or receiving radio signals. These devices are designed to be connected to a computer via a fast interface like USB or Ethernet. The computer will then be a host multiple softwares that can generate or receive and process data via / from the SDR.

These are very versatile devices that can be used for a wide range of applications like creating a simple FM radio receiver, a GSM base station, a GPS receiver or any simple or complex R.F. based system.

We can observe an example block diagram of an SDR in the next figure. It's composed of an USB port for communicating with the PC, an Field Programmable Gate Array (FPGA) for processing the data and an R.F. frontend for (de)modulating the signals and send/receive them through an antenna (usually SMA connector).

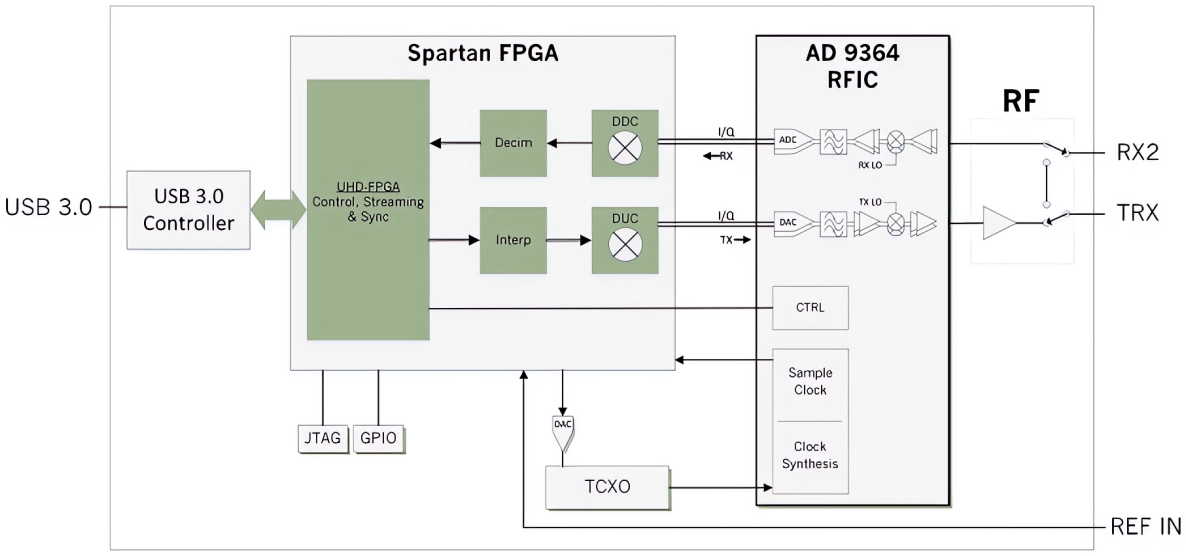


FIGURE 2.4: Example block diagram of an SDR Source: www.ettus.com ref : URL17

CHAPTER 3 : GPS OVERVIEW

In this chapter, we will explore the GPS system, its history, its architecture, the technologies involved, the signals it uses. We will explore the different frequency bands and codes used by the system. However the actual explanation of how the system actually works will be done in the next chapters.

3.1. BRIEF HISTORY

The GPS system was developed by the United States Department of Defense and became fully operational in 1995. It was created to provide a precise and continuous navigation system for the military. The system was opened to civilian use in the 1980s and has since become a global utility. It was the first GNSS to be developed and is still the most widely used and supported by the majority of devices.

3.2. SYSTEM ARCHITECTURE

The GPS system consists of three segments:

- The user segment
- The space segment
- The control segment

a. User segment

The user segment is composed of the receivers that are used to determine the position, velocity and time. These receivers are used in a wide variety of applications, including aviation, agriculture, construction, maritime, mining, public safety, transportation, etc.

b. Control segment

The control segment[3] is composed of a network of ground facilities that are used to monitor and control the GPS satellites. These facilities are used to track the satellites, upload updated navigation data, and ensure the overall health of the satellite constellation. They might

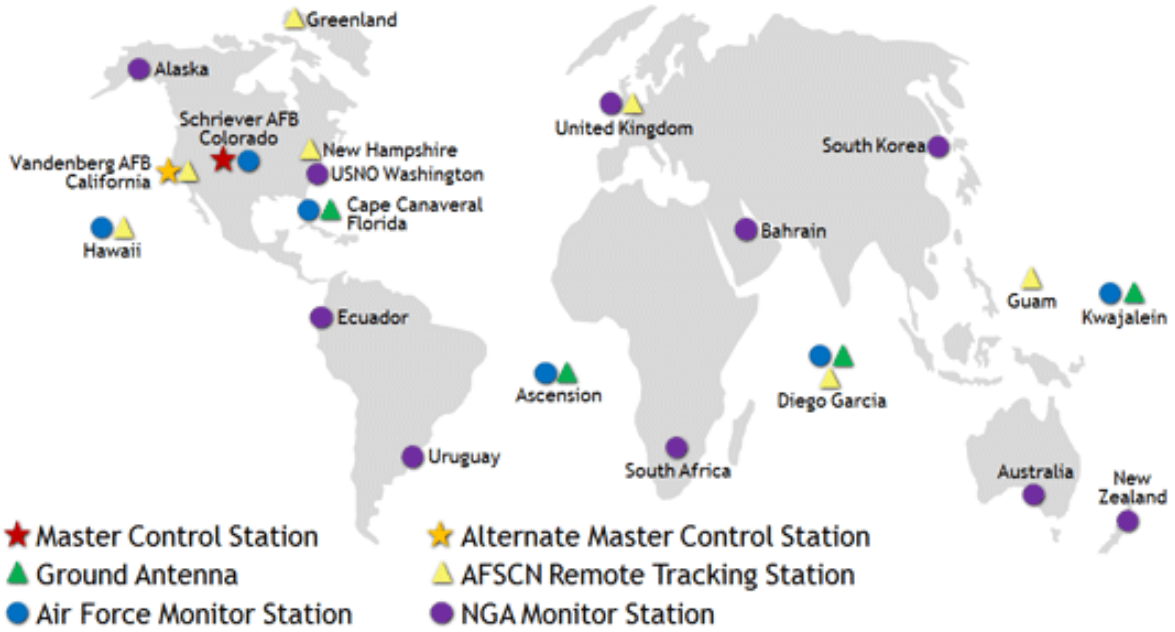


FIGURE 3.1: World map of ground facilities of the Control segment of GPS Source: www.gps.gov ref: URL04

c. Space segment

The Space segment[4] is composed of a constellation of 31 operational sattelites as of 15 Auguest 2023[5]. Each of these satellite (Space Vehicle (S.V)) is in a Medium Earth Orbit (MEO) at an altitude of approximately 20,200 km. These satellites are arranged into six equally-spaced orbital planes, with an inclination of 55 degrees in relation to the equator.

[3] **noauthor_gpsgov_nodate**.

[4] **noauthor_gpsgov_nodate**.

[5] *List of GPS satellites*. In: *Wikipedia*. Mar. 13, 2024. URL: https://en.wikipedia.org/w/index.php?title=List_of_GPS_satellites&oldid=1213499951#PRN_to_SVN_history (visited on 03/16/2024).

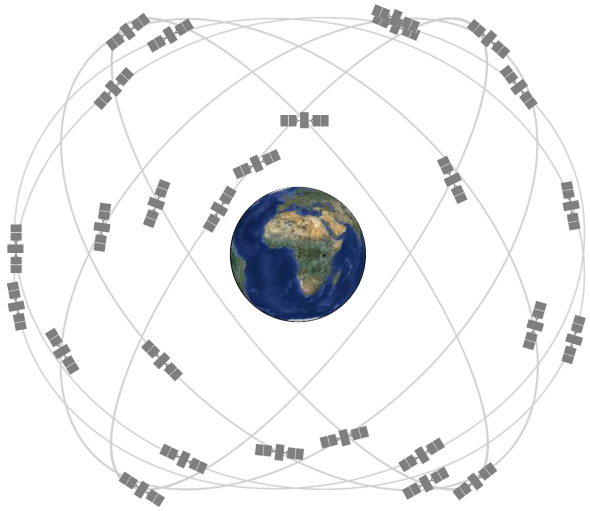


FIGURE 3.2: GPS constellation Source:www.gps.gov ref: URL01

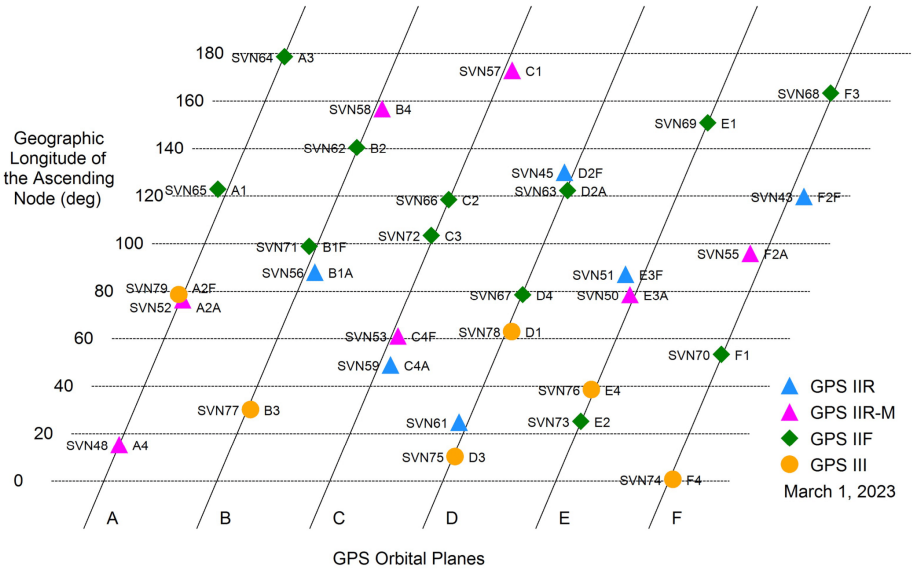


FIGURE 3.3: GPS Orbital planes Source:www.navcen.uscg.gov ref: URL02

Each of these S.V are placed in a Sub-Geostationary orbit, making them rotate two times a around earth each day. These S.V are placed in such a way that at any time, anywhere on Earth, there are at least **six** of them visible with an average of **nine** visible in the sky. Note that these numbers are acheived with the current constellation, but a minimum constellation of 24 sattelites is required for the system to work.

There have been five main satellite models developed for the GPS system, each with different capabilities and improvements over the previous

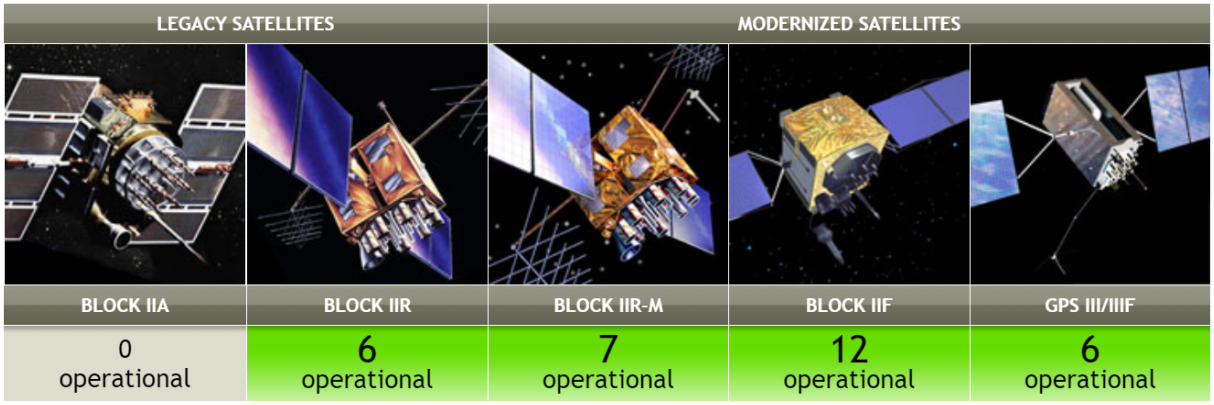


FIGURE 3.4: Current and Future Satellite Generations Source: www.gps.gov ref: URL03

3.3. BASIC GPS PRINCIPLE

The GPS constellation broadcasts signals constantly to the Earth. These signals are used by receivers to determine a position, velocity and time. Time is given as a direct value but position is computed using a process called **trilateration**. This process is based on the time it takes for the signals to travel from the satellites to the receiver and the known position of the satellites (Given in the broadcast messages). We'll explain things from a single bit data to the frames messages to the end user position.

3.4. SATELLITE HARDWARE

Each GPS satellite is equipped with **four** atomic clocks, two rubidium and two cesium. These atomic clocks allow us to generate a **Fondamental Frequency** ($F0$) of 10.23 MHz. This frequency is then multiplied to generate the **L1**, **L2** and **L5** frequencies.

3.5. SIGNALS AND FREQUENCIES

a. Frequency bands

As mentionned earlier, the GPS system broadcasts three signals, each on a different frequency band. These bands are **L1**, **L2** and **L5**.

- The **L1** band, centered at **1575.42** MHz ($F = 154 \cdot F0$).
- The **L2** band, centered at **1227.60** MHz ($F = 120 \cdot F0$).
- The **L5** band, centered at **1176.45** MHz ($F = 115 \cdot F0$).

b. L1 Codes

Four different codes are sent on the **L1** band, two of them are used for civilian use, the **C/A** and **P(Y)** (Precision) codes. One of them is reserved for the military, the **M-code**. The last one is an evolution of the L1 signal : the **L1C**. This signal is designed to be interoperable with other GNSS systems like Galileo.

c. L2 Codes

The **L2** band is used to send two different code, the **L2 CM** and the **L2 CL** codes. These are mostly used for dual-frequency receivers to remove the ionospheric delay. This works by taking

the difference between the two signals and by knowing the actual speed of each frequency in the same medium.

d. L5 Codes

The **L5** band is the third civilian signal[6], it has been introduced to meet demanding requirements for safety-of-life transportation. It is provided as a future proof signal for aviation use. Combined with the C/A code, it provides accuracy and robustness via ionospheric correction and redundancy.

This code is given as a new way for aviation to increase capacity and fuel efficiency within United States of America (USA) airspace, railroads, highways and waterways.

3.6. PRN CODE GENERATION

Each satellite sending glsca code is allocated a unique Pseudo-Random Noise (PRN) code, this code is used to identify the satellite and to allow the receiver to distinguish between the signals of different satellites. The PRN code is a **Gold code** that is generated by a **Linear Feedback Shift Register** (LFSR). This code is a pseudo-random sequence of 1023 bits that repeats every millisecond. This code is then modulated with the carrier frequency to generate the signal that is broadcasted by the satellite.

Each S.V is assigned a unique identifier from which a PRN can be generated. As explained in 3.5 figure, each ID is correspond to a pair of "taps", these taps are the bit positions in the polynomial generator. There is 2 polynomial generator, used together to generate the PRN code.

The generated codes are **Gold codes** or **Gold sequences** so they have a small cross-correlation between them. This means that the receiver can easily distinguish between the signals of different satellites.

[6] *GPS: The Global Positioning System.* URL: `https://www.gps.gov/` (visited on 03/11/2024).

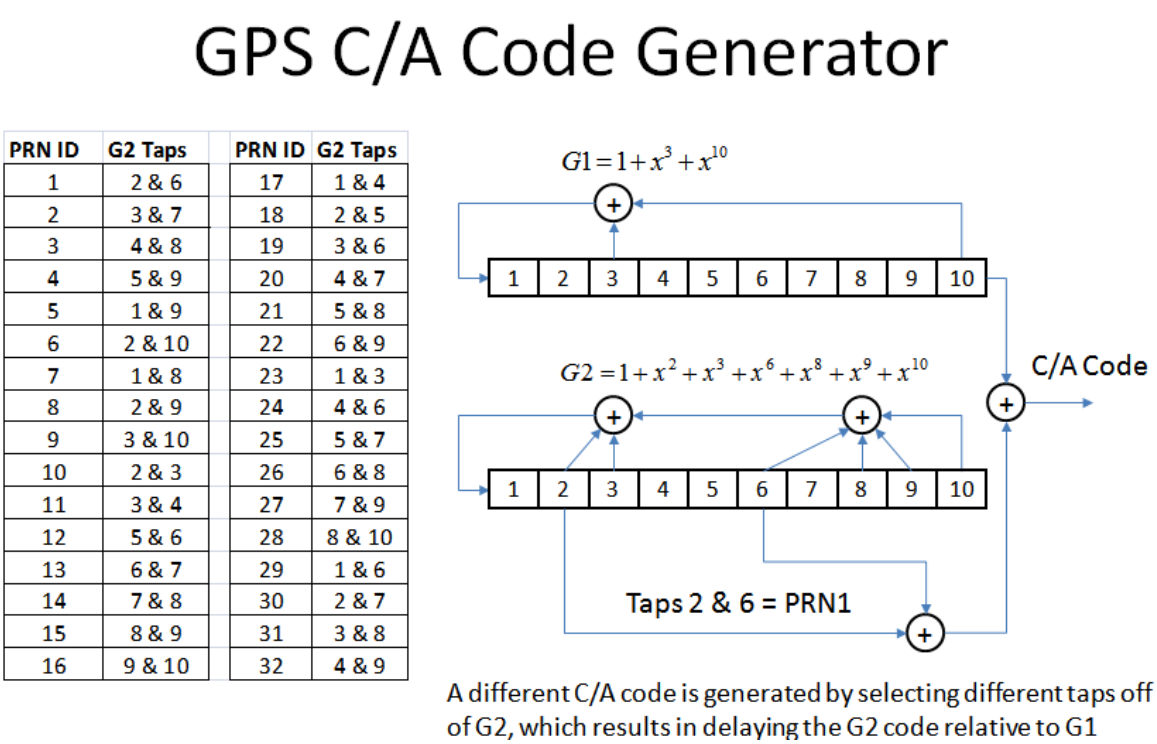


FIGURE 3.5: PRN code generators Source: https://i.stack.imgur.com ref: URL013

CHAPTER 4 : GPS CHARACTERISTICS

There is some accuracy limitations or to the GPS system. Some of them are due to the system itself, others are due to the environment and the receiver. In this chapter, we will explore some of these limitations and the solutions that have been developed to mitigate them.

4.1. ENVIRONMENTAL LIMITATIONS

The GPS accuracy can be affected by 4 main environmental factors. Even small errors in the time measurement can lead to large errors in the position calculation. We'll explore these factors and possible correction methods.

a. Atmospheric effects

The GPS signal is affected by the atmosphere. Some of these layers can change the way the signal propagates, thus affecting the time it takes for the signal to reach the receiver. The main layers that affect the signal are the ionosphere and the troposphere.

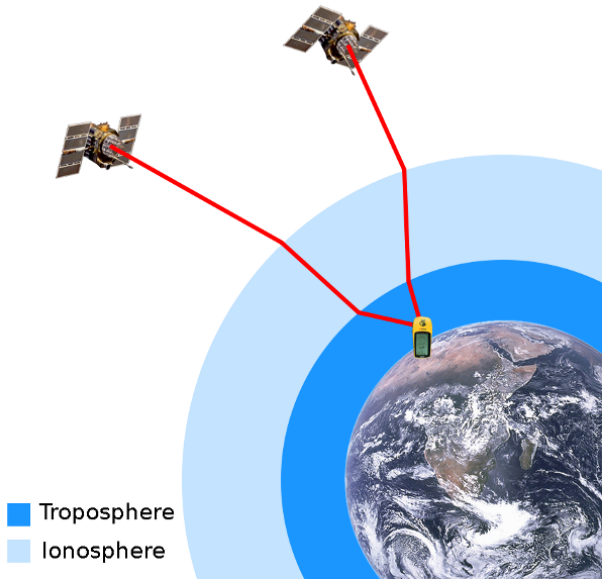


FIGURE 4.1: Earth atmosphere layers deformation representation Source: https://commons.wikimedia.org/ ref: URL08

Issues with the ionosphere can be mitigated by using dual-frequency receivers. The ionosphere affects the signal by delaying the signal, this delay is proportional to the frequency of the signal. By using two different frequencies, the receiver can calculate the delay and correct

the position.

There also exists some models that can predict the ionosphere delay depending on the satellites positions but these models are not perfect and can't be used in all situations.

b. Multipath

Multipath is a phenomenon that occurs when the signal is reflected by an object before reaching the receiver. This reflection can cause the signal to take a longer path to reach the receiver, thus affecting the time it takes for the signal to reach the receiver.

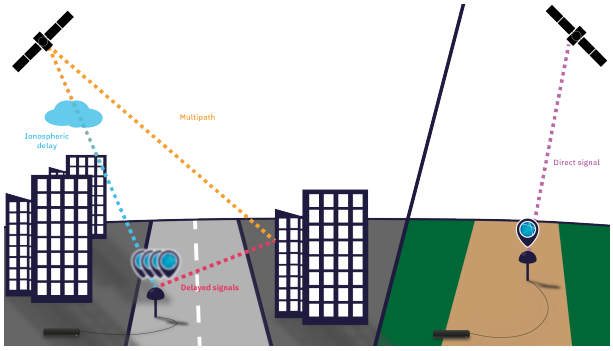


FIGURE 4.2: Multipath explicative diagram Source: https://syntony-gnss.com/ ref: URL08

In the example in Figure 4.2, we can see a direct signal on the right that directly goes to a receiver, and a reflected signal on the left that goes to the receiver after being reflected by a building. This reflected signal will take longer to reach the receiver, thus affecting the time measurement, and thus the position calculation.

This issue is still a problem today, and there are no perfect solutions to mitigate it because we're not able to predict the position of all the reflecting objects.

c. Ephemeris error

The GPS system uses the position of the satellites to calculate the position of the receiver. This means that the actual position of the satellite is important. However the position and orbital parameters and corrections made by the satellites are not absoulutely perfect and can have some errors.

There is a system in place to correct these errors, the Satellite-Based Augmentation System (SBAS) system. This system uses ground stations to monitor the position of the satellites and to send corrections to the satellites. These corrections are then broadcasted to the receivers.

d. Satellite clock drift

Even though the S.Vs are equipped with atomic clocks, they are not perfect and can drift a little but this drift is actually compensated by software. We will also mention that the relativistic effects also have to be taken into account when calculating the time difference between the receiver and the satellite (The time in the satellite reference frame is different than the time in the receiver reference frame since it's moving at a different speed and is in a different gravitational field).

4.2. AUGMENTATION SYSTEMS

The base GPS with only the L1 C/A code has an accuracy of about **10 meters.** This is not enough for some applications, so some augmentation systems have been developed to improve the accuracy of the system. One limitation is that the receiver often needs to be compatible with these systems to use them.

a. Differential GPS

The Diffential GPS (DGPS) is a system that uses a reference ground station[7] located at a known position to calculate the errors in the GPS system. The ground station knows exactly where it is, so it can determine the errors of the received signals and then broadcast these errors to the receivers. The receivers can then correct their position using these errors. This augmentation needs ground stations to be installed at maximum 400 km from the receiver we want to augment.

The final accuracy with DGPS is about 3cm.

b. RTK

The Real Time Kinematic (RTK) works on the same principle as the DGPS. It's usually a complete system with 2 devices, a base station and a rover. The base station gets its position from a known source and then calculates the errors in the GPS system. The base station then sends these errors to the rover, which can then correct its position using these errors. The correction is usually given via R.F. or Global System for Mobile Communications (GSM).

[7] **noauthor_23_nodate**.

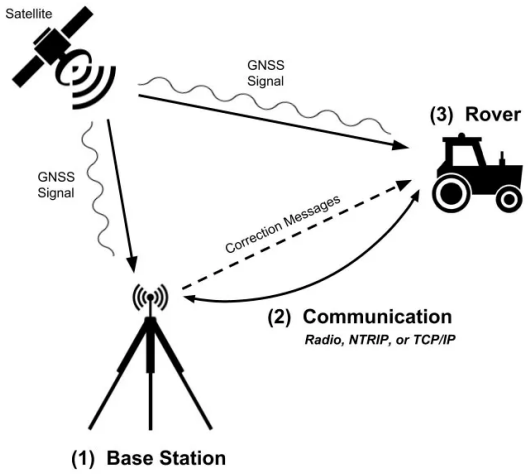


FIGURE 4.3: RTK Diagram Source: https://i0.wp.com ref: URL10

In 4.3 figure, we can observe a fixed base station, receiving GNSS signal and sending corrections to the rover. The rover receive both GNSS signals and the corrections from the base station.

FIGURE 4.4: Photo of an existing RTK system (South INNO7 rover base set) Source: https://globalgpssystems.com ref: URL11

c. SBAS

The SBAS uses a network of ground stations to monitor the GPS system and to send corrections to Geostationary satellites [8] that then broadcast these corrections to the receivers. This system is used to improve the accuracy of the GPS system in a large area, like a continent.

There actually is four main SBAS:

- Wide Area Augmentation System (WAAS) for North America
- European Geostationary Navigation Overlay Service (EGNOS) for Europe
- Multi-functional Satellite Augmentation System (MSAS) for Japan
- GPS Aided GEO Augmented Navigation (GEO) for India

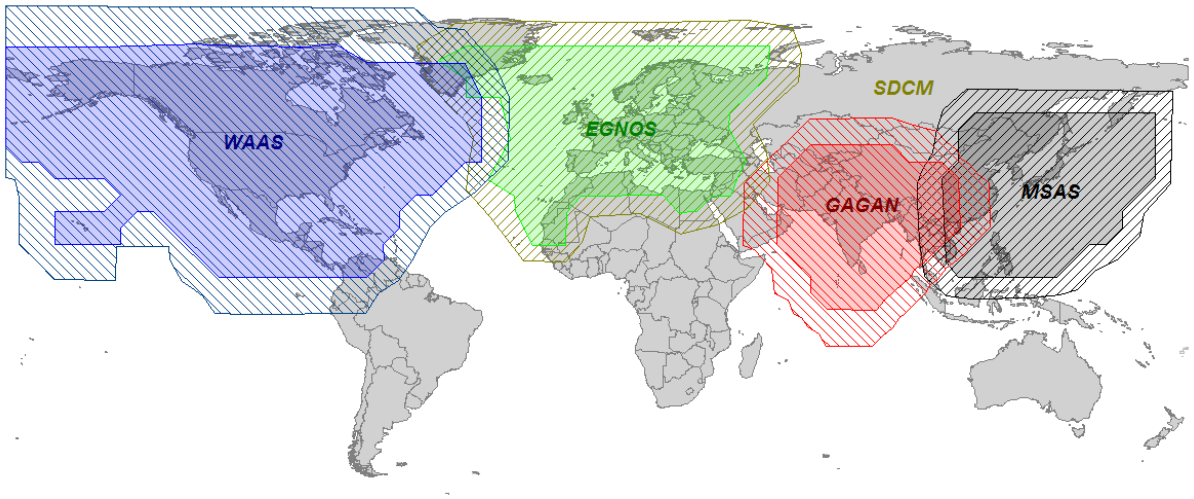


FIGURE 4.5: Map of the main SBAS systems Source: https://www.reseau-teria.com ref: URL12

[8] A geostationary satellite is a satellite that is placed in a geostationary orbit, this means that the satellite is always at the same position relative to the Earth.

CHAPTER 5 : HOW GPS COARSE ACQUISITION WORKS

This chapter will explain how the GPS C/A code works. How a single bit is received what's actually sent by the satellite, how the receiver correlates the received signal with the local replica of the signal, and how the receiver can determine the time of flight of the signal.

5.1. DATA MODULATION

When sending the actual data on the **L1** band, we're combining a carrier wave with the C/A code and the navigation data.

The navigation data actually contains:

- The time at which the data was sent
- The **ephemeris** data, which is a set of data that describes the precise position of the satellite sending the data
- The **almanac**, which is a set of data that describes the coarse position of all the satellites in the constellation
- The satellite health/accuracy
- ionospheric model for correction

Here we can see how the data is modulated and sent.

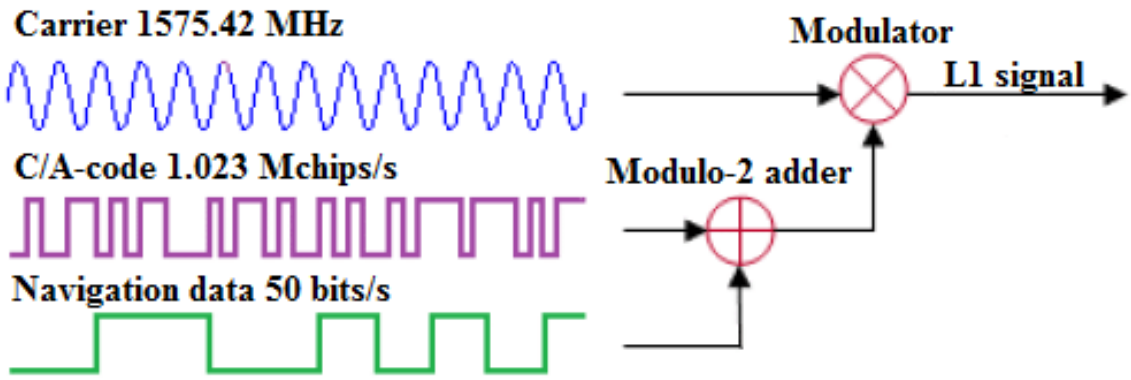


FIGURE 5.1: Navigaiton data modulation Source: www.e-education.psu.edu ref: URL14

5.2. CODE CORRELATION

As described in previous chapters, the GPS system uses a CDMA system to transmit the signals. This means that all the satellites are transmitting on the same frequency at the same time. The receiver needs to be able to distinguish the signal from each satellite. This is done by using a unique code for each satellite. The receiver knows the code of each satellite and can then correlate the received signal with the local replica of the signal. As described in the next figure, the correlation is done by multiplying the received signal with the local replica of the signal. The result of this multiplication is then integrated over a period of time. If the received signal is the same as the local replica, the correlation will be at its maximum. If the received signal is different, the correlation will be lower. This is how the receiver can distinguish the signal from each satellite.

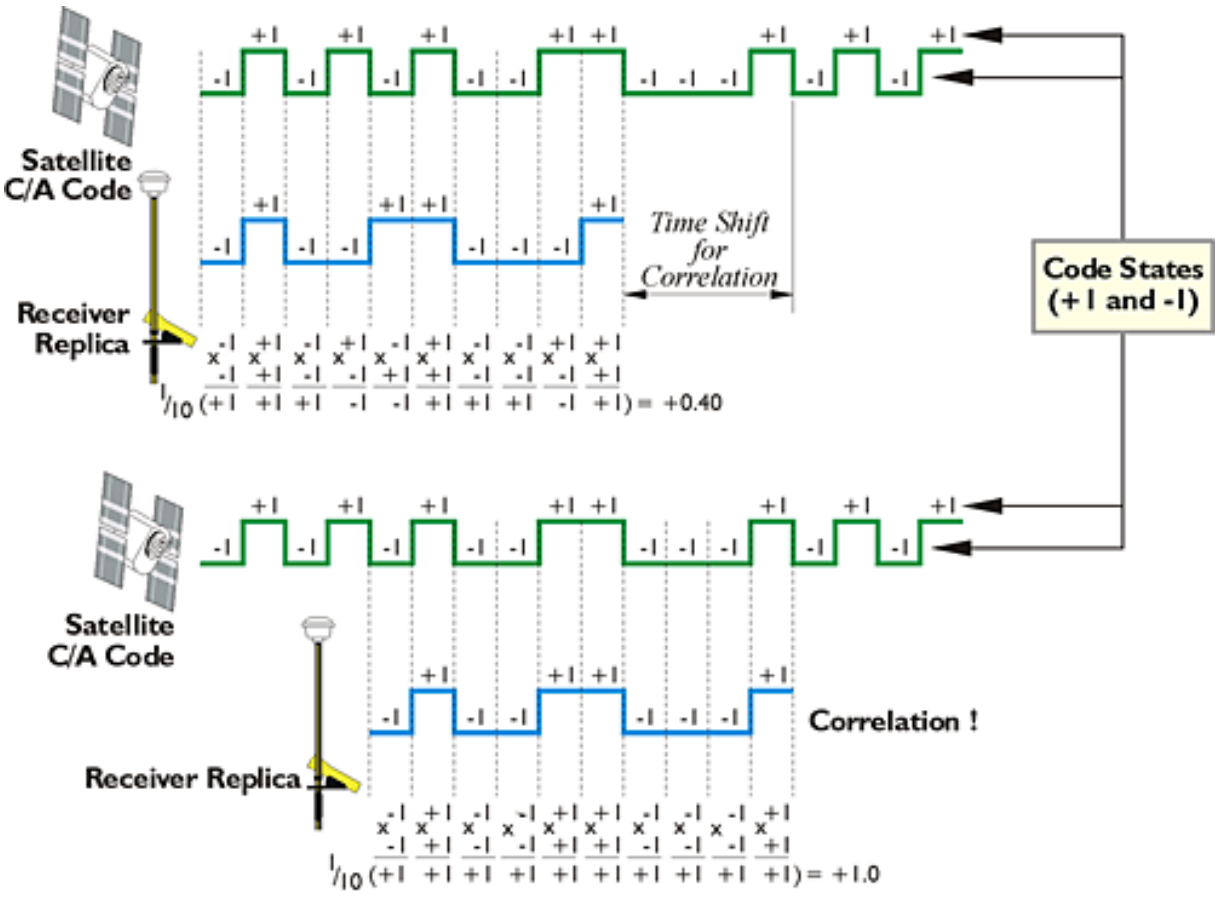


FIGURE 5.2: C/A PRN Correlation Source: www.e-education.psu.edu ref: URL14

5.3. FRAME DATA

The data sent by the satellite is divided into frames. Each frame contains 1500 bits and is sent at a rate of 50 bits per second. The frame is divided into 5 subframes, each divided into 10 words of 30 bits (300 bits per subframe). The full frame is sent every 30 seconds. The subframes one to three are sent fresh every frame but the subframes 4 and 5 contain the almanac data and are sent in multiple frames over time (25 frames for subframe 4 and 25 frames for subframe 5). The receiver can then reconstruct the full almanac data by combining the data from the different frames.

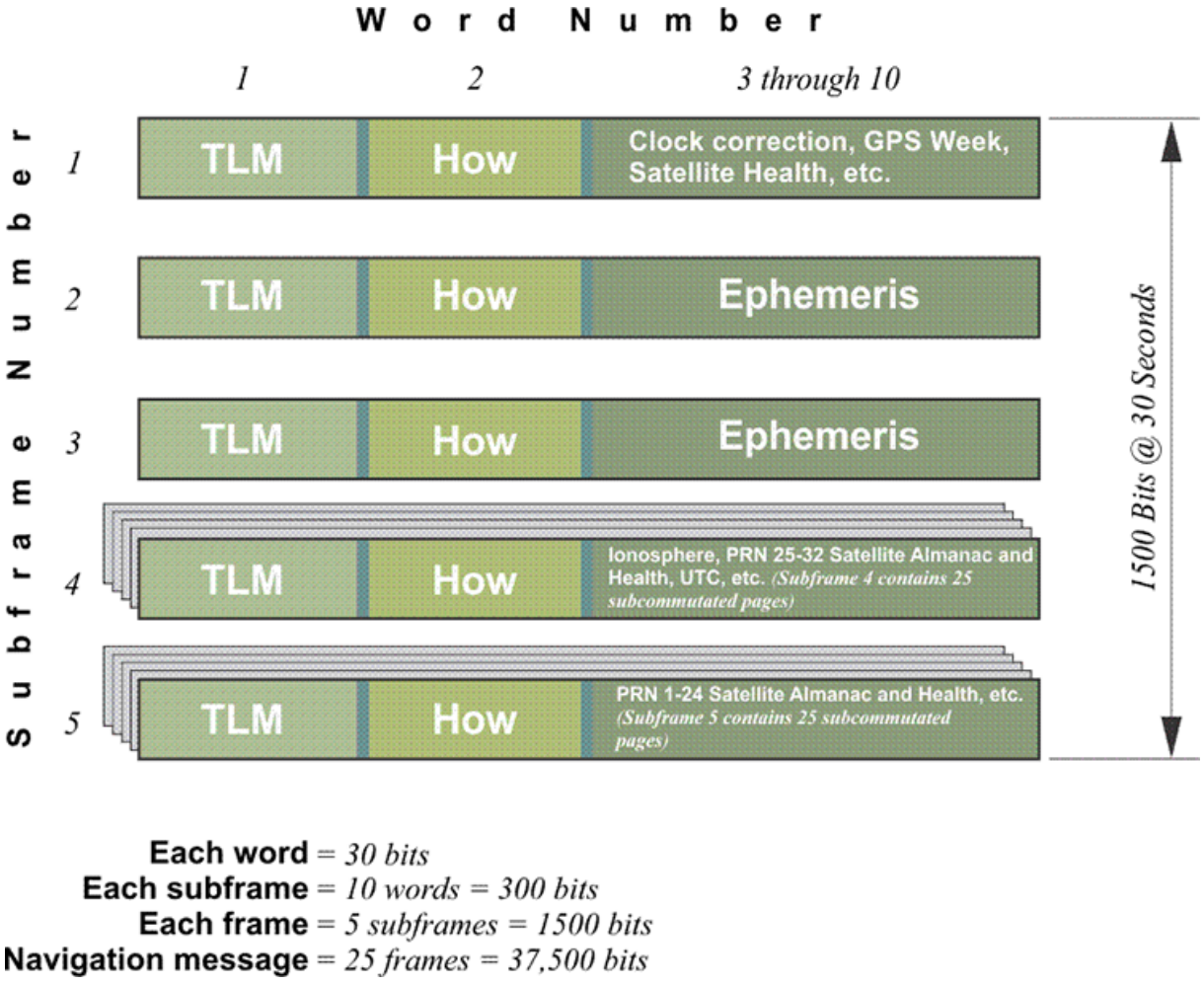


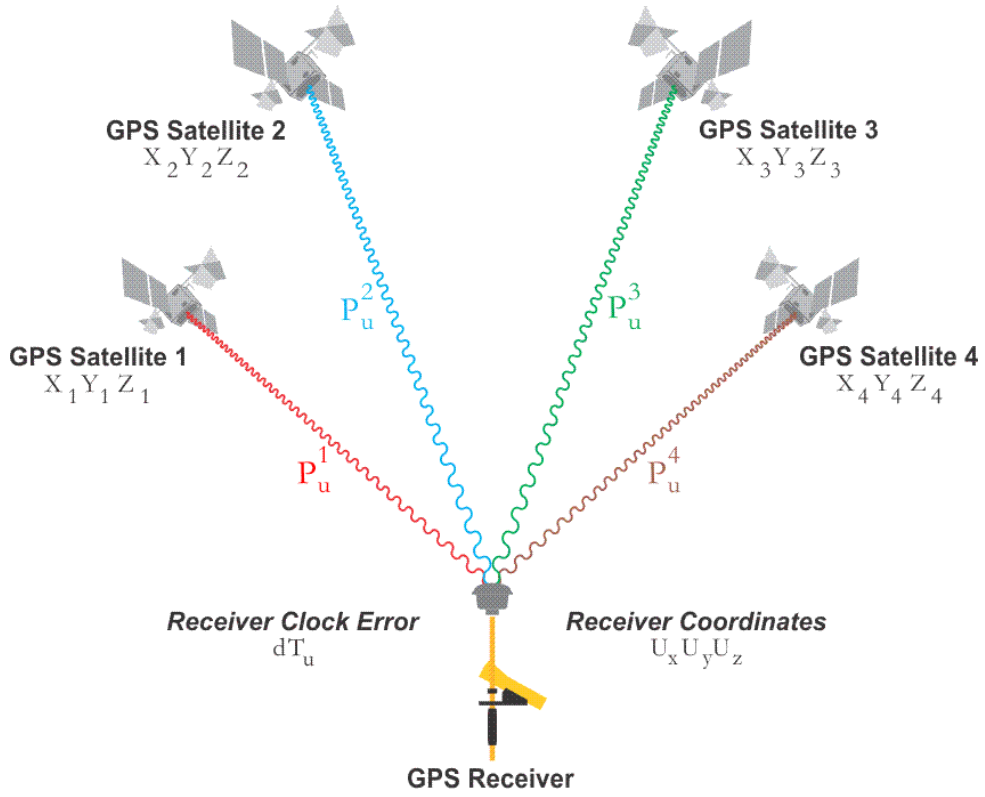
FIGURE 5.3: C/A PRN Correlation Source: www.e-education.psu.edu ref: URL15

5.4. TIME AND DISTANCE CALCULATION

We want to compute the distance between the satellite and the receiver. We know the speed of light and the time at which the signal was sent. We could compare it to the time at which the signal was received to get the time of flight of the signal to finally compute the distance,

knowing the speed of light. However, this would mean that we have two absoulutely precise and perfectly synchronized oscillators in the receiver and the satellite.

It would be kind of impossible nor would it be economically viable to have atomic clocks in every receiver devices. This is why the GPS actually uses the position and time of four satellites to solve a four equations system to determine the position of the receiver. This is called **differential positioning**.



$$p_u^1 = \sqrt{(X_1 - U_x)^2 + (Y_1 - U_y)^2 + (Z_1 - U_z)^2} + c(dT_u)$$

$$p_u^2 = \sqrt{(X_2 - U_x)^2 + (Y_2 - U_y)^2 + (Z_2 - U_z)^2} + c(dT_u)$$

$$p_u^3 = \sqrt{(X_3 - U_x)^2 + (Y_3 - U_y)^2 + (Z_3 - U_z)^2} + c(dT_u)$$

$$p_u^4 = \sqrt{(X_4 - U_x)^2 + (Y_4 - U_y)^2 + (Z_4 - U_z)^2} + c(dT_u)$$

FIGURE 5.4: Differential positionning equations Source: www.e-education.psu.edu ref: URL16

CHAPTER 6 : EXPERIMENTATIONS

This final chapter will describe the different experimentations that were conducted. Two of those are simple GPS readings with a commercial GPS receiver and with an SDR. The last one will be a spoofing attack on the commercial receiver.

6.1. EQUIPEMENT USED

a. USRP B200 SDR

There is a multitude of possible SDR to chose from with different capabilities. We need one capable of transmitting and receiving on the GPS L1 frequency, this means the device needs to be able to transmit at 1575.42 MHz with a bandwidth of at least 30 Mhz.

We had one with these capabilities at the lab, the USRP B200. The device characteristics are as follows:

- Frequency range: 70 MHz - 6 GHz
- Bandwidth: 56 MHz
- Sample rate: 61.44 MS/s
- USB 3.0 interface

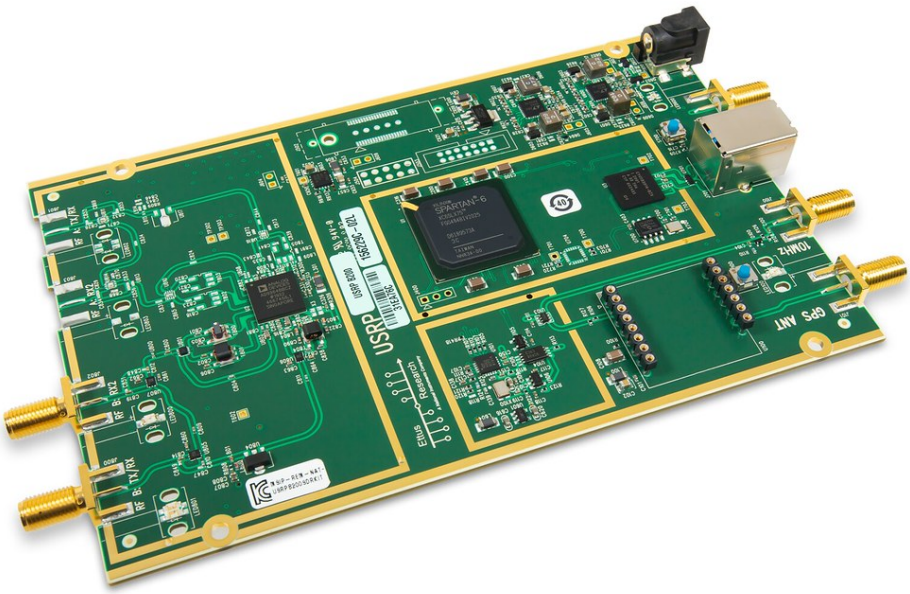


FIGURE 6.1: USRP B200 without a case Source: shop.trenz-electronic.de ref: URL18

b. UBLOX Max-M10s - Commercial GPS receiver

In order to test on hardware as close to a real-world, we tried to chose a GPS receiver as close as possible to the one used in the drone from the Windshape Paper[9]. We needed a development board with the same chip and an SMA connector instead of an on board antenna, in order to connect the receiver directly to the SDR. The drone used in the paper is a Parrot Anafi, which uses a UBLOX UBX-M8030. We couldn't find the exact model as a development board but we found the UBLOX Max-M10s which is relativly close to the UBX-M8030 and has a development board with an SMA connector: The SparkFun GPS-18037[10].

[9]Guillaume Catry et al. "Development of a Free-Flight Wind Test Facility Featuring a GNSS Simulator to Achieve Immersive Drone Testing". In: *AIAA SCITECH 2022 Forum*. AIAA SCITECH 2022 Forum. Place: San Diego, CA & Virtual. American Institute of Aeronautics and Astronautics, Jan. 3, 2022. ISBN: 978-1-62410-631-6. DOI: 10.2514/6.2022-2052. URL: https://arc.aiaa.org/doi/10.2514/6.2022-2052 (visited on 03/11/2024).

[10]**noauthor_gps-18037_nodate**.

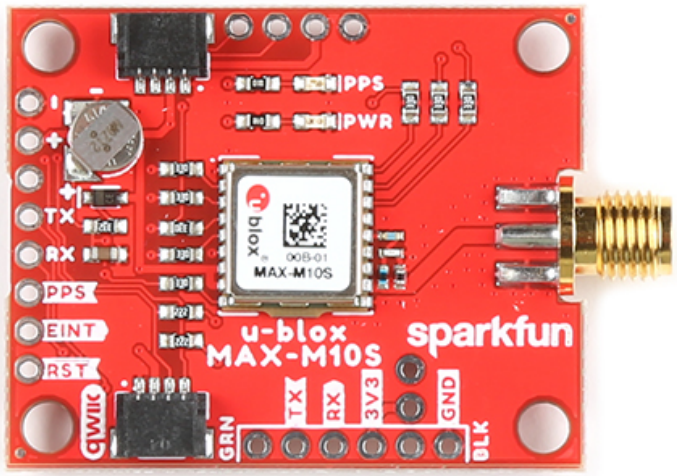


FIGURE 6.2: MAX M10s Dev board Source: https://www.mouser.ch ref: URL19

c. Antennas

We used two different antennas for the experimentations. The first one is active and specific to GPS frequency[11] , the other one is a simple retractable antenna. We're testing these two antennas to assess the need for an active GPS antenna. Note that the active antenna has some filtering and amplification circuitry to improve the signal quality but need a power source to operate, this means that we need an external bias tee, powered by direct current source.

[11] *ANT-GPS-SH2-SMA TE Connectivity / Linx Technologies | Mouser*. Mouser Electronics. URL: https://www.mouser.ch/ProductDetail/712-ANT-GPS-SH2-SMA (visited on 03/22/2024).

FIGURE 6.3: Active GPS antenna Source: https://www.swiss-green.ch ref: URL20

FIGURE 6.4: Passive retractable antenna Source: www.passion-radio.com ref: URL21

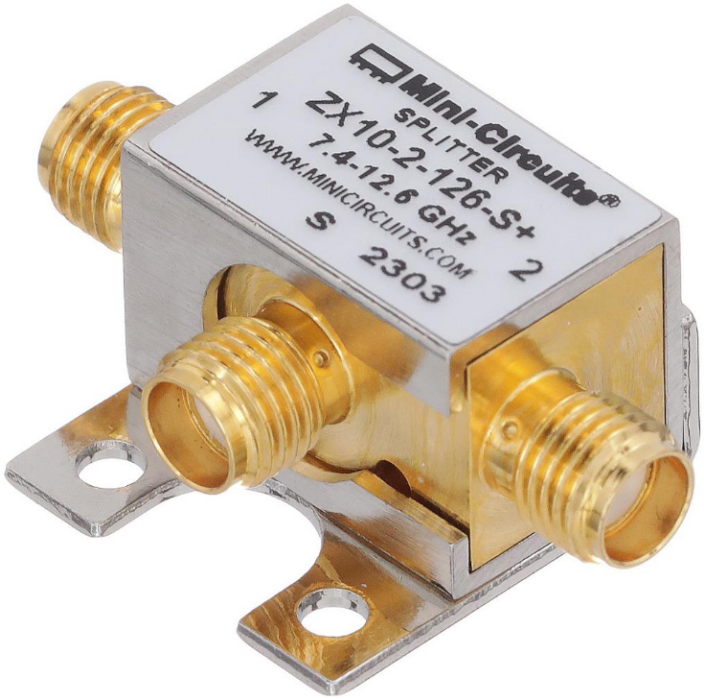


FIGURE 6.5: Bias tee to power the active antenna Source: www.artisantg.com ref: URL22

6.2. SOFTWARES USED

The advantage of using an SDR is the possibility to use a wide range of softwares to generate or receive signals. In order to get a position fix with the SDR, we used the GNURadio Companion[12] to try to record raw data and get a position, we also used the GNSS SDR software[13] to read actual position directly with the SDR as a frontend.

For the commercial GPS receiver, we mostly used a python script we made to read the serial data formatted as NMEA sentences.

The final spoofing part was done with existing spoofing software, Multi sdr gps sim[14].

6.3. RECEIVING POSITION WITH SDR AS RECEIVER

In this first experiment, we'll try to read raw data with gnu radio companion and then get a position fix with GNSS SDR[15]. For this, we will connect the USRP B200 to the active GPS antenna (herself powered by the bias tee) and record raw data with GNU radio.

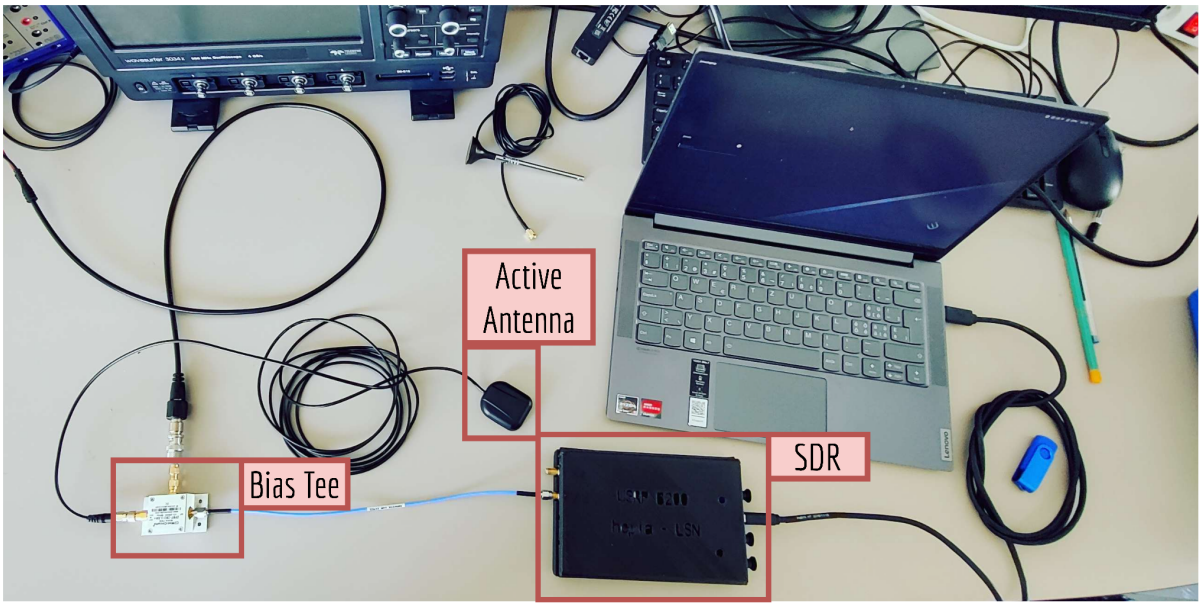


FIGURE 6.6: First experiment setup with active antenna Source: Made by Jonas Stirnemann

[12] *GNU Radio - The Free & Open Source Radio Ecosystem · GNU Radio*. GNU Radio. URL: https://www.gnuradio.org/ (visited on 03/22/2024).

[13] Carles Fernández-Prades. *GNSS-SDR*. GNSS-SDR. Feb. 13, 2024. URL: https://gnss-sdr.org/ (visited on 03/22/2024).

[14] Mictronics. *Mictronics/multi-sdr-gps-sim*. original-date: 2021-02-10T17:14:17Z. Mar. 21, 2024. URL: https://github.com/Mictronics/multi-sdr-gps-sim (visited on 03/22/2024).

[15] Fernández-Prades, *GNSS-SDR*.

a. Position fix with raw data

Here is a schematics of the GNU radio flowgraph used to record the raw data:

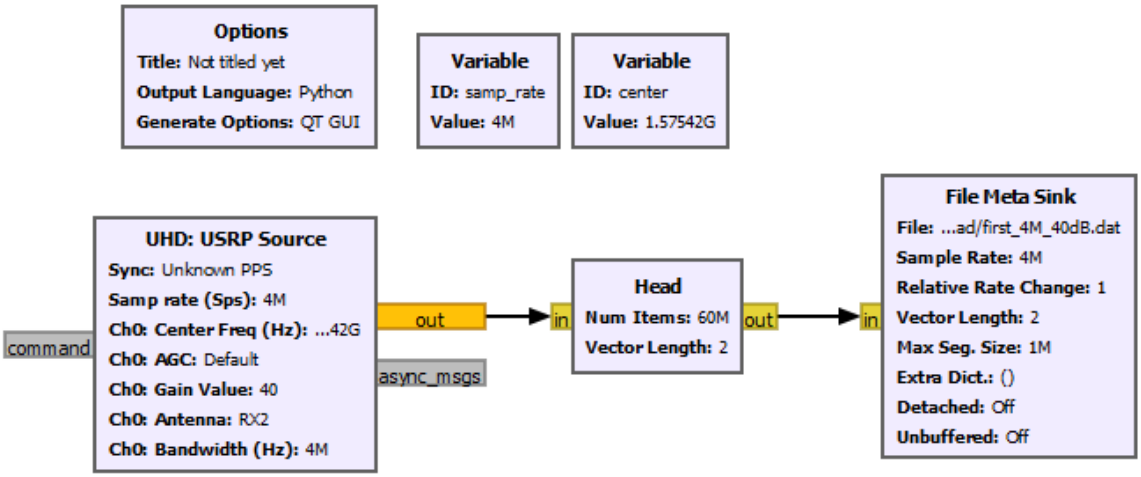


FIGURE 6.7: Gnu radio schematics for recording raw data Source: Made by Jonas Stirnemann

After recording some raw data into a file for 5 minutes with the antenna at the window with a relativly correct line of sight of the sky, and after setting everything like it's shown at [16]. I tried to get a position fix with GNSS SDR, but I couldn't get a fix. I tried to change the settings, i tried changing the data format but i could not get the fix.

```
gnss-sdr --config_file=./raw_to_fix.conf
```

FIGURE 6.8: Start GNSS SDR to get a fix from raw data Source: Made by Jonas Stirnemann

b. Position fix with sdr as direct front end

Since i could not get a fix with the raw data, i tried to get a fix directly with the USRP B200 as a frontend. The GNSS SDR software provides a way to configure the USRP as a frontend and get a fix directly from the SDR (Configuration in appendix).

[16] https://gnss-sdr.org/my-first-fix/

After this configuration, I could get a fix with the USRP B200 as a frontend, when finishing the fix, the software saves some files formatted for Google Maps or similar software. So i could check what position fix i had gotten. We can observe the path of the read position in red and the actual position marked as putple dot, even though the accuracy is not our purpose here, we can see that the position goes as far as a 100 meters from the actual position.

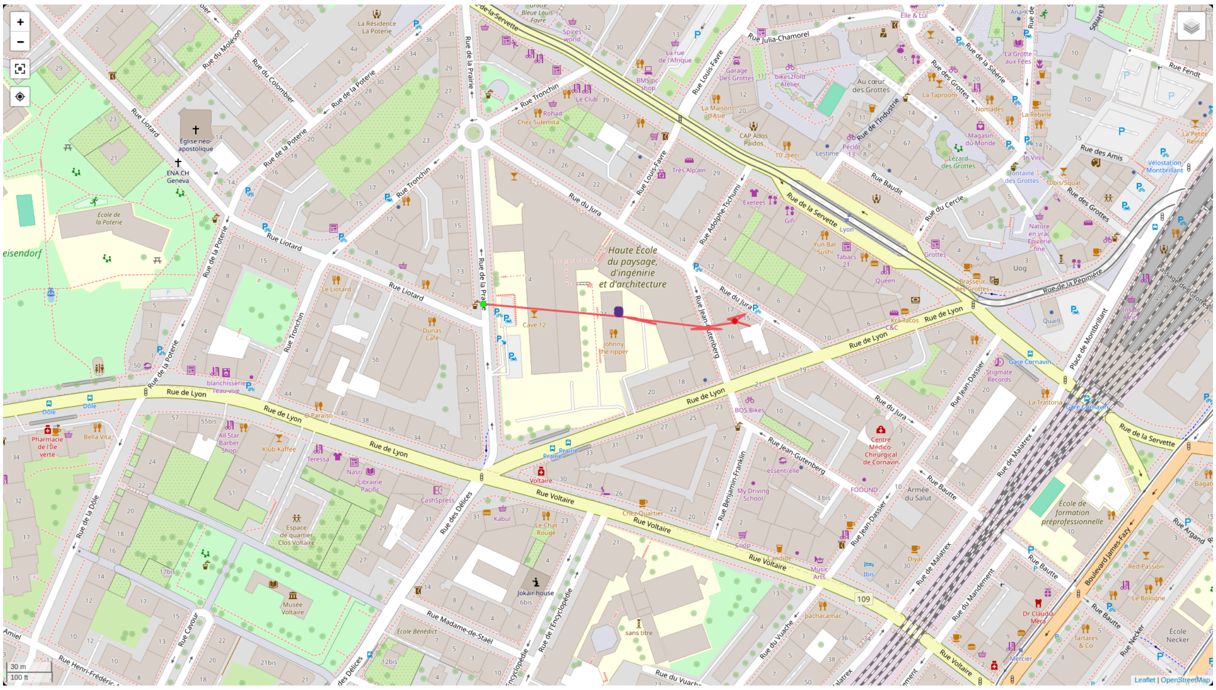


FIGURE 6.9: First position fix GNSS SDR Source: Made by Jonas Stirnemann

6.4. RECEIVING POSITION WITH COMMERCIAL GPS RECEIVER

In this second experiment, we'll try to read the position from the commercial GPS receiver with a simple python script reading NMEA data. We'll setup the passive antenna onto the commercial receiver. The receiver has to be externally powered (3.3V by a bench power supply) and is connected via a USB to serial adapter to the computer. The antenna has been placed at the window with a relativly correct line of sight of the sky with a lenght of around 19 cm (according to wavelength of the GPS signal).

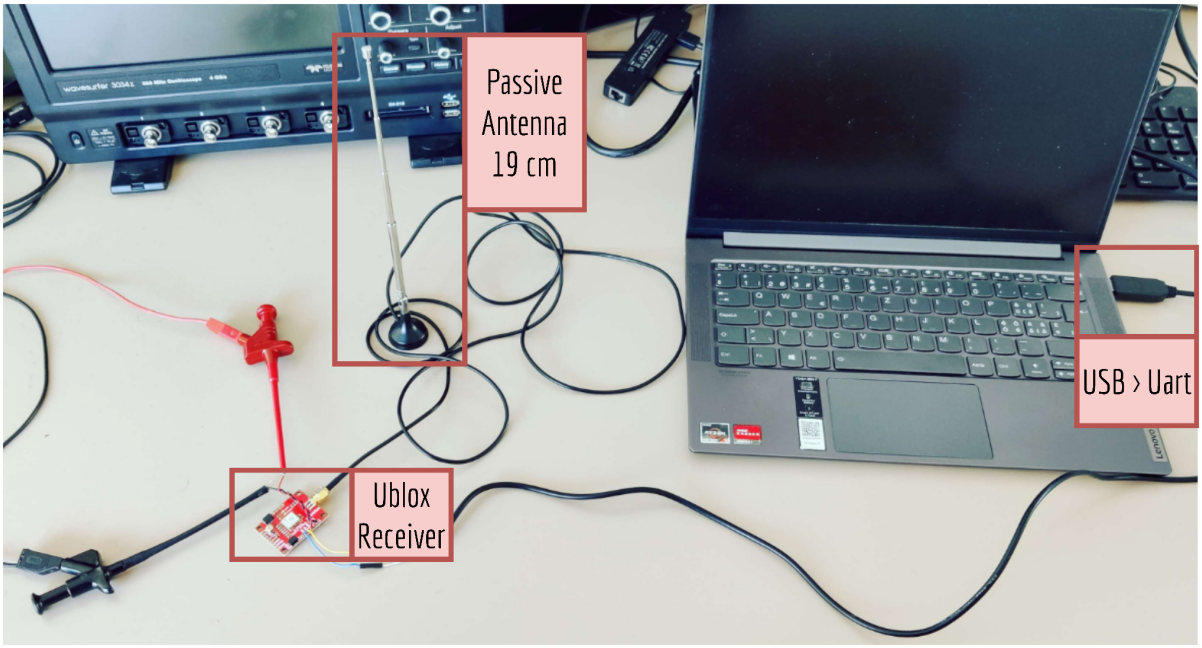


FIGURE 6.10: Second Experimentation setup with passive antenna Source: Made by Jonas Stirnemann

a. NMEA data format

The NMEA data format is a standard for GPS data, it is a set of sentences that are sent by the GPS receiver to the computer. The most important sentences are the GGA, RMC and GSA. The GGA sentence contains the position fix, the RMC sentence contains the recommended minimum data for GPS, and the GSA sentence contains the satellite data. The GGA sentence is the most important for us, it contains the latitude and longitude of the receiver.

I used a simple python library called pynmea2[17] to read the NMEA data from the serial port and extract the GGA sentence. The received positions are then renderer in an HTML Open Street Map where we can observe the positions.

[17] *NMEA 0183*. In: *Wikipedia*. Feb. 4, 2024. URL: `https://en.wikipedia.org/w/index.php?title=NMEA_0183&oldid=1203390440` (visited on 03/11/2024).

```python
# This will read NMEA sentences from the GPS receiver
# Get the Status of the fix
# Get the Latitude and Longitude
# Get the Time and Date
# Get the Height
# Get the Speed
# Get the number of satellites
# Show a OpenStreetMap with all the position we've read


import serial
import pynmea2
import folium


FILENAME_RAW = "raw_data.txt"
FILENAME_PROCESSED = "processed_data.txt"

def dec_to_dms(deg):
    """
    Convert degrees to degrees, minutes, seconds as string
    """
    d = int(deg)
    m = int((deg - d) * 60)
    s = (deg - d - m / 60) * 3600.00
    return f"{d}°{m}'{s:07.4f}"


def plot_map(positions):
    """
    Plot a map with the given latitude and longitude
    """
    m = folium.Map(location=[positions[0][0], positions[0][1]], zoom_start=30)

    folium.PolyLine(positions, color="red", weight=2.5, opacity=1).add_to(m)

    m.save('map.html')

if __name__ == '__main__':
    from datetime import datetime

    f_raw = open(FILENAME_RAW, "w")
    f_raw.write(f"——— RAW DATA {datetime.now()} ———\n")

    f_processed = open(FILENAME_PROCESSED, "w")
    f_processed.write(f"——— PROCESSED DATA {datetime.now()} ———\n")

    ser = serial.Serial('/dev/ttyUSB0', 9600, timeout=1)

    positions = []

    try:
        while True:
            try:
                data = ser.readline().decode('utf-8')
            except UnicodeDecodeError:
                continue

            f_raw.write(data)

            #print(f"{data[0:6]} =/= $GNGGA")
            if data[0:6] == '$GNGGA':
                msg = pynmea2.parse(data)
                positions.append((msg.latitude, msg.longitude))

                print(f"Fix Status: {msg.gps_qual}", )
                print(f"Latitude: {msg.latitude}", )
                print(f"Longitude: {msg.longitude}", )
                print(f"Time: {msg.timestamp}", )
                print(f"Height: {msg.altitude}", )
                print(f"Number of Satellites: {msg.num_sats}", )

                f_processed.write(f"——— Measurement n°{len(positions)} \n", )
                f_processed.write(f"Fix Status: {msg.gps_qual}\n", )

        pass
```

FIGURE 6.11: NMEA python script reading positions Source: Made by Jonas Stirnemann

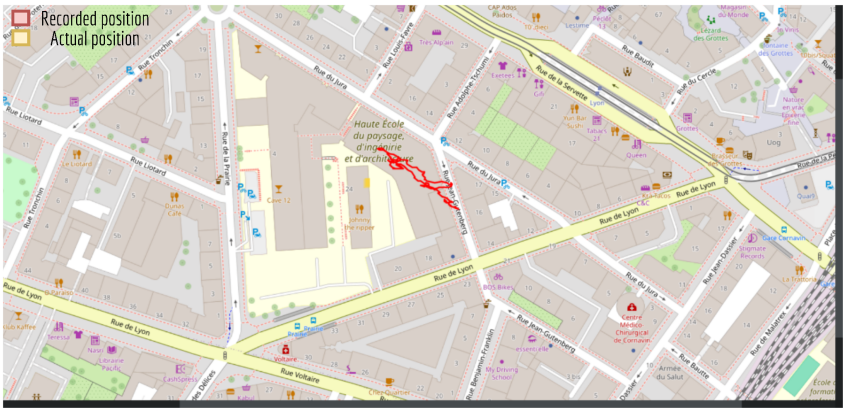


FIGURE 6.12: Second experiment map With the positions Source: Made by Jonas Stirnemann

We can observe the path of the read position in red and the actual position marked as orange dot, the accuracy looks really good (around 20m), especially considering we used the passive antenna.

6.5. SPOOFING GPS POSITION

Now that we can easily read position from the commercial GPS receiver, we can try to spoof the position with the Multi SDR GPS simulator[18]. This software allows us to generate a GPS signal and send it to the USRP B200. We can then send a fake position to the commercial GPS receiver.

a. Battery issue

I've tried spoofing multiple times but could not get a consistant fix with different spoofed position. It looked like the first time worked but when i changed the spoofed position, the receiver could not get a fix anymore. I've tried to change lots of parameters, i changed the power output from the spoofer but i still could not get a fix.

After some time, i realized that the battery of the commercial GPS receiver was allowing it to store the last ephemeries and almanac data for quicker fix when turned on. But this meant that the receiver would be confused if a completly different position was sent to it. So i decided to remove the battery from the receiver board and it worked every time!

The battery is marked with a purple square, you have to pop it out of the PCB with a knife, scalpel or a screwdriver.

[18]Mictronics, *Mictronics/multi-sdr-gps-sim*.

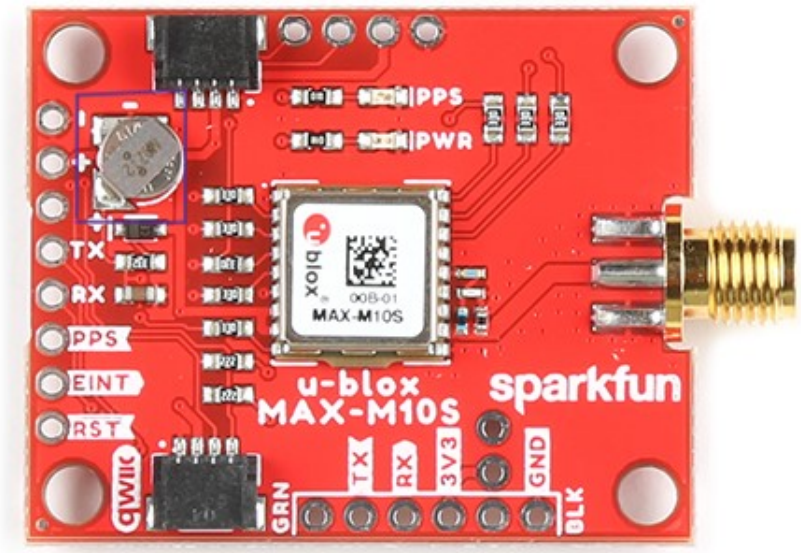


FIGURE 6.13: GPS receiver dev board battery marked for removal Source: Made by Jonas Stirnemann

b. Spoofing position

In order to setup the spoofing, we have to download the current day almanac data from NASA [19]. Then we have to get Multi SDR GPS simulator and start a spoof scenario with the usrp B200 as a frontend.

For example we chose the Toulouse Museum as a spoofed position, the coordinates are : (43.594198064081695, 1.4493044146176086).

[19] `https://cddis.nasa.gov/archive/gnss/data/daily/`

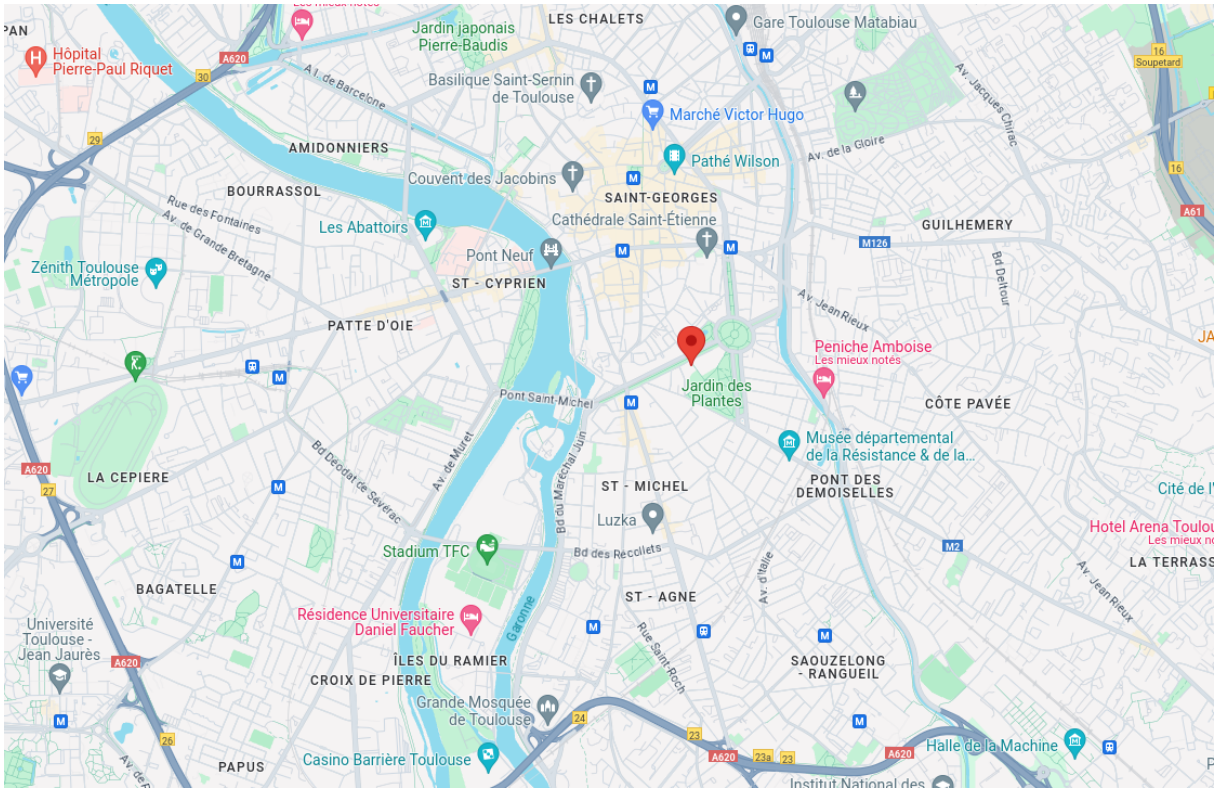


FIGURE 6.14: Real Toulouse Museum position Source: Made by Jonas Stirnemann

```
./gps-sim -e brdc0720.24n -l 43.594198064081695,1.4493044146176086,400 -d 300 --radio usrp
```

FIGURE 6.15: Command to start the spoofing Source: Made by Jonas Stirnemann

Now we can read the position from the commercial GPS receiver and see that the position is spoofed to the Toulouse Museum. After some time, WE GOT A FIX !

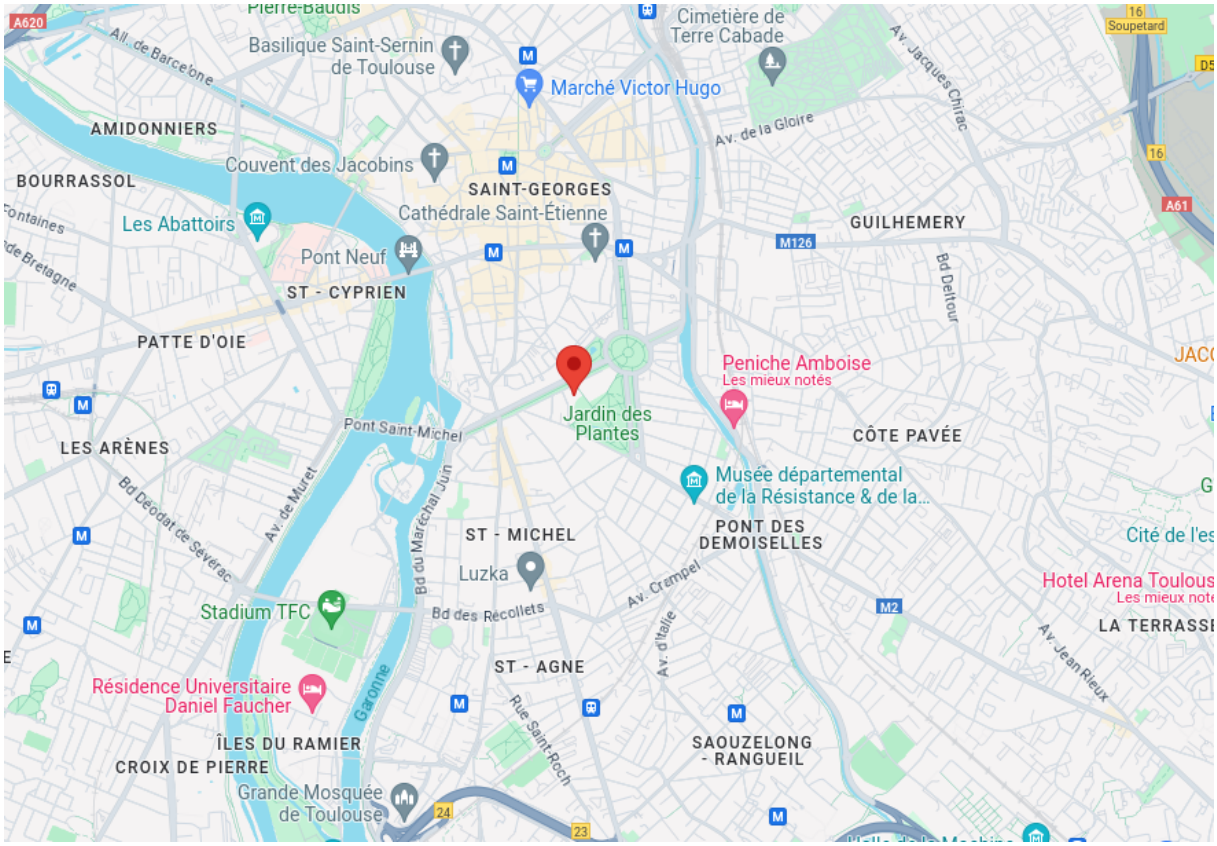


FIGURE 6.16: Spoofed Toulouse Museum position Source: Made by Jonas Stirnemann

We tried the same with a Viking museum in Oslo, Norway, the coordinates are : (59.90771024065185, 10.683988972697675).

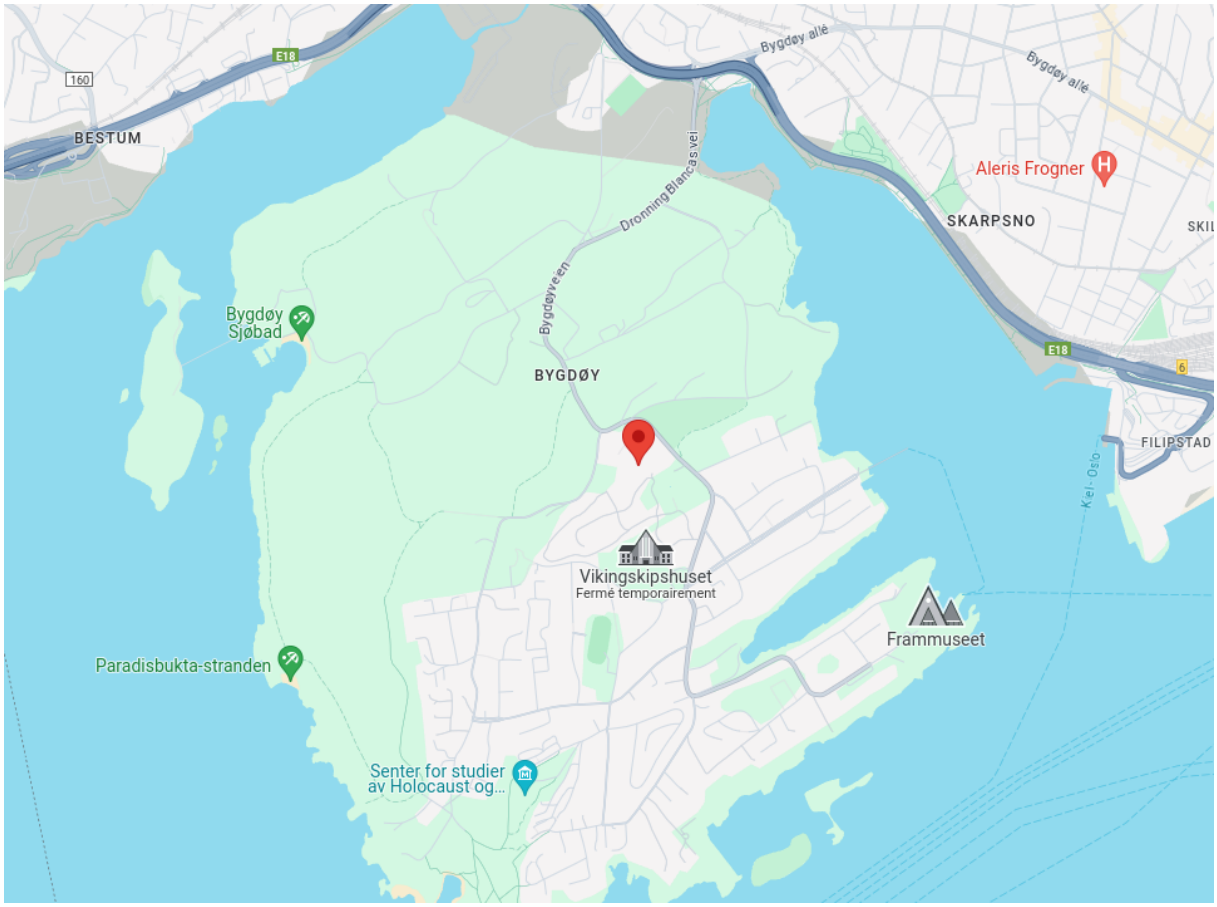


FIGURE 6.17: Real Oslo Vikings museum Museum position Source: Made by Jonas Stirnemann

Which then got us a spoofed position really close to the real one.

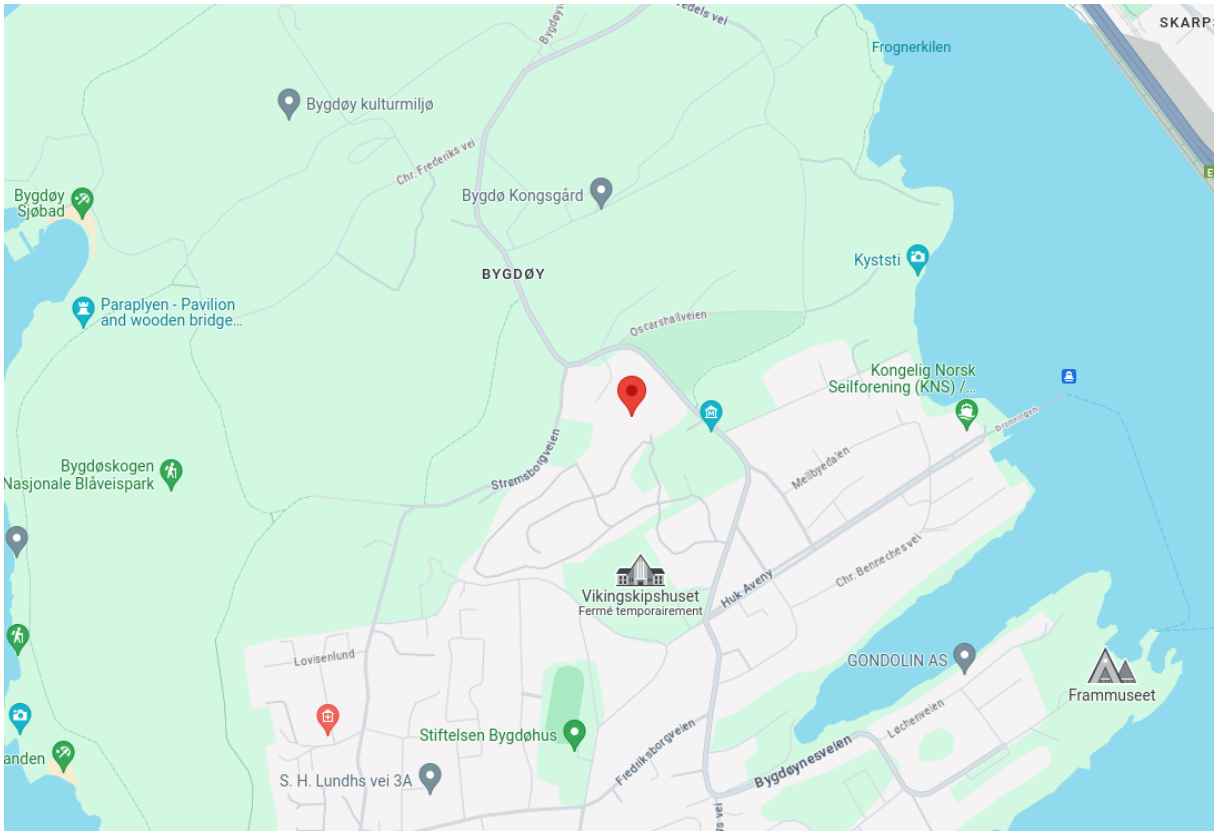


FIGURE 6.18: Spoofed Oslo Vikings museum Museum position Source: Made by Jonas Stirnemann

CONCLUSION

The project set out to explore the potential applications of the USRP B200 in spoofing a commercial GPS receiver. Following an extensive research phase, we conducted experiments to assess the project's viability. Initially, our experiments involved comparing the performance of a commercial GPS receiver with that of the SDR in obtaining a position fix. As anticipated, the commercial receiver performed admirably, while the SDR exhibited lower accuracy, likely due to factors such as its algorithm or atmospheric correction model.

This project proved to be a fascinating learning journey, particularly in delving into RF, GNSS, and SDR technologies, which were previously unfamiliar to me despite my background in electronics. The experience of applying new concepts in a practical setting and conducting thorough research was both rewarding and enlightening.

In the end, our efforts bore fruit as we successfully obtained a position fix using the SDR and effectively spoofed a commercial receiver, thus demonstrating the feasibility of our approach. Looking ahead, there are opportunities for further refinement and enhancement of the system. These may include incorporating features such as support for multiple constellations and frequencies to enhance compatibility with a wider range of UAVs. Moreover, there is potential in developing a user-friendly interface to streamline control and operation, as well as integrating the spoofing system into Windshape's testing environment for comprehensive evaluation and validation.

By continuing to build upon the foundation established in this project, we can unlock even greater possibilities in the field of GPS spoofing and its applications.

APPENDIXS

APPENDIX 1

Gnu Radio Configuration for position fix on raw data

```
[GNSS-SDR]

;######### GLOBAL OPTIONS ##################
GNSS-SDR.internal_fs_sps=2000000

;######### SIGNAL_SOURCE CONFIG ############
SignalSource.implementation=File_Signal_Source
SignalSource.filename=/home/your-username/work/data/2013_04_04_GNSS_SIGNAL_at_CTTC_SPAIN.dat
SignalSource.item_type=ishort
SignalSource.sampling_frequency=4000000
SignalSource.samples=0

;######### SIGNAL_CONDITIONER CONFIG ############
SignalConditioner.implementation=Signal_Conditioner
DataTypeAdapter.implementation=Ishort_To_Complex
InputFilter.implementation=Pass_Through
InputFilter.item_type=gr_complex
Resampler.implementation=Direct_Resampler
Resampler.sample_freq_in=4000000
Resampler.sample_freq_out=2000000
Resampler.item_type=gr_complex

;######### CHANNELS GLOBAL CONFIG ############
Channels_1C.count=8
Channels.in_acquisition=8
Channel.signal=1C

;######### ACQUISITION GLOBAL CONFIG ############
Acquisition_1C.implementation=GPS_L1_CA_PCPS_Acquisition
Acquisition_1C.item_type=gr_complex
Acquisition_1C.pfa=0.01
Acquisition_1C.doppler_max=10000
Acquisition_1C.doppler_step=250
Acquisition_1C.blocking=true

;######### TRACKING GLOBAL CONFIG ############
Tracking_1C.implementation=GPS_L1_CA_DLL_PLL_Tracking
Tracking_1C.item_type=gr_complex
Tracking_1C.pll_bw_hz=40.0;
Tracking_1C.dll_bw_hz=4.0;

;######### TELEMETRY DECODER GPS CONFIG ############
TelemetryDecoder_1C.implementation=GPS_L1_CA_Telemetry_Decoder

;######### OBSERVABLES CONFIG ############
Observables.implementation=Hybrid_Observables

;######### PVT CONFIG ############
PVT.implementation=RTKLIB_PVT
PVT.positioning_mode=Single
PVT.output_rate_ms=100
PVT.display_rate_ms=500
PVT.iono_model=Broadcast
PVT.trop_model=Saastamoinen
PVT.flag_rtcm_server=true
PVT.flag_rtcm_tty_port=false
PVT.rtcm_dump_devname=/dev/pts/1
PVT.rtcm_tcp_port=2101
PVT.rtcm_MT1019_rate_ms=5000
PVT.rtcm_MT1077_rate_ms=1000
PVT.rinex_version=2
```

APPENDIX 2

GNSS Sdr configuration for USRP B200 as frontend

```
[GNSS-SDR]

;######### GLOBAL OPTIONS ##################
GNSS-SDR.internal_fs_sps=4000000

;######### SIGNAL_SOURCE CONFIG ############
SignalSource.implementation=UHD_Signal_Source
SignalSource.item_type=cshort
SignalSource.sampling_frequency=4000000
SignalSource.freq=1575420000
SignalSource.gain=40
SignalSource.subdevice=A:A  ; <- Can be A:0 or B:0
SignalSource.samples=0

;######### SIGNAL_CONDITIONER CONFIG ############
SignalConditioner.implementation=Signal_Conditioner

;######### DATA_TYPE_ADAPTER CONFIG ############
DataTypeAdapter.implementation=Pass_Through
DataTypeAdapter.item_type=cshort

;######### INPUT_FILTER CONFIG ############
InputFilter.implementation=Fir_Filter
InputFilter.input_item_type=cshort
InputFilter.output_item_type=gr_complex
InputFilter.taps_item_type=float
InputFilter.number_of_taps=11
InputFilter.number_of_bands=2

InputFilter.band1_begin=0.0
InputFilter.band1_end=0.48
InputFilter.band2_begin=0.52
InputFilter.band2_end=1.0

InputFilter.ampl1_begin=1.0
InputFilter.ampl1_end=1.0
InputFilter.ampl2_begin=0.0
InputFilter.ampl2_end=0.0

InputFilter.band1_error=1.0
InputFilter.band2_error=1.0

InputFilter.filter_type=bandpass
InputFilter.grid_density=16
InputFilter.sampling_frequency=4000000
InputFilter.IF=0
```

```
;######### RESAMPLER CONFIG ############
Resampler.implementation=Pass_Through

;######### CHANNELS GLOBAL CONFIG ############
Channels_1C.count=8
Channels.in_acquisition=1

;######### ACQUISITION GLOBAL CONFIG ############
Acquisition_1C.implementation=GPS_L1_CA_PCPS_Acquisition
Acquisition_1C.item_type=gr_complex
Acquisition_1C.coherent_integration_time_ms=1
Acquisition_1C.pfa=0.01
Acquisition_1C.doppler_max=5000
Acquisition_1C.doppler_step=250
Acquisition_1C.max_dwells=1
Acquisition_1C.dump=false
Acquisition_1C.dump_filename=./acq_dump.dat

;######### TRACKING GLOBAL CONFIG ############
Tracking_1C.implementation=GPS_L1_CA_DLL_PLL_Tracking
Tracking_1C.item_type=gr_complex
Tracking_1C.extend_correlation_symbols=10
Tracking_1C.early_late_space_chips=0.5
Tracking_1C.early_late_space_narrow_chips=0.15
Tracking_1C.pll_bw_hz=40
Tracking_1C.dll_bw_hz=2.0
Tracking_1C.pll_bw_narrow_hz=5.0
Tracking_1C.dll_bw_narrow_hz=1.50
Tracking_1C.fll_bw_hz=10
Tracking_1C.enable_fll_pull_in=true
Tracking_1C.enable_fll_steady_state=false
Tracking_1C.dump=false
Tracking_1C.dump_filename=tracking_ch_

;######### TELEMETRY DECODER GPS CONFIG ############
TelemetryDecoder_1C.implementation=GPS_L1_CA_Telemetry_Decoder

;######### OBSERVABLES CONFIG ############
Observables.implementation=Hybrid_Observables

;######### PVT CONFIG ############
PVT.implementation=RTKLIB_PVT
PVT.positioning_mode=Single
PVT.output_rate_ms=100
PVT.display_rate_ms=500
PVT.iono_model=Broadcast
PVT.trop_model=Saastamoinen
PVT.flag_rtcm_server=true
PVT.flag_rtcm_tty_port=false
PVT.rtcm_dump_devname=/dev/pts/1
PVT.rtcm_tcp_port=2101
PVT.rtcm_MT1019_rate_ms=5000
PVT.rtcm_MT1077_rate_ms=1000
PVT.rinex_version=2
```

APPENDIX 3

https://github.com/ThePurpleOne/gnss_spoofing

BIBLIOGRAPHY

ADALM-PLUTO Evaluation Board \textbar Analog Devices. URL: https://www.analog.com/en/resources/evaluation-hardware-and-software/evaluation-boards-kits/adalm-pluto.html (visited on 03/11/2024).

ANT-GPS-SH2-SMA TE Connectivity / Linx Technologies | Mouser. Mouser Electronics. URL: https://www.mouser.ch/ProductDetail/712-ANT-GPS-SH2-SMA (visited on 03/22/2024).

Automated Testing and Assurance Solutions - Spirent. URL: https://www.spirent.com/ (visited on 03/11/2024).

Catry, Guillaume et al. "Development of a Free-Flight Wind Test Facility Featuring a GNSS Simulator to Achieve Immersive Drone Testing". In: *AIAA SCITECH 2022 Forum*. AIAA SCITECH 2022 Forum. Place: San Diego, CA & Virtual. American Institute of Aeronautics and Astronautics, Jan. 3, 2022. ISBN: 978-1-62410-631-6. DOI: 10.2514/6.2022-2052. URL: https://arc.aiaa.org/doi/10.2514/6.2022-2052 (visited on 03/11/2024).

Fernández-Prades, Carles. *GNSS-SDR*. GNSS-SDR. Feb. 13, 2024. URL: https://gnss-sdr.org/ (visited on 03/22/2024).

— *My first position fix*. Feb. 26, 2024. URL: https://gnss-sdr.org/my-first-fix/ (visited on 03/11/2024).

Flores, Anthony and Dylan Nicholas. "DISTRIBUTION STATEMENT A. Approved for public release; distribution is unlimited". In: (2021).

GNU Radio - The Free & Open Source Radio Ecosystem · GNU Radio. GNU Radio. URL: https://www.gnuradio.org/ (visited on 03/22/2024).

GPS Constellation \textbar Navigation Center. URL: https://www.navcen.uscg.gov/gps-constellation (visited on 03/11/2024).

GPS: The Global Positioning System. URL: https://www.gps.gov/ (visited on 03/11/2024).

HackRF One - Great Scott Gadgets. URL: https://greatscottgadgets.com/hackrf/one/ (visited on 03/11/2024).

Henthorn, Stephen et al. "The effect of ADC resolution on concurrent, multiband, direct RF sampling receivers". In: *2021 IEEE Global Communications Conference (GLOBECOM)*. GLOBECOM 2021 - 2021 IEEE Global Communications Conference. Place:

Madrid, Spain. IEEE, Dec. 2021, pp. 1–6. ISBN: 978-1-72818-104-2. DOI: 10.1109/GLOBECOM46510.2021.9685641. URL: https://ieeexplore.ieee.org/document/9685641/ (visited on 03/11/2024).

Home. URL: https://www.nuand.com/ (visited on 03/11/2024).

International, Rohde {\textbackslash}textbackslash\& Schwarz. *Branchenführende Technologiekompetenz*. URL: https://www.rohde-schwarz.com/ch/startseite_48230.html (visited on 03/11/2024).

Kassas, Zaher M., Jahshan Bhatti, and Todd E. Humphreys. "A graphical approach to GPS software-defined receiver implementation". In: *2013 IEEE Global Conference on Signal and Information Processing*. 2013 IEEE Global Conference on Signal and Information Processing (GlobalSIP). Place: Austin, TX, USA. IEEE, Dec. 2013, pp. 1226–1229. ISBN: 978-1-4799-0248-4. DOI: 10.1109/GlobalSIP.2013.6737129. URL: http://ieeexplore.ieee.org/document/6737129/ (visited on 03/11/2024).

Kim, Tae-Hee et al. "Analysis of performance of GPS L1 signal generator in GPS L1 signal". In: *2014 14th International Conference on Control, Automation and Systems (ICCAS 2014)*. 2014 14th International Conference on Control, Automation and Systems (ICCAS 2014). Oct. 2014, pp. 1006–1009. DOI: 10.1109/ICCAS.2014.6987921. URL: https://ieeexplore.ieee.org/document/6987921 (visited on 03/11/2024).

L1 Frequency - an overview \textbar ScienceDirect Topics. URL: https://www.sciencedirect.com/topics/mathematics/l1-frequency (visited on 03/11/2024).

LabSat Real-Time Plus. URL: https://www.labsat.co.uk/index.php/en/products/labsat-real-time-plus (visited on 03/11/2024).

LimeSDR. URL: https://limemicro.com/products/boards/limesdr/ (visited on 03/11/2024).

List of GPS satellites. In: *Wikipedia*. Mar. 13, 2024. URL: https://en.wikipedia.org/w/index.php?title=List_of_GPS_satellites&oldid=1213499951#PRN_to_SVN_history (visited on 03/16/2024).

Mictronics. *Mictronics/multi-sdr-gps-sim*. original-date: 2021-02-10T17:14:17Z. Mar. 21, 2024. URL: https://github.com/Mictronics/multi-sdr-gps-sim (visited on 03/22/2024).

Nguyen-Tan, Tang et al. “GPS Signal Reception and Spoofing Based on Software-Defined Radio Devices”. In: *2022 RIVF International Conference on Computing and Communication Technologies (RIVF)*. 2022 RIVF International Conference on Computing and Communication Technologies (RIVF). Dec. 2022, pp. 513–517. DOI: 10.1109/RIVF55975.2022.10013839. URL: https://ieeexplore.ieee.org/document/10013839 (visited on 03/11/2024).

NMEA 0183. In: *Wikipedia*. Feb. 4, 2024. URL: https://en.wikipedia.org/w/index.php?title=NMEA_0183&oldid=1203390440 (visited on 03/11/2024).

osqzss/gps-sdr-sim: Software-Defined GPS Signal Simulator. URL: https://github.com/osqzss/gps-sdr-sim (visited on 03/22/2024).

RJ. *L1, L2, and L5 GPS Signals: What Do They Mean?* Jan. 25, 2024. URL: https://equatorstudios.com/l1-l2-and-l5-gps-signals-what-do-they-mean/ (visited on 03/11/2024).

RTKLIB: An Open Source Program Package for GNSS Positioning. URL: https://www.rtklib.com/ (visited on 03/11/2024).

Satellite navigation. In: *Wikipedia*. Feb. 23, 2024. URL: https://en.wikipedia.org/w/index.php?title=Satellite_navigation&oldid=1209879496 (visited on 03/16/2024).

Satellite positioning using USRP B205mini-i and GNSS-SDR · Satoshi Takahashi. URL: https://s-taka.org/en/gnss-sdr-with-usrp-b205/ (visited on 03/11/2024).

Srinivasan S, Prasanna and Shiju Sathyadevan. “GPS Spoofing Detection in UAV Using Motion Processing Unit”. In: *2023 11th International Symposium on Digital Forensics and Security (ISDFS)*. 2023 11th International Symposium on Digital Forensics and Security (ISDFS). May 2023, pp. 1–4. DOI: 10.1109/ISDFS58141.2023.10131729. URL: https://ieeexplore.ieee.org/document/10131729 (visited on 03/11/2024).

Stanford-NavLab/gnss_lib_py. original-date: 2021-06-29T20:42:49Z. Mar. 15, 2024. URL: https://github.com/Stanford-NavLab/gnss_lib_py (visited on 03/22/2024).

Timing SDR recordings with GPS – Daniel Estévez. Mar. 29, 2022. URL: https://destevez.net/2022/03/timing-sdr-recordings-with-gps/ (visited on 03/11/2024).

VisualGPS, LLC. URL: https://www.visualgps.net/ (visited on 03/11/2024).