UAB

Master's Thesis

Master in Telecommunication Engineering

Development of a GPS Multi-Antenna Calibrated Platform Using COTS Components

Pablo Alejandro Noguera Chacón

Supervisor: Gonzalo Seco-Granados

Dpt. Telecommunications and System Engineering

Escola Tècnica Superior d'Enginyeria (ETSE)
Universitat Autònoma de Barcelona (UAB)

July 2016

El sotasignant, Gonzalo Seco-Granados, Professor de l'Escola Tècnica Superior d'Enginyeria (ETSE) de la Universitat Autònoma de Barcelona (UAB),

CERTIFICA:

Que el projecte presentat en aquesta memòria de Treball Final de Màster ha estat realitzat sota la seva direcció per l'alumne *Pablo Alejandro Noguera Chacón*.

I, perquè consti a tots els efectes, signa el present certificat.

Bellaterra, *1 de juliol de 2016*.

Signatura: *Gonzalo Seco-Granados*

Resum:
La vulnerabilitat de les senyals GNSS vers interferències de radiofreqüència i spoofing *s'ha convertit en una preocupació en aplicacions de radionavegació que requereixen un alt nivell de precisió i seguretat. Aquesta manca de precisió dels receptors GPS convencionals ha donat pas al desenvolupament de sistemes GPS multi-antena els quals integren diversos* front-ends *sincronitzats mitjançant un oscil·lador comú. L'objectiu principal d'aquest projecte es el desenvolupament d'una plataforma GPS de fàcil ús, calibrada i de baix cost utilitzant* Software-Defined Radios *(SDRs) i antenes GPS comercials. El* hardware *encarregat d'enregistrar les senyals captades per les diverses antenes GPS està format per tres* Universal Software Peripherial Radios *i un ordinador personal. La metodologia que es proposa per tal d'aconseguir una sincronització de fase entre els diferents* front-ends *consisteix en l'ús de les diferències de fase observades d'una senyal de referència per tal de compensar el desfasament existent en les senyals captades per cada USRP. En comptes de fer servir un* array *d'antenes totalment calibrat i amb una geometria fixa, l'*array *que es proposa està format per tres antenes GPS comercials. Aquest enfocament permetrà a l'usuari de canviar la geometria de l'*array *de tal manera que es podrà adaptar a les necessitats de l'aplicació.*

Resumen:
La vulnerabilidad de las señales GNSS frente a interferencias de radiofrecuencia y spoofing *se ha convertido en una preocupación en aplicaciones de radionavegación que requieren un alto nivel de precisión y seguridad. Esta falta de precisión en los receptores GPS convencionales ha dado lugar al desarrollo de sistemas GPS multi-antena los cuales integran distintos* front-ends *sincronizados utilizando un oscilador común. El objetivo principal de este proyecto es desarrollar una plataforma GPS de fácil uso, calibrada y de bajo coste utilizando* Software-Defined Radios *(SDRs) y antenas GPS comerciales. El* hardware *encargado de grabar las señales captadas por las distintas antenas GPS está formado por tres* Universal Software Peripheral Radios *y un ordenador personal. El método propuesto para conseguir una sincronización de fase entre los distintos* front-ends *consiste en el uso de las diferencias de fase observadas de una señal de referencia para compensar el desfase existente en las señales captadas por cada USRP. En vez de utilizar un* array *de antenas totalmente calibrado y con una geometría fija, el* array *que se propone está formado por tres antenas GPS comerciales. Este enfoque permitirá al usuario cambiar la geometría del* array *de modo que se adapte a las necesidades de la aplicación.*

Summary:
The vulnerability of GNSS to RF interference and spoofing has become more and more a concern for navigation applications requiring a high level of accuracy and reliability. This lack of accuracy of the GPS conventional receivers has triggered the development GPS multi-antenna systems, which integrate three or more GPS front-ends referenced to a common oscillator. The main purpose of this project is to develop a low cost and ease of deployment GPS calibrated multi-antenna platform by using both Commercial-Off-The Shelf (COTS) software-defined radios (SDRs) and GPS antennas. The signal collection hardware built to collect array data sets for each antenna consists of three Universal Software Radio Peripheral (USRP) and one Personal Computer (PC). The method proposed to achieve phase synchronization between the different front-ends consists of using a phase difference of arrival mechanism based on a synthetic GPS signal reference transmitter. The antenna array proposed is composed by three COTS GPS antennas instead of using a custom designed and fully calibrated antenna array. This approach allows the user to change the desired layout depending on the application field.

TABLE OF CONTENTS

INDEX OF TABLES

INDEX OF FIGURES

INTRODUCTION

BACKGROUND AND OBJECTIVE

Nowadays, the Global Positioning System (GPS) is the greatest exponent of satellite radionavigation systems. In short, the GPS is a satellite-based radionavigation system developed and operated by the U.S. Department of Defense (DOD) that permits land, sea and airbone users to determine their three-dimensional position, velocity, and time 24 hours a day, anywhere in the world with a precision and accuracy far better than other radionavigation systems available (U.S.C.G.N.C., 2016). Since GPS was developed by the U.S. Department of Defense and its main purpose was to be used in military applications, its accuracy in terms of civil users positioning is restricted. The DOD intentionally degrades the signal by altering the satellite clock or broadcasting less accurate ephemeris parameters that leads to worst pseudo range accuracy and therefore errors in user positioning. Furthermore, if we also consider that the GNSS satellite signal power is low (-130 dBm with satellite visibility) and that it might be attenuated 20-25 dB depending on the user environment, GPS signals can be considered weak and vulnerable to deliberate or unintentional Radio Frequency (RF) interference (MacGougan, 2003). A GPS receiver should be capable of mitigating the effects of Radio Frequency (RF) interference and multipath signals by giving them a reasonably correct position within 5-10m under these conditions.

The vulnerability of GNSS to RF interference and spoofing has become more and more a concern for navigation applications requiring a high level of accuracy and reliability. This lack of accuracy of the GPS conventional receivers has triggered the development GPS multi-antenna systems, which integrate three or more GPS front-ends referenced to a common

oscillator. The development of GPS multi-antenna systems has resulted in another leap in GPS applications. The information provided by various antennas may be used to the location of emitting sources, that is to say, to detect the direction of arrival of an incident wavefront, as well as to performing spatial filtering techniques, also called beamforming. The beamforming techniques are used to enhance the quality of the signals of interest by rejecting the possible interference sources. Obviously, this is useful in GPS applications where the interest is to maximize signals traveling by light of sight and cancel any possible reflection. In fact, antenna arrays have a significant number of applications in RADAR, SONAR, bioengineering, astronomy, etc. Not that long ago, researchers developed a system for spoofing[1] attack detection based on the observation of GPS signals' carrier phase differences between multiple antennas (Montgomery, Humphreys, & Ledvina, 2009).

The main purpose of this project is to develop a low cost and ease of deployment GPS calibrated multi-antenna platform by using both Commercial-Off-The Shelf (COTS) software-defined radios (SDRs) and GPS antennas. The signal collection hardware built to collect array data sets for each antenna consists of three Universal Software Radio Peripheral (USRP) and one Personal Computer (PC). In order to obtain time and frequency synchronized signals, the different SDRs will be referenced to a common oscillator. Furthermore, with the purpose of achieving phase calibrated samples, a phase correction method based on using a phase difference of arrival mechanism based on a synthetic GPS signal reference transmitter is suggested. The antenna array proposed is composed by three COTS GPS antennas instead of using a custom designed and fully calibrated antenna array. This approach allows the user to change the desired layout depending on the application field.

THESIS ORGANIZATION

This project is divided into four main chapters. Before their appearance, some information about the background and its significance is presented. The development of the present project is disclosed all along the four mentioned chapters.

[1] Spoofing: the act of broadcasting a fake GPS signal to fool a device into thinking it's somewhere else, and/or at a different point in time.

Chapter 1 includes a review of the most important points referring to the GPS positioning system. It incorporates a review of the history of this system and a brief description of its functioning. Moreover, the basic information about a GPS receiver functioning is included. To finish, there is a concise description about the antenna array basic concepts.

Chapter 2 —as 3 and 4— contains the development of the purpose. This chapter is dedicated to the multi-antenna platform development. As a log book, it explains the evolution of the platform's hardware, the used methods to create a calibrated multi-antenna receiver (both phase and time) and the obtained data after carrying out some tests.

In Chapter 3 the reasons why the geometry of the presented antennas array was chosen are explained. It also includes the cause why the model of antenna was elected and the presentation of the mechanical conditions of the array. In addition, it incorporates a test whose main objective is to show the possible undesirable effects on the estimated parameters by the receiver caused by the usage of different antennas and their interaction in a reduced area.

Talking about Chapter 4, it comprises a data collection carried out with the created platform in order to look for a possible utility for the recorded data. In this specific case, the possibility of using registered data by an antenna array to detect the direction of arrival of the GPS satellite signals is evaluated.

Finally and as a kind of final act, the conclusions drawn all along the realization of this experimental project are presented. As a final point, there are some annexes with complementary information about the used software.

CHAPTER 1

GPS SYSTEM FUNDAMENTALS

1.1 GNSS HISTORY

A GPS is a global positioning system which is a part of the Global Navigation Satellite Systems (GNSS); GNSS is understood as the series of radionavigation systems[2] —i.e. GPS, GLONASS[3] and Galileo[4]. This type of systems consist of an artificial satellites constellation that transmits signals which provide spatial and temporary positioning services in any part of the world.

The GNSS system development starts in the 1970s with the appearance of the American TRANSIT system designed by the US Navy and whose main operation principle was based on the Doppler Effect. The satellites sent signals at a particular frequency and, because of their relative movement, the receiver received the same signals with a slight variation on its frequency. The main change that the mentioned system introduced was its capacity to calculate the user's position with an accuracy of 10 meters. In spite of this, its main disadvantage was the fact that the user had to rest immobile for a long time so this system turned out to be slow and impractical (Bonnor, 2012).

Some years later, a worldwide revolution on the satellite positioning systems was originated and with this, the GPS system appeared. In their early days, the GPS system consisted of a 24

[2] Radionavigation: Navigation guided by electromagnetic waves.

[3] GLONASS: Satellite navigation system developed by the former Soviet Union.

[4] Galileo: Satellite navigation system developed by European Union.

satellites constellation that orbited the Earth and it was originally designed to be used only for military applications, so that, its control was under the US Department of Defense (National Coordination Office for Space-Based Positioning, Navigation, and Timing, 2016). It was not until the mid -90s that its civil applications were taken into account when the US government carried out some studies with the aim of analysing whether this technology was convenient to be used for this purpose or not. What is more, after reaching multiple agreements between the US government and other nations in the world, a global coverage was obtained and the GPS become the main fully operational satellite navigation system.

Nowadays, in the civil sphere, GNSS systems are used inside the International Civil Aviation field as positioning and recognition systems and as a key element of the communication, navigation and security systems for the air-traffic control. Other important uses at civil level are: navigation/orientation handheld devices for hiking, embedded devices for automotive industry, maritime transport, location systems for emergencies and hydrological calculations among other applications (National Coordination Office for Space-Based Positioning, Navigation, and Timing, 2016).

1.2 GPS OVERVIEW

As explained before, GPS is an all-weather satellite radionavigation system developed by the U.S. DOD[5] to meet the military requirements of world-wide positioning and time keeping. The overall system is composed of three major segments: space segment, control segment and user segment. The space segment consists of 30 (24 at the beginnings) radio navigation satellites which are deployed in 6 circular orbital planes with a radio of 26.600 km, an inclination of 55 degrees angle with respect to the equator and with a period of approximately 11 hours and 58 minutes. The satellite orbits were designed in such a way that at least four satellites could be visible at any time and location on the Earth's surface. The control segment consists of a network of tracking stations, one master control station situated in the Colorado Springs and a set of terrestrial antennas spread all over the world. The main tasks of the control segment are to track and monitor the satellite performance, compute the clock corrections and periodically upload satellite ephemeris and other system data to all satellites for retransmission to the user segment (Seco-Granados, Lopez-Salcedo, & Vicario, 2012). The user segment is composed of

[5] DOD: Department of Defense

the collection of GPS receivers. The main tasks of this segment are to receive, decode and process the signals emitted by the spatial segment in order to obtain the required information according to the use that is making the GPS system, usually calculate the user's position.

The ground positioning system uses three different frequency bands from the Ultra High Frequency (UHF) region to transmit the satellite signals. The first frequency band (L1) is centered at 1575.42 MHz and is used to transmit C/A (Coarse Acquisition) and P (Precise) codes. The second band (L2) is centered at 1226,60 MHz and is used to broadcast uniquely the P code. The third band (L5) uses a broadcasting frequency of 1176.45 MHz and it is used to transmit the new civilian signal also known as safety-of-life signal.

As commented above, the GPS has simplified the positioning and navigation of both military and civil users in a continuous way since its launch in 1978. Nowadays, this positioning system offers two levels of service: standard positioning system and precise positioning system. The Standard Positioning Service (SPS) is a continuous positioning service which is available for all users (military, private and commercial). For security reasons, the DOD degrades the SPS service precision using a technique called Selective Availability (SA). This technique manipulates the data of the navigation message so that the DOD can reduce the precision of the satellite signals and, hence, causing positioning errors. The Precise Positioning Service (PPS) is a positioning service of a great precision dedicated to military applications, it is available globally and continuously only for authorised users. This type of service is designed to have a minor accuracy error in comparison to the one provided by the SPS service. The offered accuracy is obtained by using cryptographic techniques as Anti-Spoofing (AS); this one avoids the fact of obtaining data from the broadcasted signals by encrypting methods and the mentioned SA which wakens the signals used by the SPS service.

Although GPS system was initially developed to fulfil military positioning applications, it is on its civilian applications where the technology has increased exponentially. The need for obtaining less errors during the localisation calculation using this positioning system has led researches to develop new reception platforms and processing techniques for this satellite signals. This is how the multi-antenna receiving platforms development is born; it integrates three or more GPS antennas into one system with a proper antenna configuration in space.

1.3 GPS RECEIVER

This section will be focused on the analysis of the hardware and the software that form the GPS receiver, whose basic purpose is to determine the user's position by analysing the signals that the GPS constellation transmits. As seen in Figure 1, in order to calculate the position and to obtain an analysis of the received signals parameter, some tasks will have to be sequentially carried out.

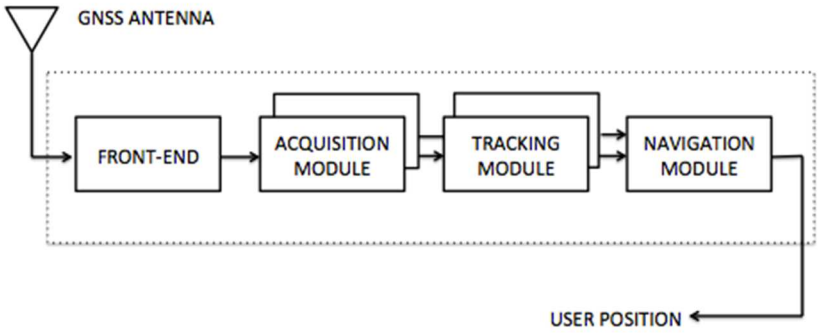

Figure 1 GPS receiver components

Since the user's position calculation is not part of this project's scope, only the three first modules —front-end, acquisition and tracking modules— will be briefly introduced.

1.3.1 FRONT-END MODULE

A front-end is the first module that forms a GPS receiver. Its main task is to down-convert the received signal to an intermediate frequency or directly to baseband and to carry out an analogue/digital conversion of the mentioned signal. Consequently, a constant flow of discrete and quantified samples are obtained (Seco-Granados, Lopez-Salcedo, & Vicario, 2012).

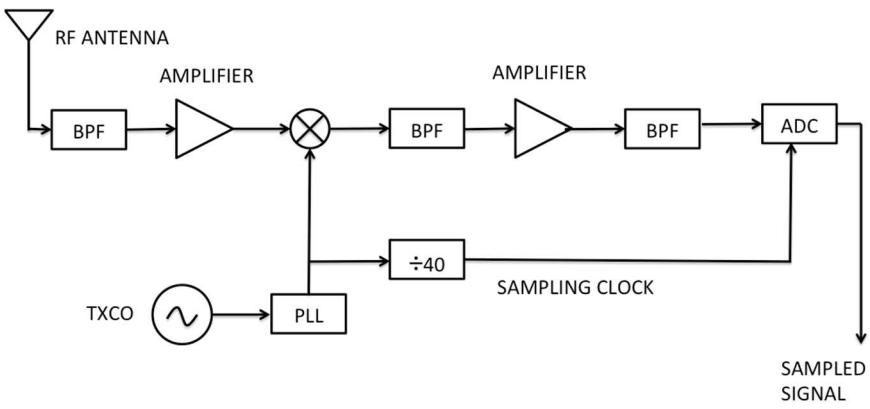

Figure 2 Front-end module

As seen in Figure 2, this module basically consists of the components below:

- **Band-pass filter (BPF):** acts as a frequency selector, allowing just a particular frequency range to pass. The most used filter is the band-pass, as opposed to a low-pass or high-pass filter, it provides additional frequency selectivity. Antennas usually have a low frequency selection range, so they tend to admit non-desired signals which are outside the frequency range; that is why it is important to eliminate these non-desired signals, otherwise, out of band signals sources could enter the front-end and saturate later-stage component following the antenna.
- **Amplifiers:** the only task of this device is, as its name shows, amplify the magnitude of the extremely weak incidental signal to a level practical for the analogue-to-digital conversion.
- **Local Mixer/Oscillator:** the main task of this device is to translate the input 1575.42 MHz RF carrier —in the L1 GPS band case— to a lower intermediate frequency (IF) signal, preserving the signal modulation structure (Borre, Akos, Bertelsen, & Rinder, 2007). The main reason why there is a need of lowering the input signal frequency is to adapt the frequency to those ranges used by the analogue/digital converser module. Generally, the large majority of oscillators cannot generate the desired frequency for GPS signals in L1 band on their own, so in order to carry out this process, some components are needed. As seen in Figure 2, generally, in order to get the desired intermediate frequency this device is combined to a Phase Lock Loop (PLL).

  In order to achieve translate the RF carrier to a lower IF, a mixer is needed. The result of mixing the local oscillator signal and the incident signal may be expressed as:

$$d(t)\cos(w_1 t)\cos(w_2 t) = \frac{d(t)\cos\big((w_1 - w_2)t\big)}{2} + \frac{d(t)\cos\big((w_1 + w_2)t\big)}{2} \quad (1)$$

  where w_1 and w_2 are the carrier and the local oscillator frequencies, respectively. It is obvious that the desired intermediate frequency is the difference term of the expression, the sum frequency is simply a consequence in this case. The second filter depicted in the cascade of Figure 2, which follows the mixer, is used remove the sum term and select only the desired difference frequency.

- **Analog to digital converter (ADC):** this is the last component of the front-end module and, as its name indicates, it converts the incident analog signal to digital samples. The main function of the ADC is to quantify[6] the incident analog signal with a specified number of bits. Spilker et al. (1996) established that the highest number of bits used by the receptor in the quantification process, the lower signal degradation is produced. Thermal noise produced by hardware has, as a consequence, a noisy ADC input signal and therefore amplifiers for increasing the desired signal level are needed.

1.3.2 ACQUISITION MODULE

Once the analog signals have been digitalized, the first task of the GPS software receiver is to identify all satellites visible to the user. The acquisition must determine the following properties of the signal:

- **Frequency:** Due to satellite relative motion to the receiver, the frequency of the signal emitted by this can differ from its nominal value, causing a Doppler effect. The Doppler frequency shift may vary depending on the satellite velocity and the user motion. In the case of having a static receptor, the maximum deviation caused by this effect may amount to ±5 KHz, whereas for the case of a moving receiver the deviation can be up to ±10 KHz (Bao & Tsui, 2005).
- **Code Phase:** The code phase denotes the point in the current data block where the C/A code begins. If a data block of 1 ms is examined, the data include an entire C/A code and thus one beginning of C/A code.

There are several methods to determinate these two signals parameters and most of them are based on using the C/A code correlation properties. One of the most important properties of the C/A codes is the result of its cross-correlation. The high peaks of autocorrelation and the low peaks of cross-correlation are used to identify the presence of a particularly satellite signal over the full samples received.

The received signal is a combination of signals from all *n* visible satellites and may be expressed as:

[6] Quantify: The quantification method consists of sampling the incident analog signal with approximate finite values.

$$s = s^1(t) + s^2(t) + \cdots + s^n(t) \quad (2)$$

In order to acquire satellite *k*, the incoming signal *s* is multiplied with a local generated C/A code corresponding to the satellite *k.* The cross-correlation between C/A codes for different satellites implies that signals from other satellites are nearly removed by this procedure. This procedure is continuously repeated applying different levels of code delay and Doppler frequency in order to avoid removing the desired signal.

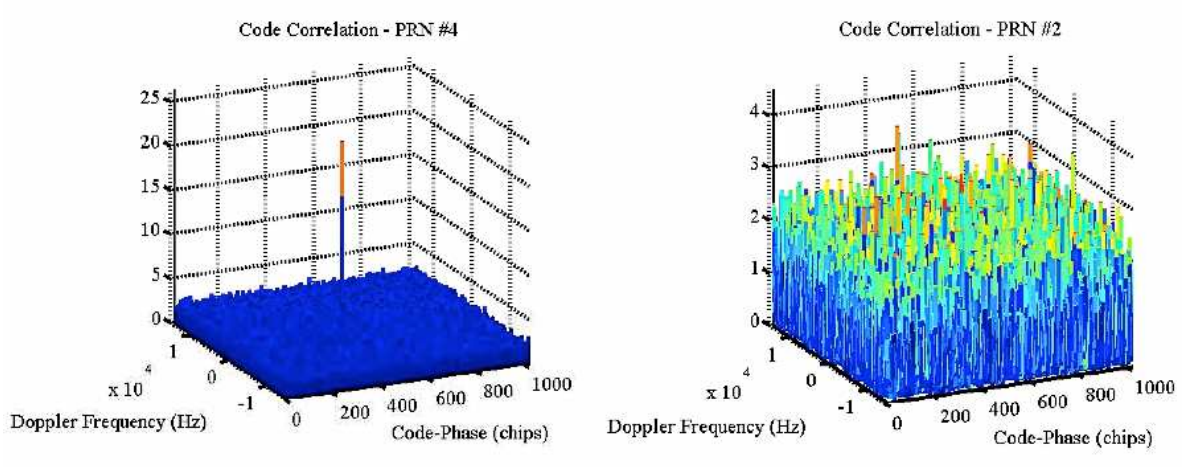

Figure 3 Left: Acquisition plot for PRN #4 Right: Acquisition plot for PRN#2

After performed the multiplication with the local generated code, the signal is mixed with a locally generated carrier wave. This procedure removes the carrier wave of the received signal by mixing the received signal with a frequency equal local generated signal. As mentioned before, the Doppler Effect may vary the satellite signal frequency up to ±10 KHz and therefore different frequencies within this area must be tested.

The acquisition procedure works as a search procedure. For each of different frequencies 1023 different code phases are tried and when all possibilities of both parameters are tried, a search of maximum value is performed. If the maximum value exceeds a determined threshold, the satellite is acquired with the corresponding frequency and phase shift. As shown in Figure 3, if the desired satellite signal is present a clear correlation peak is easily recognizable. On the opposite, if the desired satellite signal is not present, the correlation peaks do not exceeds the minimum threshold and therefore the absence of this satellite on the sky view is ensured.

1.3.3 TRACKING MODULE

Once finalized the code acquisition stage, tentative values of code phase and doppler frequency are obtained. The main purpose of tracking stage is to refine the values estimated on the acquisition module. Typically, the tracking module improves the frequency and code phase values and keeps the track of these as the signal properties change over time. To do this, two architectures in parallel are required, one dedicated to the code tracking and the other one focused on the carrier frequency/phase tracking.

1.3.3.1 CARRIER TRACKING

The main purpose of the carrier tracking is to determine the phase and the frequency of the incoming GPS signal. In order to track the carrier wave signal, the implementation of a phase lock loop (PLL) is necessary. The GPS PLL architecture should be insensitive to 180º phase shifts due to navigation data bits transitions. This property ensures that if navigation data bit transition occurs, the PLL will still track the signal and nothing will happen. The most common carrier tracking loop that fulfills this phase shifts insensitivity requirement is the Costas loop, Figure 4.

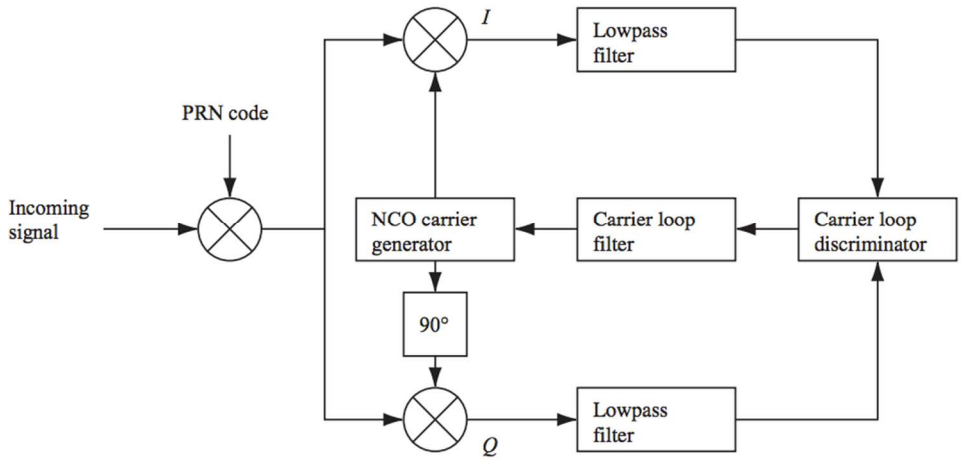

Figure 4 PLL: Costas loop used to track the carrier wave

Following the block diagram from left to right it may be seen that the first multiplication is the product between the input signal and the local carrier wave and the second multiplication is between a 90º phase-shifted carrier wave and the input signal. The main goal of this kind of PLL architecture is to keep all the energy in the in-phase arm (I) and that is why some kind of

feedback to the oscillator is needed. If the generated local code replica perfectly matches with the code signal incoming, the in-phase arm follows the expression:

$$D^k(n)\cos(w_{IF}n)\cos(w_{IF}n+\varphi)=\frac{1}{2}D^k(n)\cos(\varphi)+\frac{1}{2}D^k(n)\cos(2w_{IF}n+\varphi) \quad (3)$$

where φ is the phase difference between the phase of the input signal and the phase of the local replica of the carrier phase. On the other hand, the multiplication in the quadrature arm (Q) follows the expression:

$$D^k(n)\cos(w_{IF}n)\sin(w_{IF}n+\varphi)=\frac{1}{2}D^k(n)\sin(\varphi)+\frac{1}{2}D^k(n)\sin(2w_{IF}n+\varphi) \quad (4)$$

The low pass filter block eliminates the two terms with the double intermediate frequency and the remained signals may be expressed as:

$$I^k=\frac{1}{2}D^k(n)\cos(\varphi) \quad (5)$$

$$Q^k=\frac{1}{2}D^k(n)\sin(\varphi) \quad (6)$$

It can be seen that in order to find a term to feed back to the carrier phase oscillator, the phase error of the local carrier phase replica can be found as:

$$\varphi=tan^{-1}\left(\frac{Q^k}{I^k}\right) \quad (7)$$

In order to minimize the phase error, the quadrature-phase arm correlation should tend to zero and the in-phase correlation must be maximum. To finish, the output of the phase discriminator is filtered to predict and estimate any relative motion of the satellite and to estimate the Doppler frequency. Notice that the arctangent is not the unique discriminator type but is the most precise of the Costas discriminators.

1.3.3.2 CODE TRACKING

In a GPS receiver, the code tracking loop is performed by a Delay Lock Loop (DLL). The main goal for a code tracking loop is to keep the tracking of a determined code in the signal. The output of such code tracking loop is a perfectly aligned replica of the code. In order to generate the signal error, which is used as a reference to apply the necessary corrections and align the local replica with the incoming signal, the DLL correlates the input signal with three different replicas of the code also known as Early, Prompt and Late replicas.

Following the DLL block architecture in Figure 5 from left to right, the first step is converting the C/A code to baseband. This is achieved by multiplying the incoming signal with a perfectly aligned local replica of the carrier wave. Later, the signal is multiplied with three code replicas equally spaced ± ½ chip. Having performed this second multiplication, the three outputs are integrated and dumped. The resulting outputs of these integrations are nothing but a numerical values that indicates how much the code replica correlates with the code of the input signal.

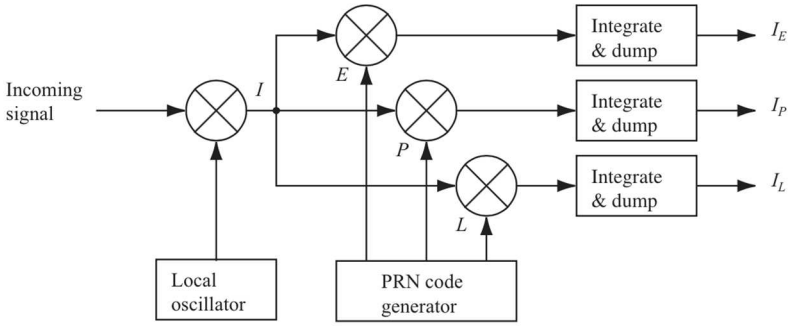

Figure 5 DLL: Basic code tracking loop diagram

Then, the three correlation outputs I_E, I_P and I_L are compared to see which one provides the highest correlation. In case of having a highest correlation in the early or late outputs, the local carrier code phase must be increased or decreased so that the I_P output achieves the highest possible correlation value. Figure 6 (a) shows a clear example of code tracking, where the late code has the highest correlation value, therefore the code phase must be decreased.

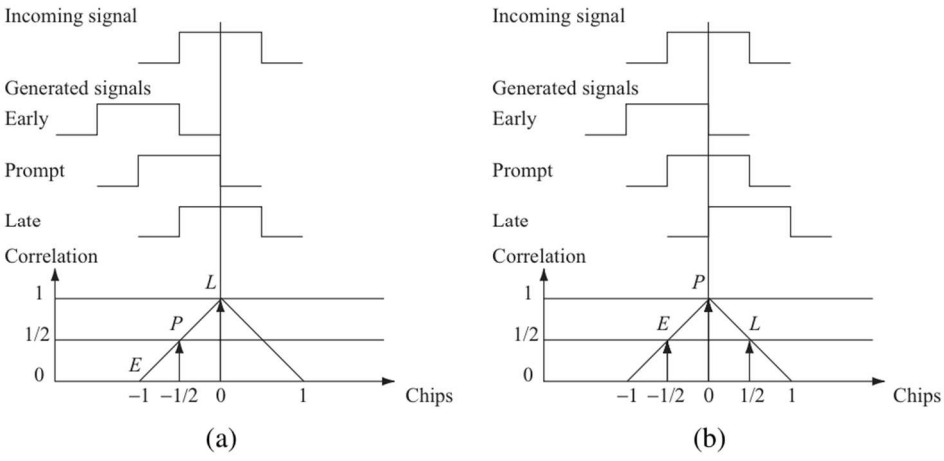

Figure 6 Code tracking.

In Figure 6 (b) the highest peak is located at the prompt replica, and the early and late replicas have equal correlation. Therefore a properly code phase tracking may be ensured.

Assuming that the local carrier is perfectly aligned with the incident signal, the three correlators' outputs may be expressed following the equations below:

$$I_P(\tau) = \begin{cases} 1 - \dfrac{|\tau|}{T_c}, & |\tau| < T_c \\ 0, & |\tau| \geq T_c \end{cases} \tag{8}$$

$$I_E(\tau) = I_p(\tau + \frac{\Delta}{2}) \tag{9}$$

$$I_L(\tau) = I_p(\tau - \frac{\Delta}{2}) \tag{10}$$

where T_c is the chip time, τ is the delay code error and Δ is the early-late separation, which use to be equal to T_c. From these three correlators' outputs, different kinds of discriminators are defined. The following lines are just a brief explanation of the most common DLL discriminators:

- **Coherent discriminator:** the most common coherent discriminator is known as early-minus-late (E-L) discriminator. As its name suggests consists of taking as signal error the difference between the early and late correlator and may be expressed as:

$$\Delta\tau_{E-L}(\tau) = \frac{1}{2}\frac{[I_E(\tau) - I_L(\tau)]}{P} \tag{11}$$

- **Non-coherent discriminator:** it is used when the hypothesis of having a local carrier perfectly aligned is not accomplished. In this particular case, the correlator output is being affected by different effects: an attenuation α due to the correlation integration in presence of frequency residual errors and a complex rotation θ_ϵ due to the integration of complex samples. Both effects are included in the following equation:

$$I_P(\tau) = \begin{cases} \alpha\left(1 - \dfrac{|\tau|}{T_c}\right)e^{j\theta_\epsilon}, & |\tau| < T_c \\ 0, & |\tau| \geq T_c \end{cases} \tag{12}$$

  In this case the attenuation suffered by the signal in the correlator output is irreparable. Therefore, in order to avoid this attenuation, the impact must be previously evaluated by adjusting adequately the correlator integration time. Concerning the phase term, in order to neutralize the possible complex rotation, the following discriminator is proposed:

$$\Delta\tau_{E-L,env}(\tau) = \frac{1}{2}\frac{[y_E(\tau)]^2 - [y_L(\tau)]^2}{[y_E(\tau)]^2 + [y_L(\tau)]^2} \tag{13}$$

This discriminator is nothing but a variance of the early-minus-late discriminator and is known as normalized early-minus-late envelope. With purpose of reducing the discriminator sensitivity to possible changes in the input signal amplitude, this discriminator includes a normalization term.

1.4 ANTENNA ARRAY: BASIC CONCEPTS

The main objective of this section is to give the reader an overview of the essential characteristics of an antenna array, describing its structure, the used nomenclature and the model of the incidental signals into the array. Moreover, the most common geometries will be presented.

1.4.1 SPACE DIVERSITY

Array refers to the antennas set situated in different spots of the space. This antennas or sensors set extracts information from the incident wave's field of its vicinity and it can be used to improve the reception of the signals of interest. This will allow, for example, the determination of the direction of arrival of an incident wavefront, the estimation of signal emitting sources, etc.

Arrays processing refers to any method that coherently uses the received information by the different sensors conforming the array. Any signal processing is based on the obtainment of diversity and on its combination. During array processing, spatial diversity is obtained by means of a spatial sampling of the incidental waves' field. The information obtained by diverse sensors receiving the same signals can be basically used for: the emitting sources location, i.e. the determination of the direction of arrival of the incident waves and the creation of a spatial filter, also known as beamforming.

Taking into account that a spatial frequency corresponds to a direction of arrival, the array can be used as a spatial filter, allowing some spatial frequencies and blocking others. This array characteristic allows the fact of invalidating interfaces and maximising the desired signal. An obvious utility is the case of the GNSS signals, where it is intended to maximise the direct signal and to invalidate all others in order to get the minimum error when measuring the pseudo-

ranges. That is why arrays have found numerous applications in radar, sonar, seismology, wireless communications, radio astronomy, acoustics, and biomedicine, etc.

1.4.2 DATA MODEL

Before talking of array geometries is necessary to make the assumption that a plane wave propagating through an isotropic space is received by an antenna array that consists of N sensors located on the space points r_i, $i = 1..N$, Figure 7.

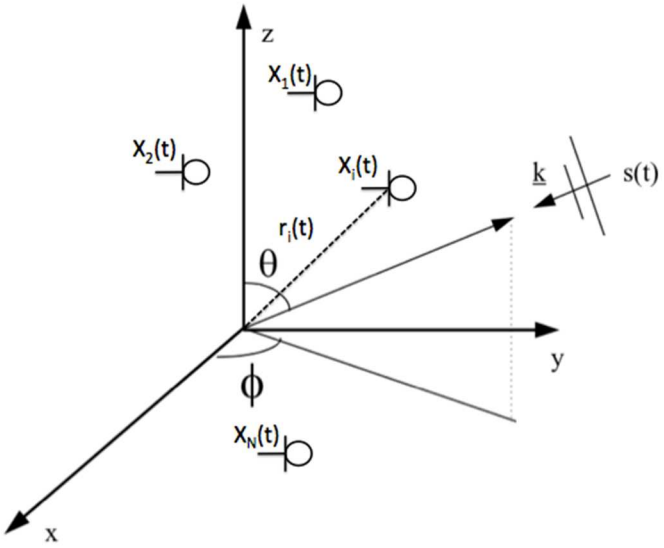

Figure 7 Antenna array receiving signals

Each of the antenna array sensor has a reception diagram $g_i(\theta, \phi, \mathrm{p})$, where θ and ϕ are the angles that defines the direction of arrival of the wave and p is a generic parameter that specifies the dependence with the incident wave polarization.

Taking into account that the receiver performs phase and quadrature processing, in this brief explanation the incident signals will be treated as its low-pass equivalent signal. This assumption allows handling the expressions on a simpler manner and it does not involve loss of generality. The low-pass equivalent signal on each sensor may be expressed as a linear combination of the signals emitted by each source and a noise term and may be expressed as:

$$x_i(t) = \sum_{k=1}^{m} s_k(t - \tau_{i,k}) g_i(\theta_k, \phi_k, p_k) e^{-j2\pi f_0 \tau_{i,k}} + n_i(t) \qquad (14)$$

$$\tau_{i,k} = \frac{k_k r_i}{2\pi f_0} \tag{15}$$

where

- s_k: Low-pass equivalent signal of the kth source being received from a reference point. The reference point is one sensor of the array or a near point.
- $\tau_{i,k}$: Signal propagation time of the source *k* from the reference point to the *i* sensor.
- k_k: Vector wavenumber associated to the signal *k*
- r_i: Position vector of the *i* sensor relative to the reference point.
- f_0: Operational center frequency
- $n_i(t)$: Low-pass equivalent noise of the *i* sensor.
- $g_i(\theta_k, \phi_k, p_k)$: Sensor gain of the *k* source wave.

The phase terms appears when replacing t by $t - \tau_{i,k}$ in the general analytic signal expression $e^{-j2\pi f_0 t}$.

1.4.3 NARROWBAND ARRAY SIGNALS

The fact of considering that the incident signals on the array are narrow band is to assume that the temporary delay of the low-pass equivalent of the received signals on the different sensors that form the array can be ignored. This approximation will be correct, provided that the inverse of the maximum delay experimented by the low-pass between the two edged sensors on the array is much higher than its bandwidth and it will depend on the array's aperture according to:

$$\frac{1}{\tau_{max}} \gg B \tag{16}$$

$$\tau_{max} = \frac{D_{max}}{c} \tag{17}$$

where D_{max} is the maximum size of the array. So that, the condition of narrow band may be expressed according to:

$$B * D_{max} \ll c \tag{18}$$

In the case of working with GPS signals, the narrow band condition is widely accomplished and thus, the following approximation can be carried out:

$$s_k(t - \tau_{i,k}) \approx S_k(t) \tag{19}$$

Using this approximation, it is assumed that the delay between the different sensors is uniquely represented by the phase difference of the carrier signal. The group of signals in all sensors can be expressed in vector form following the expressions below:

$$x(t) = [x_1(t), x_2(t), \dots, x_N(t)]^T \tag{20}$$

$$n(t) = [n_1(t), n_2(t), \dots, n_N(t)]^T \tag{21}$$

$$x(t) = \sum_{k=1}^{m} s_k(t) a(k_k, p_k) + n(t) \tag{22}$$

where $a(k_k, p_k)$ is the array manifold associated to the k signal and it characterizes the response of the array of a particular signal.

$$a(k_k, p_k) = \left[g_1(\theta_k, \phi_k, p_k) e^{-j2\pi f_0 \tau_{1,k}}, \dots g_N(\theta_k, \phi_k, p_k) e^{-j2\pi f_0 \tau_{N,k}}\right]^T \tag{23}$$

Consequently, vector $x(t)$ can also be expressed through a product of matrices:

$$x(t) = As(t) + n(t) \tag{24}$$

where

$$s(t) = [s_1(t), s_2(t), \dots, s_m(t)]^T \tag{25}$$

$$A = [a(k_1, p_1) : a(k_2, p_2) : \dots : a(k_1, p_1)] \tag{26}$$

Assuming that sensors in the array are symmetrical, the gain can be expressed as a multiplier factor of the steering vector and it only represents a modification of the signal's amplitude value. Moreover, due to it is aimed to study the array's behavior uniquely according to its geometry, it will be supposed that sensors have unity gain for all directions and polarizations. In this case, the steering vector takes a common expression which will be used from now on:

$$a(k_k) = \left[e^{-j2\pi f_0 \tau_{1,k}}, \dots, e^{-j2\pi f_0 \tau_{N,k}}\right]^T \tag{27}$$

1.4.4 ARRAY GEOMETRY

In order to introduce the reader to the most basic array configurations, this section is dedicated to give a basic overview of some kinds of array geometries.

Figure 8 shows the uniform lineal array (ULA), which is the most basic configuration of the antenna arrays distributions. This kind of array geometry consists of a distribution of N sensors equally spaced a distance d and situated along a single axis. ULA is the most basic configuration of the antenna arrays distributions.

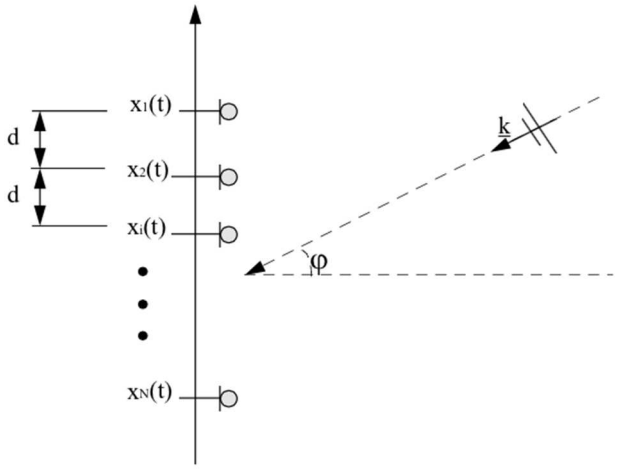

Figure 8 ULA Geometry

Taking as reference the first sensor of the array, the steering vector may be expressed as:

$$a(\varphi) = a(u) = \left[1, e^{-j1u}, \dots, e^{-j(N-1)u}\right]^T \tag{28}$$

$$u = 2 * \frac{\pi d}{\lambda} * sin(\varphi) \tag{29}$$

where λ is the signal wavelength, d the distance between the adjacent sensor, φ the elevation angle where the signal is received and u the phase difference of the incident signal between the adjacent antennas. Notice that since the array geometry has just one dimension, it only allows discriminating incident signals with angles different to its axis. However, there are several applications such as GPS where 2D signals discrimination is required. In these particular cases the usage of another kind of sensor distribution is needed.

The Uniform Circular Array (UCA) is a classic kind of 2-D array geometry and, as may be seen in Figure 9, is composed of N sensors located along a circumference of radius R with an azimuth or angle to the *x-axis* denoted with γ_i. The center of this circumference matches the origin of the coordinate system and for reference purposes is taken as the phase origin point. The antenna array performance directly depends on the number of the sensors it is composed of; the most sensors used the better performance.

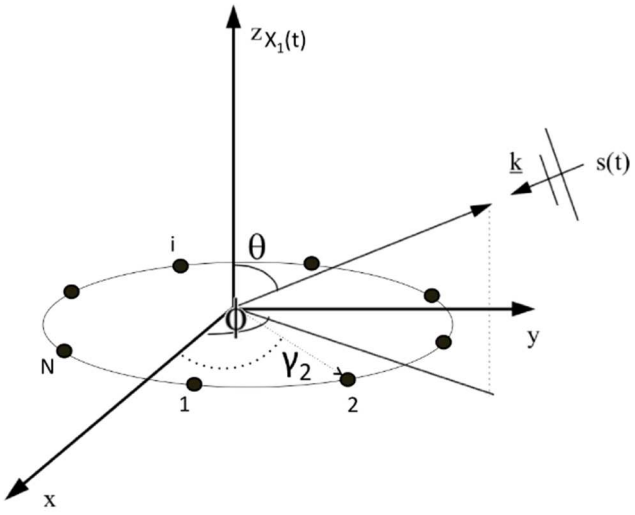

Figure 9 Uniform Circular Array geometry

In this case, the array manifold may be expressed as:

$$a(\theta,\phi) = \begin{bmatrix} e^{\frac{j2\pi R}{\lambda} sen(\theta)\cos(\phi-\gamma_1)} \\ \vdots \\ e^{\frac{j2\pi R}{\lambda} sen(\theta)\cos(\phi-\gamma_N)} \end{bmatrix} \quad (30)$$

1.5 MULTI-ANTENNA SYSTEM APPLICATION

Multi-antenna systems provide measurements from different antennas with a fixed and known separation between them. These extra measurements can be used for different purposes. In particular, as this project is focused on determining the location of a GPS satellite, these extra measurements will be used to compute the direction of arrival (DOA) of a GPS wavefront.

In an ideal configuration and in a clean environment, the measurements received at each antenna should be the same except for the effect caused by the separation between the antennas. Hence, the analysis performed in this section will be based on the differences between the measurements received at each antenna. The underlying concept is based on the estimation of direction of arrival using carrier phase observables.

Figure 10 represent a two antennas ULA array receiving the signals emitted by a GPS satellite.

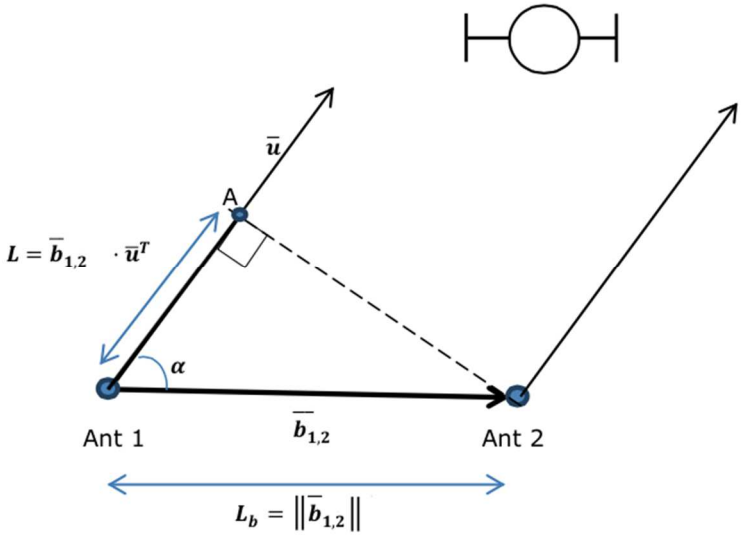

Figure 10 DOA estimation with ULA array

From the graph, $\overline{b_{1,2}}$, is the baseline vector that conforms the two antennas and α represents the satellite elevation angle. This last term can be defined in terms of the scalar product as the projection of the baseline on the line of sight of the satellite:

$$\alpha = arccos\left(\frac{L}{L_b}\right) = arcos\left(\frac{\overline{b_{1,2}}\,\bar{u}^T}{\|\overline{b_{1,2}}\|}\right) = arccos\left(\frac{\Delta\varphi_{1,2}\,\lambda}{2\pi\|\overline{b_{1,2}}\|}\right) \tag{31}$$

Since the Antenna 2 and the point A of the Antenna 1 line of sight, are at same wavefront, the magnitude L can be expressed as the difference between the GPS true ranges of both antennas:

$$L \equiv \Delta R_{1,2} = R_1 - R_2 \tag{32}$$

In order to determine the satellite elevation basing on GPS signals parameters estimated, the measurements that provide most accurate values are the carrier phase observables. Unlike the psudoranges observables, the carrier phase could provide measurements with a precision of centimeters. The observation equation for RAW carrier phase measurements is given as:

$$\varphi_i = R_i + common\ errors + \tau_i + noise + \lambda\, N_i \tag{33}$$

where $common\ errors$ are related to troposphere, ionosphere, orbit and satellite clock errors and are virtually identical for both antennas (meters level), τ is the receiver clock and hardware bias (meters level), $noise$ is the thermal noise and multipath (cm level) and N is the integer ambiguity (cycles).

Neglecting noise for simplicity and notation, the phase simple differences between both antennas could be defined as:

$$\Delta\varphi_{1,2} = \varphi_1 - \varphi_2 = \Delta R_{1,2} + \Delta\tau_{1,2} + \lambda\,\Delta N_{1,2} \tag{34}$$

Since the $common\ errors$ are virtually identical for both antennas, the term is canceled by performing the difference. From the equation, two terms are still disturbing: the differential receiver clock bias ($\Delta\tau_{1,2}$) and the differential integer ambiguity ($\Delta N_{1,2}$). In principle, in a configuration where both receivers of the baseline share the same oscillator, one could think that the differential clock bias should cancel out; however, additional hardware biases make this hypothesis difficult to fulfil in practice. In order to solve this problem, the differential measurement approach can be based on double rather than single differences. For such purpose, one satellite is selected as pivot (typically the one with the higher elevation), then the double differences are defined as the difference between the single differences for both satellites:

$$\Delta\varphi_{1,2}^{j,pivot} = \Delta\varphi_{1,2}^{j} - \Delta\varphi_{1,2}^{pivot} = \Delta R_{1,2}^{j,pivot} + \lambda\,\Delta N_{1,2}^{j,pivot} \tag{35}$$

With this approach, the double differential receiver clock has disappeared, since it is virtually the same for both single differences. Notice that by computing the double differential equations the DOA of the satellite signal cannot be solved, but only the differential direction of arrival (DDOA).

CHAPTER 2

MULTI-ANTENNA RECEIVER PLATFORM

Depending on the application, a multi-antenna system may require different levels of synchronization across multiple SDR's devices. Since the main objective of the proposed system is to create an easy of deployment multi-antenna receiver platform, this document will introduce the hardware and software environment used and the methods applied in order to obtain synchronization between three USRP2 devices.

2.1 RECEIVER SYSTEM APPROACH

As it may be seen in Figure 11, the receiver system is composed of three main blocks. The first block is formed by an antenna array, which is composed of three commercial off the shelf GPS antennas. The second block is composed of an array of three USRP2 radios and their corresponding synchronization hardware and the third block includes the software frontend and Matlab processing scripts.

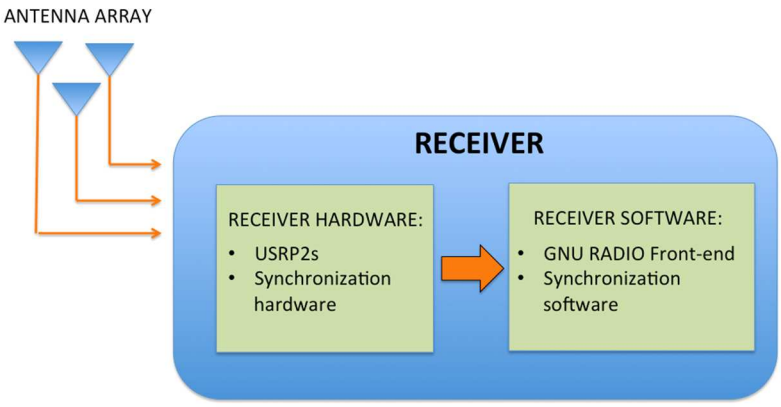

Figure 11 Receiver system approach

The following sections describe the details of each block.

2.2 TIME SYNCHRONIZATION CIRCUIT TOPOLOGY

Before talking about the hardware description it is necessary to know the different options the Universal Software Radio Peripherals offers regarding to the system synchronization. Depending on the synchronization option chosen, different hardware would be necessary. The following section will introduce the synchronization solution chosen in order to obtain time-aligned samples from three different USRPs.

Nowadays there exist two synchronization methods that allow the user transmit or receive time-aligned samples:

- MIMO: MIMO cable.
- External reference: 10 MHz and 1 PPS signals.

The appropriate selection of the synchronization method totally depends on the user requirements. Both methods allow the user to obtain time-aligned samples but the second one offers more flexibility in terms of creating a scalable system. MIMO configuration by itself supports the synchronization of two front-ends, whereas using and external reference configuration the user can create a synchronized system formed by up to 16 USRPs, and obtain high accuracy synchronization if a rubidium source is used as the external reference oscillator. Since one of the main goals of this project is to create a scalable system, the external reference synchronization option was chosen.

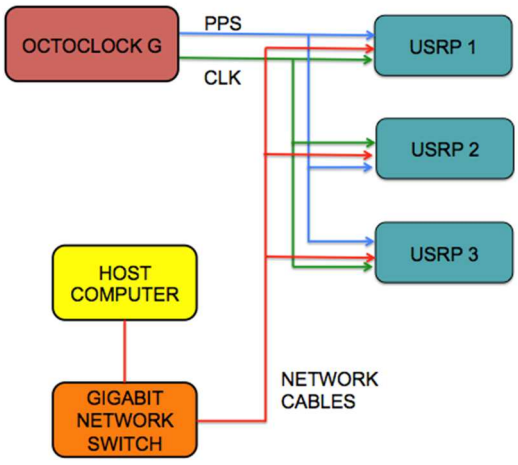

Figure 12 Synchronization hardware topology

Figure 12 illustrates the hardware topology. On the one hand, in order to distribute the external reference signals to the USRPs, an Octoclock-G device will be used. The Octoclock-G is nothing but a high-accuracy timing reference and distribution system, and it will be the responsible for distributing the 10 MHz and 1 PPS signals generated from an internal GPS-disciplined, oven-controlled oscillator (GPSDO), to the front-ends. The GPSDO serves as an accurate reference that allows users to time-align multi-channel systems across the globe within 25 ppb. On the other hand, in order to establish communication and control with the SDRs, a host computer and a gigabit switch are needed. Depending on the host computer operating system, the communication with the front-ends can be performed through different software. In this particular case, the communication will be performed through GNU Radio software working under Ubuntu.

2.3 HARDWARE DESCRIPTION

As shown in Figure 12, in order to achieve time synchronization between several USRPs, external hardware is needed. The receiver hardware it is the responsible for sampling, demodulating and digital storing the signals received. The following sections describe the main characteristics of each receiver hardware component.

2.3.1 USRP2 FRONT-END

A software-defined radio is a system whose components such as modulators/demodulators, filters and amplifiers are controlled by software instead of hardware. As it may be seen in Figure 13, the basic SDR consists of two blocks, one centered on the analog signal conditioning and another focused on the digitalization and processing of the analog signal. Concretely on this project the SDR model used was the USRP2, which correspond to the second generation of USRP devices created by the company Ettus Research.

The USRP can be divided into two main blocks: Daughterboard and Motherboard. On the one hand, the first block amplifies and converts the incoming L-band signal to intermediate frequency or directly to baseband using a broadband I/Q down converter. Both I and Q channels include a linear amplifier with variable gain control, a down conversion mixer and a low pass

filter with both gain and cutoff frequency controls. The filtered signals are further amplified by a differential amplifier before being sent to the Motherboard.

On the other hand, the Motherboard module digitalize the signal with two high speed ADC at 100 MS/s. Both analog-to-digital converters are driven by an internal 100 MHz local oscillator which can be trained by an external reference source through a phase locked loop. Multiple USRP2 radios can be synchronized to form a coherent multiple receiver array by locking the 100 MHz internal local oscillator to an external reference clock and a 1 pulse per second input via a phase-locked loop for precise timing applications.The center part of the USRP2 board is the Motherboard which performs the high sampling rate data processing. The standard configuration in the receiver signal path includes complex mixing which down converts the baseband signal to zero center frequency, decimators that down-sample the signals and a half-bandwidth low-pass filters that further reduce the signal spectral bandwidth by half.

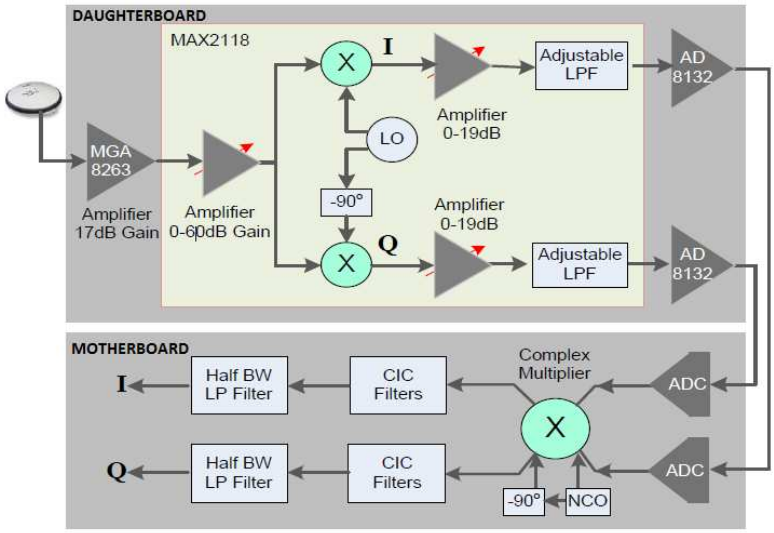

Figure 13 USRP2 block diagram

Finally, the USRP2 combines the I and Q components into a complex signal which is transmitted to a personal computer trough gigabit Ethernet interface.

There are several Daughterboards available for USRP2, each one of them with a different operating frequency range. The reason why the SBX was chosen as the Daughterboard on this project was because of its operating frequency range. SBX Daughterboard has an operational frequency range from 400 MHz to 4.4 GHz, so it totally fulfills GNSS frequency requirements.

Furthermore, regarding synchronization features it offers better performance in terms of automatic software phase alginment.

2.3.2 OCTOCLOCK-G

Building a multi-antenna receiver require multiple USRP2 SDRs sampling multiple antennas at the same time. As mentioned earlier, USRP2 has the ability to train its 100 MHz internal local oscillator to an external reference. By training the local oscillators in all three radios using the same external reference it is possible to achieve sampling synchronization in the receiver array. Each USRP2 trains its local oscillator by using an external 10 MHz reference clock and a pulse per second signal. The 10 MHz reference clock is used to provide the SDRs a common external frequency reference and the 1 PPS signal is used to synchronize the sample time across devices.

To produce a common 10 MHz reference clock and 1 PPS signal an Octoclock-G was used. As may be seen in Figure 14, this device is nothing but a clock distribution module that can provide both frequency and time synchronization for up to eight USRPs by amplifying and splitting an internal GPSDO (GPS-disciplined oven-controlled oscillator). The GPSDO serves as an accurate reference that can be synchronized to the GPS timing standard, this allows to time-align multi-channel systems within accuracy better than 25 ppb.

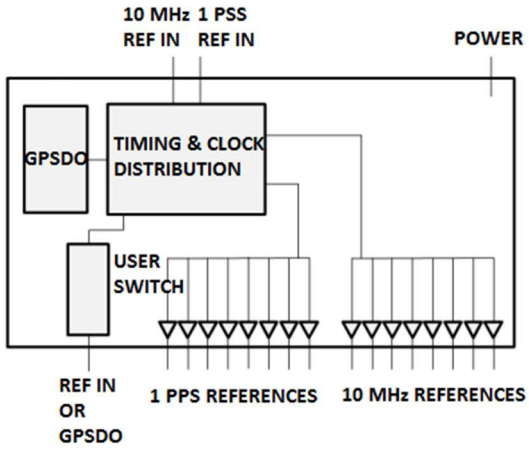

Figure 14 Octoclock-G block diagram

Depending on the USRP model, the synchronization input signals levels may vary. Table 1 presents the Octoclock-G output signals and the USRP requirements regarding input voltage and power levels for both synchronization signals.

	1 PPS SPECIFICATION	EXTERNAL REF. SPECIFICATION
OCTOCLOCK-G	5Vpp	6.9 dBm
USRP2 INPUT	5Vpp	5 to 18 dBm

Table 1 External references specifications

In order to distribute the synchronization signals to all the SDRs avoiding possible phase offsets caused by cabling and connectors, equal length cabling and same number of adaptors on each front-end were required.

2.3.3 SWITCH

As explained before, the Universal Software Radio Peripheral uses the gigabit Ethernet interface to establish communication with a personal computer. In this project up to three USRP were connected to a single personal computer, so the usage of a gigabit switch was necessary. Is important to note that some gigabit Ethernet switches —at least at consumer grade— does not fulfills its specifications, bugs in hardware design may cause fatal packet losses. Therefore, not all the gigabit Ethernet switches on the market would work properly under the continuous stress produced by the different USRPs data transmissions. In brief, to avoid problems regarding data loss and following the manufacturer's recommendations, the switch Netgear GS108 was chosen.

Figure 15 Netgear GS108 switch

Figure 15 shows the frontal panel of the GS108 switch. As may be seen, it consists of 8 gigabit copper ports with 2Gbps bandwidth per port that would allow the connection up to 7 USRPs working simultaneously.

Notice that in order to avoid packet losses due to a possible switch saturation or link congestion, is important to limit the maximum data rate to be transmitted. As may be seen in the Figure 12, depending on the USRP output data rate and format, a possible bottle neck could be produced

on the link that connects the switch and the personal computer. Following the diagram depicted in Figure 12, the maximum transfer rate was limited to 1Gbps and was imposed by the maximum data rate of Cat5e Ethernet chords used on this project. The USRP data transfer rate (DTR) may be expressed following the expression:

$$USRP_{DTR} = Sample\ rate\ \left[\frac{Msamp}{s}\right] * Sample\ data\ format\ \left[\frac{bit}{samp}\right] \quad (36)$$

Having configured the USRPs sample rate to 5 MS/s and the sample data format to 32 bits, it is trivial to determinate that the maximum data transfer rate per USRP was 160 Mbps and the maximum system transfer rate was 480 Mbps. Therefore, with this configuration of the USRPs, the possible bottle neck was avoided.

2.4 SOFTWARE ENVIRONMENT

In order to create a useful guide that may help future researchers to record signals with the multi-antenna receiver, the following section is dedicated to define and describe the software used to establish communication with the SDR's.

2.4.1 USRP-PC COMMUNICATION

In this section, the methods and protocols used to establish communication between the SDR's and the personal computer will be defined. Specifically, the purpose of using UHD and GNU Radio platforms will be explained.

UHD is nothing but a driver that should be installed in both the PC and USRP. By contrast, GNU Radio is a free and open-source software that provides signal processing functions. In order that GNU Radio can work with the received samples, the need to establish a communication with them is needed somehow. Here is the point where UHD and GNU Radio's paths come together. GNU Radio integrates the UHD driver in form of a package which is used as an interface to connect, send and receive data between the USRP's.

- **UHD COMMUNICATION**

  The communication between the SDR's and the personal computer is carried out through IP protocol and UDP packages, which allows the user to connect multiple

devices to the same PC by using an external HUB and distinguishing the different USRP's through its IP's.

Furthermore, as explained above, it is important to differentiate the distinct parts that form the UHD system. Therefore the driver can be classified in two different groups depending on the end device:

- UHD-USRP: Driver/Image that is programmed on the USRP FPGA and it is the responsible of both the communication with the PC and the transmission/reception to the RF stage.
- UHD-PC: Driver that captures and makes available to the users the packets that are sent through the Ethernet port.

For more information regarding UHD installation, see ANNEX I.

- **GNU RADIO**

GNU Radio is a free and open-source software development toolkit that provides signal processing blocks to implement software radios. Software radio is a radio system that performs the required signal processing in software instead of using dedicated integrated circuits in hardware. The software enables the user to perform the entire signal processing graphically using the interface GNU Radio Companion (GRC). This approach enables the user to process signals simply interconnecting predefined filters, channel codes, synchronization elements, and many other elements (blocks). In case the defined blocks do not fit the user specifications they can be easily tuned or replaced.

Although the GRC signal processing is based on the interconnection of predefined blocks, the resulting software output is a python code containing the whole information regarding the graphical interface design. Through this approach the user is free to hand modify any desired parameter of the generated system code and adapt it for his convenience.

For more information regarding GNU Radio installation, see ANNEX I.

2.4.2 DESIGNING A MULTI-USRP RECEIVER SYSTEM

The following section explains how to synchronize multiple USRP devices with the goal of receiving time-aligned samples.

2.4.2.1 DEVICE SYNCHRONIZATION

Nowadays, there exist more than one method to achieve synchronization between several USRPs, however, this section particularly details how to achieve a device synchronization using an external PPS and clock signals. Notice that it is important to know your SDRs features before choosing the synchronization method to use.

In order to form an antenna array system composed of SDRs, the USRP boards must be synchronized. Originally, each board contains its own local oscillator locked to its own reference signal which will cause phase drift between channels and misalignment of samples collected at the same time in different boards. There is where a common clock and PPS reference signals take part:

- A 10 MHz reference is used to provide a single frequency reference for the devices.
- A pulse-per-second (PPS) is used to synchronize the sample time across devices.

Typically, both signals are provided by an external GPSDO but depending on the SDR model these signals may be provided from an optional internal GPSDO. The first signal provides a reference for the system clock and local oscillator, allowing this last one to be tuned to the same frequency giving a frequency synchronized system. The second signal is used to synchronously latch a time into the device. There exist several commands either to initialize the USRP's sample time either to zero or to an absolute time, such as GPS. In order to achieve synchronization on the devices of this project, it doesn't matter the exact time when you initialize the time on the SDRs, it only matters to ensure the front-ends initialize its sample time simultaneously. Having this in mind and since one of the goals is to create an easy of deployment platform, the GRC synchronization method will be explained.

GNU Radio Companion allows the user to synchronize multiple USRP's simply by creating a single block which is capable of control the whole devices. The properly configuration of this

block could be found in the Appendix I; now emphasis will be put on the method that the software uses in order to synchronize the SDR's.

In order to achieve time synchronization using external PPS and clock signals, GRC software uses the function *set_time_unknown_pps(uhd.time_spec()).* The use of this synchronization command is convenience if the user is not interested when the PPS will trigger, it will wait for a PPS on one of the devices, and then will initialize the whole devices timestamps to zero using the command *set_time_next_pps*. It is a 2-step process, and takes at most 2 seconds to complete. Upon completion, the times will be synchronized to the time provided:

- Step1: Wait for the last pps time to transition to catch the edge.
- Step 2: Set the time at the next pps (Synchronous for all boards).

It is important to notice that the *set_time_unknown_pps()* command call is only useful for a device session object with multiple USRPs in it. If the user makes separate objects for each USRP, will want to manually:

- Wait for a PPS on one of the device by calling *get_time_last_pps*,
- Call *set_time_next_pps* on all devices

2.4.2.2 MULTI-USRP RECEIVER IN GNU RADIO

As explained before, GNU Radio is a collection of tools that can be used to develop radio systems in software as opposed to completely in hardware. The software includes a graphical interface, GNU Radio Companion (GRC), which is so similar to Simulink and that allows the user to process signals simply interconnecting predefined blocks. GRC was created to simplify the use of GNU Radio by allowing the creation of python files graphically instead of creating them just in code alone. Following this approach, the Multi-USRP receiver was designed simply by interconnecting predefined blocks.

As shown in Figure 16, the GNU radio Multi-USRP receiver consists basically on five different interconnected blocks:

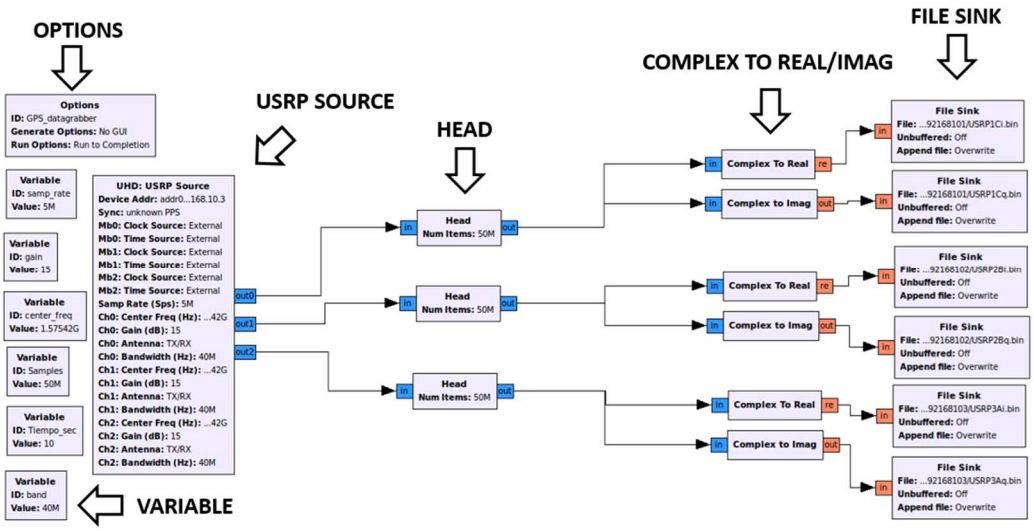

Figure 16 Multi-USRP GPS receiver

- **Options:** used to set the global parameters.
- **Variable:** in order to make the resulting python code more readable, the usage of variables to define constants or basic mathematical operations is recommended.
- **USRP Source:** this block basically acts as a USRP driver and is the one in charge of USRPs configuration and synchronization.
- **Head:** this block is used to restrict the number of samples to receive.
- **Complex To Real/Imag:** the USRP digitalize the RF signal acquired in form of a complex stream (IQ data). This block filters the incoming IQ signal depending on the interested component.
- **File sink:** This block is used to save the incoming USRP digitalized data in a binary file.

Once the graphical design has been done, a python file containing the full code generated graphically with GRC is created. The Figure 17 shows the basic Multi-USRP receiver parameters that can be easily hand changed by the user in order to test different configurations.

```
##################################################
# Variables
##################################################
self.samp_rate = samp_rate = 5*1000000
self.Tiempo_sec = Tiempo_sec = 10
self.gain = gain = 25
self.center_freq = center_freq = int(1.57542e9)
```

```
self.band = band = int(40e6)
self.Samples = Samples = Tiempo_sec*samp_rate
##################################################
# Blocks
##################################################
self.REAL_3 = blocks.file_sink(gr.sizeof_float*1, "/home/USRP3i.bin", False)
self.REAL_3.set_unbuffered(False)
[…]
self.IMAG_3 = blocks.file_sink(gr.sizeof_float*1, "/home/USRP3q.bin", False)
self.IMAG_3.set_unbuffered(False)
```

Figure 17 Python variable code

For more information regarding the design of a Multi-USRP front in GNU Radio, see ANNEX I.

2.5 TESTING THE MULTI-USRP RECEIVER

Having explained the time synchronization circuit topology and the hardware and software environment used to synchronize multiple front-ends, the full system was tested following the diagram depicted in Figure 18.

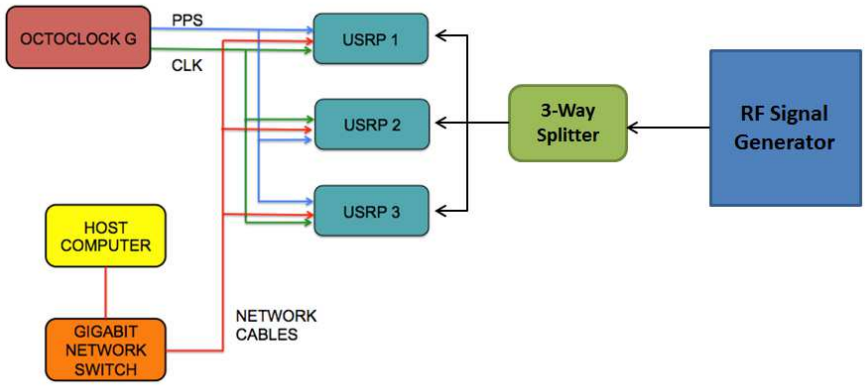

Figure 18 Synchronization test diagram.

As seen in the diagram, a sinusoid tone was generated using a signal generator and was distributed to the front-ends employing a 3-way splitter and matched length cables of the same type and connectorization. The samples were synchronously recorded using the Multi-USRP receiver (Figure 16) and stored in two different files for each USRP. The first file contains the in-phase (real) signal component and the second one the quadrature (imaginary) signal component.

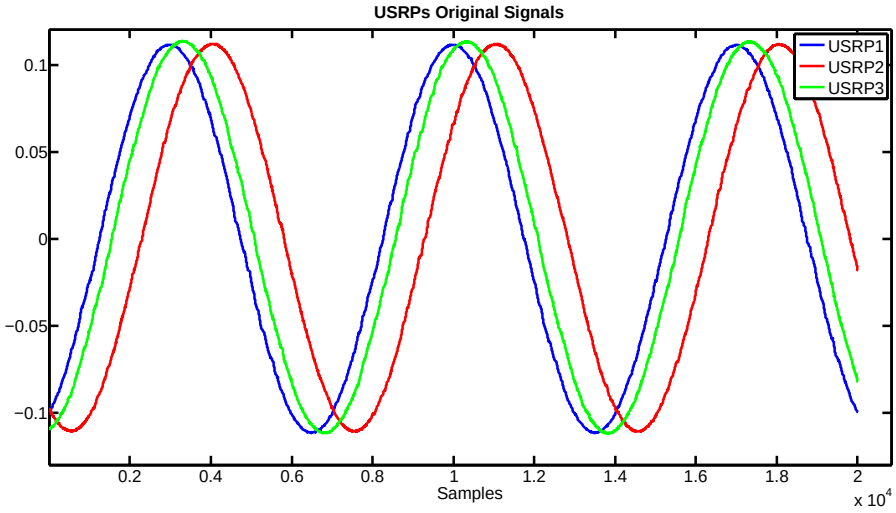

Figure 19 USRP real part data acquired.

Figure 19 shows the real data acquired by each USRP. It is noticeable that the tones acquired by the different front-ends are the same frequency but phase differenced. This phase differences are due to phase ambiguities caused by phased-locked loops which are used for up and down-conversion and may change randomly each time the SDRs are tuned. There are more components aside from local oscillators that contribute to phase errors; filters, mixers and other components produce phase offsets that may vary with time, temperature and mechanical conditions.

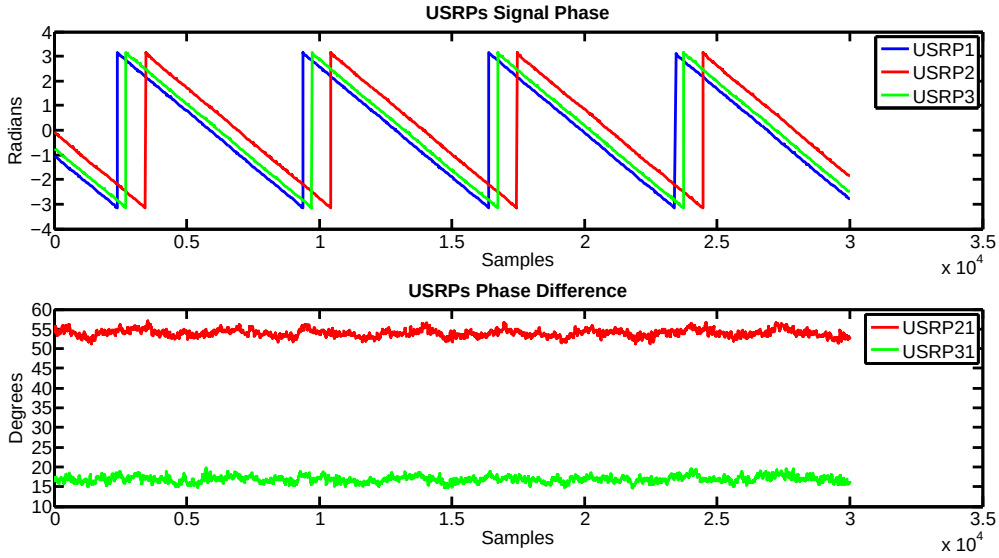

Figure 20 USRPs Phase differences

Figure 20 shows the signal phase of each USRP and the phase differences between the signal acquired with USRP2 and USRP3 compared with the signal acquired by USRP1. USRPs phase differences are may be expressed by:

$$\varphi_{21} = \varphi_2 - \varphi_1 \tag{37}$$

$$\varphi_{31} = \varphi_3 - \varphi_1 \tag{38}$$

where φ_x is the signal phase of each USRP and φ_{x1} is the phase difference between USRPX and USRP1. It is noticeable that, at least a priori, the signal phase differences are constant along a short period of time, being the difference φ_{21} approximately 55 degrees and φ_{31} almost 17 degrees. These phase differences changes randomly after each retune; therefore if phase alignment is required, this random phase offset must be measured and compensated in software. Furthermore, as depicted in Figure 21, depending on temperature and mechanical conditions, USRP phase offsets may be affected producing as a result phase offset variations over time. Thus, in order to obtain time and phase aligned samples, it is necessary to perform a periodic phase calibration.

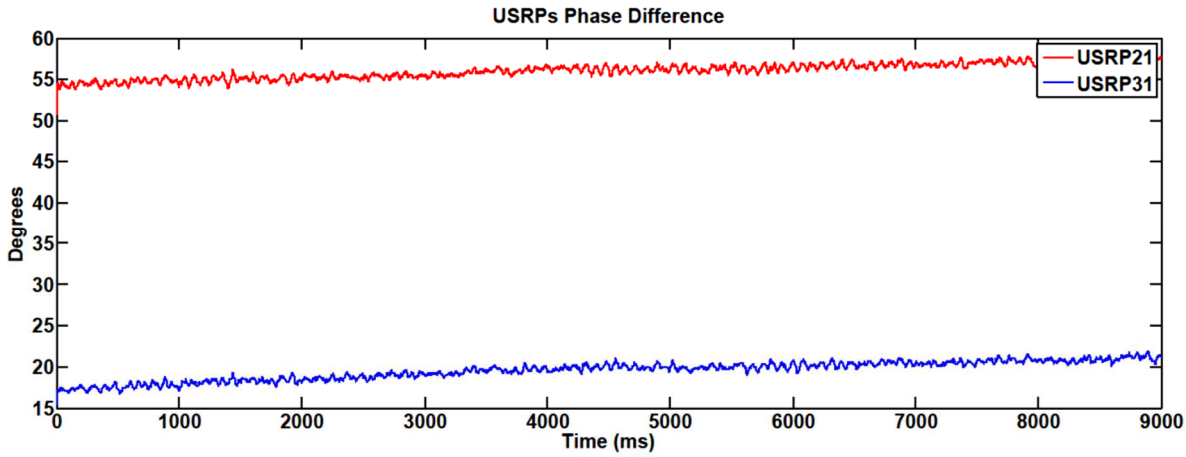

Figure 21 USRPs phase differences time variance

The phase calibration method proposed is performed following the phase correction approach described below:

$$\varphi_2 = \varphi_2 - \varphi_{21} = \varphi_1 \tag{39}$$

$$\varphi_3 = \varphi_3 - \varphi_{31} = \varphi_1 \tag{40}$$

Both formulas take as reference the USRP1 signal phase, so that once performed the phase correction, the 3 SDRs signals phases would be the same. In order to phase correct the complex data recorded by each USRP it would be enough to phase calibrate the signals by multiplying the recorded data per a complex exponential following the expression below.

$$USRPX_{DATA\_ALIGNED} = USRPX_{DATA} * e^{-j\varphi_{X1}} \quad (41)$$

Figure 22 shows the front-ends phase differences after applying (41). It is noticeable that after performing the phase correction method, the phase differences tend to zero, which means that the USRPs signals were phase aligned correctly.

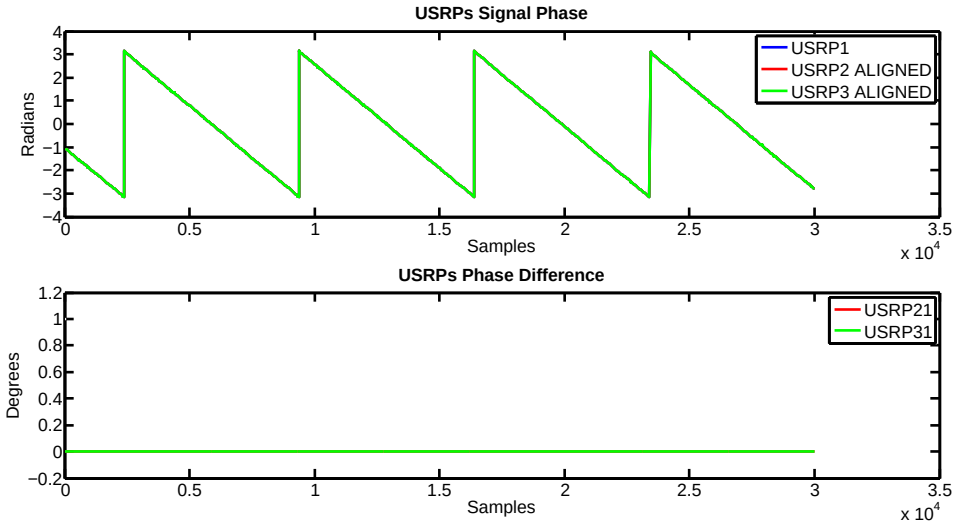

Figure 22 USRPs Phase differences after correction

Finally, Figure 23 shows the comparison between real data acquired by each USRP before and after performing the phase alignment.

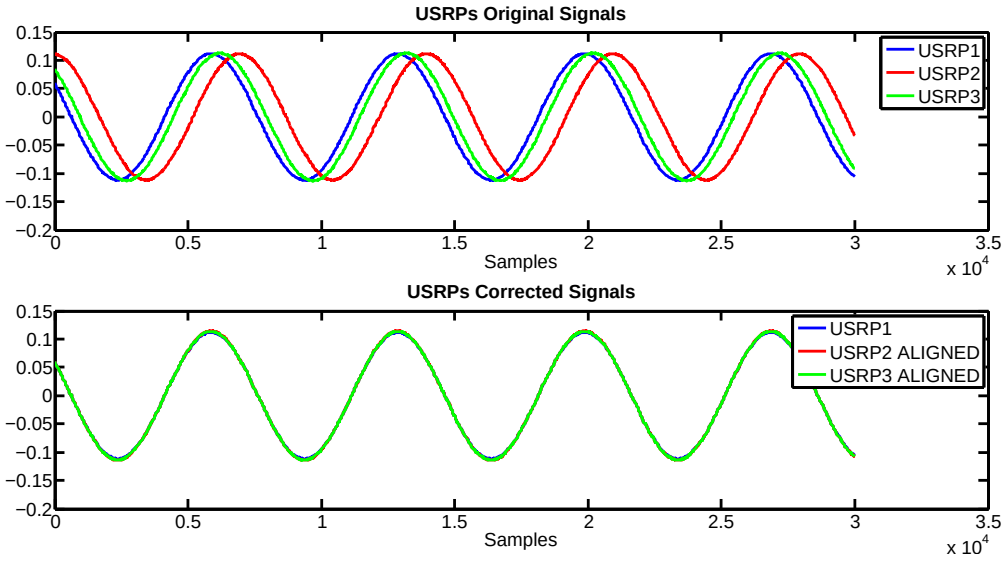

Figure 23 USRPs real signal data corrected

Having executed the phase compensation method, a time and phase alignment between the signals acquired by the different front-ends can be ensured.

2.6 TIME AND PHASE ALIGNMENT CIRCUIT TOPOLOGY

In light of the results achieved in the last section, a periodic phase calibration is needed if time and phase alignment is a requirement. In order to perform samples phase alignment, a phase correction method should be carried out. Depending on the USRP model, this phase correction can be performed in software simply by modifying the python code obtained after generating the flow graph in GRC (Figure 17). But, since the SDRs model used on this project does not support phase compensation by software commands, this process should be carried after the data collection by performing data phase alignment in software.

Nowadays, there exist more than one way to achieve phase-aligned samples by using an external calibration signal. The most used phase correction method is based on directly applying an external calibration signal to the second port of each of the USRP boards in the array and thus discover the phase ambiguity without a calibration step. A clear disadvantage of using this phase correction technique is the need to make a clear distinction of the target signal from the calibration tone. This calibration signal has to be separable in some way from the target signal so that after performing software filtering the target signal can be clearly differentiated.

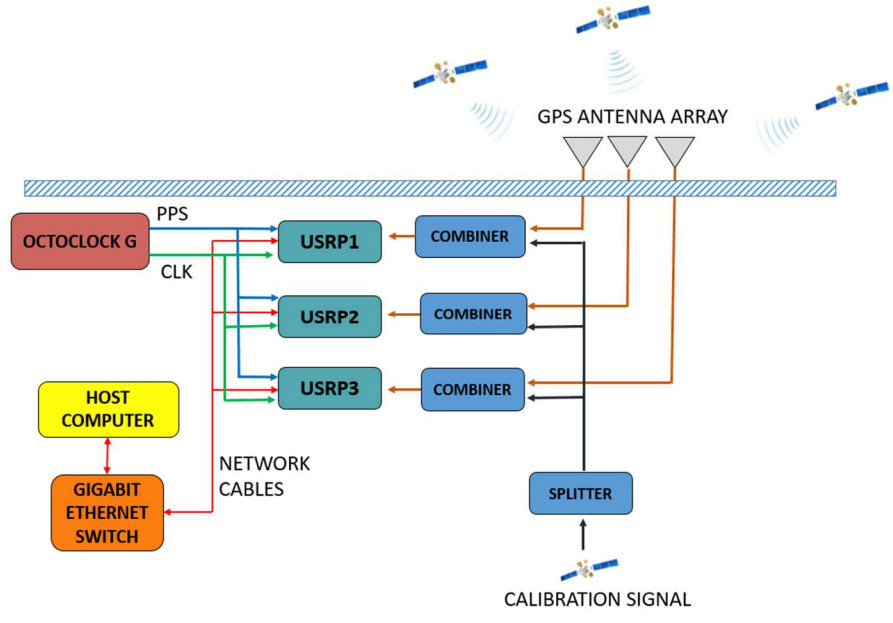

Figure 24 Time and Phase synchronization circuit approach

As may be seen in Figure 24, considering that the multi-antenna receiver created has GPS applications, a synthetic GPS signal is proposed as the phase calibration signal. This GPS synthetic signal is nothing but a pure fictitious GPS satellite signal that is distributed on the

front-ends by using a splitter and matched length cables. The proposed approach allows the user to correct the USRPs phase random offsets by compensating the phase differences observed from a common calibration signal, letting the real GPS phase differences observed are due to the antenna array position. Notice that is absolutely crucial use matched length cables in order that every single front-end receive exactly the same phased signal.

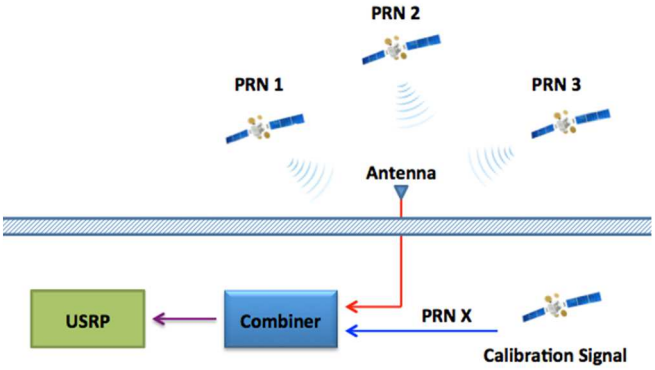

Figure 25 Detail - Calibration signal combining circuit

Figure 25 shows a detailed view of the proposed phase calibration method. Notice that in order to facilitate the diagram comprehension, the circuit presented is reduced to a single front-end. Unlike the basic calibration method explained before, this approach removes the software filtering requirement. The real GPS data acquired by the antenna is combined with the phase calibration signal using a GPS Combiner and it is injected to the RF input port of each front-end. This procedure simply adds a fictitious GPS satellite data to the real GPS data acquired by each USRP so that the impact in the receiver only involves the detection of one more GPS satellite signal.

2.7 HARDWARE UPDATE

In light of the results achieved with the time synchronization circuit (Figure 12), an external signal to perform phase calibration was needed. As may be seen in Figure 24, in order to generate and distribute the phase calibration signal, some hardware devices were added to the basic time synchronization circuit. The main characteristics of the new hardware are described on the lines below:

- **Phase calibration signal generator**: there is more than one way to generate and replay a synthetic radio frequency GPS signal: from using an expensive and unwieldy signal generator to a low-cost dedicated signal generator. Taking advantage of the available

material on the Telecommunication department, the chosen GPS signal replayer was the Labsat.

Figure 26 Labsat front panel

The Labsat is a global navigation satellite system receiver which is capable of recording and replaying real GNSS satellite signals and was designed to test and develop GPS engines. This device was designed to record and replay real GNSS data as well as to recreate the reception of GNSS signals anywhere in the world. For the latter, Labsat uses the SatGen software that allows to simulate almost any kind of test, at a set time and date, anywhere in the world. Table 2 shows the main parameters to take into account when testing a GPS front-end.

LABSAT SPECIFICATIONS	
Sampling frequency:	16.368 MHz
Quantization:	1 bit
RF Output power:	-85 dBm to -115 dBm
Output signal Frequency:	1575.42 MHz
Reference Oscillator	16.368 MHz Temperature controlled +/-2.5ppm. Long term stability +/-1ppm

Table 2 Labsat specifications

Notice that since this signal generator needs a personal computer with Windows operative system, and the operative system used for this project was Ubuntu, it was necessary to add a secondary personal computer destined uniquely to control this device.

- **Combiner:** in order to combine the phase calibration signal and the real GPS data acquired with a GPS antenna, a combiner was required. Table 3 shows the main specifications of the combiner used on this project.

C21 COMBINER SPECIFICATIONS	
Frequency range:	1-2 GHz
In/Output Impedance:	50 Ω

Gain:	1.5 dB
Max RF Input:	30 dBm

Table 3 C21 combiner specifications

- **Splitter:** in order to distribute the phase calibration signal to all the front-ends, a splitter was required. The chosen splitter was a one-input, four-output GPS device and was designed to split the received input up to four devices. Table 4 shows the main specifications of the splitter used on this project.

S14 SPLITTER SPECIFICATIONS	
Frequency range:	1-2 GHz
In/Output Impedance:	50 Ω
Gain:	21 dB
DC IN	3-16 V
Max RF Input:	0 dBm

Table 4 Splitter specifications

Having explained the basic specifications of the full hardware used, some tests were carried out in order to check whether the signal combination approach worked properly or not.

2.8 SIGNAL CALIBRATION TEST

In order to check the calibration signal influence over the real GPS data acquired by the antennas, three tests were performed following the diagram depicted in Figure 27.

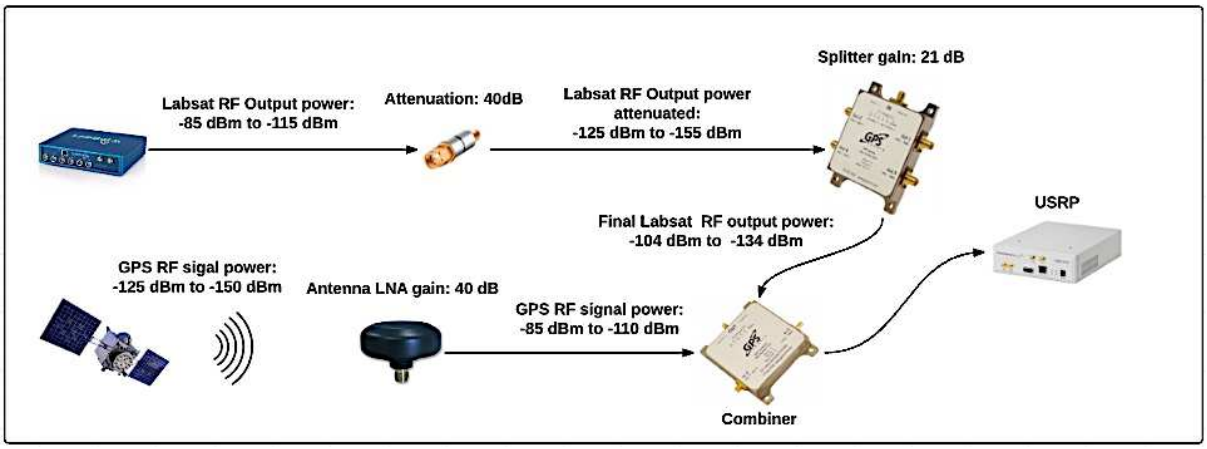

Figure 27 Testing the phase calibration signal influence

For the purpose of having a pure GPS fictitious signal in terms of carrier to noise density, the phase calibration signal reference was generated with 51 dB-Hz of $CN/0$ and with an output power of -85 dBm, for all tests. The attenuation stage was the only parameter that changed all along the experiments. So that, the phase calibration signal output power was the only difference between the tests.

The first test was performed using an external attenuation of 15 dB on the Labsat signal, which implies a phase calibration signal output power of -100 dBm. As depicted in Figure 27, the calibration signal attenuated was distributed to the SDRs through a 4-way splitter and mixed with real GPS data, acquired with a GPS antenna, using combiners. Notice that in order to obtain some reference values of pure real GPS signals, GPS data was recorded directly from the antenna and some seconds later the phase calibration signal was combined.

FIRST TEST				
Antenna Signal		Antenna Signal + Labsat Signal (-85 dBm) - 15 dB attenuation		Difference
SATELLITE	CN0 [dB-Hz]	SATELLITE	CN0 [dB-Hz]	Difference [dB-Hz]
PRN 2	43,93491651	PRN 2	×	SIGNAL LOST
PRN 8	49,83028449	PRN 8	35,67901354	**14,15127095**
PRN 11		PRN 11	50,99369985	LABSAT SIGNAL
PRN 12	45,66811795	PRN 12	×	SIGNAL LOST
PRN 20	43,15276969	PRN 20	×	SIGNAL LOST
PRN 21	49,62707137	PRN 21	38,64132342	**10,98574796**
PRN 25	44,40402218	PRN 25	32,35961984	**12,04440234**
PRN 26	48,24356271	PRN 26	14,38860072	**33,85496199**
PRN 29	45,17784497	PRN 29	×	SIGNAL LOST
PRN 31	48,93985645	PRN 31	34,45964303	**14,48021343**
			Average loss [dB-Hz]=	***17,10331933***

Table 5 First test: GPS real data and fictitious GPS signal combination

As shown in Table 5, after applying 15 dB of attenuation on the phase calibration signal, an average loss of 17 dB-Hz is observed over real GPS data. This outstanding loss on the carrier to noise ratio may imply high degradation and even the tracking loss of many signals. From this results, it can be determined that a higher attenuation of the phase calibration signal is required in order to achieve a minimal impact over real GPS data. For the second test, 30 dB of attenuation on the phase calibration signal was used.

SECOND TEST				
Antenna Signal		Antenna Signal + Labsat Signal (-85 dBm) - 30 dB attenuation		Difference
SATELLITE	CN0 [dB-Hz]	SATELLITE	CN0 [dB-Hz]	Difference [dB-Hz]
PRN 5	42,3749404	PRN 5	36,46315537	**5,911785027**
PRN 8	47,29482168	PRN 8	45,86512347	**1,42969821**
PRN 11		PRN 11	46,78277149	**LABSAT SIGNAL**
PRN 16	49,77107927	PRN 16	46,15927438	**3,611804888**
PRN 18	47,75629808	PRN 18	46,46618577	**1,290112311**
PRN 20	41,75217096	PRN 20	39,49332839	**2,258842566**
PRN 21	41,92197055	PRN 21	40,79519273	**1,126777827**
PRN 25	46,31504373	PRN 25	39,27912463	**7,035919105**
PRN 26	48,33101147	PRN 26	45,90242647	**2,428585002**
PRN 29	46,75012859	PRN 29	44,77443197	**1,975696626**
PRN 31	41,69270223	PRN 31	45,17530875	**-3,482606521**
			Average loss [dB-Hz] =	***2,706922156***

Table 6 Second test: GPS real data and fictitious GPS signal combination

As it appears from Table 6, after applying 30 dB on the attenuation stage, an average loss of 2,7 dB/Hz over real GPS data is observed. Comparing these results to the ones from the first test, a visible improvement is considerable. It is also noticeable that the phase calibration signal carrier to noise density decreased from 51 dB/Hz to 46,7 dB/Hz which means that the attenuation stage causes a degradation of the calibration signal quality in terms of C/N0; the highest attenuation used, the lowest carrier to noise density measured.

THIRD TEST				
Antenna Signal		Antenna Signal + Labsat Signal (-85 dBm) - 40 dB attenuation		
SATELLITE	CN0 [dB-Hz]	SATELLITE	CN0 [dB-Hz]	Difference [dB-Hz]
PRN 5	40,23325895	PRN 5	41,73837754	**-1,505118585**
PRN 8	48,32646265	PRN 8	47,65925523	**0,66720742**
PRN 11		PRN 11	42,04820998	**LABSAT SIGNAL**
PRN 16	45,16923649	PRN 16	43,29931955	**1,869916938**
PRN 20	39,18154283	PRN 20	38,01318868	**1,168354156**
PRN 21	46,68288852	PRN 21	45,6525642	**1,03032432**
PRN 23	41,03681212	PRN 23	41,72086212	**-0,684049992**
PRN 25	47,21864519	PRN 25	44,85259878	**2,366046407**
PRN 26	49,97778815	PRN 26	49,68344534	**0,294342804**
PRN 29	50,3592092	PRN 29	48,61433339	**1,744875806**
PRN 31	47,76299549	PRN 31	46,08133325	**1,681662234**
			Average loss [dB-Hz] =	***0,863356151***

Table 7 Third test: GPS real data and fictitious GPS signal combination

Finally, the third test stands for using an attenuation stage of 40 dB. Table 7 features real GPS data degradation after combining 40 dB attenuated phase calibration signal. In this case, in contrast to the previous tests, an average loss of 0,86 dB/Hz over real GPS data is detected. Moreover, once again, the phase calibration signal decreases from 51 dB/Hz to 42,04 dB/Hz. The mentioned result may be considered a remarkable decrease on the calibration signal but 42,04 dB/Hz of C/N0 is comprised inside the standard parameters of strength GPS signal. In

light of the obtained results, a minimum impact on the real GPS signal can be ensured by using an attenuation stage of 40 dB over the phase calibration signal.

2.9 PHASE CALIBRATION METHOD TEST

Once verified —in the previous section— that the fact of using an adequate attenuation stage in the calibration phase signal is crucial to combine this signal to other received by a GPS antenna, a test of the phase correction method functioning was carried out.

Up to this point, it has been demonstrated that by attenuating 40dB of the calibration phase signal, a minimum impact over the real GPS signals is reached and, moreover, it is possible to obtain a powerful level of the calibration signal in terms of carrier to noise density. With a view to verifying that it is really possible to use an external signal to correct the existing random phase offset between the signals received by the USRPs, some tests using the circuit shown in Figure 28 were carried out.

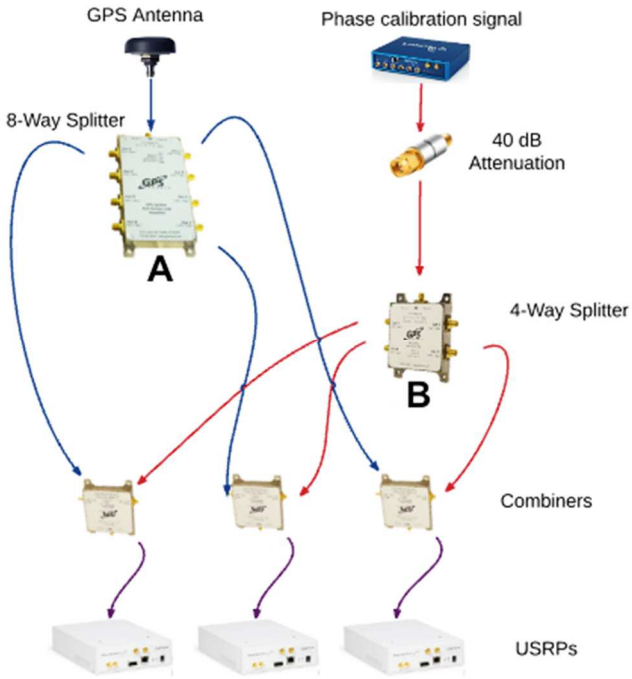

Figure 28 Phase calibration test circuit

On the one hand, as shown in Figure 28, the circuit consists of a phase calibration signal generator, an attenuation stage of the same and a 4-way splitter in order to distribute this signal to the different combiners. On the other hand, the circuit also has a GPS antenna whose signal is routed to combiners using an 8-way splitter. Finally, the signals coming from the two splitters

are combined and distributed to the different front-ends using combiners. The fact of processing the registered data obtained from the different front-ends wants to verify that the signals recorded by the devices are exactly the same for the purpose of ensuring the synchronization in time and phase of the received signals by the different USRPs —once rectified the random phase offset of the receivers.

In order to perform an acquisition and tracking of the obtained GPS signals by the different USRPs, a MATLAB-based software receiver was used. Specifically, in order to compute the phase differences between the various USRPs signals recorded, the Integrated Carrier Phase (ICP) tracking output was used. Although the ICP is often used to smooth code pseudorange for improving accuracy of positioning, in this case, it was used to determine possible phase differences among the USRPs recorded data.

Firstly, with a view to determine the existing random phase offset between the different USRPs, the phase difference among the registered signals by the distinct devices was calculated using the following equations:

$$\varphi_{L21} = \varphi_{L2} - \varphi_{L1} \tag{42}$$

$$\varphi_{L31} = \varphi_{L3} - \varphi_{L1} \tag{43}$$

where φ_{LX1} is the phase difference of the calibration signal between USRPX and USRP1 and φ_{LX} is the *ICP* of the calibration signal recorded by each device. As seen in (42) y (43), the phase differences are calculated by taking the signal phase registered by one front-end as a reference, so the ICP differences observed are relative to the carrier phase estimated by the USRP1. The detected calibration signal phase difference by each device is equivalent to the existing random phase offset among the various USRPs and it needs to be continuously compensated in order to guarantee sample phase alignment between the different front-ends.

Once calculated the current random offset between the USRPs in (42) y (43), it was compensated using the equations below:

$$\varphi_{L2}{}' = \varphi_{L2} - \varphi_{L21} = \varphi_{L1} \tag{44}$$

$$\varphi_{L3}{}' = \varphi_{L3} - \varphi_{L31} = \varphi_{L1} \tag{45}$$

where $\varphi_{LX}{}'$ is equal to the calibration signal phase registered by the USRPX once compensated. By extracting the random phase offsets it was possible to get the same calibration signal for each device recording, so that the phase differences between the rectified signals and the reference one were zero. Equations below clearly show the reasoning applied:

$$\varphi_{L2'1} = \varphi_{L2'} - \varphi_{L1} = 0 \tag{46}$$

$$\varphi_{L3'1} = \varphi_{L3'} - \varphi_{L1} = 0 \tag{47}$$

where $\varphi_{LX'1}$ is equal to the calibration signal phase difference among the recorded an phase rectified samples by USRPX and the reference front-end.

Once defined the phase correction method, its impact over real GPS signals obtained by the antenna was studied. The following equations show the expected effect on the satellite signal phase received by the antenna:

$$\varphi'_{S2} = (\varphi_{S2} - (\varphi_{L2} - \varphi_{L1})) \tag{48}$$

$$\varphi'_{S3} = (\varphi_{S3} - (\varphi_{L3} - \varphi_{L1})) \tag{49}$$

where φ'_{SX} is the satellite signal phase received by USRPX after modifying the random phase offset. So that, the satellite signals phase difference —once rectified— can be defined by:

$$\varphi'_{S21} = \varphi'_{S2} - \varphi_{S1} = (\varphi_{S2} - \varphi_{S1}) - (\varphi_{L2} - \varphi_{L1}) \tag{50}$$

$$\varphi'_{S31} = \varphi'_{S3} - \varphi_{S1} = (\varphi_{S3} - \varphi_{S1}) - (\varphi_{L3} - \varphi_{L1}) \tag{51}$$

where φ'_{SX1} is the real GPS satellite signals difference between USRPX and USRP1.

Once established the phase correction method, it was implemented by combining a phase calibration signal in the form of a fictitious GPS satellite with real GPS signals recorded by a unique antenna. As seen in Figure 28, in order to check the proper functionality of the phase rectifying method, some tests were carried out using the same GPS antenna so that, all the received signals by different front-ends were exactly the same. This test wanted to prove that, once applied the phase compensation method on the received signals by two of the USRPs—using (50) and (51)—, a phase difference close to zero should be obtained both to satellite signals and reference signals, as the following equations show:

$$\varphi'_{S21} = (\varphi_{S2} - \varphi_{S1}) - (\varphi_{L2} - \varphi_{L1}) = \varphi_{OFFSET21} - \varphi_{OFFSET21} = 0 \tag{52}$$

$$\varphi'_{S31} = (\varphi_{S3} - \varphi_{S1}) - (\varphi_{L3} - \varphi_{L1}) = \varphi_{OFFSET31} - \varphi_{OFFSET31} = 0 \quad (53)$$

As seen, theoretically, it should be possible to check that the existent phase offset between two different USRPs is exactly the same for any registered signal —by applying strictly the same signals to the 3 different front-ends input.

Down below, the following example shows the obtained results after applying (50) and (51). The PRN code assigned to the fictitious satellite, the one that was used as a phase calibration signal, corresponds to PRN 17; the rest of the codes make reference to the real GPS signals obtained using the antenna.

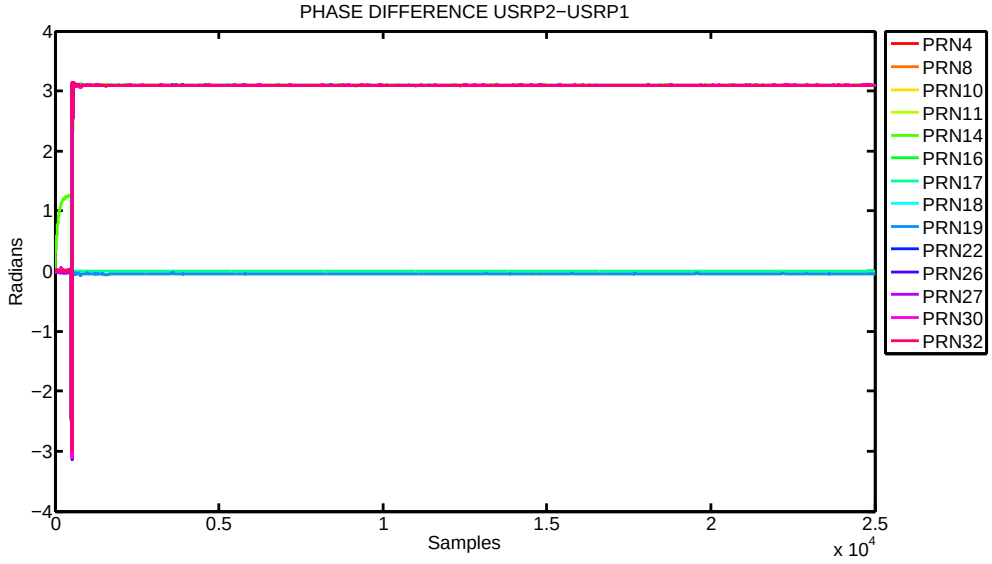

Figure 29 Phase Difference USRP2-USRP1

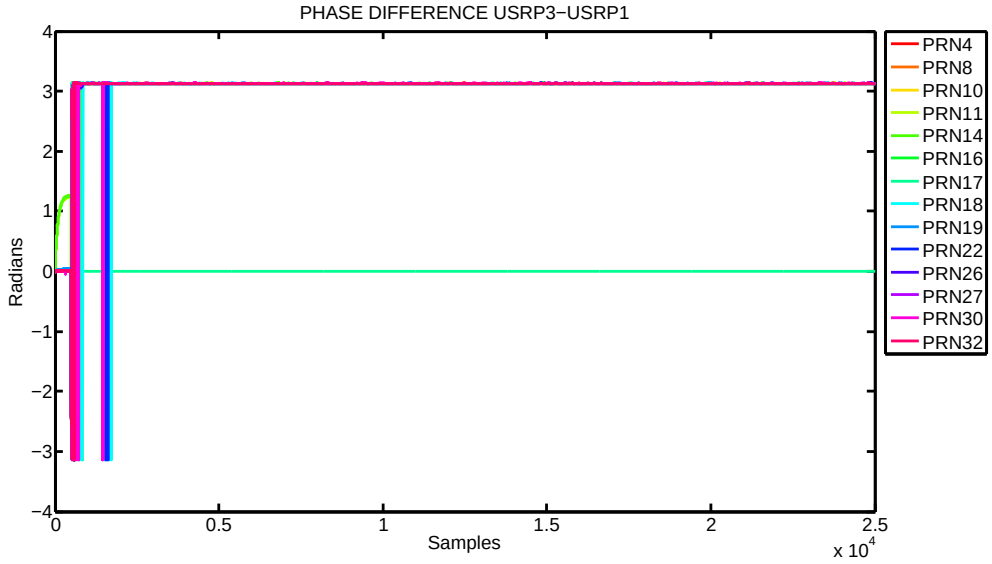

Figure 30 Phase difference USRP3-USRP1

As Figure 29 and Figure 30 show, the obtained results are not the expected ones. As these graphs present, the phase differences are not close to zero for some satellites. Instead of this, there exist π radian gaps between the expected phase difference and the obtained one; remarkable fact because, as commented above, all distributed signals to the different USRPs were exactly the same. This lack of precision on the phase difference calculation is due to the Phase Locked Loop (PLL) of the used receiver. The employed Matlab based receptor uses the Costas loop to estimate the received signal phase. The mentioned receptor —depending on the signal disturbances and the way the loop starts up— can randomly estimate a signal phase converging in values with a $\pm\pi$ difference. The ambiguity is intrinsic because it contains a signal with bits of +1 or -1 and, consequently, it causes an ambiguity of 180º in the signal phase estimation.

For the purpose of correcting this phase ambiguity caused by the PLL of the receptor, the method shown in Figure 31 was used.

```
% Navigation data bit differences detection USRP2-USRP1
if NavData_Ant2(count2,:) - NavData_Ant1(count1,:) == 0
Sat_ind2_1 = 1;
end
% Navigation data bit differences detection USRP3-USRP1
if NavData_Ant3(count3,:) - NavData_Ant1(count1,:) == 0
Sat_ind3_1 = 1;
End
% Navigation data bit differences detection USRP3-USRP2
if NavData_Ant3(count3,:) - NavData_Ant2(count2,:) == 0
Sat_ind3_2 = 1;
end

% Antenna 2 Phase correction
if Sat_ind2_1 == 0 && Sat_ind3_2 == 0
Phase_Antenna2(count2,:) = wrapping(Phase_Antenna2(count2,:)+pi,pi);
end
% Antenna 3 Phase correction
if Sat_ind3_1 == 0 && Sat_ind3_2 == 0
Phase_Antenna3(count3,:) = wrapping(Phase_Antenna3(count3,:)+pi,pi);
end
% Antenna 1 Phase correction
if Sat_ind2_1 == 0 && Sat_ind3_1 == 0
Phase_Antenna1(count1,:) =wrapping(Phase_Antenna1(count1,:)+pi,pi);
end
```

Figure 31 Phase ambiguity correction

This method basically involves the use of the Navigation Data Bit of the received signal as an indicator of possible errors on the estimated phase. The 180º ambiguity of the Costas PLL can be resolved by referring to the bit detection result of the navigation data bit demodulation. If the navigation data bit estimated in two signals is the same, then the carrier phase indicated by the Costas PLL is correct. On the opposite, if the data navigation data bit is inverted, then the carrier phase indicated by the Costas PLL phase can be corrected by adding 180º to the faulty signal phase. Figure 32 and Figure 33 shows the results acquired after applying the phase ambiguity correction method.

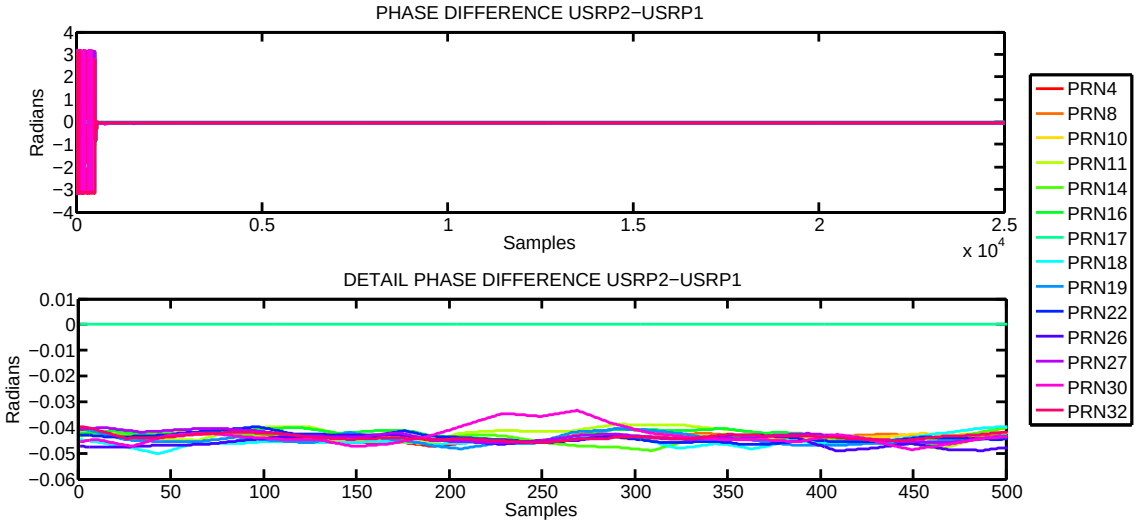

Figure 32 USRP2-USRP1 Phase differences after applying phase ambiguity correction

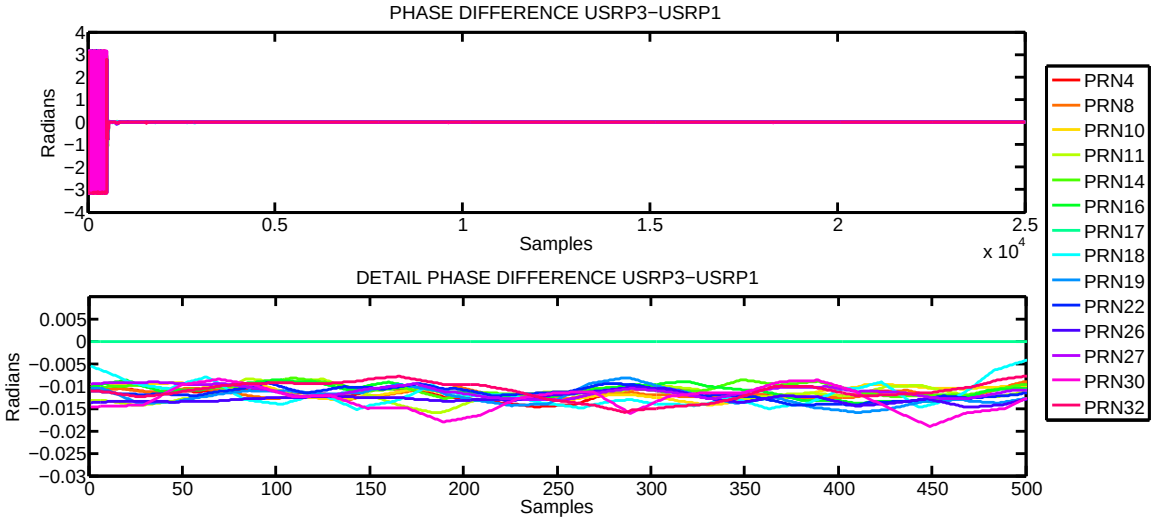

Figure 33 USRP3-USRP1 Phase differences after applying phase ambiguity correction

As reflected in the graphs above, the obtained results after applying the phase ambiguity correction accord to the expectations. Although, at first glance, it seems that the lack of

precision on the double differences calculation has been solved after correcting the phase ambiguity, it is worth noticing that, in case of analyzing obtained results in detail, there exist some errors which vary depending on the used front-ends.

On the one hand, for the case of the phase differences between USRP2 and USRP1 (Figure 32), it may be observed that there exist an error of $-0.04\,rad$ in the satellite signal phase differences. On the other hand, for those differences between USRP3 and USRP1 (Figure 33), it can be deduced that there exist an error of $-0.01\,rad$ in the satellite signal phase differences. The reason why the satellite signal phase differences are not exactly zero is due to the use of different splitters model both on the signal from the antenna and the one from the calibration phase signal generator. So that, it could be said that the obtained values on the results are product of the phase differences on the different splitters outputs. The equations below —based on (50) and (51)— show the applied reasoning:

$$\varphi'_{S21} = \left[\varphi_{OFFSET21} + \varphi_{SPT\_A21}\right] - \left[\varphi_{OFFSET21} + \varphi_{SPT_{B21}}\right] = \varphi_{SPT\_A21} - \varphi_{SPT\_B21} \qquad (54)$$

$$\varphi'_{S31} = \left[\varphi_{OFFSET31} + \varphi_{SPT_{A31}}\right] - \left[\varphi_{OFFSET31} + \varphi_{SPT_{B31}}\right] = \varphi_{SPT\_A31} - \varphi_{SPT\_B31} \qquad (55)$$

where parameters φ_{SPT_AX1} and φ_{SPT_BX1} make reference to the phase difference of the splitter A and splitter B ports, respectively. As shown, obtained values in satellite signal phase differences correspond to the existent phase differences of the splitters' multiple outputs that were used to distribute signals in the circuit shown in Figure 28.

After seeing that an external signal can be used to calibrate the current random phase offset of each USRP, the temporary synchronization of the registered samples by each front-end was checked. One way to check the samples synchronization is to use the Code Phase Tracking output parameter as an indicator of a possible time lag among the registered signals. A GPS receiver determines the travel time of a signal from a satellite by comparing the pseudo random code it's generating, with an identical code in the signal from the satellite. The receiver "slides" its code later and later in time until it syncs up with the satellite's code. The amount it has to slide the code is equal to the signal's travel time. Code Phase parameter is nothing but these propagation times expressed in chips or meters and is particularly used to solve for the user position using trilateration.

Graphs in Figure 34 show the code phase differences calculated based on the results obtained from the recorded signal tracking by USRP1 as reference values. Two different scenes can be distinguished: the first one reflects the code phase difference of PRN17 calibration signal and the second one reflects one of the real satellite signals, PRN32.

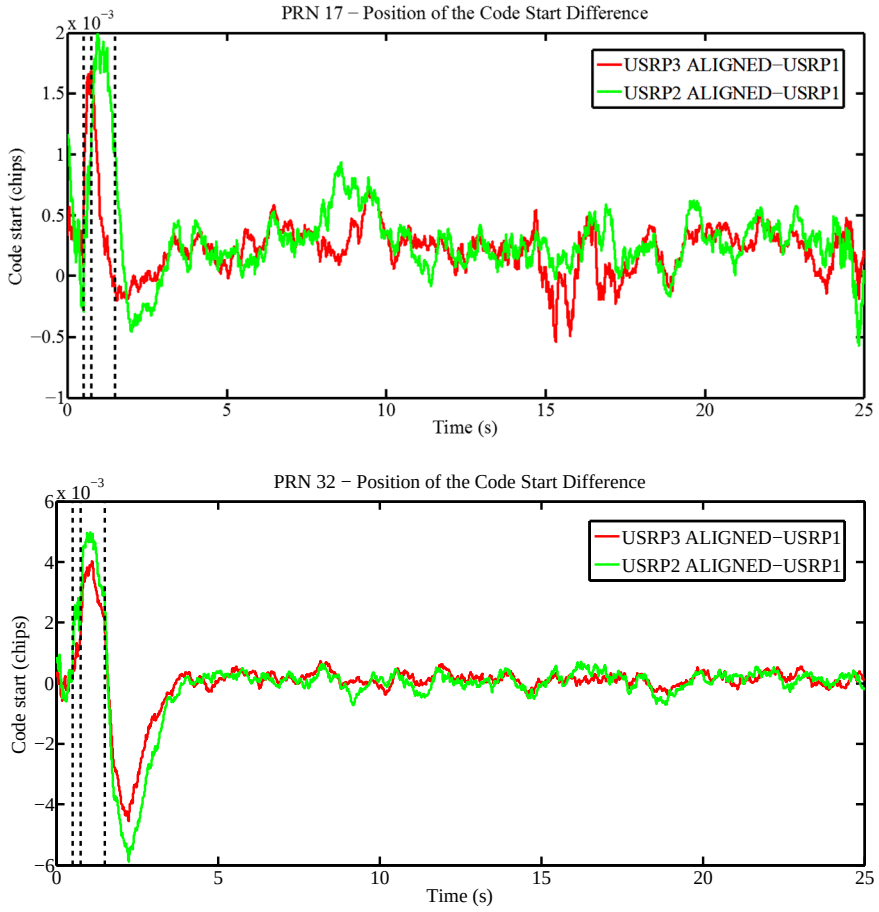

Figure 34 Position of the Code Start Difference

According to the graphs, it can be deduced that, although differences are not constant, values are practically zero. Table 8 shows a summary of the observed code phase difference standard deviation. As it may be seen, there exist fluctuations of the order of centimeters on the code phase differences and, although they should be ideally zero, the low standard deviation observed indicates that the data points tend to be close to the mean and hence to zero. These little errors may be caused due to the thermal noise on the first stage amplifier of the different front-ends. The DLL correlates the GPS incident signal with local replicas with the objective of achieving a local replica perfectly aligned with the incident carrier wave. The noise mismatch between the different front-ends may cause discrepancies on the delay locked loop (DLL) and therefore differences on the code phase estimations. The low standard deviation and

the near to zero mean observed in the code phase differences is a good indicator of temporary synchronization between the front-ends.

	PRN 17	PRN 32
USRP2-USRP1	6.17 [cm]	5.57 [cm]
USRP3-USRP1	6.51 [cm]	5.11 [cm]

Table 8 Code phase differences standard deviation

To conclude, even though there exist variations in signal phases from the different splitters, the correction of the existing random phase offset in the different front-ends by using a calibration phase signal was ensured. Moreover, the phase estimation ambiguity of the Costas loop could be solved by using the navigation data bit as an indicator of possible phase shifts in the carrier phase estimation. Furthermore, it has also been possible to prove temporary samples synchronization of the registered signals by analyzing the signals code phase differences.

CHAPTER 3

ANTENNA ARRAY DESIGN

This chapter will analyze the components that make up the antenna array designed for this project and it will also justify its geometry. Moreover, all registered data obtained by these antennas will be analyzed in order to detect possible undesired effects resulting of the antennas mutual interaction.

3.1 ANTENNA ARRAY GEOMETRY

It is well understood that in order to solve one-dimensional direction of arrival of a wav front, Uniform Linear Arrays (ULA's) have been studied extensively (Hua, Sarkar, & Wiener, 1991). For two-dimensional wave arrival information, a 2-D array of sensors is needed. A conventional 2-D array is the L-Shaped Array (LSA), which consists of two ULA's connected orthogonally at the edge of each array. In this project a simple structure L-shaped 2-D array formed by the perpendicular interconnection of two ULA's array is presented.

The choice of the LSA model within the many existing geometrical arrays was basically determined by the limitation of the number of front-ends available in the university, which directly limit the number of antennas. Specifically, the university had three available front-ends and, therefore, since one of the aims of this project was to determine the two-dimensional direction of arrival of GPS signals, the design of the antenna array was drastically reduced to a triangular geometry. Within the many possibilities in designing a triangular array, the right-angle triangle was chosen.

Researchers concluded that for 2-D wave direction of arrival finding, the L-shaped array has the higher accuracy potential than many other simple structured arrays (Hua, Sarkar, & Wiener, 1991). Figure 35 shows the antenna array configuration, which uses the x-y plane and consists of three radiating elements. Considering that —for referencing purposes— both linear arrays share the element placed at the origin, it could be determined that each linear array consists of two elements.

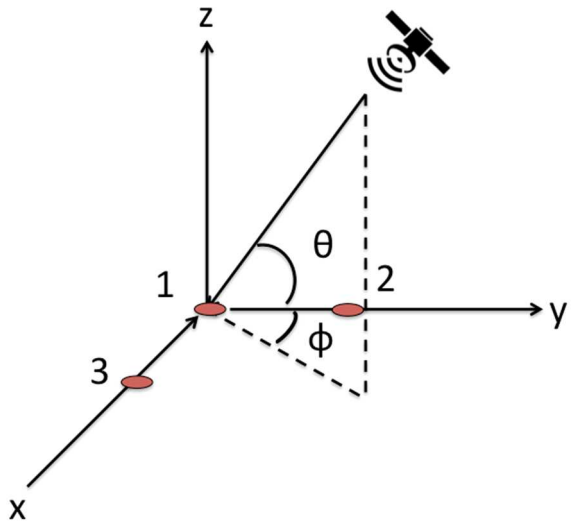

Figure 35 Geometrical configuration of the antenna array

Figure 35 shows an uncorrelated narrowband source impinging on the L-shaped antenna array from far-field, where the source has an elevation angle of θ and an azimuth angle of ϕ, with $\theta(0 \leq \theta \geq \pi/2)$ and $\phi(0 \leq \phi \geq 2\pi)$. The signal derived by any sensor can be represented as a phase-delayed version of the original signal impinging on the reference sensor. As shown in Figure 36, in this particular case, the reference sensor is the one situated on the origin of the coordinate system. The equations that relate the differential phase measurements can be expressed as:

$$\begin{bmatrix}\varphi_{21}\\ \varphi_{31}\end{bmatrix} = \frac{2*\pi}{\lambda} * \begin{bmatrix}x_1 & y_1\\ x_2 & y_2\end{bmatrix} * \begin{bmatrix}u\\ v\end{bmatrix} + \begin{bmatrix}N_2\\ N_3\end{bmatrix} + \begin{bmatrix}\vartheta_2\\ \vartheta_3\end{bmatrix} \quad (56)$$

where φ_{21} and φ_{31} stand for the phase difference between the Antenna2-Antenna1 and Antenna3-Antenna1, respectively. As it may be seen in Figure 36, since the antenna array uses the *x-y* plane, the values (x_1, y_1) and (x_2, y_2) are nothing but the coordinates of the two antennas relative to the reference one; N_2 and N_3 are the possible phase ambiguities; ϑ_2 and ϑ_3 represent the possible phase shift between antennas and u and v are the so-called directional cosines that depend on the DOA of the satellite as follows:

$$u = \cos(\theta) * \sin(\phi) \quad (57)$$

$$v = \cos(\theta) * \cos(\phi) \quad (58)$$

being θ and ϕ the elevation and azimuth of the satellite, respectively.

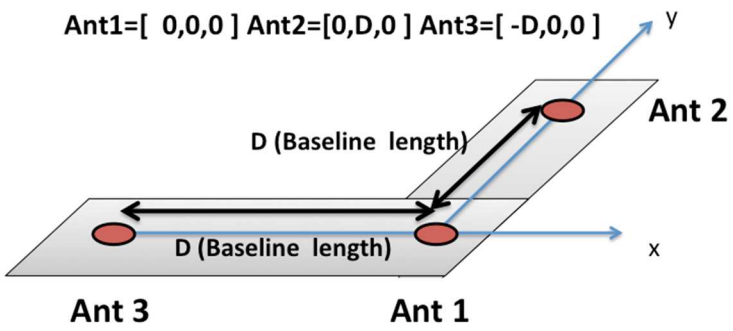

Figure 36 Antenna array geometry

If the antenna separation is smaller than half the wavelength, the ambiguity terms can be ignored. Otherwise, they are in general not null and must be considered in order to solve the equations. Likewise, the term that represents the possible phase shift has to be considered depending whether the system has been calibrated or not. Depending on the explicitly need of considering the ambiguity terms and/or the residual terms, four situations are possible:

1. $N_k = 0$ and $\vartheta_k = 0$

   If there are neither ambiguity nor residual terms, the system of equations can be directly solved and the DOA can be easily computed.

2. $D > \frac{\lambda}{2}$ and ϑ_k = Known value

   If the antennas are more than half the wavelength apart but the system is calibrated, it is possible to find a discrete set of solutions and even find the correct point in the ambiguity space is not very large. The solutions lead to a discrete set of possible DOAs or to a unique one if the ambiguity is solved.

3. $D \leq \frac{\lambda}{2}$ and $\vartheta_k \neq 0$

   If the antennas are up to half the wavelength separated but the system is not calibrated, the equations cannot be directly solved because there are more unknowns than equations. There are two ways to proceed:

a. Using satellite signals whose DOA is certainly known to determine ϑ_k. Having performed this procedure the system can be considered calibrated and therefore, it can be solved in the same way of the first studied case.
b. Computing the differences between equations corresponding to two satellites. This procedure is known as 'double difference estimation'. The unknown variables in the resulting equations are the differential DOA (DDOA), which can be obtained since the residual phase shifts have disappeared by computing the difference. In this particular case the DOA of each signal cannot be obtained, but only the DDOA between each pair of signals.

4. $D > \frac{\lambda}{2}$ and $\vartheta_k \neq 0$

   If the antennas are separated more than half wave length and the system is not calibrated, there are two ways of proceeding:

   a. Using a satellite whose DOA is certainly known. This allows to measure the residual terms but the ambiguities would remain as unknowns since they are different for each satellite. In this case the solving method would be the same as in the second case.
   b. Computing the difference between equations corresponding to two satellites. This case falls back to the second studied case but instead of solving the DOA, the DDOA would be estimated.

It is clear that the best possible case would be the first one because it provides a unique and unambiguous identification of the DOA. The other cases present some limitations and the result of many of them is a set of possible DOAs or DDOAs. In order to perform DOA estimation of the GPS signals impinging the L-shaped antenna array designed on this project, the third case of DOA estimation will be evaluated in the next chapter.

3.2 COTS ANTENNA SELECTION

The need of flexible antenna array geometries has led to the researchers to evaluate the possibility of using Commercial Off-The-Shelf (COTS) antennas as elements of an antenna array. With this freedom in array geometry design using COTS components, the important issue of possible undesirable effects on performance resulting from the interaction of the elements that conform the array is raised. As the characteristics of antenna model affect the array

performance, it is important to evaluate the COTS antenna characteristics before selecting the model to use.

The COTS antenna selection is usually determined by high gain and great out-of-band rejection. Table 9 shows the benchmarking study of three models of commercial off-the-shelf antennas based on its main specifications.

Spec / Antenna	WS3978-HR	3978D-HR	WS3997
RF Frequency	1572.5~1.578 MHz	1575.42±10 MHz	1572.5~1.578 MHz
Antenna Gain	@90º 3 dBic @20º -2.0 dBic	@90º 3 dBic @20º -2.0 dBic	@90º 3 dBic @20º -2.0 dBic
LNA Gain	40 dB	40 dB	30 dB
Noise figure	3.1 dB	0.5 dB	1.5 dB
Out-of-band rejection	± 40 MHz: 80 dB	± 40 MHz: 35 dB	± 40 MHz: 40 dB

Table 9 COTS antenna comparative

The three antennas are equipped with SAW filters for rejecting out-of-band signals and have also Low Noise Amplifiers (LNA) embedded. As may be seen in Table 9, the three models of COTS antennas present high gain LNA and great out-of-band rejection. However, since these antennas were designed to function in a stand-alone manner, there are no mutual coupling specifications.

Basing on the mutual coupling study performed by Chen (2012) and in the antenna LNA gain and its out-of-band rejection, the antenna model selected to form part of the antenna array designed on this project was the model WS3978-HR.

3.3 ANTENNA ARRAY SOLUTION

From the mechanical point of view, the main goal of this project was to build up a flexible antenna array in terms of distance between its components. This distance flexibility would allow the user to perform tests with different separations between the antennas before defining their final position for the data recording. As may be seen in Figure 37, the three antennas that

conform the array solution were placed at the vertices of a rectangular triangle with variable length sides, taking into account that the reference point is the vertex of the right angle of the rectangle triangle. The minimum length of the rectangular triangle antenna array was set to be half the wavelength and the maximum length was set to one wavelength. Having in mind that this project is focused on the reception of GPS L1 signals and that the center frequency of this band is 1.57542GHz, the minimum and maximum separation between the antennas was 9.5 cm and 19 cm, respectively.

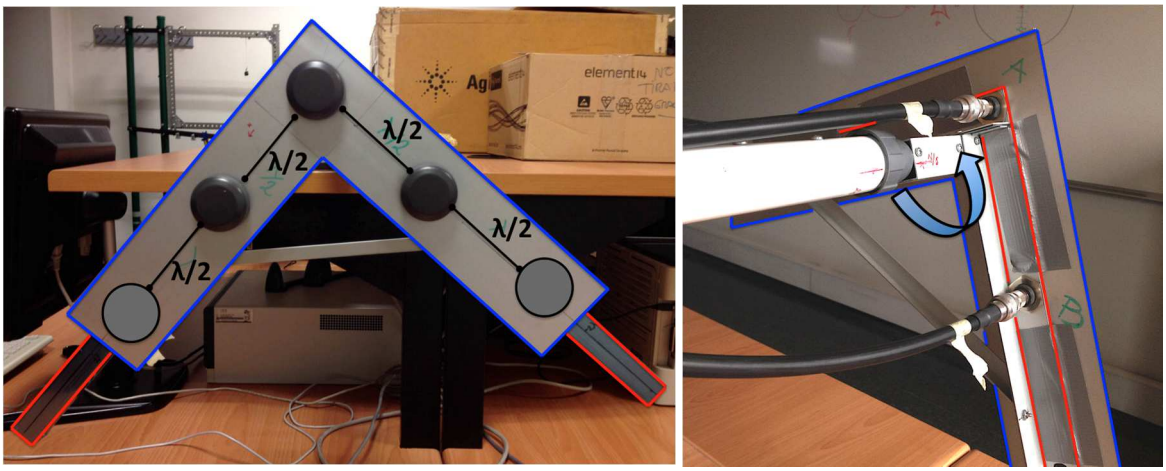

Figure 37 Antenna array solution

The blue surface in Figure 37 stands for an aluminum ground plane and was designed so that each antenna could have the minimum flat surface needed to work properly regardless the position on the array. In order to take advantage of all virtually ground plane benefits, the minimum ground plane area per antenna established by the manufacturer is to have 10 cm diameter of metallic flat surface without any hole or imperfection (PCTEL Inc, 2015). As may be seen in Figure 37, the ground plane designed for the antenna array (blue surface) was mounted over a fixed iron structure (red surface) that provides stability and guarantees the antenna array geometry.

As shown in Figure 37, the antenna array designed on this project allows a discrete set of positions for the antennas. The separation between these discrete positions is half wavelength in each of the two perpendicular directions. Notice that the mechanical condition of the array ground plane allows the user to change this maximum inter-element separation up to 2 wavelengths, but for the scope of this project the maximum antenna separation chosen was enough.

To finish, one of the most interesting points regarding the array geometry is its rotational capability. The antenna array was designed to have two swiveling baselines that allow the user to guide the array 360 degrees horizontally. This feature could become very useful in order to check the antenna array signal phase reception. A clear example of the usage of this feature will be shown on the next chapter.

3.4 ANTENNA ARRAY RECEPTION PARAMETERS ANALYSIS

After having verified that the existing random phase offset between the USRPS can be corrected by using an external calibration signal, in order to detect any possible resulting effect due to the usage of different antennas on the various front-ends, the following test was carried out:

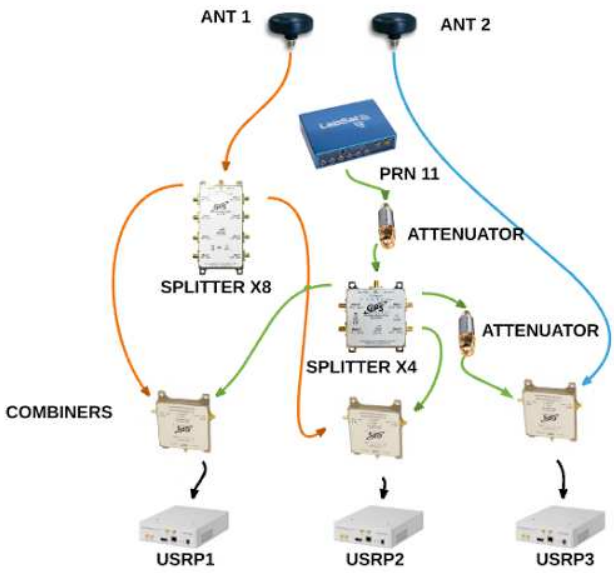

Figure 38 Multi-antenna analysis circuit

As seen in the diagram from the Figure 38, with the purppose of detect any possible influece of receiving GPS signals with different antennas, the data collection was carried out by using two different antennas of the same model. As commented before, the COTs antennas used on this project were designed to work in a stand-alone manner so any possible influence of a radiating element working simultaneously in a short distance was not described on the product datasheet. The first antenna signal was splitted and disttributed through an eight-output splitter to the combiners associated to USRP1 and USRP2 and the second one was directly connected to the combiner linked with USRP3. The phase calibration signal generated for this particular

tests corresponds to the PRN11 signal and was distributed to the combining stage through a four-output splitter.

It is important to note that the antennas used on this test were exactly the same model and were pretested individually to check any possible differences at least in terms of signal quality. From this test, no empirical evidence of possible differences on the antennas performance was concluded. Thus, this study commenced with the premise that ―at least working in a stand-alone manner― the antennas were identical elements.

First of all, in order to compare the quality of the signals received by different antennas in terms of power to noise density, the C/N0 of different signals was studied.

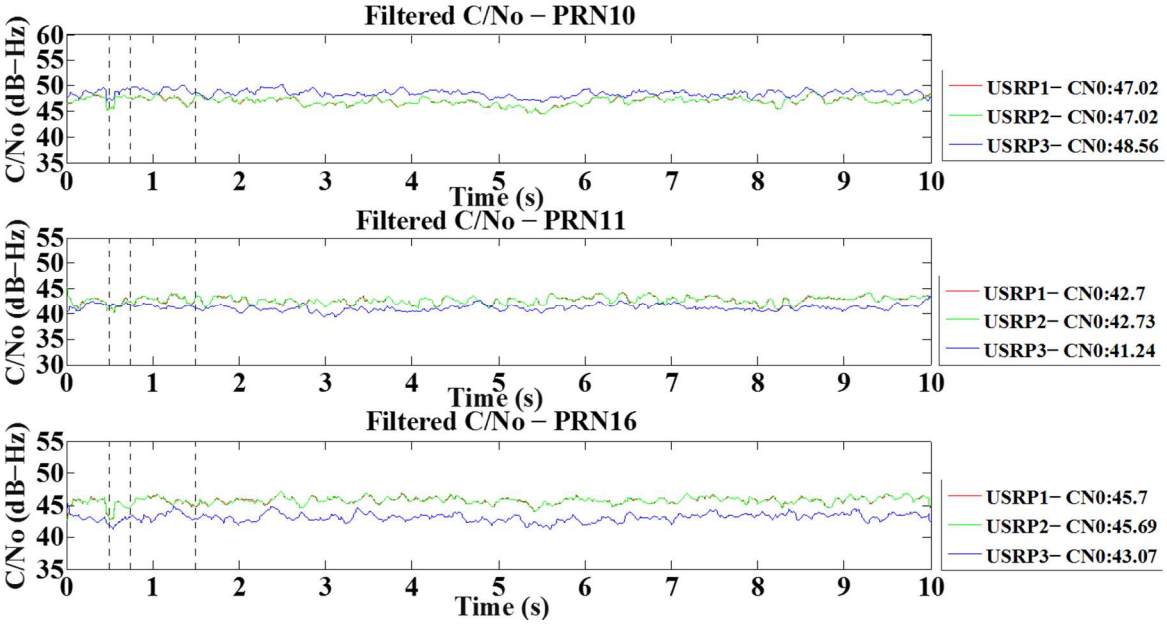

Figure 39 Multi-antenna carrier to noise ratio analysis

From the graphs in Figure 39, PRN10 and PRN 16 stand for real GPS signals and PRN11 is the signal PRN code related to the phase calibration signal. From the graph above, it is easy to determine that the C/N0 estimated for the three signals is equal for the two front-ends that share the same antenna. This results has revealed that the possible noise induced by the combining stage may be neglected. Therefore, it could be determined that the most striking differences among the C/N0 of the signals received by the different USRPs are due to the location of the different antennas. The most logical explanation of this C/N0 variances in the signals recorded by the USRPs that did not share the same antenna could be caused by the multipath. In the absence of multipath, the C/No measured in all antennas should be very similar to one another.

But in the presence of multipath, an antenna where multipath happens to be combined constructively will present a higher C/No than an antenna where multipath is combined destructively. Figure 39 shows that depending on the antenna location, the multipath may affect constructively or destructively to the real GPS real signals, provoking as a result variations on the C/N0 estimated between ±2dB-Hz. Ideally, since the phase calibration signal was distributed to all the front-ends using the same hardware components, the power estimated for PRN 11 signal received by the three USRPs should be exactly the same. In practice, as may be seen in the graph above, there exist variations in the C/N0 estimated for PRN11 and may be caused by the different noise applied on the combining stage. Since the radiating elements used on this project were active antennas, the noise contribution of the signals received and amplified by them could impact on a different way in the combiners, provoking a slight attenuation on the calibration signal.

Second of all, with the purpose of study the impact of using different antennas on the signal delay estimated by the receiver, the differences in the signals code phase were evaluated. Graphs in Figure 40 show the code phase differences calculated based on the results obtained from the recorded signal tracking, taking into account that the estimated parameters of the signals recorded by the USRP1 were used as reference values.

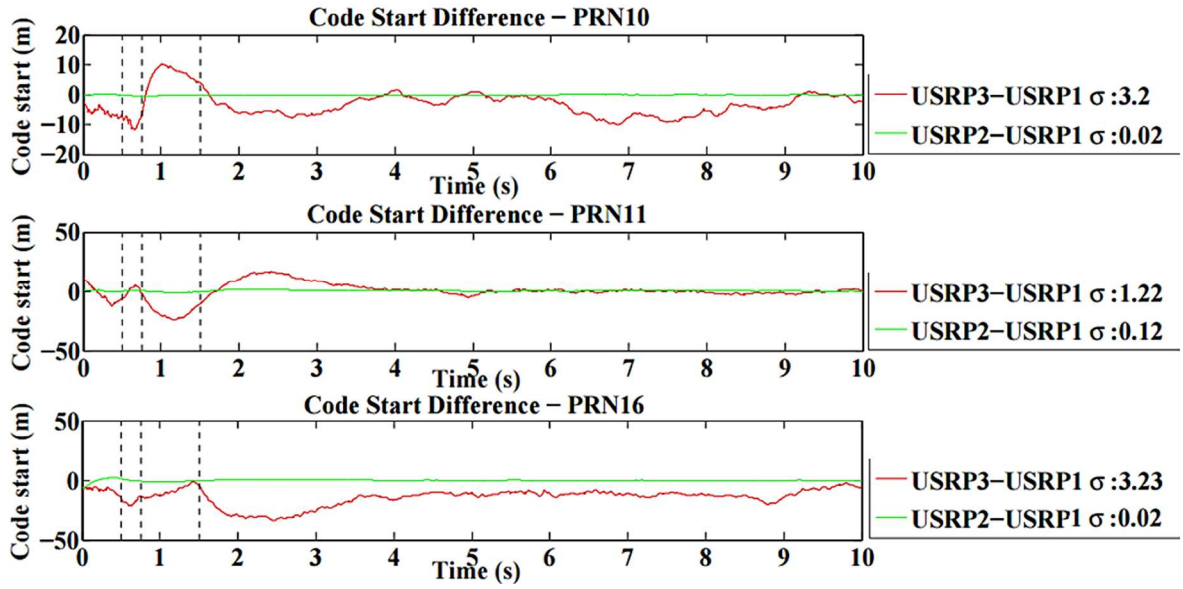

Figure 40 Multi-antenna code phase differences

On the one hand, analyzing the low standard deviation and the near to zero mean of the code phase differences observed between the signals received by the front-ends that shared the same antenna— USRP2 and USRP1—, it could be determined that any possible noise added by the

antenna amplifier in the signals was cancelled by applying the difference. Theoretically, if both front-ends were working under the same conditions, the standard deviation observed between the code phases of the signals should be zero. In practice the hardware does not work under the same conditions and the different thermal noise on the first stage amplifier of the front-ends may cause noise mismatch between the different front-ends and hence discrepancies on the code phase estimations. On the other hand, in light of the results obtained from the code phase differences between USRP3 and USRP1 —each front-end linked with a different antenna —, it could be determined that the noise mismatch between the antennas may cause discrepancies on the delay locked loop (DLL) and therefore differences on the code phase estimations. As shown in Figure 40, these noise mismatch —added to other factors— may provoke fluctuations on the code phase differences of the order of meters, e.g. as may be seen in the code phase difference results obtained from the PRN11, the noise mismatch between the different antennas may increase the standard deviation of the code phase differences form 12 cm to 1,22 meters. The noise induced by each antenna is added to the phase calibration signal in the combination stage and provokes this undesirable effect.

Regarding the code phase differences of the real GPS signals, the fluctuations observed may change from 2 cm to 3.23 meters. This huge gap in the code phase differences is obviously a contribution of different factors. Since the real GPS signal was received by two different antennas separated a distance *d*, the difference of location of the antennas contribute to the code phase differences. Furthermore, as mentioned Kim (2005), the mutual coupling between two antennas working in a short distance may vary the frequency response of the antennas. Exact antenna element can have rather different frequency response, depending on the adjacent element which dictate the mutual coupling environment of the antenna element. These different frequency responses lead to differing effects on the code phase of the received GPS signals.

Finally, with the purpose testing the performance of the phase correction method specified in the previous chapter, the phase differences between the distinct signals recorded by the front-ends were computed. As may be seen in Figure 41, after applying the phase correction method, the phase differences obtained from the signals recorded with the USRPs that shared the same antenna —USRP2 and USRP1— were the expected ones. In the PRN 11 case, the phase differences tend to zero, as supposed to be. Otherwise, in the real signals case, the obtained phase difference tend to 0.13 radians. As explained in the previous chapter, these differences

were not null because different models of splitters were used to distribute the signals coming from the antenna and from the phase calibration signal.

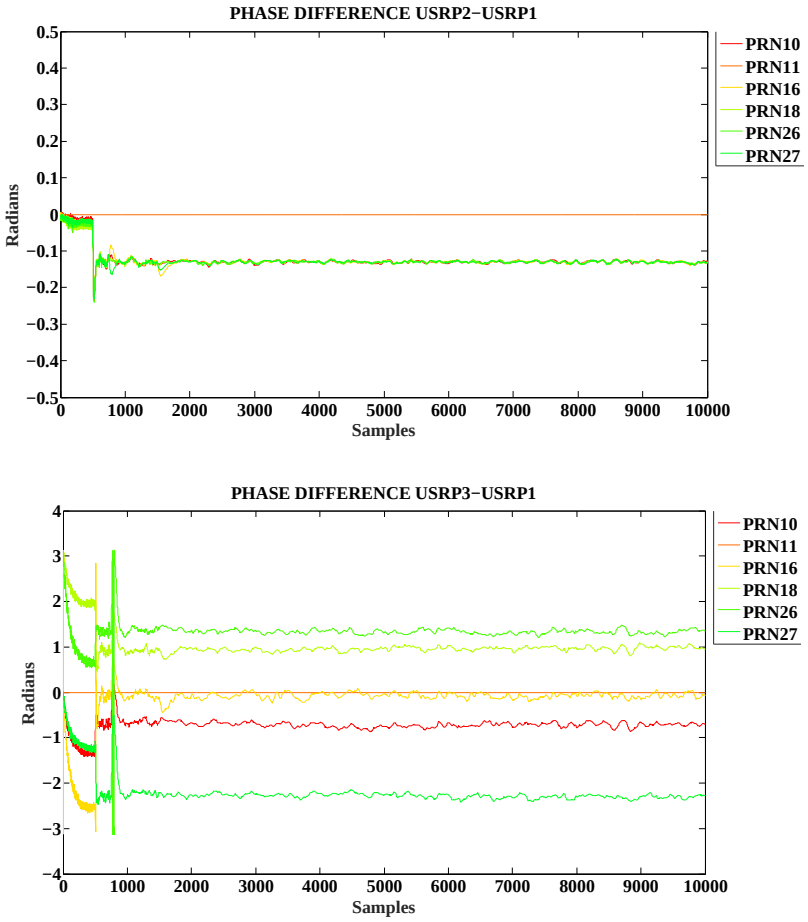

Figure 41 Multi-antenna phase differences

Talking about the phase difference resulting from the signals registered by USRP3 and USRP1 —each one linked with a different antenna—, after applying the phase correction method on the signal registered by USRP3, the observed differences in Figure 41 are due to the antennas location. The following chapter will analyze in detail the possibility of using these phase differences to calculate the direction of arrival of real GPS signals.

In brief and in one hand, after performing this test using different antennas in order to receive satellite signals, the influence of these antennas over the parameters estimated by the receiver has been observed. On the other hand, the different location of the antennas may affect the C/N0 signal received constructively or destructively depending on the incidental wave angle and depending on the multipath effect. Moreover, the noise mismatch, the mutual coupling effect and the different location of the antennas may cause alterations on the code phase estimated by the receiver —alterations may vary from centimeters to meters.

To finish, through this test it has been able to prove that the purposed phase correction method works properly once again and that, after correcting the existent USRP's random phase offset, the phase differences of the satellite signals were just caused by the reception of GPS signals from a distinct location (that is, the already studied equations No.50 & 51[7]).

[7] See equations (50) and (51) on page 47.

CHAPTER 4

DOA ESTIMATION

The following chapter is dedicated to analyze possible applications of the antenna array data recorded. The antenna array used on this project provides measurements from three different antennas placed on top of a mast with a fixed and known separation between them. These extra measurements could be used for different purposes but, as the project was focused on determining DOA estimation, the objective of this section was to analyze these extra measurements and the possibility of using them to perform DOA detection.

4.1 SCENARIO

During this project, real GPS signal data was recorded simultaneously with three synchronized USRPs each receiving from different antennas (Ant1, Ant2 and Ant3). As it may be seen in Figure 36, the three antennas that conform the array were fixed on ground plane in a body reference system and conformed two orthogonal baselines, the first one formed by Ant2-Ant1 and the second one by Ant3-Ant1.

As shown in Figure 42, in order to analyze the multi-antenna recorded data, four data recording tests were performed placing the antenna array on the roof of the UAB Engineering School, an area with a complete visibility and a clear surrounding without any type of obstructions that may block the direct signal emitted by the GPS satellite constellation. The lack of these obstructions ensured the reception of clearly signals without any significant attenuation.

Figure 42 Antenna array location

Each data collection test was performed with ten minutes of difference with respect to the first recorded sample. The reason why 4 tests were carried out was basically to analyze the phase differences calculated for each satellite signal for a specific lapse of time and thus, evaluating both the front-ends and antennas performance regarding phase stability. Table 10 shows the basic parameters that describe both the localization of the data collection and the date and time it was collected.

Time Zone:	**GTM+1**
Data collection date:	17/02/2016
Location:	41°30'02.2"N, 2°06'46.2"E
Antenna spacing:	$\lambda/2$
Test1 Data Collection Time:	16:00 H
Test2 Data Collection Time:	16:10 H
Test3 Data Collection Time:	16:20 H
Test4 Data Collection Time:	16:30 H

Table 10 Data recorded information

In order to avoid ambiguities when calculating the direction of arrival of the incident signals, it is important to take into account that a half-wave length between elements separation when recording the signals was used, i.e.: 9.5 cm approx. Moreover, as explained in the chapter

above, the fact that the mechanical conditions of the antenna array designed for this project allow the user to orientate the baselines for convenience, simplifies the DOA estimation. Figure 43, shows a clear example of this feature —the baseline formed by the antennas connected to USRP1 and USRP2 was orientated to North-South axis.

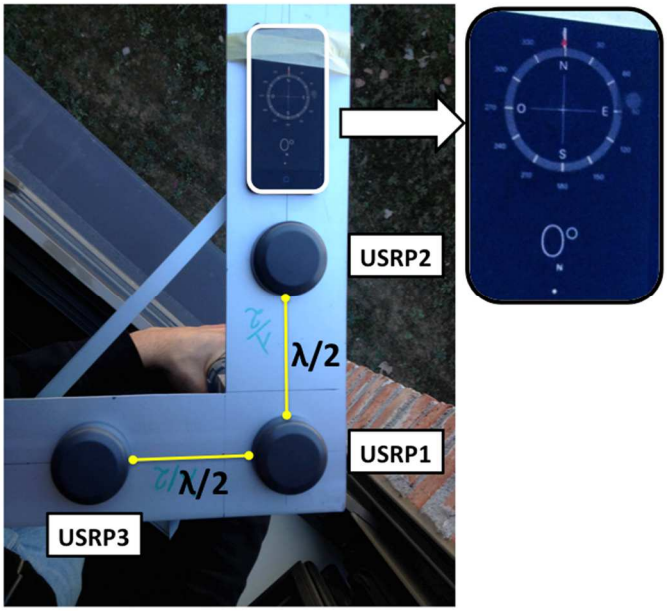

Figure 43 Antenna array direction

After analyzing both the localization where the data recording tests were taken and the separation between the different antennas conforming the array, a theoretical calculation of the phase differences received by the antennas was carried out.

4.2 THEORETICAL CALCULATION OF THE PHASE DIFFERENCES

Before calculating the phase differences of the GPS signals recorded by the antenna array, a theoretical analyze of the expected results was carried out. For this purpose, the calculation of the direction of arrival was performed inversely. That is, knowing (a priori) the satellites position in terms of elevation, azimuth and the location and orientation of the antenna array, the phase differences that had to be theoretically obtained were calculated.

In order to determine the GPS satellite constellation location at the time of the data recording, the online Trimble GNSS Planning Tool was used. This online tool is used to program GNSS observations. It also allows users to predict the satellite position in relation to their situation for a concrete time lapse.

In order to compute GNSS satellite position, Trimble GNSS planning tool uses on input the latest almanac data files provided by the U.S. Coast Guard's Navigation Information Service (NIS) so that the satellite localization computed by the tool is highly reliable data.

The satellite position during the data recording is particularly clear in Figure 44. The ascending scale of blue markers represents the temporal evolution of the satellites location while the red ones show the orientation of the antennas. As may be seen, the baseline formed by the antenna associated with USRP 2 and USRP1 was orientated following the North-South axis while the one formed by the antenna associated with USRP3 and USRP1 was orientated following West-East axis.

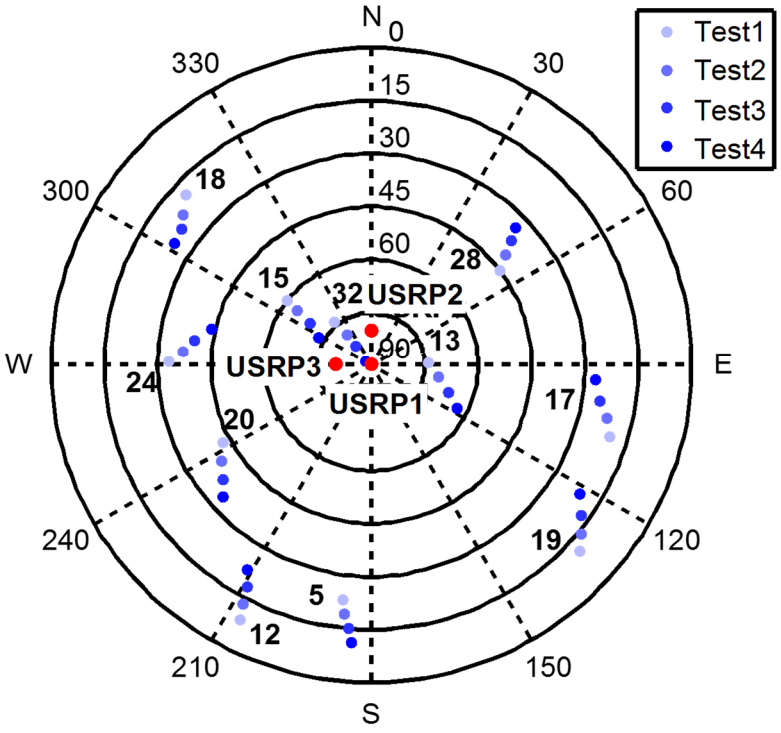

Figure 44 Satellite location

As may be seen in Figure 44, the antenna array was oriented in a way that, at least during the first test, the baseline formed by the antennas associated with USRP3-USRP1 frontends was aligned with PRN 24 and PRN 13 satellites. This allowed the baseline formed by USRP2-USRP1 to be totally perpendicular to the signals coming from the mentioned satellites, thus achieving a perfect scenario to analyze the behaviour of the antennas that make up the array.

In order to calculate the theoretical phase differences of the GPS signals, the following expressions were used:

$$\varphi_{31} = -\sin(\phi) * \cos(\theta) * \frac{2\pi d}{\lambda} \tag{59}$$

$$\varphi_{21} = \cos(\phi) * \cos(\theta) * \frac{2\pi d}{\lambda} \tag{60}$$

where d is the antenna separation and ϕ and θ stand for the satellite azimuth and elevation, respectivelly. As explained before, the separation among antennas used all along the data recording was half-wave length, i.e.: 9.5cm approximately, and both the elevation of the satellites and their azimuth were values obtained through the online platform Trimble GNSS Planning Tool. Figure 45 graphically shows the calculation of the theoretical phase differences obtained from each of the performed tests.

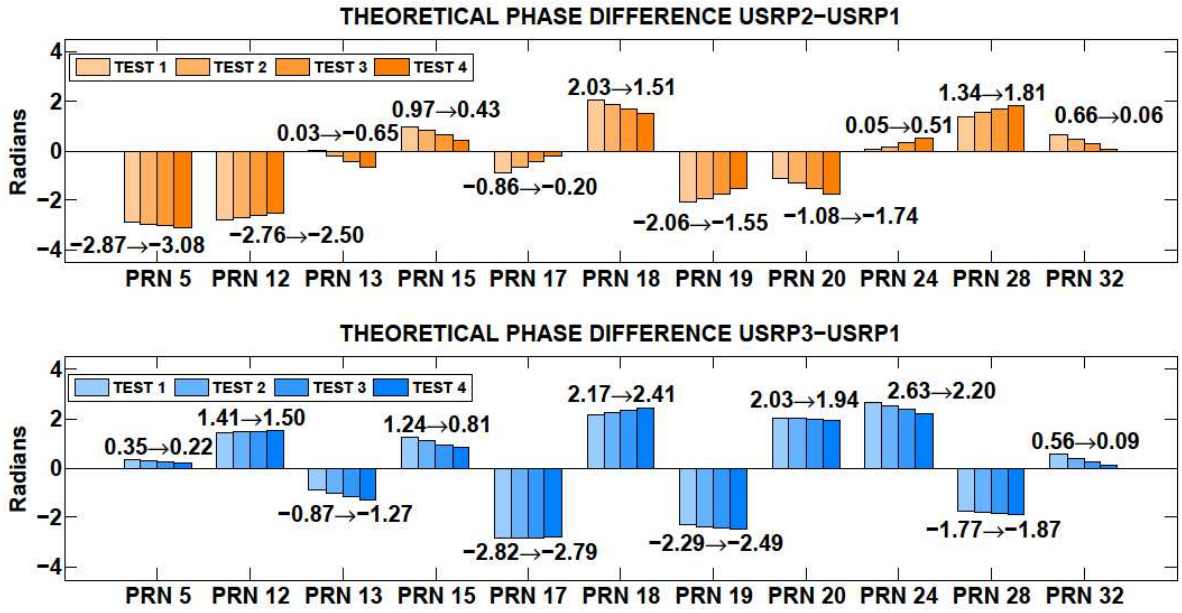

Figure 45 Theoretical phase differences

The most outstanding cases to be commented are the temporary evolution of the phase differences between USRP2 and USRP1 of the signals of satellites PRN 13 and PRN 24. As shown in Figure 45, for the first test performed, both satellites were situated in such a manner that the signals coming from them perpendicularly impinged the baseline formed by USRP2 and USRP1 front-ends. As time went by, it was possible to observe that this perpendicularity was lost because of the satellites' movement. Carefully observing the data obtained in Figure 45, the influence of the satellite position on the obtained phase differences can be seen. When the impinging signal was perpendicular to the baseline formed by the antennas associated to

USRP2-USRP1 front-ends, an almost null phase difference was obtained but this difference increases while satellite position changes.

It is also important to note the expected phase difference of the signal emitted by satellite PRN 32. As seen in Figure 45, the path of the satellite went over the bisector of the straight-angle that forms the antennas array, placing itself almost over the zenith of the antenna array in the last test. As expected, the values of the obtained phase differences both for the baseline formed by USRP2-USRP1 and for USRP3-USRP1 were basically the same, tending to zero when the satellite had an elevation of, approximately, 90 degrees.

4.3 ANALYSIS OF C/N0 OF THE DATA ACQUIRED

In order to check the quality of the signal registered by the three USRPs, the carrier to noise density of the signals recorded was evaluated. This C/N0 analysis was focused on the study of two main concepts: the first one was based on the analysis of the C/N0 evolution along a concrete period of time and the second one was based on the comparison of the signal's strength acquired by each SDR antenna.

On the one hand, the first study gave an overview of how much the satellite signal strength may have changed over time. Figure 46 shows the evolution of the carrier to noise density of the signals acquired by each USRP along four tests, which —as shown above in Table 10— were performed with 10 minutes of separation between each data collection.

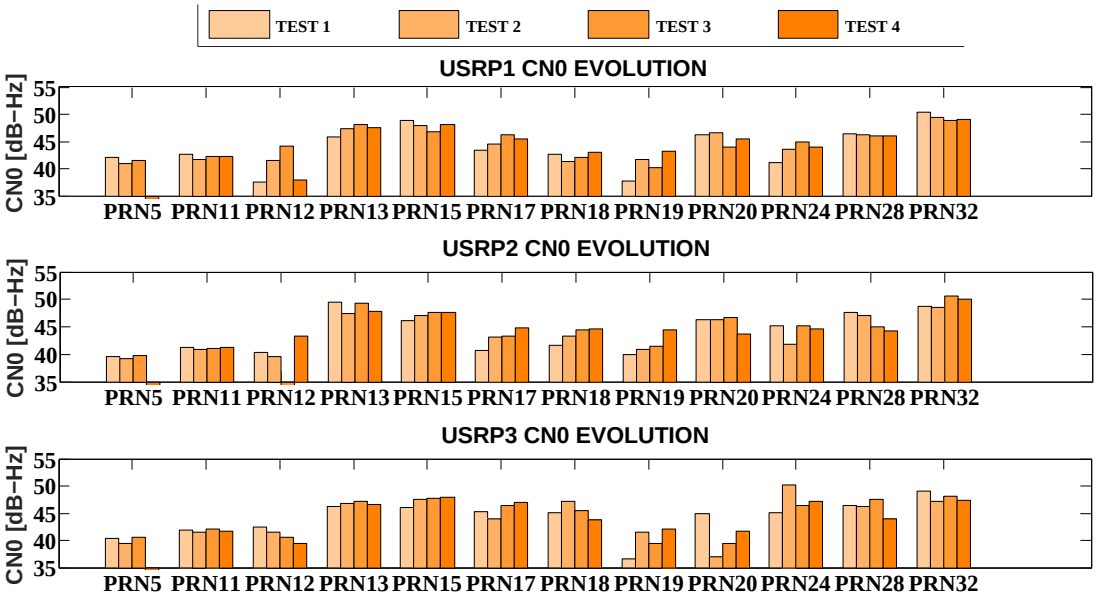

Figure 46 C/N0 evolution

From the graph above, it may be determined that the C/N0 of the signals broadcasted by satellites with high elevation did not have a substantial change along the data collection. Looking at the satellite location (Figure 44), it could be said that satellites whose elevation did not lower more than 60 degrees did not suffer substantial changes along time. As shown in Figure 46, the carrier to noise density of the signals emitted by the satellites PRN 32, PRN 15 and PRN 13 was practically stable along time and the little differences observed could have been due to standard deviation of the C/N0 estimation itself.

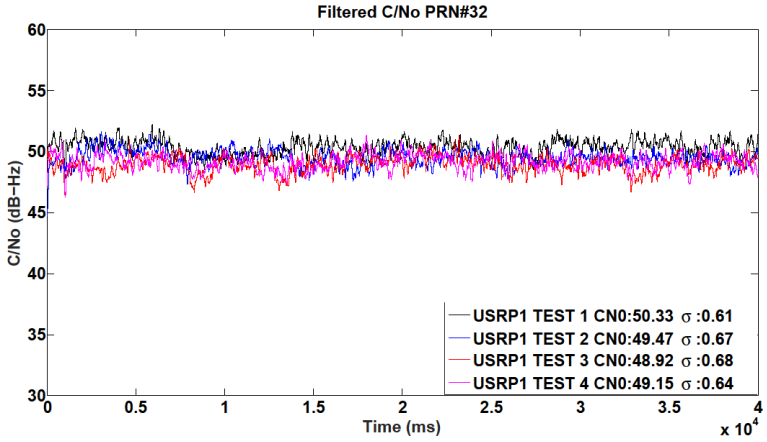

Figure 47 USRP1 - PRN 32 C/N0 Evolution

For instance, Figure 47 shows the carrier to noise density time evolution of the signal broadcasted by the satellite 32. As may be seen, the standard deviation of the PRN 32 signal may have varied from 0.61dB-Hz to 0.68 dB-Hz and may have been the cause of the little differences observed in the C/N0 estimated in Figure 46.

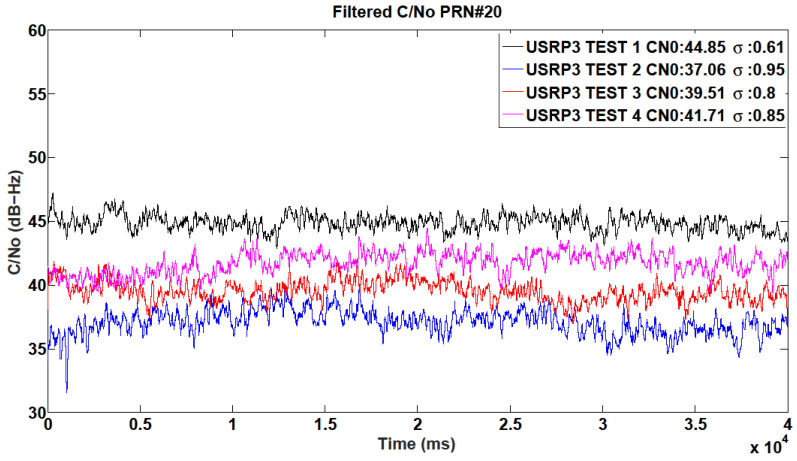

Figure 48 USRP3 - PRN20 C/N0 Evolution

On the other hand and unlike the previous case, the signals emitted by satellites with a lower elevation may have suffered differences on the carrier to noise density estimated along a period of time. A clear example of this phenomenon may be seen in the C/N0 evolution of the satellite signals PRN 12, PRN 19 and PRN 20 (Figure 46). Figure 48 shows, as an example, the

evolution of the C/N0 estimated for the satellite signal PRN 20 registered by the USRP3. From the graph, it is noticeable that the carrier to noise density varied from 37 dB-Hz to 44.85 dB-Hz. This notorious variances may be caused by the changing multipath contribution on the antenna. As shown in Figure 46, the antenna where satellite signal multipath happened to be combined constructively presents a higher C/No than the one where multipath was combined destructively.

To finish, in order to compare —at least qualitatively— the satellite signal strength registered by each antenna of the array, Figure 49 shows the carrier to noise density evolution of the different satellite signal strength registered by each USRP.

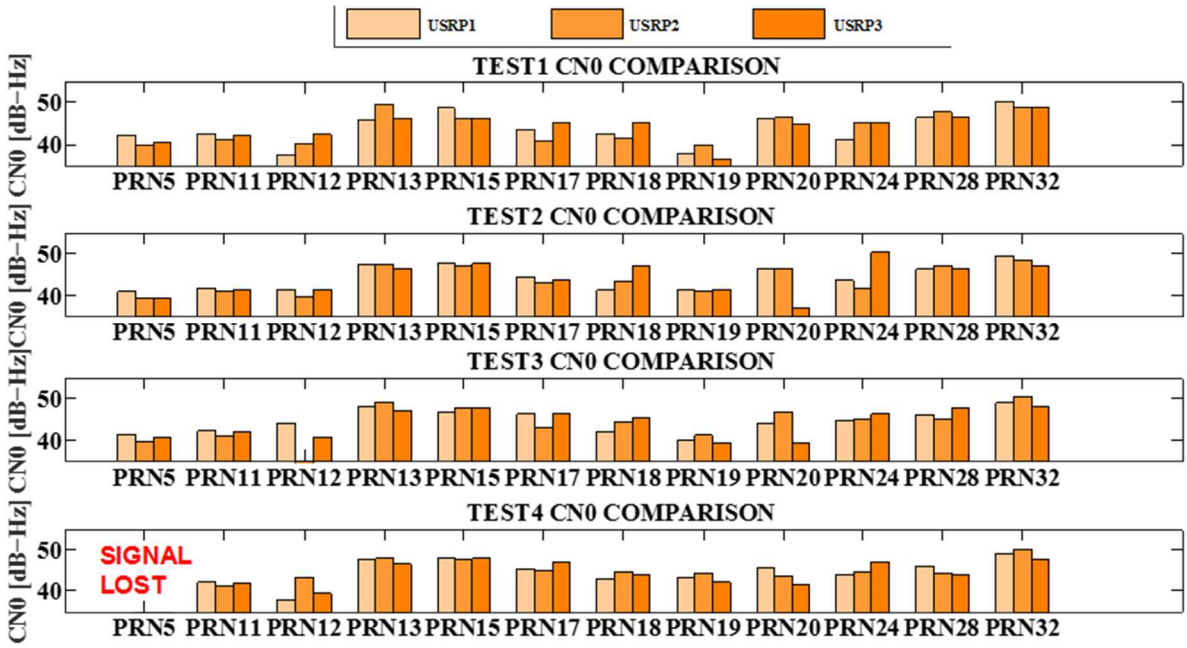

Figure 49 C/N0 antenna comparison

As shown in Figure 49, comparing the carrier to noise density received by each antenna that compose the array, it could be determined that the signal quality of the satellites with high elevation —PRN 13, PRN 15 and PRN 32— did not have significant changes along the tests. This results are a clear evidence that those signals received by each antenna were in a direct light of sight and thus, in the light of the lack of multipath effect, the C/N0 estimated on the signals received by each USRP remained practically equal along time. On the contrary, satellites with lower elevations suffered important changes on the carrier to noise density estimated over time. Figure 50 shows a good example of these C/N0 variations, in this particular case, the quality of PRN 20 and PRN 24 was evaluated.

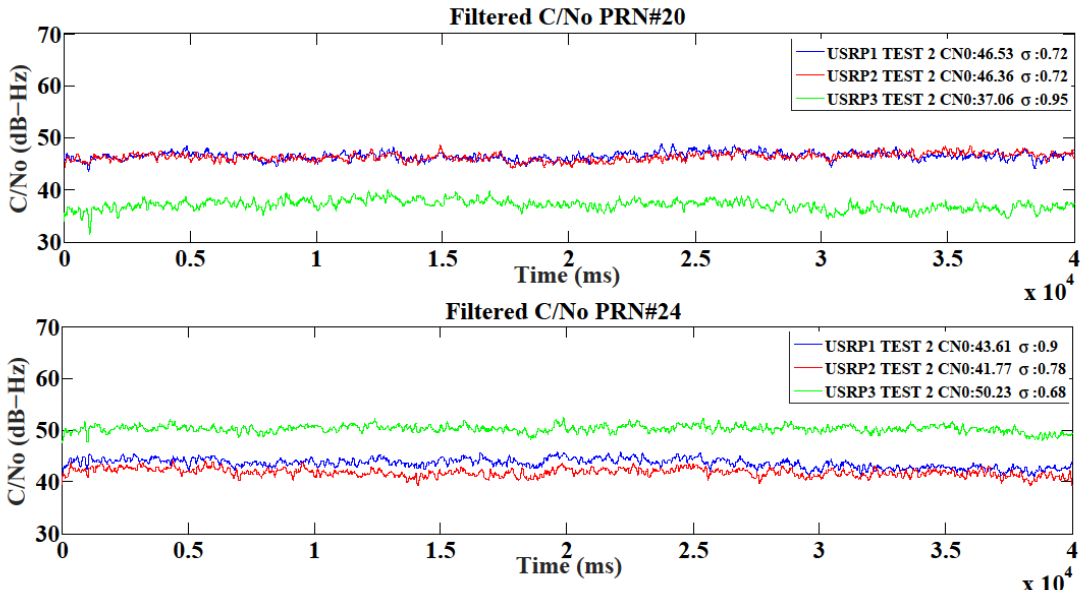

Figure 50 PRN 20 and PRN 24 C/N0 comparison

As shown in Figure 50, depending on the direction of arrival of the signals, the multipath contributed constructively or destructively on the C/N0 estimation. For instance, in the case of receiving the PRN 20 signal, the multipath affected the third antenna destructively causing a signal decrease of approximately of 9 dB-Hz with respect to the signals received by the two other antennas. In contrast, PRN 24 signal received by the third antenna presented an increase of approximately 9 dB-Hz in the C/N0 estimated, with respect to the signal strength registered by the two other antennas. This unpredictable changes on the evolution of the estimated C/N0 show that the multipath may cause a positive or negative impact on the carrier to noise density estimation and it may vary along time depending on the signal direction of arrival.

4.4 ANALYSIS OF SIMPLE DIFFERENCES CARRIER-PHASE

Having checked the influence of the GPS satellite location on the carrier to noise density estimation, the phase differences of the signals recorded by the antenna array were analyzed. This study aims to identify the behavior of the phase differences observed in order to use this information to determine the direction of arrival of GPS satellite signals. The following graphs show the phase differences of the most significant satellite signals observed on the first performed test. Specifically —as commented on the theoretical analysis—, the satellite signals that could provide the most relevant information regarding the expected phase differences were PRN 13, PRN 24 and PRN 32. The terms $\Delta\phi$ and $\Delta\phi_T$ on the graphs stand for the observed phase difference and the expected theoretical phase difference, respectively.

First of all and before starting the analysis, it should be emphasized that the differences observed on the satellites signal phases were product of the different geographical location of the antennas, since the USRPs were previously synchronized in time and phase by using an external oscillator and a phase calibration signal.

Figure 51 shows the phase differences observed on the satellite signal PRN 13 registered by the three distinct antennas that conform the antenna array, taking the antenna linked to USRP1 as a reference. Taking into account that the satellite location (Figure 44) was practically perpendicular to the baseline formed by USRP2 and USRP1, the expected phase difference on the signal acquired by the antennas linked to both SDRs was approximately 0.03 radians. In contrast, the difference on the phase signal registered by the baseline formed by USRP3 and USRP1 was expected to be approximately -0.87 radians. As may be seen in the graph, both values obtained after processing the data collected on the first test did not match the expected ones.

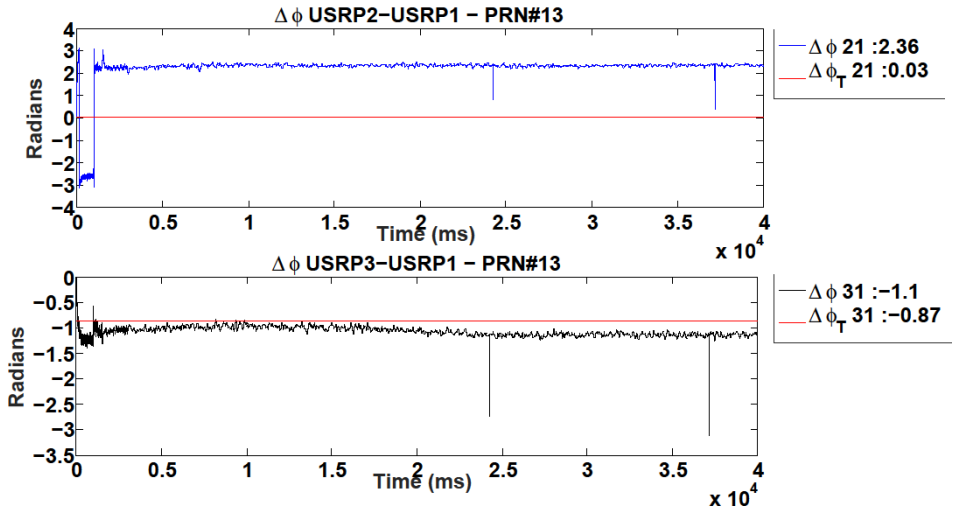

Figure 51 PRN 13 - Phase difference

The difference between the expected values and the obtained results differs depending on the antenna baseline, from 2.33 radians in the case of USRP2-USRP1 to -0.23 radians in the case of USRP3-USRP1. In order to determine if the differences between the expected values and the obtained ones could be caused by hardware biases —i.e. possible mismatch in the antenna cable lengths or difference on the connectors used to adapt the signal to the different RF ports—, the phase differences of two more signals were analyzed. If the differences between the expected values and the computed ones were product of hardware biases, the observed phase differences between the expected values and the obtained ones should be constant

regardless the signal direction of arrival. Figure 52 shows the phase differences computed from the signal broadcasted by the satellite 24.

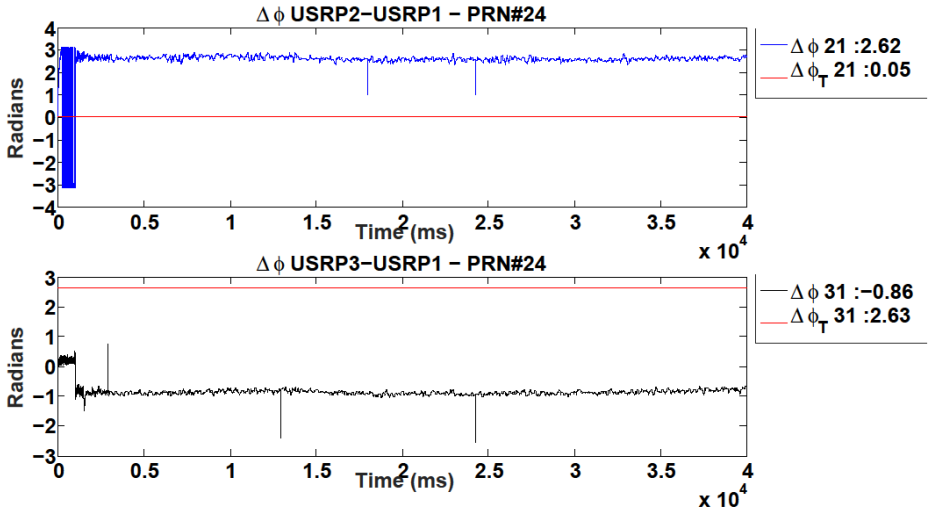

Figure 52 PRN 24 - Phase difference

Once again, the satellite location (Figure 44) was practically perpendicular to the baseline formed by USRP2 and USRP1; the expected phase differences on the signal acquired by the antennas linked to both SDRs was approximately 0.05 radians. In contrast, the difference on the phase signal registered by the baseline formed by USRP3 and USRP1 was expected to be approximately 2.63 radians. Once more, both values obtained after processing the collected data did not match the expected ones. In this case, the difference between the expected and the computed values was approximately 2.57 radians in the case of the phase difference between USRP2-USRP1 and -3.49 radians in the case of USRP3-USRP1. To finish, the phase differences of PRN 32 signal were analyzed.

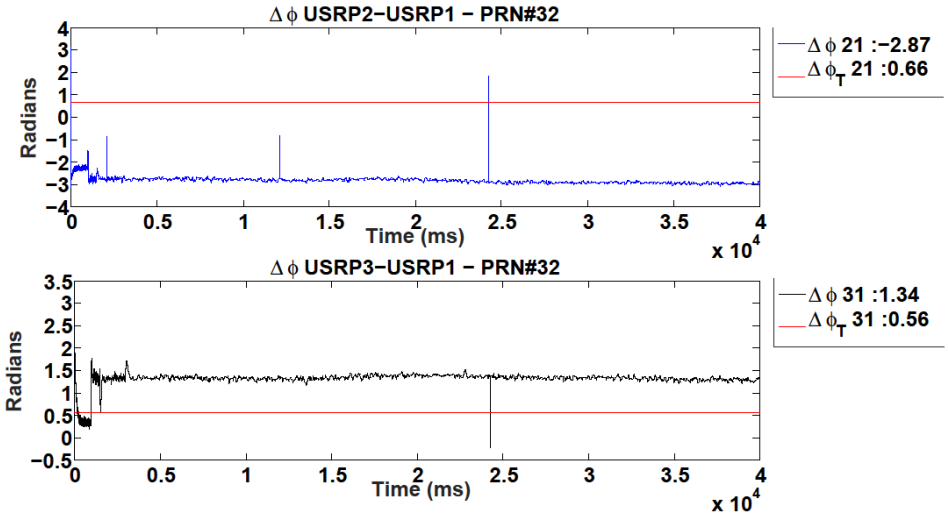

Figure 53 PRN 32 - Phase difference

Taking into consideration that the satellite location was practically over the bisector of the straight-angle that forms the antenna array, the expected phase differences on both baselines were practically the same: 0.65 radians in the case of comparing the phase of USRP2 and USRP1 and 0.56 radians in the baseline formed by URP3-USRP1. The error between the theoretical phase differences and the obtained results went from -3.54 radians in the case of USRP2 and USRP1 to 0.78 radians in the case of USRP3-USRP1.

In light of this results, it could be determined that the obtained phase mismatch was not the result of possible hardware biases, since the obtained errors were not constant regardless the signal analyzed. A summary of the different errors obtained for each received signal is shown in the chart below.

PRN	USRP2-USRP1 [rad]	USRP3-USRP1 [rad]
5	0,67583948	0,24201377
12	1,37820898	1,16606937
13	2,32686805	-0,2336851
15	-3,6348823	1,28482146
17	2,21340426	4,80661507
18	-2,6348039	-3,9062761
19	0,90814954	5,3122597
20	1,51508073	-4,407598
24	2,5715007	-3,4967054
28	-3,3935895	4,75843471
32	-3,5377424	0,78180409

Table 11 Phase differences errors

As may be seen from Table 11, the obtained errors are not constant and they directly depend on the direction of arrival of the incident signal. These obtained errors may have been caused by mutual coupling effect introduced by other antenna elements that conform the array (Danesh, Sokhandan, Zaeri-Amirian, & Lachapelle, 2014). The impact of the actual physical response of the used antennas —which were neglected by assuming isotropic receiving elements— may also have impacted on the phase response of the antenna elements. Each of the individual antenna elements in the array had a different phase center movement depending on the azimuth and the elevation of the incident signal. This phase response pattern of each individual antenna may have differed due to the configuration of the adjacent antenna elements and may have depended on the mutual coupling environment (Kim, De Lorenzo, & Gautier,

2004). In order to correct this phase errors caused by nothing but the interaction of the antenna elements, the antenna array should be phase calibrated in an anechoic RF chamber so that the antenna array will receive a signal from a precisely known direction. Since antenna phase calibration was not included on the scope of this project, this antenna phase adjustment will be lead to future researchers who might be interested in improving the multiplatform designed on this project.

Figure 54 shows a clear example of the evolution of the phase differences of the acquired signals depending on the hardware calibration level. Red bars indicate the phase differences observed before performing the phase calibration, the orange ones stand for the phase differences obtained after applying the calibration process using an external phase calibration signal and yellow bars depict the theoretical phase differences.

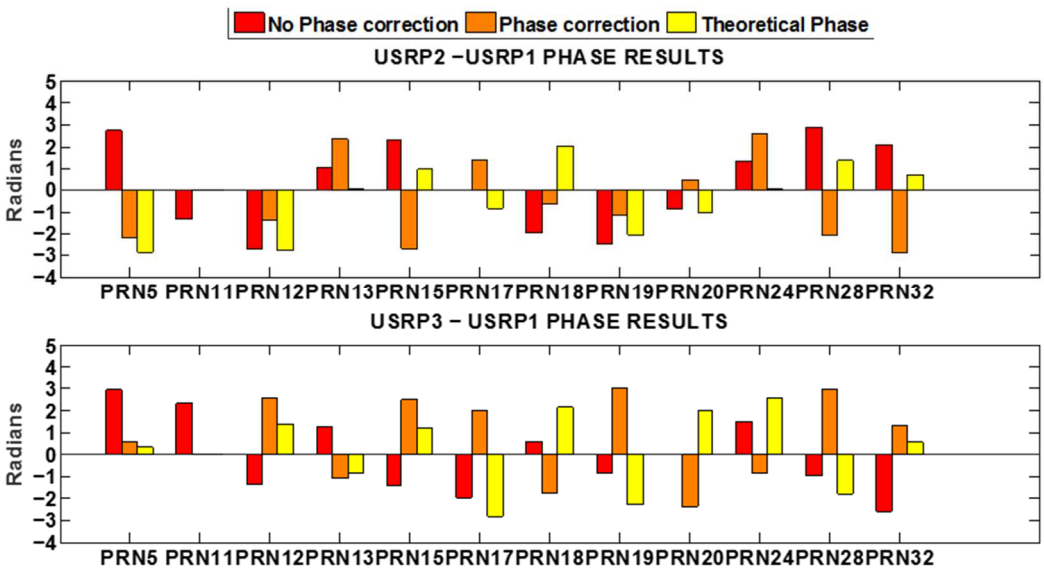

Figure 54 Test 1 - Signal phase correction process

The phase differences performed directly on the signals recorded by the USRPs (red bars) include the front-ends random phase offsets. As explained before, this random phase offset changes every time the front-ends are retuned and this is the reason why an external phase calibration signal was needed. After performing the phase calibration process, the random phase offsets of the SDRs were compensated and the phase misalignment between all USRPs was caused due to the geographical location of the antennas that conform the array (orange bars). Since the phase response of the antennas was not the expected due to phase center variations which were provoked by the mutual coupling effect, the phase differences had to be recalibrated in order to balance the antennas phase errors (yellow bars). After having checked

that there existed errors on the computed phase differences which could have been result of the physical antenna interaction, it was trivial to determine that it was not possible to compute the direction of arrival of the incident signals basing on the obtained results. In order to compensate the phase errors of the antennas, the errors obtained on the first test —Table 11— were used to calibrate the antennas phase errors obtained in the signals recorded during the four tests. The results obtained from this procedure were used to compute the DOA of the GPS signals. Evidently, since the antennas phase errors may vary differently depending on the signal angle of incidence, the computed phase differences were not expected to be accurate values but they would be enough to explore the evolution of the phase differences obtained over time. Figure 55 shows the phase difference evolution of the baselines after performing the antennas error compensation.

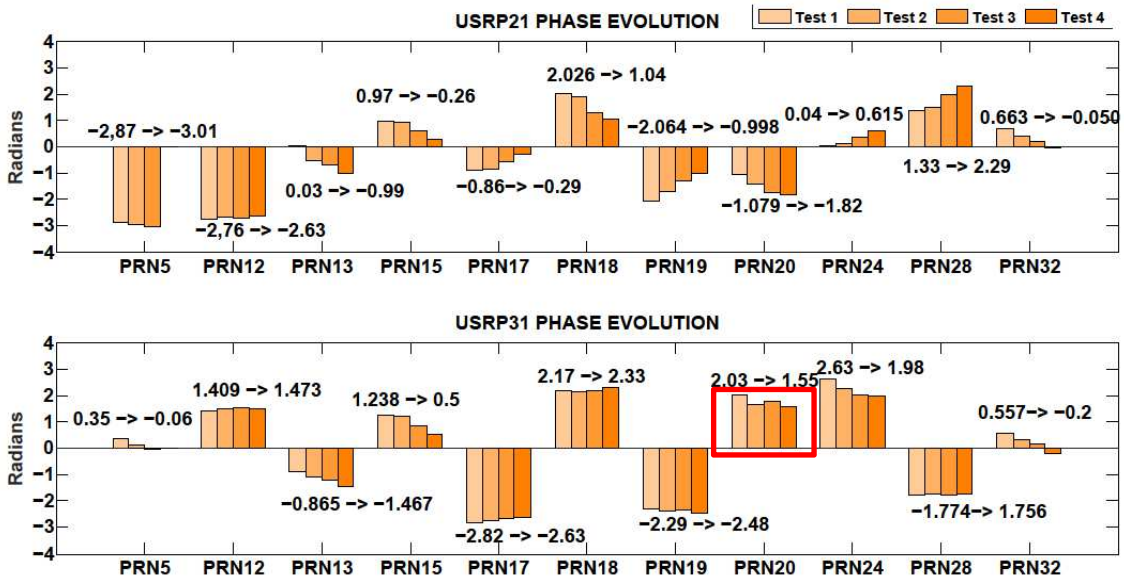

Figure 55 Antenna array phase differences evolution

Comparing the theoretical phase differences in Figure 45 and the real values obtained in Figure 55, it is noticeable that the phase difference evolution of both scenarios matches the phase differences —at least in terms of tendency. The only satellite signal whose phase differences did not follow a linear tendency was PRN 20 when computing the phase differences from the baseline formed by USRP3-USRP1. The answer to this behavior may be seen in the carrier to noise density estimated for this satellite (Figure 44). Since the carrier phase estimation directly depends on the shape of the correlation peak and thus on the signal carrier to noise density, the big fluctuations on the C/N0 computed by USRP3 may have caused an unreliable estimation on the signal carrier phase.

In light of the obtained results after performing the antenna phase error compensation, the direction of arrival of real GPS signals was performed following the expressions derived from (59) and (60):

$$\phi = \arctan\left(\frac{-\varphi_{31}}{\varphi_{21}}\right) \tag{61}$$

$$\theta = \sqrt{\varphi_{21}^2 + \varphi_{31}^2} * \frac{\lambda}{2\pi d} \tag{62}$$

Figure 56 shows the obtained results after applying (61) and (62), the rising range of colors represents the temporal evolution of the DOA estimation and each data point corresponds to the estimated direction of arrival for an individual data test. Comparing the obtained results to the real GPS location (Figure 44), it is noticeable that the estimated azimuth and elevation were imprecise values. As explained before, the antennas phase response may have varied depending on the incidence signal angle therefore, since the satellites were in a constant movement, the error estimated in the first test could change over time. This variances on the phase error of the antennas provoked a lack of accuracy on the estimated azimuth and elevation.

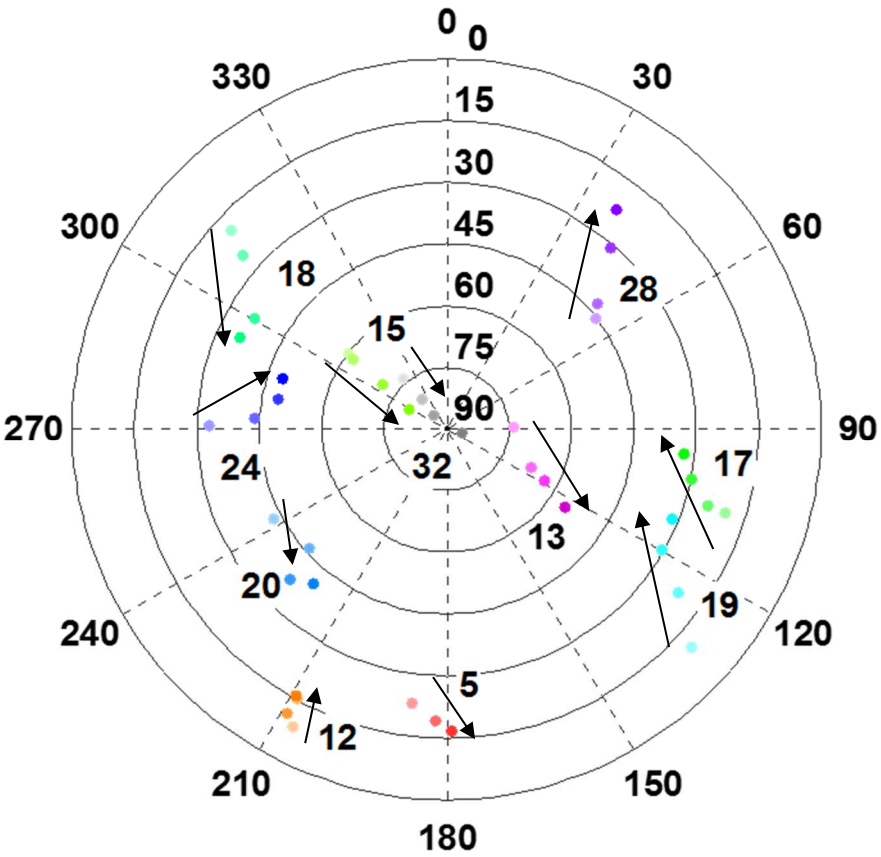

Figure 56 Direction of arrival estimation.

Although the computed azimuth and elevation were not reliable values in terms of accuracy, they were enough to predict, at least, the satellites motion. As may be seen in Figure 56, all the satellites' orbits were properly predicted unless for the case of PRN 20 signal that, as seen before, suffered incidents on the estimated phase differences with the baseline formed by USRP3 and USRP1.

While carrying out this study, it has been possible to prove that there exist fluctuations on the estimated carrier to noise density and these may vary depending on the satellite location. That is, for those satellites with a high elevation, no remarkable changes were observed; contrary, for those with a lower elevation, significant changes on the estimated C/N0 were appreciated. Moreover, it has also been possible to prove that important differences on the quality of the signal received by each antenna of the array may exist and this fact may have been caused by either constructive or destructive contribution of the multipath effect.

Referring to the phase differences obtained after the phase calibration process, it has been possible to check that the mentioned process was not enough to get a phase calibration of the whole multiplatform. After finding out that there existed some differences on the antennas phase response due to the mutual coupling effect, it was possible to stablish that it is not viable to determine the direction of arrival of the satellite signals because it is extremely necessary to know, a priori, the location of the signal emitting source and thus the direction of arrival of the signals to, subsequently, compensate the phase errors of the antennas.

Assuming the antenna array designed on this project cannot be used in DOA applications due to differences on the antennas phase response, the developed system might have applications in the GPS signals spoofing detection. Theoretically, the system designed should be capable of detecting a static spoofer emitting source by observing the evolution of the carrier phase differences computed. As showed in Figure 55, the carrier phase differences between the signals registered by the SDRS followed a linear behavior due to satellite continuous motion. This linear evolution on the phase differences observed may be used to detect a possible emitting source transmitting from a static location.

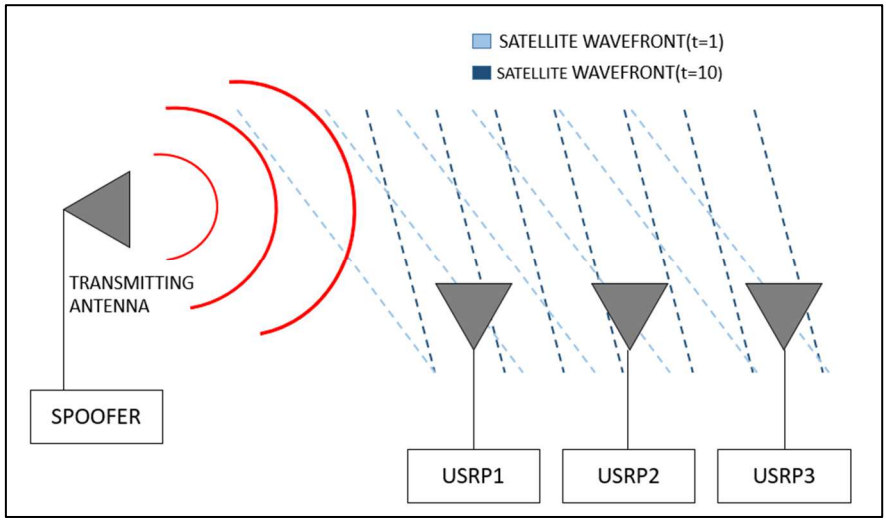

Figure 57 Static spoofing detector

As may be seen in Figure 57, in case of receiving a signal emitted by a static source, the phase differences registered by the antenna array would be constant along time, which is not the case when receiving real GPS satellite signals. If a signal phase difference were constant during an extended period of time, a static spoofer attack might be determined.

SUMMARY & CONCLUSIONS

The main purpose of this project was to develop a GPS calibrated multi-antenna platform by using both COTS software-defined radios and GPS antennas. The embryonic idea was the fact of using the basic hardware and software needed to simultaneously register signals broadcasted by the satellites which conform the GPS constellation. This hardware was initially formed by a personal computer, three Universal Software Radio Peripheral, an external oscillator and an antenna array built from commercial GPS antennas.

While developing the present project, it was possible to check that the basic hardware from the beginning was not enough to get time and phase samples synchronization between the front-ends. After performing some tests, it could be determined that there exist random phase offsets in the signals registered by each USRP and that these may drift over time, so a constant phase calibration was necessary. The proposed calibration method in this project consisted on using a phase difference of arrival mechanism based on a synthetic GPS signal reference transmitter. In order to correct these phase differences of the signals registered by the USRPs, it was required to make a hardware update; to do so, a synthetic GPS signal generator was added along with one splitter (to distribute the phase calibration signal to the SDRs) and three combiners (to combine the phase calibration signal with the real GPS data acquired by the antenna). Once more, after testing the new system and getting to combine the phase calibration signal with the signals received by the GPS antennas, it was possible to notice that there existed some errors on the phase estimated by each USRP. These errors were caused by the carrier tracking loop of the software receiver and they had to be solved using the navigation data bits of the GPS signals as indicators of possible phase shift on the carrier phase estimated.

Finally, a flexible L-shaped antennas array made up of three COTS GPS antennas was designed and some tests of the whole system were carried out in order to determine the direction of arrival of real GPS signals basing on the phase differences of the signals received by each antenna. After analyzing the obtained results from the tests it was possible to check the fragility of the GPS signals against possible multipath effects. The obtained results were used to compare the carrier to noise density of the signals received by each antenna and, thanks to this, to determine that the quality of the signals received by each antenna may drastically vary depending on the positive or negative contribution of the mentioned multipath effect. Moreover, it was possible to observe that there exist some differences on the phase response of the different antennas due to the mutual coupling effect provoked by the different antennas interactions.

Given that to correct the antennas phase errors it is strictly necessary to know a priori the location of the emitting source, it could be determined that the antenna array developed in this project was not able to determine the direction of arrival of the GPS satellite signals. However, in order to check the behavior of the phase response of each antenna, these errors were corrected and it was possible to see that, as far as possible, the phase response of the antennas followed a linear behavior, so at least the prediction of the satellite movement based on the phase differences received by the antenna array was achieved.

As said before, although it was impossible to compute the DOA of the signals registered by the platform, the hardware and the methods used in this project might be used for the detection of GPS static spoofer attacks. Moreover, since the calibration process performed throughout this project guarantees at least the reception of time and phase aligned samples, the platform developed might be used as a testing tool for any other antenna array.

Following and as a kind of final act, there is a list of possible future improvements for the designed platform:

- Calibrate the antennas in an anechoic chamber. This improvement would allow to correct the phase errors of the antennas and on this way to model the mutual coupling effect of array layout on the antennas phase response.

- Take into account and solve the main array uncertainness like: platform orientation perturbations, possible turbulences and inclination of the platform during operations. This would help to obtain a better performance of the antenna array.
- Use radio frequency absorbing materials to isolate the antennas that conforms the array. This improvement could be useful to reduce the mutual coupling effect between the antennas.
- Using a more recent version of the phase calibration signals generator. The Labsat version used for this project was Labsat 1.0; this requires the exclusive use of a personal computer just to control its functioning. Therefore, the fact of using a more recent version would allow to eliminate the use of this secondary personal computer.
- Another improvement related to the point above would be the implementation of a Windows virtual machine in the Ubuntu operative system of the personal computer that manages the communication with the USRPs. That is, if it was not possible to use a more recent version of the Labsat (as said before), this virtual machine would control the Labsat 1.0 and would allow to eliminate the necessity of a secondary personal computer.

REFERENCES

Bao, J., & Tsui, Y. (2005). *Fundamentals of Global Positioning Systems Receivers, A Software Approach.* . New Jersey: John Wiley & Sons .

Bonnor, N. (2012). A brief History of Global Navigation Satellite Systems. *The Journal of Navigation*.

Borre, K., Akos, D. M., Bertelsen, N., & Rinder, P. (2007). *A Software-Defined GPS and Galileo Receiver*. Boston: Birkhâuser.

Chen, Y.-H. (2012, September). A Study of Geometry and Commercial Off-The-Shelf (COTS) Antennas for Controlled Reception Pattern Antenna (CRPA) Arrays . *roceedings of the 25th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS 2012)*.

Danesh, S., Sokhandan, N., Zaeri-Amirian, M., & Lachapelle, G. (2014). Precise Calibration of a GNSS Antenna Array for Adaptive Beamforming Applications. *Sensors 14*, 9669-9691.

Hua, Y., Sarkar, T. k., & Wiener, D. D. (1991, February). An L-Shaped Array for Estimating 2-D Directions of Wave Arrival. *IEEE Transactions on Antennas and Propagation*.

Kim, U. S. (2005). Analysis of Carrier Phase and Group Delay. *Proceedings of the 18th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2005)*, 635-642.

Kim, U., De Lorenzo, D., & Gautier, J. (2004). Phase Effects Analysis of Patch Antenna CRPAs. *Proceedings of the 17th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2004)*, 1531-1538.

MacGougan, G. D. (2003). *High Sensitivity GPS Performance Analysis in Degraded Signal Environments.* Retrieved January 3, 2016, from University of Calgary: http://www.ucalgary.ca/engo_webdocs/GL/03.20176.GMacGougan.pdf

Montgomery, P., Humphreys, T., & Ledvina, B. (2009). Receiver-Autonomous Spoofing Detection: Experimental Results of a Multi-Antenna Receiver Defense against a Portable Civil GPS Spoofer. *Proceedings of the 2009 International Technical Meeting of The Institute of Navigation, Anaheim*, 124-130.

National Coordination Office for Space-Based Positioning, Navigation, and Timing. (2016, February 5). *Official U.S. Government information about the Global Positioning System (GPS) and related topics*. Retrieved 2016, from GPS.gov: http://www.gps.gov/

PCTEL Inc. (2015). *PCTEL Performance Critical.* Retrieved 2016, from http://www.antenna.com/artifacts/201168Install_WiSys_Perm.pdf

Racelogic Ltd. (2015). *SOFTWARE & FIRMWARE*. Retrieved 2016, from Labsat: http://www.labsat.co.uk/index.php/en/customer-area/software-firmware

Seco-Granados, G., Lopez-Salcedo, J. A., & Vicario, J. L. (2012, February). El sistema GPS. *Sistemas de Radionavegación*.

Spilker, J. J., Parkinson, B. W., & Axelrad, P. (1996). *Global Positioning System: Theory and Applications* (Vol. 1). Washington.

U.S.C.G.N.C. (2016). *U.S Coast Guard Navigation Center*. Retrieved February 15, 2016, from http://www.navcen.uscg.gov/?pageName=GPSmain

ANNEX I

UHD – GNU RADIO INSTRUCTION MANUAL

In order to ensure the correct use of the developed GPS multi-antenna system, the following sections are dedicated to describe step by step the data recording method.

1.1 UHD+GNU RADIO LINUX INSTALLATION GUIDE

Since the software installation may become a challenge, the following section will detail how to set up correctly the software environment.

The following installation method has been tested with Ubuntu, however, it must be able to function under any other Linux distribution. The step-by-step installation guide is listed below:

1. Copy and create the installation script from the official website.
2. Change the file permissions using the command *"chmod a+x build-gnuradio"*, where *build-gnuradio* is the name of the script.
3. Execute the script: *./build-gnuradio*

It is important to emphasize that internet connection is required along the installation in order to connect with the UHD and GNU Radio servers.

1.1.1 LOADING THE UHD DRIVER IN THE USRP

Once installed GNU Radio and UHD drivers in the personal computer is necessary to load the UHD drivers (images) in the front-ends. The step-by-step loading driver guide is given below:

1. Insert the USRP SD-card in the PC reader/writer.

2. Open a terminal and introduce the following path: *cd /usr/local/share/uhd/utils.*
3. Once on the directory, execute as root the following application: *sudo ./usrp2_card_burner_gui.py*.
4. Introducing the properly root password, the SD card Burner program will be opened (Figure 58). At this point, the correct drivers should be chosen from the following route: */usr/local/share/uhd/images.*
5. Once the properly images have been chosen, the drivers must be burned on the SD-card by simply clicking on "*Burn SD Card"*.

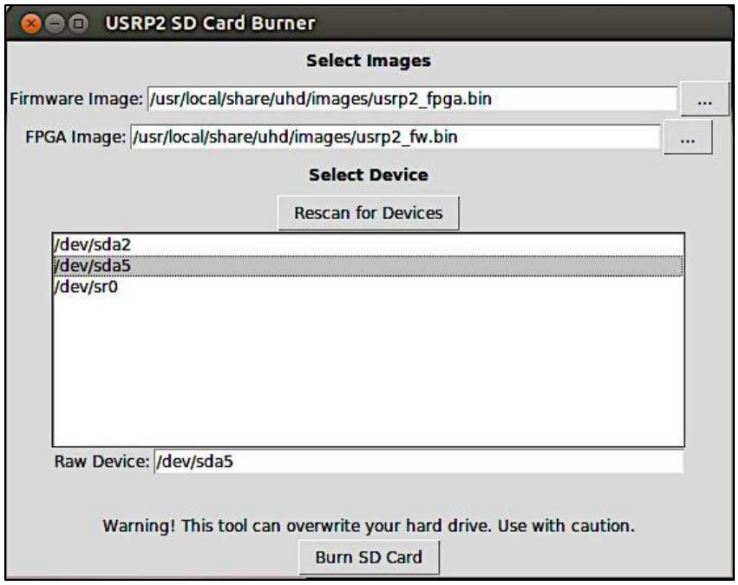

Figure 58 SD Card Burner Program

1.1.2 PC NETWORK CONFIGURATION

As commented before, since the communication between the SDR's and the personal computer is carried out through IP protocol, a previous PC network configuration is required. In order that the PC and the USRP's can work under the same network, it is necessary to create a new wired connection. For that, the following steps must be carried out:

1. Open the Network connections editor.
2. Create a new-wired network configuration by clicking "*Add"*.

Figure 59 Network configuration

3. Since the default IP address of the USRP's is 192.168.10.2, the personal computer IP must be within the same range but it cannot be the same. A valid example may be set the PC IP address to 192.168.10.5.

Figure 60 Personal computer IP configuration

4. In order to use multiple USRPs on the same host computer, the front-end default IP address must be changed. For this, the following commands should be executed:

   cd <install-path>/lib/uhd/utils

   sudo ./usrp2_recovery.py --ifc=eth0 --new-ip=192.168.10.3

   Notice that this procedure assumes that the USRP IP address is unknown; therefore, this procedure should be performed attaching a single front-end to the personal computer.

Having executed the steps above, the compatibility between the SDR's and the PC is ensured.

1.2 DESIGNING A BASIC USRP RECEIVER IN GRC

First of all, in order of giving an overview of GNU Radio software's usage for communicating and configure the front-ends parameters, the most basic example of communication method GNU Radio-USRP will be explained.

As explained before, GNU Radio is a collection of tools that can be used to develop radio systems in software as opposed to completely in hardware. The software includes a graphical interface, GNU Radio Companion (GRC), which is so similar to Simulink and that allows the user to process signals simply interconnecting predefined filters and many other elements (blocks). GRC was created to simplify the use of GNU Radio by allowing the creation of python files graphically instead of creating them just in code alone. The designing guide of a basic receiver using a USRP device is listed below:

1. Open the GRC software by executing the following command on the terminal: *sudo gnuradio-companion.* Note that the administrator password is required.

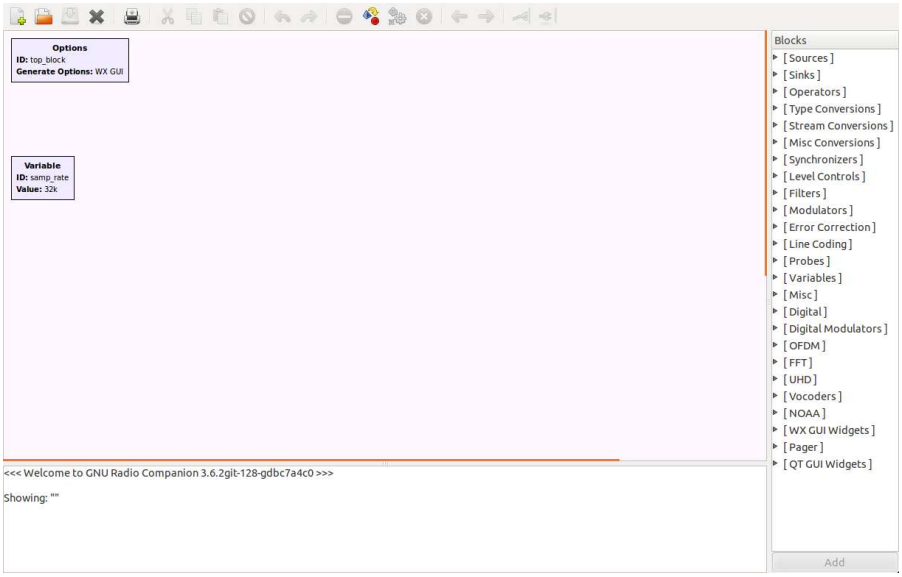

Figure 61 GRC software interface

2. In order to work with the generated python file under the Linux terminal, the Options block (Upper left block, Figure 61) must be configured to do not show anything by the graphical environment. The Figure 62 shows the properly options block fields configuration. The most important fields are the following:

    - **ID:** The Phyton file will be generated under this name.
    - **Generate Options:** *No GUI* (Do not show anything by graphical environment.)

- **Run Options:** *Run Completion*
- **Realtime Scheduling:** *ON*

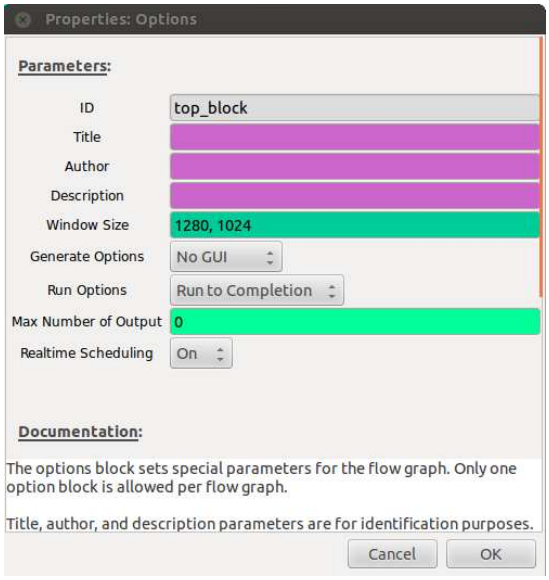

Figure 62 Options block properties

3. In order to establish communication with the USRP device, a block driver is necessary. This block can be found under the name of UDH: USRP SOURCE, in the Block menu (Right menu, Figure 61). The configuration properties for a simple receiver are shown in Figure 63 and the most important parameters are the following:

   - **Output Type:** Select the data type of the stream.
   - **Device Address:** As explained above, the USRP devices are identified by means of its IP addresses.
   - **Sync:** This field is used to synchronize some USRP devices. In this case should be set to *"Don't sync"*.
   - **Clock, Time source:** In this case should be set to *default*. This means that the device will use its own internal clock.
   - **Sample Rate:** Set the desired sample rate.
   - **Center frequency:** Choose the reception center frequency.
   - **Channel Gain:** Choose the desired gain depending on the USRP daughterboard.
   - **Channel Antenna:** Select the particular reception channel: *RX/TX*.
   - **Channel Bandwidth:** Choose the desired Bandwidth depending on the daughterboard.

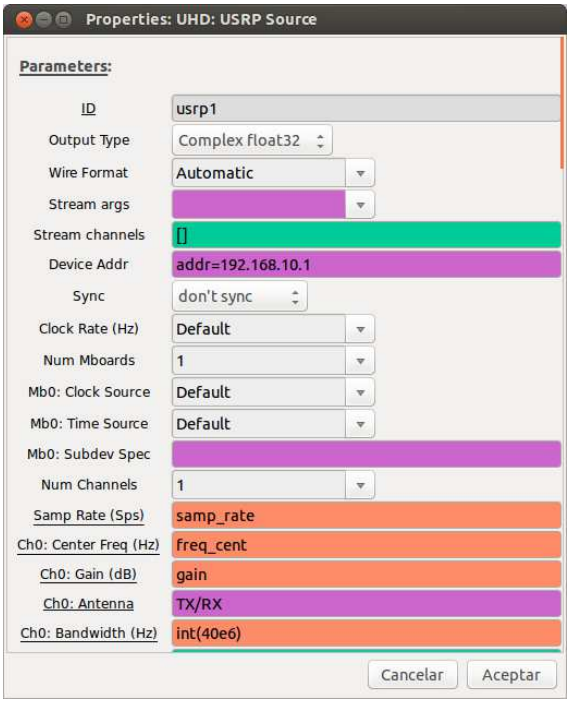

Figure 63 USRP Source Properties

4. In order to make the resulting python code more readable and easy to manage, the USRP reception parameters are created as variables (Left blocks, Figure 67). The variable block can be found under the name of VARIABLE, in the Block menu (Right menu, Figure 61).

5. The USRP digitalize the RF signal acquired in form of a complex stream (IQ data), this digital signal should be divided depending on the component in two different streams. To do this, two different conversion blocks are needed. Both blocks can be found under the name of COMPLEX TO REAL and COMPLEX TO IMAG in the Block menu (Right menu, Figure 61).

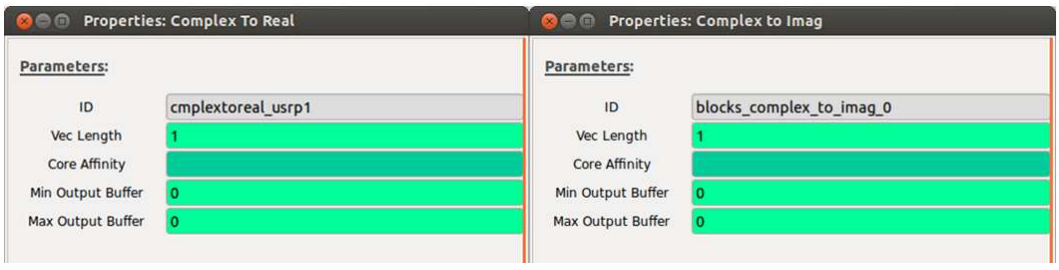

Figure 64 Complex to Real & Complex to Imag Properties

6. With the purpose of saving the data acquired by the USRP, a couple of write stream to file block are needed. This block can be found under the name of FILE SINK, in the Block menu (Right menu, Figure 61). The most relevant parameters to configure are

the name, the path file location and the input data type. Notice that the input data type must be the same as the USRP output data type.

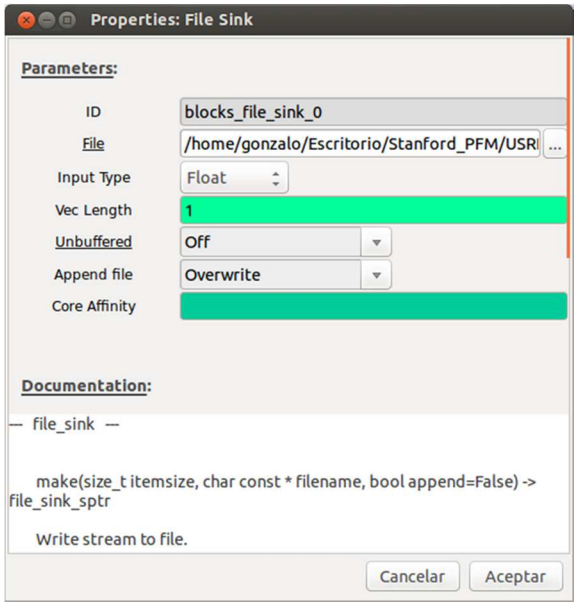

Figure 65 File sink properties

7. Having in mind to receive a certain number of samples, a block which restricts the samples received is needed. This block can be found under the name of HEAD, in the Block menu (Right menu, Figure 61). The samples value should be the number of samples to record. Therefore, knowing the reception time and the sampling frequency, it is trivial to determine this parameter.

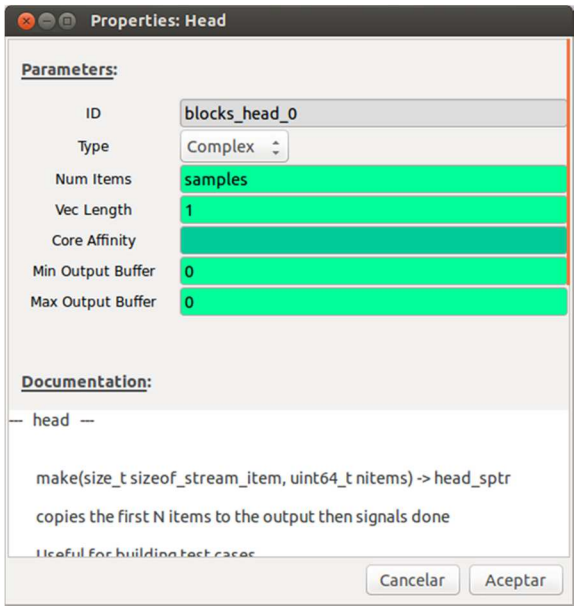

Figure 66 Head block properties

8. Once the whole blocks have been properly configured, the USRP basic receiver is created simply by linking the blocks together.

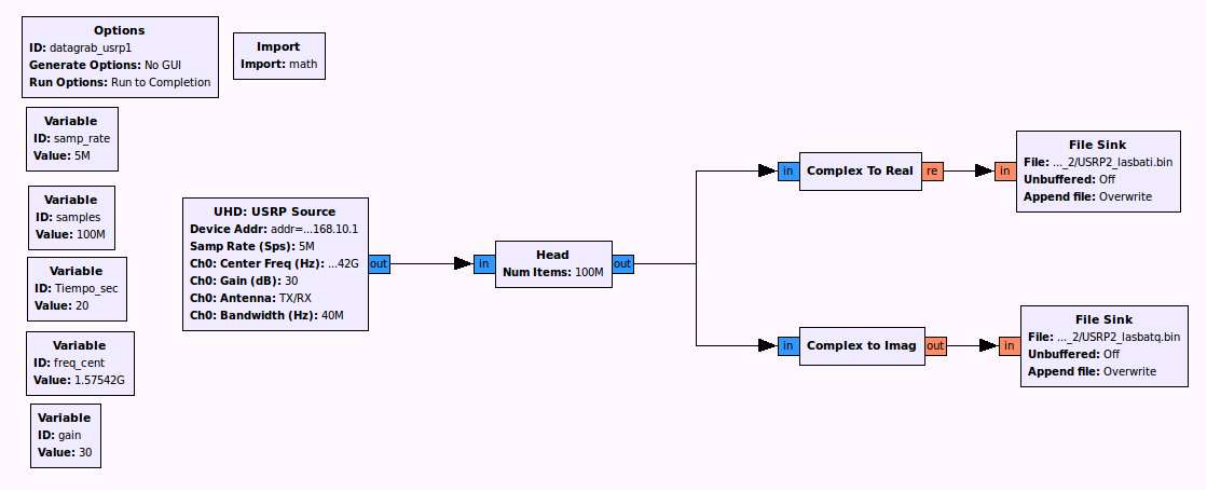

Figure 67 Basic USRP receiver

9. Finally, by clicking on *Generate the flow graph* (Figure 68) a python file is generated. This file contains the python source code for the flow graph and it can be easily executed from a Linux terminal.

Figure 68 Generate the flow graph

1.3 DESIGNING A MULTI-USRP RECEIVER

Having in mind the USRP basic receiver model (Figure 67), a Multi-USRP receiver is defined. The whole blocks used to create a Multi-USRP receiver is proportional to the number of USRPs used; this means that each USRP requires the same blocks used to build the basic receiver. Therefore, since the blocks that conforms the Multi-Receiver are exactly the same as the used in the basic receiver, this section details only the Multi-USRP block configuration.

In order to establish communication with the USRP's devices, a block driver is necessary. This block can be found under the name of UDH: USRP SOURCE in the Block menu (Right menu, Figure 61). The configuration properties for a Multi-Receiver are shown in Figure 69 and the most important parameters are the following:

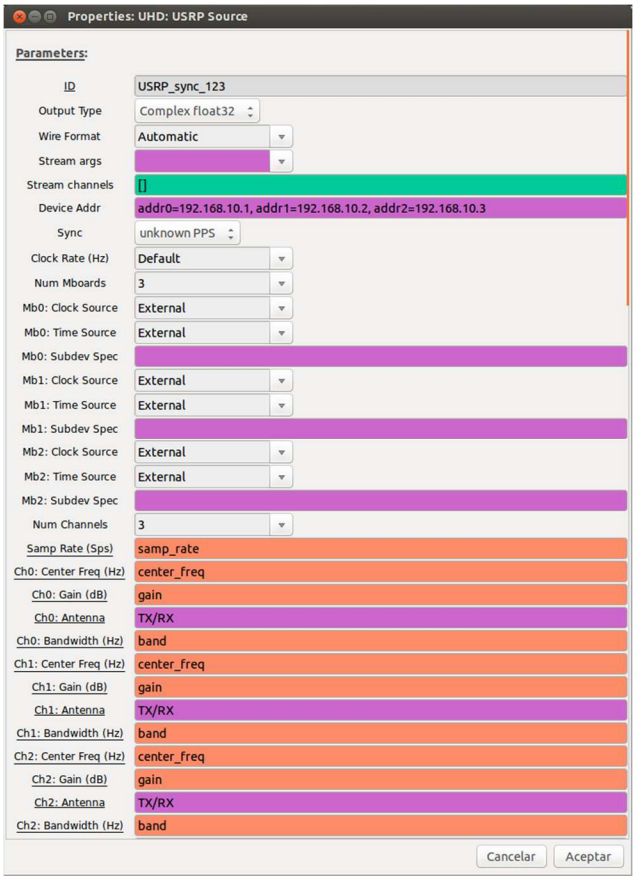

Figure 69 Multi-USRP Configuration

- **Output Type:** Select the data type of the stream.
- **Device Address:** USRP devices are identified by means of its IP addresses. In order to use several USRPs working together is essential to have previously changed its default IP addresses. The different SDRs are recognized following:

  addr0=192.168.10.1,addr1=192.168.10.2,addr2=192.168.10.3

- **Sync:** This field must be set to *unknown PPS* in order to synchronize the USRP devices using an external clock and PPS signals.
- **Clock, Time source:** In this case, set to *default*. This means that the device will use its own internal clock rate.
- **Subdev Spec:** Sub-device specification, follows the format "X:Y", being X the daughterboard slot. USRP2 model cannot stand multiple daughterboard, that is why UHD automatically select the unique sub-device found if the field is empty.
- **Sample Rate:** Set the desired sample rate.

- **Center frequency:** Choose the reception center frequency.
- **Channel Gain:** Choose the desired gain depending on the USRP daughterboard.
- **Channel Antenna:** Select the particular reception channel: RX/TX.
- **Channel Bandwidth:** Choose the desired Bandwidth.

Following the steps described above, and once the whole blocks have been properly configured, the Multi-USRP GPS receiver is created.

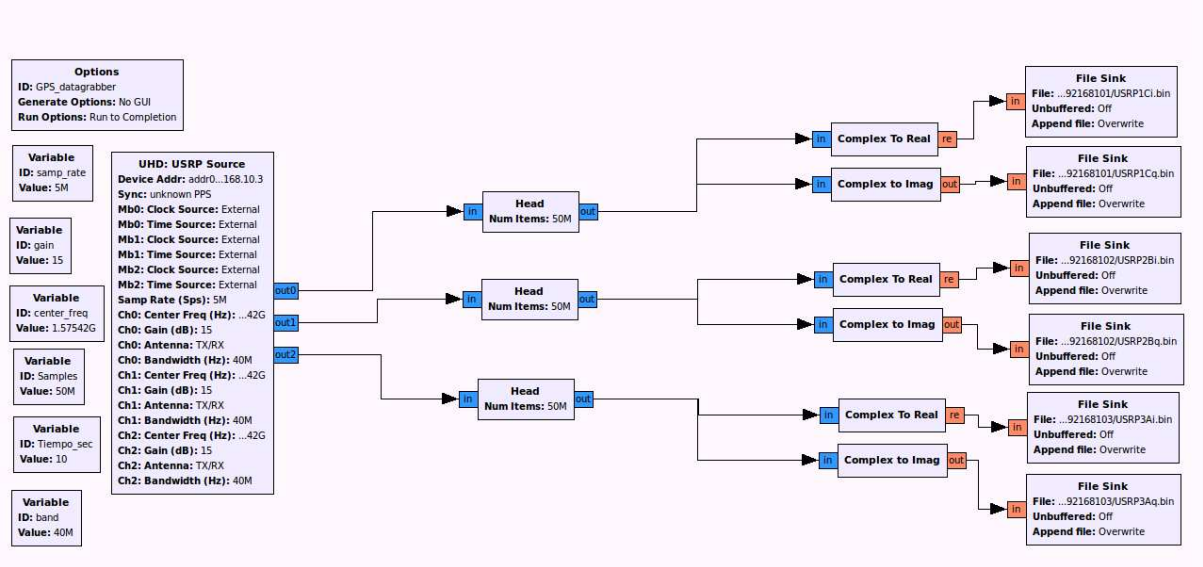

Figure 70 Multi-USRP GPS receiver

ANNEX II

PHASE CALIBRATION SIGNAL GENARATION

In order to explain how to generate the phase calibration signal employed for this project, the used method —both software and how to use it— will be introduced below:

1. Download and install Satgen v3 Software from Labsat official website. Without license, the software is restricted to run for only 120 seconds when creating GPS scenarios (Racelogic Ltd, 2015).
2. After the installation of the Satgen software, a main screen should be displayed. Satgen software allows the user to recreate different kinds of scenario receptions. In order to recreate the reception of GPS signals from a static scenario, choose the option *Static scenario* as shown in Figure 71.

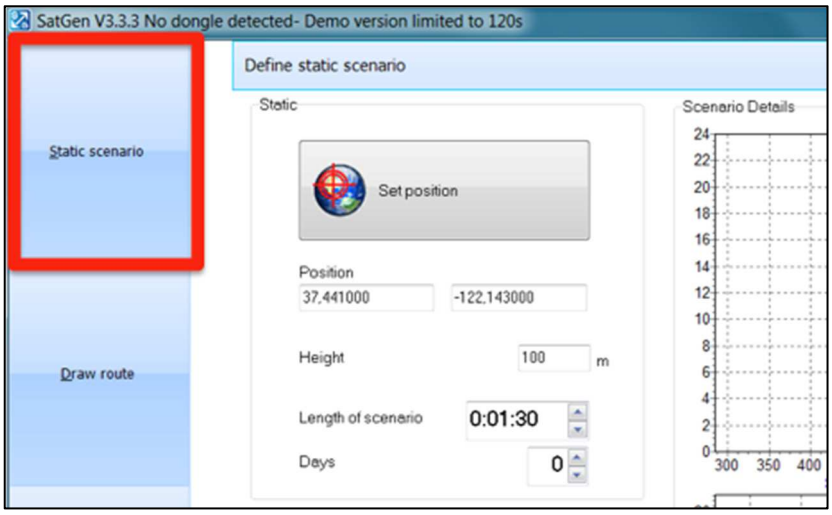

Figure 71Satgen - Static scenario option

3. As may be seen from Figure 72, there exist two options to set the scenario location: (1) by filling in by hand the *Position* form with the latitude and longitude of the desired

location expressed in decimal degrees or (2) by simply using the *Set position* button. This option allows the user to search the desired location by exploring Google Maps.

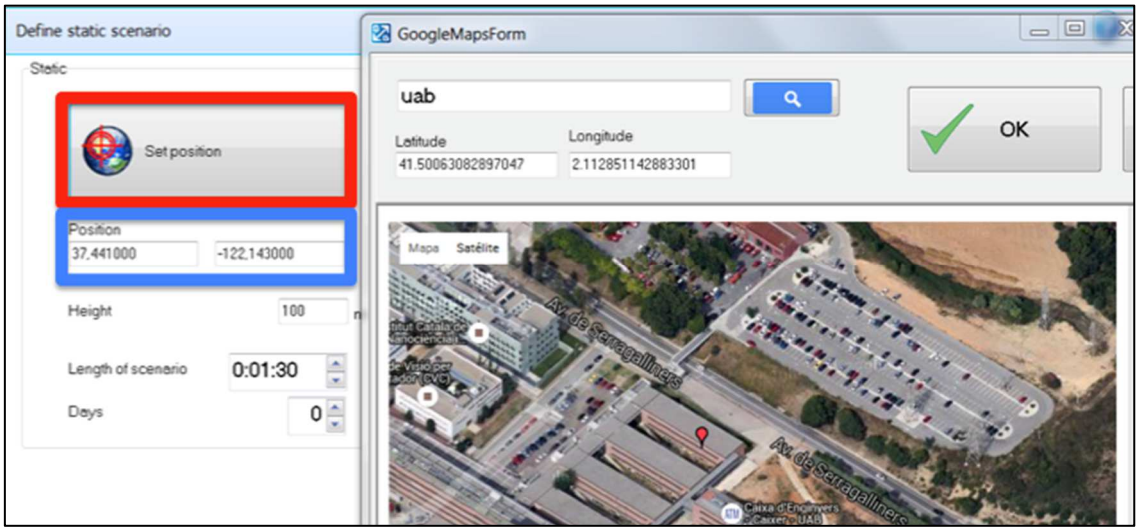

Figure 72 Setting user location

4. Having established the static user location, some signal and scenario parameters should be determined. These parameters should be established simply by filling the form depicted in Figure 73. From this form, the most important parameters to set are:

    - **Date and time of simulation:** Since the main objective of this synthetic signal generation is to combine it with real GPS data, it is important not to simulate a scenario in which satellite constellations are already present when recording real signal data.

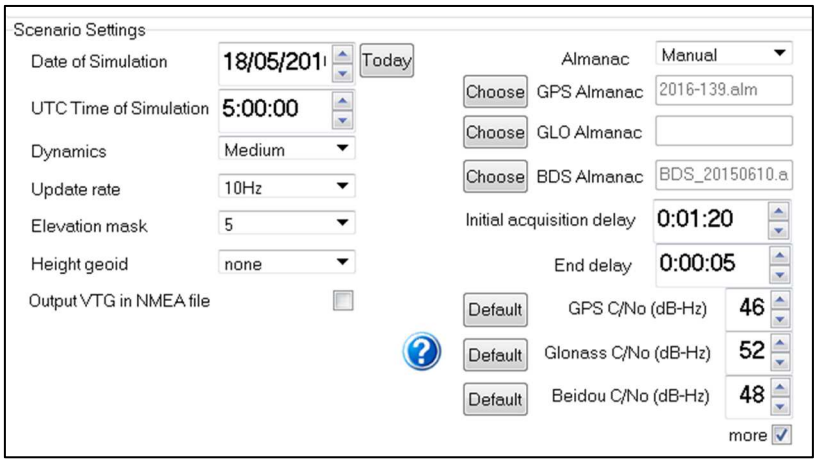

Figure 73 Date, time and signal parameters.

    - **Elevation mask:** This parameter allows the user to discriminate low orbit satellites, the most high the elevation mask is, the most discrimination there will exist.

- **Almanac:** The GPS almanac is a set of data that every GPS satellite transmits, and it includes information about the state (health) of the entire GPS satellite constellation, and coarse data on every satellite's orbit. In order to download the correct GNSS almanac file, Satgen software allows the user to choose two different options: *automatically* or *manually*.

  Choosing the *Automatic* option, the software will download the almanac file corresponding to the simulation date set by the user. In order to generate or not satellite signals from the full constellation present on the scenario, Satgen software uses the satellite information contained in the almanac file, Figure 74.

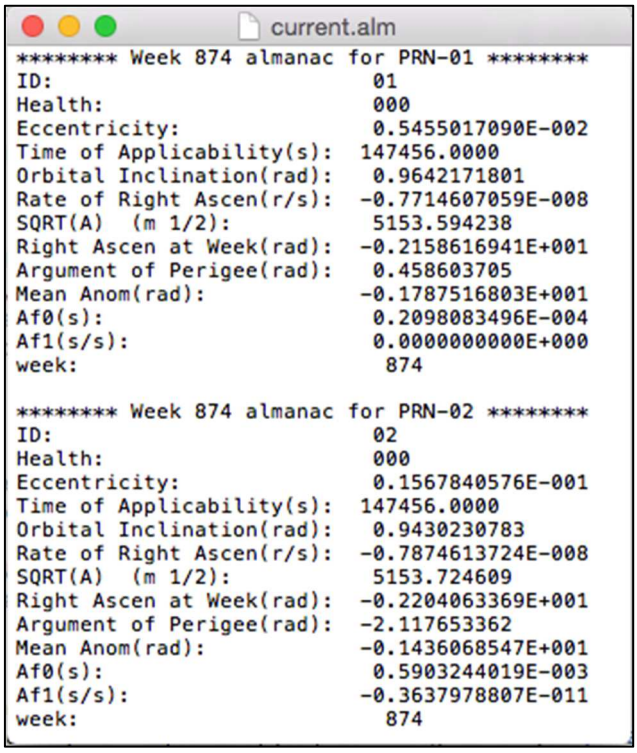

Figure 74 Almanac file information

Since the main objective of this synthetic GPS signal generation is to obtain a single satellite signal, this almanac file should be hand modified and manually loaded to the software using the option *Manual*. This almanac file can be downloaded from the U.S.C.G.N.C. (2016) and easily modified using a text editor; the resulting almanac file should contain just the information of the desired satellites.

- **GNSS C/No:** This parameter is nothing but the signal carrier to noise density, which should be set carefully depending on the application. On this project application,

the GPS signal strength was recreated with the maximum carrier to noise ratio value allowed by the software which is 51dB-Hz.

5. Finally, generate the synthetic GPS signal by clicking the *Create Scenario* button. Satgen software allows the user to create different GPS signal output files depending on the Labsat model that will be used to replay this signal.

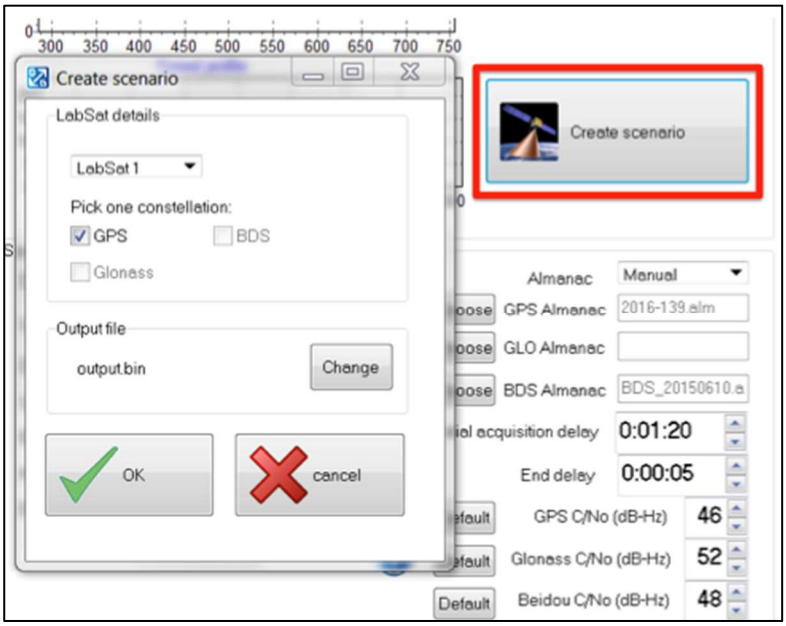

Figure 75 Creating the scenario

Having executed the steps described above, the synthetic GPS phase calibration signal would be ready to be combined in hardware with the signals acquired by the antennas.

ANNEX III

GPS MULTIANTENNA PLATFORM RECORDER

Recapitulating, at this point, the GRC software installation has been described, the GRC Multi-antenna receiver generation has been explained step-by-step and the phase calibration signal generation has been introduced in the annex before. Therefore, this section is dedicated to introduce the user how to combine both real and synthetic GPS signals using the multi-antenna platform recorder created in this project.

In order to replay the phase calibration signal created with Satgen software, a Labsat receiver is needed. This particular receiver uses the software Labsat to establish communication with the personal computer where the data created with the Satgen software is stored. Notice that this section does not pretend to be a full user guide of the Labsat receiver but it will define the clue steps to take into account in order to use the Labsat receiver to replay a GNSS signal stored in a personal computer. A brief guide of the software usage and its installation is introduced below:

1. Easily and freely download and install the Labsat software from Racelogic official website (Racelogic Ltd, 2015).
2. As may be seen in Figure 76, after the Labsat software installation, a main screen should be displayed. This interface allows the user to replay, record or configure the Labsat receiver. In order to replay signals, the replay mode should be chosen.

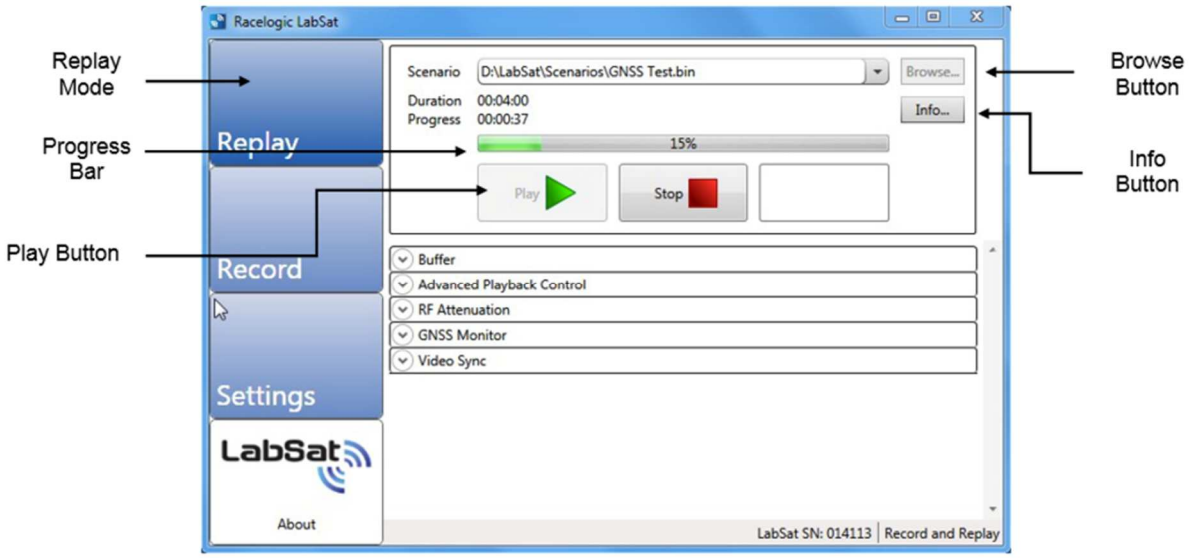

Figure 76 Labsat interface

3. From the *Replay*, tab click the *Browse* button to navigate to the phase calibration signal file created with the Satgen software.
4. In order to replay the phase calibration signal, simply click the *Play* button and check the progress bar to ensure that the signal has been reproduced correctly.

Notice that the replaying method described above will work with the first versions of Labsat receiver (LabSat & LabSat2), if the user disposes of a latest device version, the method explained may not work.

Having explained the phase calibration signal replay method, the following lines will define how to launch the GRC multi-antenna recorder platform from Ubuntu:

1. Open a command terminal by pressing: *Cntrl+Alt+T*
2. Browse to the Multi-antenna file location by tipping the command:

   cd path/Multi-antenna_python_file_location/
3. Before launching the Multi-antenna receiver and in order to improve transfer efficiency, it is necessary to increase the maximum socket buffer size of the personal computer. For this, the following commands must be executed as super user:

   sudo sysctl -w net.core.wmem_max=1048576
   sudo sysctl -w net.core.rmem_max=50000000

4. Finally, launch the Multi-antenna receiver by executing the following command:

   python Multi-antenna_python_file.py

Notice that in order to simplify the Multi-antenna launching process it could be interesting to compress the full steps described above in a single script. As depicted in Figure 77, this script must be executed as super user so that it will launch the Multi-antenna application in an easier and faster way.

```
######################################################################
# This script must be executed as Super User (sudo ./ScriptName)
######################################################################
#! /bin/bash
sysctl -w net.core.wmem_max=1048576
sysctl -w net.core.rmem_max=50000000

cd path/Multi-antenna_GRC_file_location/
python Multi-antenna_GRC_file.py
```

Figure 77 Multi-antenna launching script

It is trivial to determine that in order to combine both real GPS and phase calibration GPS signal, the applications that are broadcasting/receiving both mentioned signals need to be run at the same time.