NoSpoof – A multiple antenna phase detection approach

By

Kostadin Georgiev Biserkov

GRADUATION REPORT

Submitted to

Hanze University of Applied Science Groningen

in partial fulfillment of the requirements

for the degree of

Fulltime Honours Bachelor Advanced Sensor Applications

2015

ABSTRACT

NoSpoof – A multiple antenna phase detection approach

by

Kostadin Georgiev Biserkov

The following paper presents a research on the subject of Spoofing - a malicious attack on global navigation satellite systems, capable of replicating the genuine signals, in order to emit false data to the receiver. Together with Science & Technology and Astron, the author aimed at describing a conceptual design of a system, capable of detecting spoofing attempts through the use of a phase detection algorithm, based on the parallel observation of multiple antennas. While there are many researches done in the past, evaluating the danger of spoofing and complex methods of countering it, the paper focuses on using a method based on angle-of-approach discrimination. This choice was made due to the fact that latter uses the physical properties of the signal, and not encryption algorithms.

The research focuses on evaluating the concept, its key parameters, their importance and effect on the end results. While the end product cannot be classified as ready for direct application, it can serve as a base for further developments and improvements. A list of recommendations can be used as guidelines during any future attempt of recreation of improvement upon the conducted experiments.

DECLARATION

I hereby certify that this report constitutes my own product, that where the language of others is set forth, quotation marks so indicate, and that appropriate credit is given where I have used the language, ideas, expressions or writings of another.

I declare that the report describes original work that has not previously been presented for the award of any other degree of any institution.

Signed,

Kostadin Georgiev Biserkov

ACKNOWLEDGEMENTS

The author would like to express his sincerest gratitude's to the supervisors from both S&T and ASA - Andre Bos and Bryan Williams, for their continuous support and dedicated tutoring.

Special thanks should be given to Mr. Koos Kegel, representative of Astron, for the technical support and tutoring throughout the full length of the project.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1
RATIONALE

The following research was proposed by Science & Technology Corporation [1], a company specialized in project implementations and product development where science and technology play an important role. The technology focused on by the experts present can be split into three different areas: *Sensor Solutions, Computer Vision* and *Bioinformatics.* Within the " NoSpoof – A multiple antenna phase detection approach" project, the area of Sensor Solutions is mainly concerned as the focus is on the extraction of data from a complex system and its analysis.

The "NoSpoof" project is a part of a larger project, called "Intergalac". One of the main focuses of the latter is to develop and provide additional services to the existing and upcoming Global Navigational Satellite Systems (GNSS) [2]. In the case of "NoSpoof", the service in development is rather narrowly specified - the goal is to design, test and evaluate the concept of a system, capable of providing a reliable level of defence against signal spoofing attacks (capable of detecting the presence of the more commonly observed types. For a brief description please refer to Chapter 2[3]). In its essence, spoofing represents the attempt of a third party to interfere with the signal reception, specifically from satellite navigation systems. Unlike signal jamming, where it becomes obvious to the receiver that no valuable signal can be received, the spoofing device creates an accurate simulation of the source signal [4]. After careful adjustment within the data frames, one can "trick" the receiver into using false data, while assuming the received signal is genuine. An example can be given with ships in the open ocean - since in most cases there are no visual landmarks that can be used for navigation, most captains are entirely dependent on

their navigation equipment. If the latter can be tricked into showing false location, heading and speed, such ships could be lead off the trade routes, and in waters where they are more likely to encounter modern pirates.

There are currently spoofing protection devices and algorithms, but they are mainly developed and used by more specific users (such as military). Since the techniques used often include specific encryptions [5], it is not available for the large number of commercial users. The end goal is to provide similar options for the wider market.

Based on previous studies, there is a suggested method which should allow for detection whether the signal received is genuine or an interference attempt. Knowing the fact that while the precise content and power of the signals can be mimicked accurately, it is also important to note that since all signals are captured by a single antenna, it is impossible to recreate the direction from which the signal is received. Theoretically, by using the input from multiple antennas, placed at different locations, the direction of the incoming signal can be detected (based on phase differences), thus allowing to exclude multiple satellite signals with the same source. Another clue to the presence of a spoofing attack is to monitor the change of the phase difference over time - the satellite constellation is a dynamic system, while in most situations spoofers are on fixed location with respect to the receiver. A similar technique was used within another spoofing detection project, relying on synchronous movement of antennas[6]. While in the research "Spoofing Detection with Two-Antenna Differential Carrier Phase"[6], conducted by Mark L. Psiaki and his team, also focuses on the physical distribution of the signal, their method involves a rather sophisticated mobile structure. In addition, their research focuses on the observation of the carrier signal (with frequency 1.575GHz) - while being more precise, it requires powerful (and thus expensive) equipment.

While the target of the project is to provide a security related service for the upcoming Galileo GNSS, in its current stage the project focuses on the use of the already established GPS system. Taking into consideration the brief description given above, the main question driving the project is:

"What *are the main design parameters needed for a multiple-antenna setup, capable of utilizing Code Phase detection techniques for detection of spoofing signals of the GPS system?"*

In addition, the following secondary questions will be used to further clarify the subject.

"What *are the main design parameters to detect the Code phase accurately enough*[1] *to decrease the antenna distance to minimum? How does this reflect on the hardware requirements?*"

"*What methods and algorithms would be of benefit to the phase detection of the Code of the GPS signal?*"

[1] The Code phase detection focuses on the time difference between the received signals at the two antennas. Since this directly relates to distance (the speed of the signals is constant), the minimum time difference that can be detected by the converter must be researched.

CHAPTER 2
SITUATIONAL & THEORETICAL ANALYSIS

The focus of the project is on the security and robustness of GNSS signals mainly because of the extent to which it is used in different fields - from law enforcement and shipment vehicles to the everyday consumer electronics (smart phones, car navigation, etc.). While the main goal is for the final product to be used together with the Galileo network, for the time being the team will be making use of the already existing GPS, since only eight satellites from the Galileo have been launched.

In order to evaluate the danger that signal spoofing represents [3] one must first understand the principles of such spoofing in more detail (i.e. the device - responsible for the execution of the signal spoofing). The spoofer is based on the following main elements:

1. **GNSS software simulator** [7] : this is one of the more complex components of the system. In its essence, any widely used GNSS (Global Navigational Satellite System) [8], such as GPS, is widely known and almost any aspect of its work and data transmission system is documented and available to the masses (hence its popularity and wide usage). This feature makes the GPS satellite constellation and data transmissions relatively easy to simulate in a virtual environment[2]. Once this is achieved, the next step is to synchronize the simulation with the actual constellation, and alter the parts of the signal, used for the navigation calculations. Since the power of the received signal is rather low (due to distance, Gaussian noise etc.), the spoofer can easily simulate the power levels,

[2] GPS software simulators are commercially available and widely supported: examples are LabSat, NAVSYS, NI Global Navigation Satellite System Toolkit, and many more.

and once the receiver has locked to the spoofed signal, slightly increase the power levels. This would make locking back to the genuine signal rather impossible with the widely used techniques, currently employed by most GPS receivers. (**Note**: Spoofer devices vary in their complexity - some may not include the synchronization components, described above).

2. **Transmission hardware**: once the simulated signal is generated, it needs to be converted into an analogue signal. This is done mainly through the use of DACs (digital-to-analogue converters), in combination with certain other RF front-ends (such as filters, mixers, amplifiers and finally an antenna). Since the exact parameters of the specific GNSS must be duplicated, the hardware must be designed (or tuned) to specific frequency bandwidth (for example - in case of GPS, the signal carrier frequency we are focusing on is L1 = 1575.42MHz) [9]. It is important to note that certain components of the final signal (carrier frequency and the code itself) must be altered, in order to compensate for the Doppler shift [10], resulting from the movement of the satellites with respect to the receiver.
3. **Proximity to the target receiver**: The spoofer requires a specific location for the signal simulation, hence the device must be placed close to the targeted receiver. If the distance is larger, the simulation must be created such as to compensate for the displacement, and the hardware must also be adapted in order to ensure plausible power levels and high reception at the target receiver.

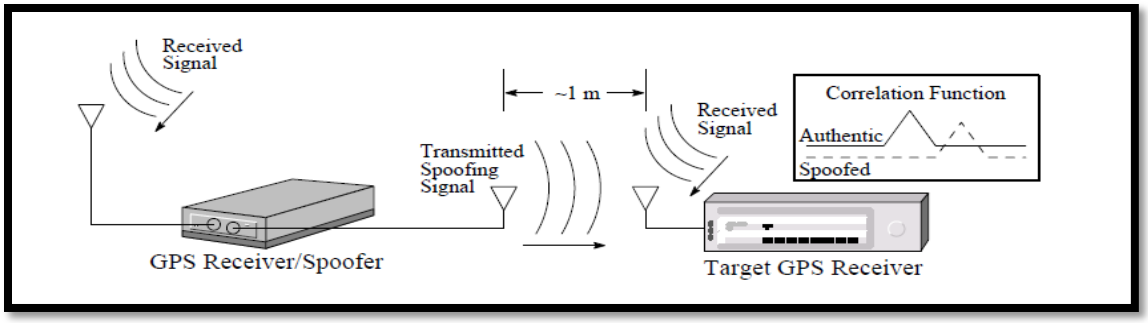

Figure 1. Spoofer Device Representation. (Kai Borre, "A Software-Defined GPS and Galileo Receiver")

Having defined the most crucial and fundamental points of the spoofing system, we must also note that there can be different types of spoofers, classified mainly according to their complexity and efficiency (Table 1). For the purpose of this project, a spoofer is defined as an immobile device with a single antenna. While it is possible to create a mobile simulator, the complexity of the system needed to compensate the resulting Doppler effect, change of location etc. will increase drastically both the cost and resources, needed to maintain the correct functionality of such a spoofer. Due to the fact that the spoofing device needs to be placed close to the receiver, it is assumed that its location in the vertical plane is close to the one of the receiver. In other words, the most common situation addressed is when the spoofer device is aligned with the horizon.

Type of spoofer	Brief description
Unsynchronized	○ The GPS simulator used does not repeatedly synchronize the simulated signal to the genuine one. ○ Least sophisticated, easily reproducible by commercially available products.
Loosely synchronized	○ It aims to accurately mimic the genuine GPS signals, with respect to time delays, Doppler shift, navigation data frames etc. ○ Requires a feedback loop from a receiver to the simulator.
Tightly synchronized	○ Similar to the loosely synchronized, but making sure that the simulated signal is with less than half a C/A code chip variation from the genuine signal. ○ Usually such spoofers focus the different PRNs in a sequence, instead of simultaneous attack. This is done in order to avoid suspicion if the overall power of the channels is carefully observed.

Table 1 Brief description of the different types of spoofers.

CHAPTER 3
CONCEPTUAL MODEL

Signal spoofing is a known threat to GNSS for many years. As a result, many different detection and protection methods have been designed and tested, each with its advantages and disadvantages [3][11][12]. The first part of the NoSpoof research phase focused on filtering those methods and picking the most plausible to implement into the commercial (public) products. While there are a lot of possible solutions considered theoretically, the most feasible and common techniques are:

Amplitude discrimination - filtration and evaluation of the signal based solely on the amplitude (power) of the incoming signal. While the genuine signal has a well-known amplitude behaviour, the spoofer can easily simulate this behaviour. As stated by Todd Humphrey and his team in "Assessing the Spoofing Threat: Development of a Portable GPS Civilian Spoofer"[3], amplitude discrimination can be directly applied on the software level of the GPS receivers currently on the market - meaning that there is no need for further change of the hardware structure. The authors however also make it clear that because this method focuses purely on the power of the received signal, it can prove to be ineffective against more complex spoofing systems. In their original analysis, the authors also do not take into account the effects of bad signal reception or signal interference could have on the robustness. In specific situations, the latter could result in altering the spoofing signal, making it more similar to the genuine one.

Angle-of-arrival discrimination - this method uses a "multiple antenna" approach - by deploying several antennas, with known geometry and distance, one can use beam-forming techniques to make a "map" of the incoming signals. The filtration of the signals is done by

evaluation of the map - the genuine satellite signals are spread throughout the visible sky, while the spoofed signals will have a single source direction. In addition, the genuine satellite signals change their location over time - unlike the spoofers focused by the author. By monitoring the time delay map over time, one could detect the presence of a spoofer if the phase differences calculated remain constant for prolonged periods of time. Most researches done on this methodology focus on the carrier signal of the GPS system[6][3]. As explained in "Assessing the Spoofing Threat: Development of a Portable GPS Civilian Spoofer", monitoring the carrier phase allows a more precise detection, due to the high frequency (1575.42MHz). The authors of the paper, however, do not give an opinion on the possibility of using a precisely measured code phase, instead of the carrier phase.

Cryptographic authentication - by embedding encrypted messages within the structure of the GPS network, each receiver could authenticate the incoming data. While this could prove to be the most secure option, it has one major disadvantage - the GPS system has to be modified. This would mean that older GPS receivers would not benefit from the change, or may even prove to be unusable.

There are other, more seldom considered techniques for the detection of spoofing attacks. For example, in "Assessing the Spoofing Threat: Development of a Portable GPS Civilian Spoofer"[3] the authors also mention that *time-of-arrival* discrimination can add basic levels of protection. Its principle is based on the fact that at the moment the spoofer system is triggered, it has no prior information on the exact GPS location of the receiver. As a result, there will be an initial delay in the data bits transmitted, which could later be compensated. Its validity is assessed by John Nielsen and his team in "GPS Vulnerability to Spoofing Threats and a Review of Antispoofing Techniques"[13]. In the research, the authors give a clear statement that this

method has strong limitations, due to the well-known structure of the GPS system and the low update rates (the C/A code repeats every millisecond, with 1023 data bits in each repetition). If the spoofer includes a start-up period, where it can perform synchronization of the data bits before the emitting of the spoofed signal is triggered.

The other methods, less commonly considered, include L1/L2 signal relative delay and GPS clock consistency check. The first is based on the fact that the signals in the two GPS bands (L1 at 1575.42MHz and L2 at 1227.60MHz) have a constant relative delay, due to the different effects that the ionosphere[14] exerts on their propagation.[13] The second observes the GPS clock information present in each PRN signal and looks for inconsistencies among the data present. Both are discarded by the authors of GPS Vulnerability to Spoofing Threats and a Review of Antispoofing Techniques"[13] due to the fact that all of the information monitored is well-known and easy to simulate by any synchronized spoofer.

While technique 1 can be implemented rather straightforward, it is also easy to deceive . It will allow certain levels of filtration, but the protection provided will be quite minimal. Technique 3 is the exact opposite - while it can filter almost all potential spoofed signals, the changes required on the GPS structure and of the receiver designs, translate into high resource demand (both financial and human-power). Because of those disadvantages, methods 1 and 3 are discarded. This leaves the option to use a phase detection technique (method 2).

Phase detection is a technique where the system uses the phase shift between two or more instances of a signal, to determine its source position (spatially) with respect to the receivers used. In its essence, a phase detector for GPS signals would consist of two (or more) antennas, placed at a known distance between each other. As the signal from a specific SV (Space Vehicle) comes from a location on the visible sky, it will be travelling at constant, well-

known speed (the signal is electromagnetic). If we observe a single data point within this signal, it will "reach" antenna one at time t_0, and shortly after that - antenna two at time t_1. The difference $|t_0 - t_1|$ would allow us to estimate what was the difference between the two paths that this specific data point "travelled", or in other words the difference in the distance SV-antenna 1 and SV-antenna 2. By knowing this small difference, and the distance between the antennas, the system could calculate a rather accurate angle, from which the signal must be coming.

Based on the information given above, the following preliminary test procedure was developed. The goal of these tests is to prove the theoretical assumptions behind the phase detection approach, assess the effects of the different parameters involved in the test, validate its reliability and give specific and well-argued recommendations for the future stages of the "NoSpoof" project. In order to explain the process itself, we must also keep in mind that there is a strong relationship between the actions taken and the hardware setup used. The following charts depict the hardware and the work flow. The explanation given below also addresses both the hardware setup, and the logic of the test procedure.

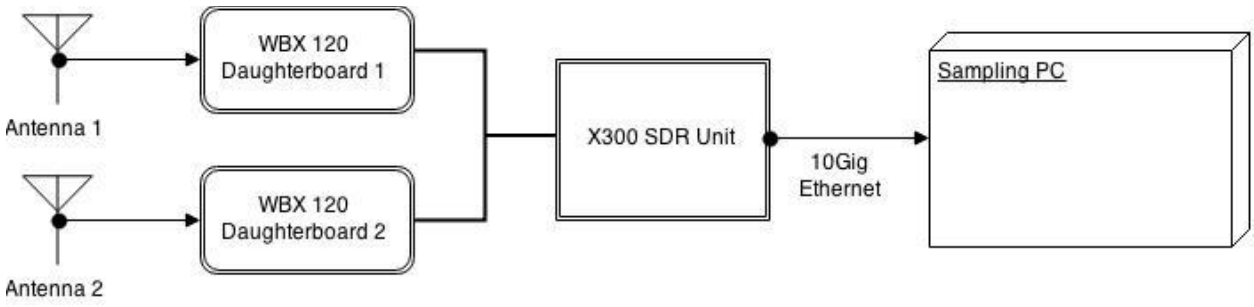

Figure 2. Hardware Setup Diagram.

The hardware setup (Fig. 2) consists of two major elements: The SDR (Software Defined Radio), and the Sampling PC. The first was purchased from Ettus Research [15], and it includes the following:

- 2 GPS Active Antennas [16]. These are responsible for the first step of the Initial stage - Signal Aquisition (Fig. 3A). Because the antennas are active, they include the first filter and amplifier.
- 2 WBX120 RF Daughterboards [17]. Each of the daughterboards allows the setup to correctly tune to a specified frequency. Following the specification of these daughterboards, we could tune to any signal from 25MHz to 2200MHz, with a bandwidth of 120 MHz. The tuning itself is done automatically, through the use of the dedicated UHD Software [18]. The board will convert the carrier signal from 1575.42MHz, to an IF (Intermediate Frequency) more suitable to the chosen ADC. The reason for this conversion lies behind the "Sampling theorem"[19]. Since the output of the mixer will have two major components - higher and lower centre frequencies (while preserving the Dopplers[10] and the C/A code encoded within the signal[9][20]), a band-pass filter[21] must be implemented (with the aim to preserve only the lower centre frequency).
- X300 SDR Unit [22]. The main board of the setup. It communicates with the daughterboards, performs the Digitalization through the use of the on-board ADCs [23]. The exact parameters of this step (sampling rate and quantization levels) were also altered during the later stages of the project, in order to define the most optimal situation.
- 10Gig Ethernet kit [24]. The main communication pipeline between the X300 and the Sampling PC. It provides an environment, which could support the data speeds involved

(initial tests were made at 100MS/s per channel, each sample consisting of two 16bit components).

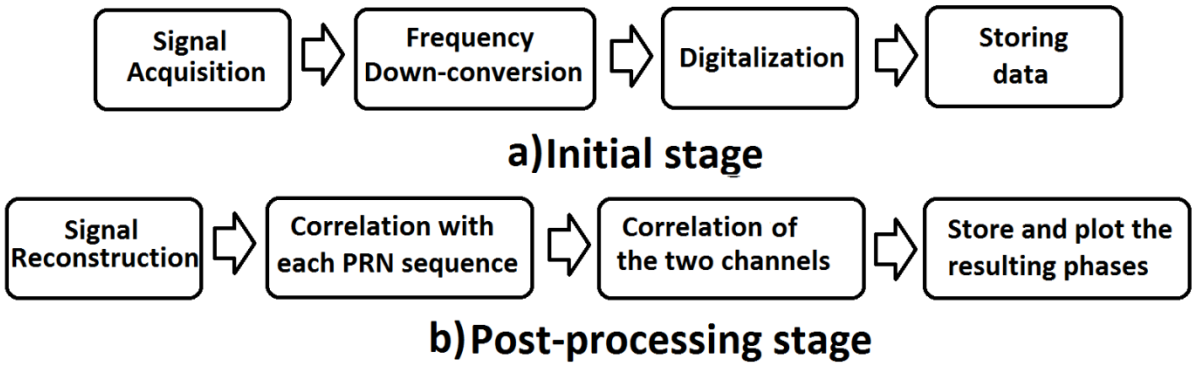

Figure 3. Full Flow Chart of the Phase Detection Algorithm

The communication between the PC and the X300 is handled by a combination of the UHD Software, provided by Ettus, and GNU Radio [25]. The flow-graph used for this specific test is shown in Fig. 4.

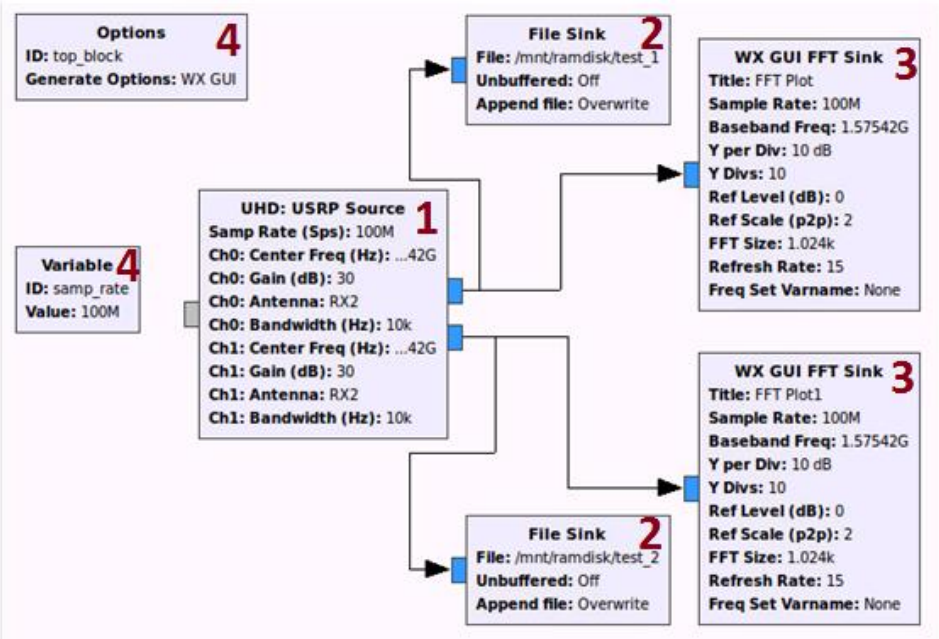

Figure 4. GNU-Radio Flow-Graph, used for the saving of the samples.

1. UHD: USRP Source - this unit represents the X300 board. It includes all of the parameters that need to be given to the hardware, in order to specify the sampling speed, number of channels, centre frequency etc. The output is two data streams, one for each of the antennas connected.
2. File Sinks - these components write the incoming data to a file. Since the writing speeds required to store all of the data are quite high (800 - 820 MB/s), we could not use ordinary Hard Drives or SSDs. The solution made was to store the data into a buffer drive, consisting of volatile memory (RAM). To do so, the following command:

   sudo mount -t tmpfs -o size=20480m tmpfs /mnt/ramdisk

   was used in Ubuntu 14.04, in order to create a local disk, consisting of 20GB of RAM memory. This local disk could maintain the writing speeds required for storing the data files (10GB each, containing up to ~12seconds of data).
3. WX GUI FFT Sink - these two components are used to visualize a constantly refreshing FFT (spectrum analysis) of the sampled data. Each frame uses a buffer of 1024 samples, and it's only used as a visual confirmation of the validity of the test, while the data is sampled and stored. It has no effect on the post-processing component of the project.
4. Internal Variables - used only for the work process of the Chart itself.

After defining the hardware, and the Initial Stage (Fig. 2 and Fig. 3A), the author focused on the post-processing component (Fig.3B). Using the binary file, created during the first stage, the program reconstructs the original signal (after the mixer and the filters). The first step is to execute a correlation with locally generated (precise) copies of the PRN sequences (*pseudorandom noise*). Since there are PRNs for each satellite, a copy of the result with each of the PRNs is stored for later use. The result of the correlation is precise alignment with reference

to the local oscillator, as well as removal of unwanted noise. This action is performed for both channels, but using the same reference clock signal. The final step is to calculate the phase between the signals at the two channels - this is done by a final correlation. All of the resulting phases (delays) are stored and later plotted. The plot should show different delays for the different satellites (due to different angle-of-arrival) if the signals tracked are genuine. In the case of a spoofer, since the source is at the same point, the delays of each satellite must have the same value. In Appendices A and B we present a brief list of explanations of the hardware, and the PC component list respectively.

CHAPTER 4
RESEARCH DESIGN

The following chapter contains information about the process of executing the conceptual model, described in the previous part. The test procedure and validation of the project will be split into several different stages, all concerning key milestones.

The first step of the preparation of the equipment consists of configuring the setup. This involved several key milestones: configuration of the Ettus hardware, the communication protocol (10Gig Ethernet) and the file storage. To accomplish the first, the quick installation guide provided by the manufacturer was followed [26]. This involved uploading the latest FPGA (Field programmable gate array)[27] image - a piece of firmware, responsible for the work of the X300 and the data streaming[28]. Once this was accomplished, the Ethernet connection was established, and two main parameters regarding the maximum data rate were changed:

sudo ifconfig eth0 mtu 9000 - changing the Maximum Transmission Unit (MTU) allows for bigger package sizes to be sent over the Ethernet frame. The default value is 1500, whereas the 10 Gig Ethernet requires 9000 (translating to 6 times larger frames).

sudo sysctl -w net.core.rmem_max=33554432 - changing the maximum buffer size while using UDP protocol. The solution to the file storage was to store the data into a buffer drive, consisting of volatile memory (RAM). To do so, the following command:

sudo mount -t tmpfs -o size=20480m tmpfs /mnt/ramdisk

was used in Ubuntu 14.04, in order to create a local disk, consisting of 20GB of RAM memory. This local disk could maintain the writing speeds required for storing the data files (10GB each, containing up to ~12seconds of data).

After the system was operational, and the communication protocols allowed higher data transfer rates, the initial testing of the equipment within a controlled (laboratory) environment were made. A signal generator [29] was used to simulate a signal at 1.5 GHz. Since the signal had relatively high power ratio (roughly -50dBm), it was clearly visible on both RF inputs.

The next stage was to place the GPS antennas on the roof of the Astron building (Fig. 5). Because of the location of the facility, this option was chosen over placing the antennas on ground level because of the presence of dense forest in the vicinity. The present setup would allow a higher percentage of visible clear sky. The final step of the hardware preparation was to attach Biased Tees [30] to the antennas. The purpose of those components is to provide 5V DC power over the SMA connector to the antenna. If they are not attached, the X300 cannot provide power to the active GPS antennas. The author strongly recommends that the tests described above are conducted in order to ensure that all equipment functions properly.

Figure 5. Antenna placement on top of the roof of Astron building, Dwingeloo

After validating that the test equipment is operational, the Matlab algorithm was prepared. The software used in this project is developed around some of the functionality of the

software GPS receiver, provided with Kai Borre's "*A Software-Defined GPS and Galileo Receiver*" (*ISBN 978-0-8176-4540-3*). The software was well documented, and the fact that it is quite modular made it easier to extract the needed sections, and alter them. Since the goal of the project is only the monitoring of the physical propagation of the GPS signal, the parts of the code which are responsible for the actual navigation calculations were removed. What was kept from the original code, was the signal acquisition component - responsible for the PRN correlation. The main changes that had to be done to these parts were connected to the fact that the sampling hardware used by his team was performing only amplitude sampling, while the Ettus Research SDR samples in complex form. This means that while the original hardware had one 8-bit number per sample, the X300 provides two 16-bit numbers (a real and an imaginary component). In addition, the X300 setup gives the samples with respect to the baseband - meaning that within the code itself, we could use the centre frequency (1575, 42MHz) directly. The last two modifications were the introduction of a second signal from the second antenna and enlarging the correlation samples from 5ms (5 repetitions of the C/A code) to various lengths between 20 and 50ms (increasing the accuracy of the results). The outcome of changing the correlation sample lengths is discussed within the results section of this paper.

Once both the hardware and the software components were finalized, initial tests of the complete system were made. Those involved small samples (up to 10 seconds) being taken and ran through the algorithms. The end result of the algorithm is the phase of each signal, with respect to the locally generated C/A codes. During the first tests a major problem with the hardware was observed, involving increased noise floor on one of the channels. In addition, the noise floor within the samples was constantly alternating. For more information, please refer to Chapter 5 *Deviations and Delays*.

Due to the fact that using a spoofer device is illegal in the Netherlands, recreating a spoofing attack directly is impossible. The alternative method of evaluating the system is to show with what accuracy and consistency can it place the direction of the incoming signals. For this purpose, test data was taken during different parts of the day, and then the constellation, built from the resulting C/A phase delays, were compared to the actual locations of the GPS satellites at this point of time. The conclusions and validity assessment are based on the evaluation of the consistency and the accuracy observed during those tests.

CHAPTER 5
DEVIATIONS AND DELAYS

As mentioned earlier in Chapter 4, a major problem with the Ettus hardware was observed during the first tests made with the finalized setup. While sampling the GPS data, the two FFT plots, used for visual confirmation of the presence of a signal, showed an unexpected anomaly on one of the signals. (Fig. 6). In order to show the presence of the increased noise floor more clearly, a test was done with the inputs connected to fixed 50Ω impedance (Fig. 7). The problem observed was a difference between the noise levels in the two channels. Since this noise is classified as Gaussian, this should not be the case.

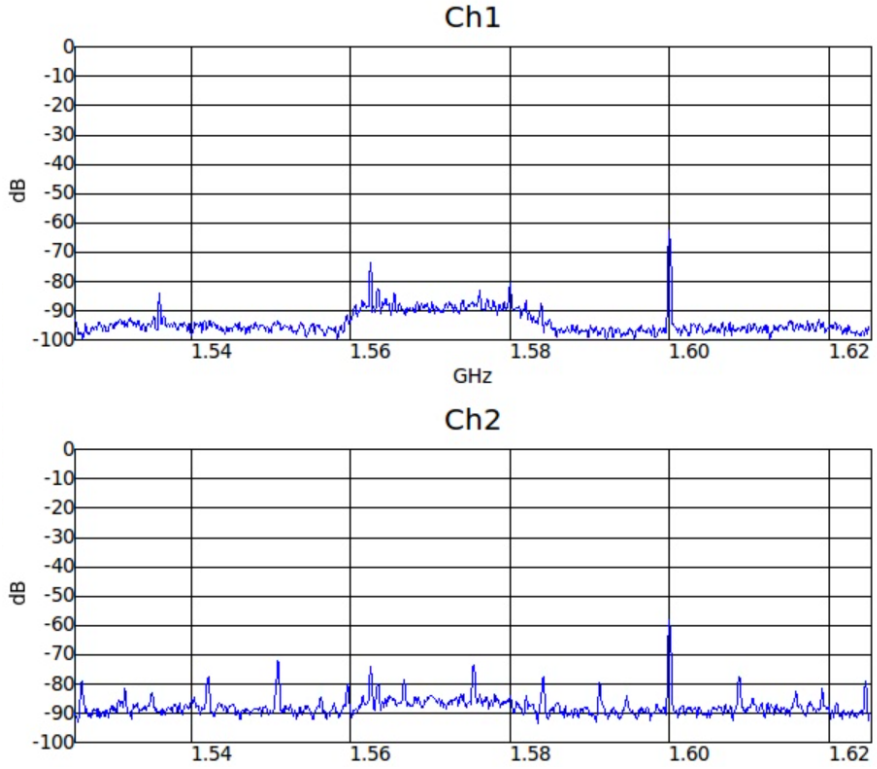

Figure 6. Raised Noise Presence while sampling GPS L1 Band at 100MS/s.

As can be seen above, Channel 1 shows clearly the captured signal between ~1.56GHz and ~1.585GHz. The average level of the noise floor is roughly -96dB, with the signal around 10dB above the noise floor (excluding individual peaks). On the other hand, Channel 2 has almost no clearly visible signal acquired. There is a slight increase in the power level between 1.56GHz and 1.58GHz, but significantly lower in comparison. The average noise floor is about -90dB, with many spurs visible across the bandwidth that were not present in the spectrum of Channel 1. It must be noted that since the input signal is identical, such a significant difference in quality of the data is not expected. The frequency spurs are also artifacts that should not be observed in the case of properly functioning system.

In order to locate the problem causing this anomaly, a few simple tests were performed. The first option explored was the presence of interference, caused because of a malfunction of one of the antennas. In order to check this, the two antennas were swapped - Antenna A, originally connected to Daughterboard A, was now connected to Daughterboard B, and vice versa. During the sampling it was obvious that this was not the cause, because the noise floor was again elevated with about 10dBm in the second channel (Fig 7). As can be seen below, Channel 2 clearly shows the unexpected frequency spurs (the spectrum shown on Channel 1 is closer to what should be expected from the equipment used in this setup).

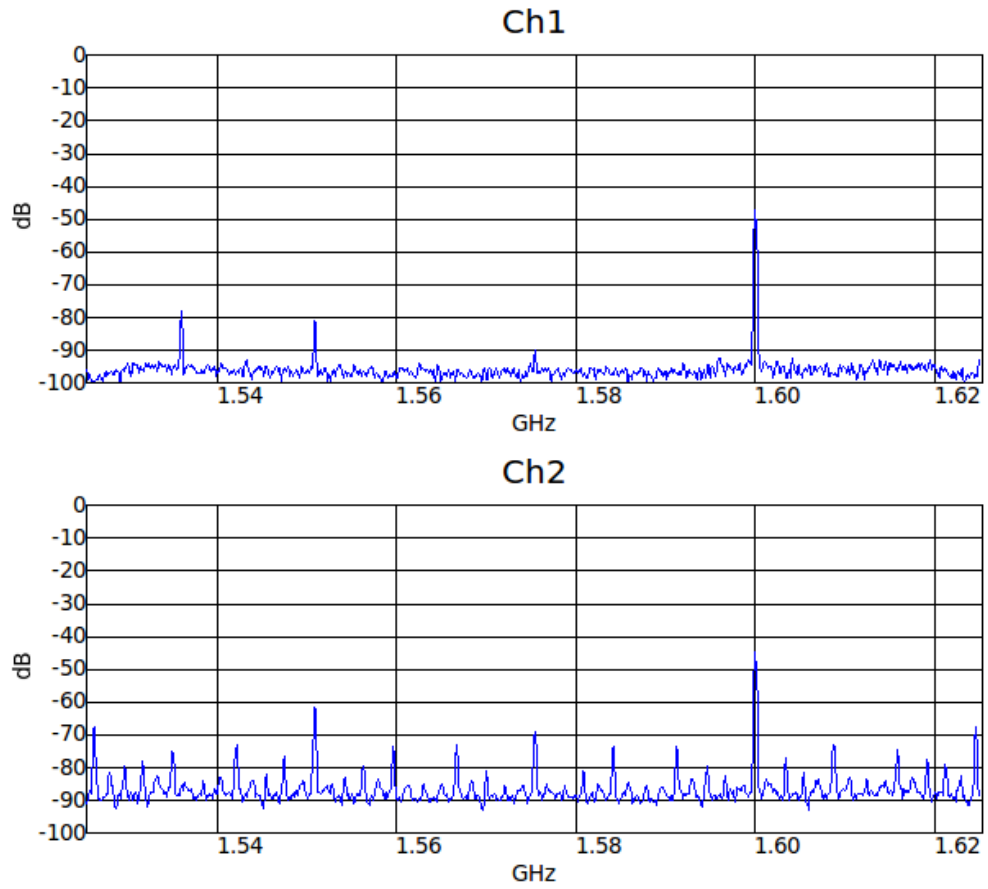

Figure 7. Raised Noise Presence while sampling a closed input (with 50Ω).

The second cause that was checked was a faulty Daughterboard. Similar to the antennas, the X300 was disassembled and Daughterboard A was placed on channel 2, while originally it was on channel 1, and vice versa. After that, a sample was taken, and it was confirmed that the Daughterboards were also not a problem - there was no change in the anomaly. At this point, the technical support at Ettus was contacted about the problem.

After thorough analysis on the situation and the equipment, it was concluded that the X300 unit has be shipped back to Ettus Research for repairs or replacement. Due to the procedures involved (shipping procedures controlled by National Instruments), it became clear that the unit will not be fixed before the deadline of the project.

ALTERNATIVE PROCEDURE DESCRIPTION

After it was concluded that the channel B of the X300 unit was unusable, two options were considered as alternative solutions. The first one involved purchasing of new equipment - an option quickly discarded due to many reasons, some of which are budget constraints, time required for the process of the invoice, delivery, setup and even the small probability of receiving another defective unit. The second option is to modify the test procedure, so that the working channel 1 could be used to sample both of the incoming signals.

The WBX120 Daughterboard, attached to channel A of the X300, has 2 RF inputs - RX/TX and RX2. Under normal conditions, each one of them can be used to sample an incoming signal. The drawback of the RF card is that it is impossible to sample both inputs simultaneously. On the other hand, it is possible to (as an example) sample the RX2 input for 10 seconds, save the data, and then perform the same action for the RX/TX for another 10 seconds. While both of the files will include valid GPS data, they will not be usable for the test described earlier, since between the two samples there is an unknown offset, which will lead to an unknown difference of the phase of each C/A code. A solution to this problem was found within the GNU Radio software, used for the sampling. Instead of using ordinary "File Sink" function (Fig. 4, element 2), the "Metadata File Sink" function was used. The main difference is that apart from saving the data in a binary file (just like the ordinary "File Sink"), it generates a separate header file, containing secondary information about the contents of the sample. The most important parameter, saved in this file, is the timestamp of the first sample within the data file. What this means is that by knowing the sampling rate, and the timestamp of the first sample of each file, the two files can be artificially realigned. This is possible due to the fact that there will be less than one minute between the two signals - the change of the GPS satellites, which can occur for

this period, is magnitudes smaller than the resolution of the equipment used. The realignment itself is done by shifting one of the signals with a number of samples, found from the difference of the timestamps. For example, if header one has timestamp of *88585.205185726*, and header two has timestamp of *88610.188349187*, we will know that there were 24.983163461 seconds between the start of the two samples. Each C/A code is repeated every 1ms, and at 100MS/s this translates to 100 000 samples. What this means, in this particular case, is that the data in file 1 must be shifted with 16346 samples. At this time, the software component (executing the phase detection within the two files) can proceed without any changes from the initial test setup.

While on theory the error introduced by such a setup will be smaller than the resolution used, in practice it becomes more complicated. Since the timestamp of the samples is generated by the X300, it uses the same reference clock as the ADC of the unit. What this means is that the resolution of the timestamp is (1s) / (200MS/s) = 5ns, while the resolution of the sampled signal is (1s) / (100MS/s) = 10ns. In the worst case, there may be 1 sample shift between the samples that cannot be picked by the setup. There are many other factors that can introduce both time delays and rounding issues, such as difference between the quality of materials used in the RX2 and the TX/RX input lines, clock cycle skips (known issue with the reference clock of the Ettus Research boards, encountered by S&T researchers) and small, unnoticed delays due to start-up and communication lag between the sampling PC and the Ettus board.

CHAPTER 6

RESULTS

After covering the preparations of the test setup, the reasons behind the change of it, and its last variation, a series of 5 to 10 second samples were taken, 30 minutes apart. Apart from the samples and the respective header files, a virtual map of the currently visible GPS satellites was taken upon each sampling. This map was taken from "https://in-the-sky.org/index.php", a web-based service which provides easy tracking of multitude of satellites. The test samples shown below were taken at Astron rooftop (latitude: 52.81, longitude:6.40).

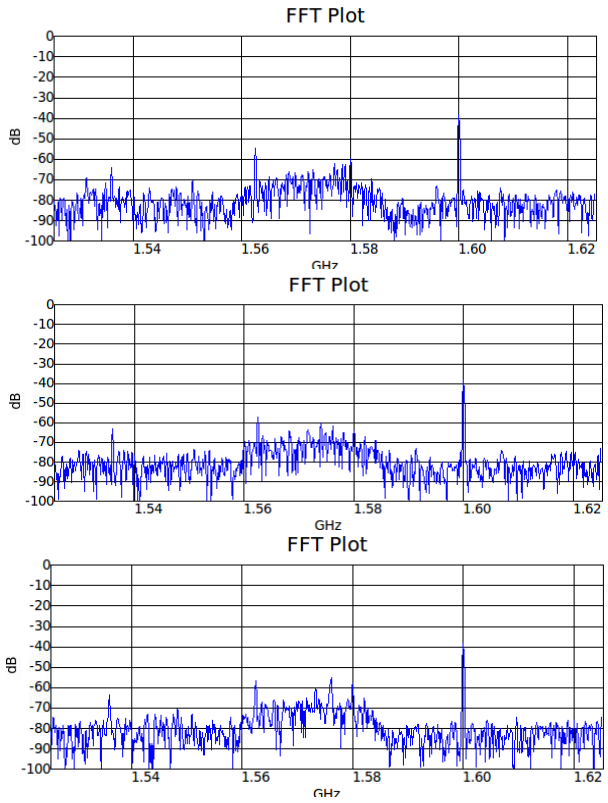

Figure 8. FFT of the raw signal (antenna A).

The first visual results on the captured results was a quick FFT plot, showing the frequency spectrum of the incoming stream. As can be seen in Fig. 8, the signal from antenna A is clear, with little variation over time. Fig. 8 displays three successive states of the FFT. It is clear that the GPS signals from the different satellites are between 1.56 and 1.58GHz, hence the increased signal power in the range. The average noise floor is about 85dB, while the average signal power observed is roughly 75dB.

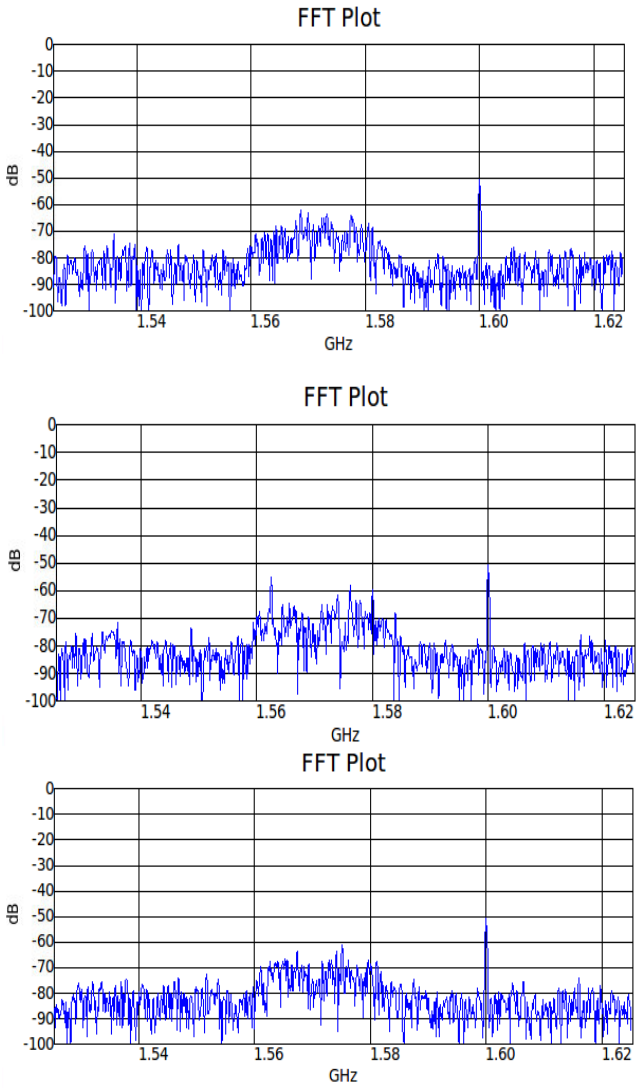

Figure 9. FFT of the raw signal (antenna B)

The signal captured from antenna B (Fig. 9) bears strong resemblance to the one shown on Figure 8, as initially expected. After short observation however, it became clear that while the sample from antenna A had a stable signal captured within the expected limits, the one for the second channel had slight variation in the power. In addition, a relatively clear and strong peak at around 1.562GHz would appear and disappear periodically (the change was visible with the naked eye). Apart from this change, the signal had little fluctuation, with a noise floor at roughly 85dB and signal power at an average of 75dB. (identical to the one observed with antenna A).

During the sampling of the two data samples, different channel gains were tested. It was quickly observed that for both channels there was no significant difference in the difference between the noise floor and the GPS signal - as long as the gain was above 10dB, the difference

was kept at roughly 10dB. As the gain increased to its maximum value (recommended 35dB), the noise floor and the signal were increasing in power, but not the difference between them. If the channel gain was set to a value between 0 and 10 dB, only the noise was visible. After the visual results were gathered, the following information was extracted from the two header files:

	Header file A	Header file B
Sampling frequency	**100000000**	**100000000**
Centre frequency	**1.5754e+09**	**1.5754e+09**
Timestamp	**8637.181137279**	**8651.125710711**

Table 2 Header file results.

Once the quick confirmation from the header file contents is made, the files were run through the algorithm. The following list of satellites was acquired:

File A	File B	Available satellites
PRN 12	**PRN 12**	**PRN 12**
PRN 14	**PRN 17**	**PRN 14**
PRN 17	**PRN 24**	**PRN 17**
PRN 24	**PRN 25**	**PRN 24**
		PRN 25
		PRN 02

Table 3 List of acquired satellites

The list of available satellites given in Table 3 is based on the information taken from the virtual map of the currently visible GPS satellites mentioned earlier in this chapter. The acquisition was performed 10 times in each file, with the use of a 20ms correlation sample. The offset between the samples used is a precise amount of C/A codes, in order to avoid introducing

additional offset. The average of the C/A code offset for each observed PRN are given in the tables below (Table 4 and Table 5):

Antenna A	C/A Phase average (rounded)
PRN 12	**41158**
PRN 14	**63578**
PRN 17	**63386**
PRN 24	**94999**

Table 4 C/A Phase for Antenna A

Antenna B	C/A Phase average (rounded)
PRN 12	**35040**
PRN 17	**60297**
PRN 24	**90757**
PRN 25	**42711**

Table 5 C/A Phase for Antenna B

The code phase presented in Tables 4 and 5 are given in number of samples. Given that the sampling rate used is 100MS/s, each sample would be equal to 10ns. (sample period = 1second / sampling frequency).

Taking into consideration the offset between the first samples of the two files (taken from the header file information, displayed in Table 2), the following phase difference values were obtained (between the signals acquired by antenna A and antenna B). The "Nan" values are because of the fact that PRN 14 and PRN 25 were not detected in both files.

PRN	Phase Difference
12	36583
14	Nan
17	39567
24	38414
25	Nan

Table 6 Phase difference between antenna A and antenna B

In order to detect if the system can detect any change within the position of the satellites over time, the same experiment was performed one hour later (Initial test was performed on 04.06.2015, at 14:00, with the second data set taken at 15:00). Given below are the gathered results.

	Header file A	Header file B
Sampling frequency	**100000000**	**100000000**
Centre frequency	**1.5754e+09**	**1.5754e+09**
Timestamp	**12318.151395887**	**12334. 187033236**

Table 7 Header files, second data pair

File A	File B	Available satellites
PRN 6	**PRN 12**	**PRN 02**
PRN 12	**PRN 14**	**PRN 12**
PRN 14	**PRN 22**	**PRN 14**
PRN 23	**PRN 25**	**PRN 24**
PRN 29	**PRN 29**	**PRN 25**
		PRN 29
		PRN 31

Table 8 List of acquired satellites, second data pair

Tables 7 and 8 present the check of the header files, and the list of acquired satellites. Because of the one hour difference, there is a slight modification in the visible satellites. Given below are the C/A phases calculated for the new data pair.

Antenna A	C/A Phase average (rounded)
PRN 6	**9679**
PRN 12	**68867**
PRN 14	**95303**
PRN 23	**81065**
PRN 29	**42725**

Table 9 C/A Phase for Antenna A, second pair

Antenna B	C/A Phase average (rounded)
PRN 12	**29409**
PRN 14	**55373**
PRN 22	**90864**
PRN 25	**50799**
PRN 29	**98435**

Table 10 C/A Phase for Antenna B, second pair 63735

PRN	Phase Difference
6	**Nan**
12	**96807**
14	**96335**
22	**NAN**
23	**Nan**
25	**NAN**
29	**91976**

Table 11 Phase difference between Antenna A and Antenna B, second pair

Since in this particular test only PRN 12 is detected in all situations, we can say that the shift observed by the system over a period of one hour is 60244 samples.

CHAPTER 7
VALIDATION

After analyzing the data shown in the previous chapter, it is clear that results obtained do not match the expectations. Theoretically, the movement over time of a satellite will not generate a larger path difference between the two antennas than the actual difference between the antennas. For example, if the antennas are 5 meters away from each other, the path from the satellite to both antennas will not differ more than 5 meters. Following the situations seen in the previous chapter, the distance between the antennas must be magnitudes above the 5 meters, defined in the test setup. In addition, the phase shift difference, occurring over time, should not change as drastically as observed, mainly due to the fact that satellites have high altitude, and the time it takes for one to make a complete passing over a given point on the planet is in most cases significantly higher than a few hours (strongly dependant on the inclination at which the satellite passes over the observer. The above statement is made for satellites that would align within at least 45 degrees from the Normal. In order to assess this behaviour, each component of the system must be validated.

SIGNAL ACQUISITION ASSESSMENT

Shortly after the test shown in the results section, there was a strong anomaly detected during signal acquisition. While the data saved using antenna B remained of the same quality, antenna A observed major interferences. The FFT chart, used as visual check, showed slight variation in the level of the acquired signal. In addition, the noise level showed periodic rises in power, not fitting the expected behaviour. In order to assess this situation, the antenna was attached to a spectrum analyser, which confirmed the observations. Because of the higher quality

of the channels of the analyser, it was observed that the signal power was pulsating - a "breathing" effect, most commonly related to the effects of multipathing. The latter is the resulting effect where a signal is bounced off different surfaces, without losing too much of its power. After a certain number of bounces, the signal reaches the antenna with a certain delay. In most cases, this results in the bounced signal interfering with the original signal, gradually rising and lowering its power. Because of the high number of satellites signals, and the shape of the roof where the antennas were mounted, it is highly probable that this is indeed the reason why one of the antennas experiences significantly worse interference then the other. The best solution would be to replace the antennas with better quality antennas, capable of isolating bounced-off signals.

All of the data used for the validation of the setup, was recorded on the 3rd and 4th of June. It includes assessment of the effects of different sampling rates, as well as overall satellite acquisition performance.

SAMPLING RATE VALIDATION.

Reflecting back on the conceptual model, higher sampling rates ultimately translate to better precision and resolution with respect to the phase detection itself. During testing, the following results were gathered for sampling speeds of 100MS/s and 200MS/s:

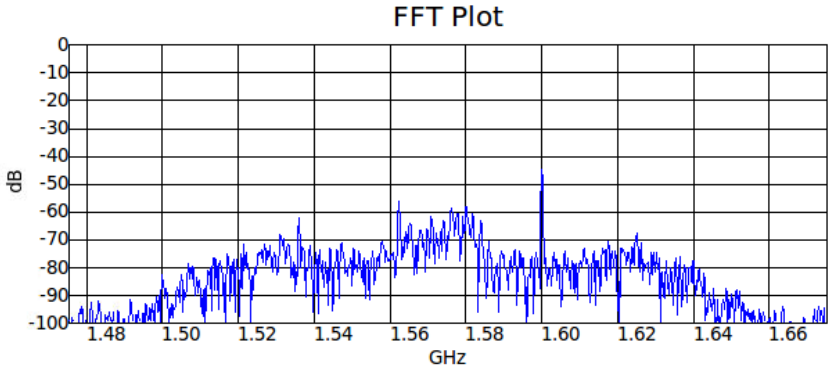

Figure 10. Raw signal at 200MS/s

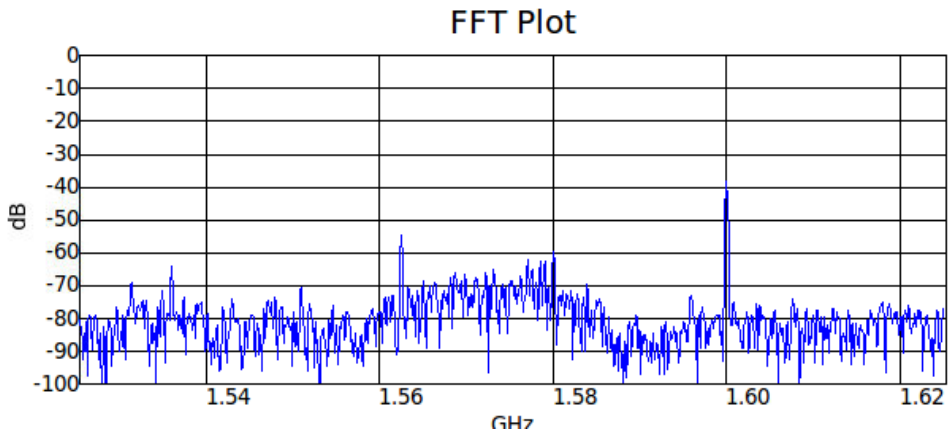

Figure 11. Raw signal at 100MS/s

While Figure 10 and 11 have some definite differences (bandwidth due to sampling rate), it is clear on both that the interval with the highest observed power is between 1.562GHz and 1.60Ghz, including the L1 centre frequency. While Figure 9 shows that with a higher sampling rate there is a need for better filtering of the resulting data, the satellite acquisition performed on the same samples showed that when the higher sampling rate is used, the errors produced due to rounding or bit shifts within the generation of the C/A codes on the local machine can negatively affect the performance of the correlation, which is responsible for the calculation of the code phase. The two tables below show number of acquired PRNs versus the number of available satellites, for the two sampling rates discussed

Number of acquired satellites at 200 MS/s	Number of acquired satellites at 100MS/s
2 out of 7	**6 out of 8**
3 out of 8	**4 out of 10**
2 out of 6	**5 out of 8**
1 out of 9	**5 out of 8**
2 out of 6	**6 out of 7**

Table 12 Number of acquired satellites - comparison between 200MS/s and 100 MS/s

In spite of requiring more sophisticated equipment to sustain the data flow, the higher sample rates also appear to reduce the performance of the satellite acquisition loop. While this could be avoided with the creation of a more accurate system, without any averaging errors (most significant current error appears when calculating the length of each sample, always rounded to higher value), the lower 100MS/s speed provides sufficient reliability, and results in better satellite recognition.

In order to complete the evaluation of the system's performance, the error accuracy of the Phase detection loop must also be evaluated. Since there is no reference code phase values to compare to and calculate an error, only the dispersion and the consistency of the data can be evaluated. Given below are the full results from the sample, presented earlier in Chapter 6. (Table 13)

PRN	C/A phase measured
12	41159
12	41152
12	41160
12	41156
12	41157
12	41154
12	41158
12	41171
12	41153
14	63591
14	63592

PRN	C/A phase measured
14	63583
14	63567
14	63559
17	63372
17	63378
17	63383
17	63395
17	63401
21	8545
24	94998
24	94991
24	95007

Table 13 Full acquisition list, Antenna A at 100MS/s

The mean values for the phase measured can be found in Table 4. Evaluating the Standard deviations for such small datasets will not be beneficial to the validation, especially in the case of PRN 24 or PRN 21. However, it can be seen that the values observed in this specific

dataset vary rather inconsistently. This disallows the opportunity of wrong calculation of the step between the samples, taken from each file. If the number of bytes that need to be skipped to reach each new sample is wrong, the offset will be gradually build over time. This variation in the measurements can be caused by several different phenomena - from the multipathing - a single strong refraction can create false correlation peak; or even the results of altering Doppler shift - while the change is rather small in comparison to the carrier frequency, if changed it could lead to small offsets within the measurements. Last but not least, the rounding errors, mentioned earlier, could also build up over time, adding to a significant number the longer the algorithm is used.

CHAPTER 8
CONCLUSION

During the development of the system it became apparent that achieving the ultimate goal of the research - live spoofing detection and filtration, will prove to be more difficult than expected. While the theory suggests a straightforward and direct approach, there are many parameters that need to be taken into consideration while creating such a system. By following the SDR approach of recreating the GPS receiver on software level, proposed by Kai Borre[9], a widely customizable and high-performing test setup was created. Despite the delays caused due to malfunction in the hardware, the procedure was altered in ways that keep the main principle of the test identical to the original plan. It must also be taken into account that while sampling rate does mean higher resolution of the gathered data and provides the possibility to decrease the physical distance between the antennas, it does reduce the quality of the satellite acquisition function. It is also recommended to use sampling speeds of around 100MS/s, in order to avoid overloading of the communication protocol or overcomplicating the post-processing unit.

Other methods of increasing the accuracy of the phase detection were considered during the research phase of the project. Knowing the precise duration of the C/A code, and its behaviour, one could use bit prediction in order to make estimations about the exact timestamp of each edge, part of the C/A code signal. This could allow the user to perform correlation sweeps with steps smaller then a single sample. Similar approach is the use of Quadratic Interpolation, in order to predict the peaks of a sine wave. To do so, one must first focus on the Carrier Signal, not only on the C/A code. Those two methods were not validated within the research itself due to their high complexity, and the unexpected delays, caused by malfunctions.

Nevertheless, based on the theoretical research done in the first phases of the project, it is recommended to perform a thorough research on the stability of bit prediction, since it is the most promising way of improving the code phase detection.

As an outcome of the research, it can be stated that the accuracy of the current setup is insufficient to determine precisely the C/A code phase with antenna distance smaller than 5 meters. However, it is validated that by using sampling speeds of 100MS/s, and widely used PRN correlation techniques, it is possible to detect the satellites currently visible, and measure their C/A code phases with small variation. While by using two antennas it is possible to make phase difference calculations, it is insufficient to map the current positions of the satellites (needed for online spoofing detection). Nevertheless, the created setup provides a good base for further development of phase-based spoofing detection algorithms. By taking into consideration the recommendations found in the next chapter, it can be expected that creating a more stable and accurate Code phase detection system is a feasible project.

CHAPTER 9
RECOMMENDATIONS

In order to improve the results gathered and analysed in this report, some equipment could be altered or changed. Given below are a number of recommendations which could lead to such an improvement.

First of all, a major improvement to the results of the phase detection algorithm would be to introduce additional signals. To do so, more antennas and equipment capable of handling three or more inputs must be purchased. The benefit of such a setup is that it would allow basic beam forming techniques. If the antennas are placed in a known geometry, the actual location of the satellite can be established in the 3D space. What that means to the output of the system is that it will be easier to distinguish between groups of satellites located in close vicinity and a possible spoofer. Another parameter that can be used to determine if a signal is genuine is the "height" of the source location with respect to the horizon. In most pratical cases the spoofer device is aligned with the horizon (due to the fact that it needs to be in close proximity, the spoofing system will be on a similar horizontal level), while the satellites are not.

In addition, during the research related to the hardware used, a plausible way of increasing the performance and detection chances of the current setup would be to decrease the quantisation level. Currently, the device is taking complex measurements of total 32 bits/sample. Using the default FPGA image (provided by Ettus) it is impossible to reduce the size of the measurements. However, using a technique known as RF Network-On-Chip [31], the device could be allowed to swap to total of 16 bits/sample. This would theoretically double the maximum sampling rate that can be used. In other words by using 200 MS/s instead of 100 MS/s

we could either increase the precision that can be achieved in the phase detection or allow us to decrease the distance between the antennas while keeping the current accuracy. However, with respect to the findings about the effects of high sampling rates on the stability of the acquisition function, this must be avoided until a better correlation technique is applied.

Another benefit of the reduction of the quantization levels would be the fact that with 16 bit samples, the system becomes highly sensible to even the lowest noise levels. By lowering the quantization levels, the resulting rounding can lead to a more robust acquisition performance, or even to a more stable phase detection output.

One major improvement for anyone who ventures into the improvement of this setup, would be to utilize the many possibility the X300 offers. While this is a sophisticated piece of equipment, offering a wide variety of options and freedom of operation, it is currently used for nothing more than an expensive multi-channel analogue to digital converter. By programming the FPGA within the unit, the need for excessive data storage can be removed, resulting in smoother running system, as well as allowing to make more direct attempt to create an online spoofing detection system.

REFERENCES

[1] Science & Technology, [Online]. Available: http://www.stcorp.nl/. [Accessed 04 February 2015].

[2] ESA, "What Is Galileo?," 27 June 2014. [Online]. Available: http://www.esa.int/Our_Activities/Navigation/The_future_-_Galileo/What_is_Galileo. [Accessed 04 February 2015].

[3] T. E. Humphreys, "Assessing the Spoofing Threat:," September 2008.

[4] L. Scott, "Spoofs, Proofs & Jamming," 2012. [Online]. Available: http://www.insidegnss.com/auto/2012-sepoct-Scott.pdf.

[5] A. El-Rabbany, "GPS Details," in *Introduction to GPS: The Global Positioning System*.

[6] M. L. Psiaki, "GNSS Lies, GNSS Truth," *GPS World.*

[7] GPS World, "What's New in GNSS Simulation," 8 May 2013. [Online]. Available: http://gpsworld.com/whats-new-in-gnss-simulation/. [Accessed 04 02 2015].

[8] "Inside GNSS," Gibbons Media and Research, LLC, [Online]. Available: http://www.insidegnss.com/. [Accessed 04 February 2015].

[9] K. Borre, " Chapter 2 : GPS Signal," in *A Software-Defined GPS and Galileo Receiver*.

[10] Wikipedia, "Doppler Effect," January 2008. [Online]. Available: http://en.wikipedia.org/wiki/Doppler_effect. [Accessed 04 February 2015].

[11] M. L. Psiaki, "GNSS Lies, GNSS Truth," *GPS World,* 2014.

[12] C. J. Wullems, "A Spoofing Detection Method for Civilian L1 GPS and the E1-B Galileo Safety of Life Service," 31 October 2010.

[13] A. Broumandan, J. Nielsen, G. Lacapelle, A.J.Jahromi, "GPS Vulnerability to Spoofing Threats and a Review of Antispoofing Techniques," 29 May 2012. [Online]. Available: http://www.hindawi.com/journals/ijno/2012/127072/. [Accessed 15 06 2015].

[14] Wikipedia, "Ionosphere," [Online]. Available: https://en.wikipedia.org/wiki/Ionosphere. [Accessed 06 06 2015].

[15] Ettus Research, "Ettus Research Home Page," National Instruments, [Online]. Available:

http://www.ettus.com/. [Accessed 20 05 2015].

[16] Ettus Research, “5-Volt Active GPS Antenna,” [Online]. Available: http://www.ettus.com/product/details/GPS-ANT-5V. [Accessed 15 03 2015].

[17] Ettus Research, “WBX 50-2200 MHz Rx/Tx (120 MHz, X Series only),” [Online]. Available: http://www.ettus.com/product/details/WBX120. [Accessed 05 February 2015].

[18] Ettus Research, “USRP Hardware Driver software (UHD),” [Online]. Available: http://code.ettus.com/redmine/ettus/projects/uhd/wiki.

[19] “Nyquist-Shannon sampling theorem,” Wikipedia, 04 April 2015. [Online]. [Accessed 07 April 2015].

[20] E. Kaplan, “Chapter 5 : Satellite Signal Acquisition, Tracking, and Data Demodulation,” in *Understanding GPS Principles and Applications 2nd Ed.*.

[21] K. Borre, “Chapter 4: GNSS Antennas and Front Ends,” in *A Software-Defined GPS and Galileo Receiver*.

[22] Ettus Research, “USRP X300 and X310,” [Online]. Available: http://www.ettus.com/content/files/X300_X310_Spec_Sheet.pdf. [Accessed 05 February 2015].

[23] Texas Instruments, “ADS62p48 datasheet,” April 2009. [Online]. Available: http://www.ti.com/lit/ds/symlink/ads62p48.pdf. [Accessed 05 February 2015].

[24] Intel, “Intel® Ethernet Converged Network Adapter X520 Product Family,” Intel Corporation, [Online]. Available: http://www.intel.com/content/www/us/en/network-adapters/converged-network-adapters/ethernet-x520.html. [Accessed 20 05 2015].

[25] GNU Radio, “Welcome to GNU Radio,” [Online]. Available: http://gnuradio.org/redmine/projects/gnuradio/wiki. [Accessed 15 03 2015].

[26] Ettus Research, “Quick Start X Series,” National Instruments, [Online]. Available: http://www.ettus.com/content/files/kb/xseries-quick-start.pdf. [Accessed 10 05 2015].

[27] Wikipedia, “Field-programmable gate array,” [Online]. Available: https://en.wikipedia.org/wiki/Field-programmable_gate_array. [Accessed 10 05 2015].

[28] Ettus Research, “USRP Hardware Driver and USRP Manual,” National Instruments, [Online]. Available: http://files.ettus.com/manual/page_images.html. [Accessed 16 06

2015].

[29] Rohde-Schwarz, “Signal Generators,” Rohde-Schwarz, [Online]. Available: http://www.rohde-schwarz.com/en/products/test-measurement/signal-generators/pg_overview_63667.html. [Accessed 15 05 2015].

[30] Wikipedia, “Bias tee,” [Online]. Available: http://en.wikipedia.org/wiki/Bias_tee. [Accessed 15 03 2015].

[31] Ettus Research, “RFNoC: Specification,” National Instruments, [Online]. Available: https://github.com/EttusResearch/uhd/wiki/RFNoC:--Specification. [Accessed 25 05 2015].

[32] MathWorks, “Matlab,” [Online]. Available: http://nl.mathworks.com/products/matlab/.

APPENDIX A:
BRIEF TEST SETUP EXPLANATION

The hardware required for the operation of the test setup includes 3 components:

- ***USRP X300***[22] **+ *2x WBX120*** [17]**-** The main sampling unit. The X300 represents a single-package Software Defined Radio unit. It provides wide frequency tuning, as well as high sampling rates (up to 200MS/s)[23]. The WBX120 daughterboards allow the main board to sample two input lines in parallel, while keeping the high sampling rates.
- ***RF front-end* -** this includes two active GPS antennas [16] (with center frequency 1575.42MHz) and two Bias Tees [30] (needed to "power" the antenna).
- ***Processing unit* -** For the use of this project, a high-end PC was put together. All of the components were chosen in order to allow storage of the large amounts of data, coming from the sampling unit. The full specifications of the build can be found in Appendix B.

The software required for the operation of the test setup includes 3 components:

- ***USRP Hardware Driver™ software (UHD™)*** [18] **-** Provides the software interface needed for the processing unit to connect to the sampling unit. Because of the high speeds required, an optic interface was used.
- ***GNU Radio*** [25] **-** Free and open-source software development toolkit that provides signal processing blocks to implement software radios. It supports both wireless communications research and real-world adio systems.
- ***Phase detection algorithms* -** A Matlab[32] script, performing signal extraction and directional calculations.

APPENDIX B:

SAMPLING COMPUTER COMPONENTS LIST

1. Motherboard: Gigabyte GA-Z97X-UD3H (109.90 EUR)

http://azerty.nl/0-5602-688182/gigabyte-z97-d3h.html

2. RAM: 32GB (4x8) Corsair Vengeance Pro @1600MHz (291.82 EUR)

http://azerty.nl/0-750-627379/corsair-vengeance-pro-series.html

3. CPU: Intel i7-4790k @4.0 GHz (quad-core) (335.90 EUR)

http://azerty.nl/0-5656-707629/intel-core-i7-4790k-4-ghz.html

4. CPU cooler: Cooler Master Hyper TX 3 EVO (22,91 EUR)

http://azerty.nl/0-976-455749/cooler-master-hyper-tx-3-evo-k.html

5. Power Supply Unit: Cooler Master G550M (62.90 EUR)

http://azerty.nl/0-1073-645239/coolermaster-gm-series-g550m.html

6. Casing: Cooler Master N300 - Midtower (35,90 EUR)

http://azerty.nl/0-1044-624119/cooler-master-n300-midtowermodel-atx-geen-voeding-atx-ps-2-nachtzwart-usb-audio.html

7. SSD: Samsung EVO Basic 250GB (121,90 EUR)

http://azerty.nl/0-1990-634348/samsung-840-evo-ssd-basic.html

8. 10GigE Ethernet card - Intel (575.00 EUR)

http://www.ettus.com/product/details/10GIGE-KIT

Total Price: 1556.23 EUR