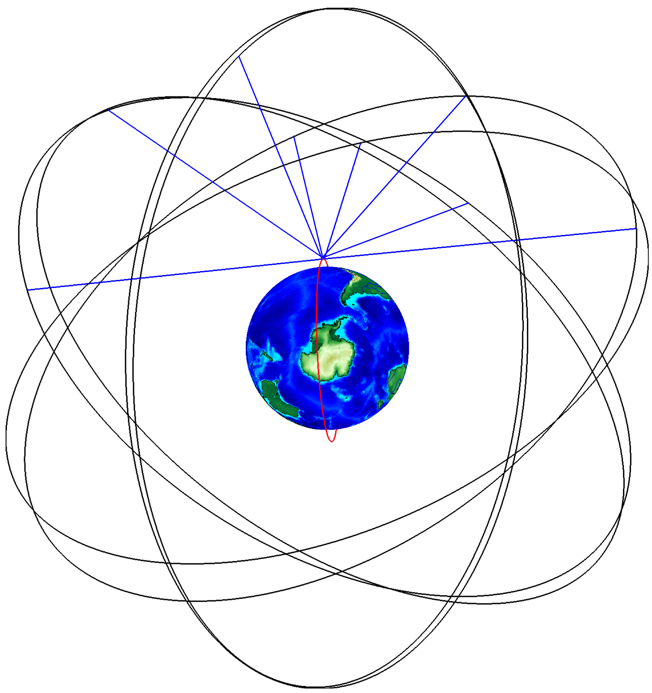
High Dynamic GPS Signal Acquisition

A Case Study in GPS Recievers on Nano-Satellites in LEO



AALBORG UNIVERSITY

Applied Signal Processing and Implementation
Group 10gr1045, Fall 2009 & Spring 2010
Faculties of Engineering, Science and Medicine
Department of Electronic Systems, Aalborg University

AALBORG UNIVERSITET

Department of Electronic Systems
Master Thesis - ASPI
Fredrik Bajers Vej 7
Telephone 96 35 86 90
http://es.aau.dk

Title:
High Dynamic GPS Signal Acquistion

Theme:
Optimal VLSI Signal Processing

Project Period:
Sep. 2009 to June 2010

Group:
10gr1045

Group Members:
Alex Aaen Birklykke

Supervisor:
Yannick Le Moullec

Co-supervisor:
Darius Plausinaitis

Publications:
5

Pages:
135

Finished:
3. June 2010

Abstract:

A low-power GPS signal acquisition algorithm is designed, evaluated and implemented for use on small-satellites in low earth orbit. In order to evaluate the high dynamic environment in relation to signal acquisition, empiric data is obtained using the SGP4 perturbation model for 24 GPS satellites and the Ørsted and AAU Cubesat missions. This shows that a maximum Doppler shift offset of $\pm$45 kHz on the L_1 carrier can be expected with a mean drift of -29 Hz/s. With an outset in these constraints a maximum likelihood estimator-based signal acquisition algorithm is proposed that enables acquisition of 98.3% of the GPS signals under continuous receiver operation. The design space of the algorithm is explored to minimize area and power of the implementation. The algorithm is realized using a semi-systolic array of add/subtract controlled accumulators. The received GPS signal ripples through the accumulators, whereas spreading codes are used to control the add/subtract behavior, effectively correlating the spreading codes with the received signal. The architecture is modeled in VHDL and implemented on a FPGA. If the quiescent power of the FPGA is disregarded, the final solution, including an active patch antenna and GPS front-end, has a power consumption of approximately 400 mW. Due to practical issues it was not possible to fully confirm the functionality of the implemented algorithm, but theory and offline simulations all work as intended.

Preface

This thesis is the result of a two semester project at the Applied Signal Processing and Implementation (ASPI) specialization at Aalborg University. As the project covers two semesters it serves as documentation for both the master and 3ed semester of the specialization. The theme of the 3ed semester is optimal VLSI signal processing, whereas the master is open. The project has been supervised by Associate Professor Yannick Le Moullec from the Technology Platforms Section at AAU and co-supervised by Postdoc Darius Plausinaitis from the Danish GPS Center at AAU.

When this project was commenced the view of GPS did not seem daunting at all. And why should it? GPS is embedded in even the most simple and small devices today, so the system is by no means presented as complex to its user. Hence, when Gomspace proposed a GPS receiver for use on one of their satellites, it seemed as a challenging but manageable project that would fit nicely to a 2 semester master project. And this was also the spirit in which the project was presented by the company. Not to say that it was totally without hesitation. As a former Junior Engineer at the company I had previous experienced the challenging nature of designing systems for space environments and as such I knew that the project would present its fair share of challenges. Also, the project would involve disciplines from different area of the science, but I found all of these aspects appealing. A master project should be challenging, I thought.

However, a few months into the project the complex nature of the GPS system started to dawn on me, and it became clear that the scope of the project was underestimated. The Global Positioning System is a huge communication system, with a multitude of different layers and components, all of which requires insight in different areas of science. Furthermore, the evaluation of space environment requires great insight in many areas of science not common in electronic engineering. E.g. positioning in space or space weather. Hence, the project turned out to be a multi-disciplinary challenge, which by far exceeded the initial (naive) expectations to the project. The decision was however made to proceed with the project, although in a slightly reduced version, with as many aspects remaining as possible. The following thesis is the distilled result of this work. However, due to the multi-disciplinary nature of the problem the number of aspects and perspective that can be applied are endless. Hence, some parts may go into great detail, whereas others only scratch the surface of a subject and it might also mean that some areas have not been covered sufficiently.

Alex Aaen Birklykke

Acknowledgements

First of all a special thanks should go to my co-supervisor Darius Plausinaitis for sharing his profound knowledge regarding the GPS system and insight in GPS receiver technology. Without this knowledge a lot of aspects regarding GPS would have remained unconsidered or unknown. Similar, gratitude goes to my supervisor, Yannick Le Moullec, who has been very helpful in proof-reading and providing guidance in relation to FPGA technology and EDA tools. Also, thanks to Gomspace for providing the GPS front-end, the AAUSAT group for lending me their GPS patch antennas and the community behind `www.opencores.org` for providing open VHDL modules, which has been used as great learning examples.

Finally, I want to thank those who did not believe that it was possible to accomplish this project. No one mentioned, no one forgotten. The lag of trust in my abilities has been a great motivational factor through out the project period.

Table of Contents

Chapter 1

Introduction

This project is based on a proposal made by Gomspace Aps. Based on meetings with the company, their proposal and use case is reproduced in the first section. As the initial proposal is very broad, it is delimited to a more suitable size, which is done in the project description section. Finally the structure of the report is presented.

1.1 The Gomspace Proposal

The following description of the use case relevant to Gomspace is based on meetings with the company. The company have not provided anything in writing and as such this is the best effort by the author to reproduce an abstract of the topics discussed in those meetings.

In short, Gomspace proposes an FPGA-based GPS receiver designed for use on small satellites in low earth orbit (LEO). By small, Gomspace refers to satellites between 1-10 kg - known popular as nano-satellites - which is their area of focus. An example of a nano-satellite is the GOMX satellites shown in Figure 1.1. This is an upcoming demonstration mission planned by Gomspace and is build around the so-called Cubesat concept. In cubesat terminology a 1 unit satellite is 10x10x10 cm in dimensions and has a weight of 1 kg. The GOMX is a 2 unit satellite which means that it is 10x10x20 cm.

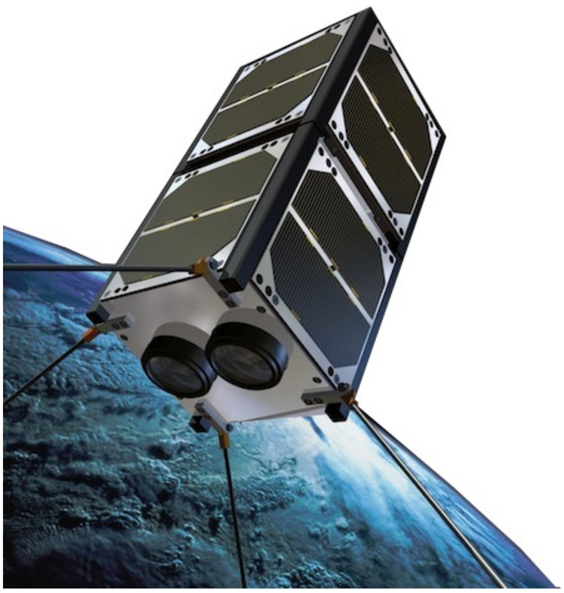


Figure 1.1: Illustration of the GOMX satellite from the Gomspace homepage. GOMX is scheduled for launch in Q1 2011.

Due to the small size of the satellite the surface area available for solar-cells is limited. This means that the power budget is small and that all satellite components have to be constructed with power in mind. This is no different for the proposed GPS receiver, and Gomspace estimates that around 1 W of instantaneous power is all that is available. Furthermore, even though the power is available it is not uncommon that the receiver or other parts of the satellite is shot down on a regular basis, either to save energy or to service other payloads on-board. Consequently the receiver has to be designed to cobe with periodic power-downs and any start-up time must be minimized.

Another important design aspect is the space environment, where both physical and dynamic factors are more extreme compared with those in terrestrial applications. E.g. according to Gomspace, the physical temperature of the satellite extremities fluctuates between -40 and 110 °C depending on whether the satellite is in the sun or shadow of the earth. The inside component temperature should be between a more moderate 0 and 30 °C. Also the relative motion between the GPS receiver (on the satellite in LEO) and the GPS constellation causes the GPS signal to be shifted in frequency, due the Doppler effect. Hence, in order for a receiver to work in these conditions the environmental factors must be included in the design process.

Company Motivation

The main motivation behind the Gomspace proposal can be divided in two 1) the GPS application it self, which could solve a number of issues in space mission, and 2) the FPGA technology, which is attracting interest from the space industry due to its flexibility and possible radiation tolerance. The following summaries some of the areas of interest mentioned by the company.

- **Space GPS:** Today a coarse estimation of the satellite orbits can be obtained by using a model known as SPG4 and two-line elements (TLE) which are available for all satellites is space and is provided for free on the Internet by NORAD (North American Aerospace Defense Command). At best the accuracy of the this method is 2 km for potion and 2 m/s for velocity [1]. With GPS the accuracy is measured in meters, which would thus provide a far better estimation of the satellites position.

- **GPS time:** The GPS system is based on its own time standard, known as GPS time. GPS time is based on very precise atomic clocks placed on each GPS satellite and the time is broadcasted from all GPS satellites to the receivers. By gathering this data it is possible to calculate the time to a very high precision, which can be used on the satellite for mission planning. E.g. most larger satellites have an on-board time-bus which provided the sub-systems with a very precise clock.

- **FPGA:** The FPGA technology has gained increasingly interest from the space industry over the last few years. The flexibility of the technology along with the availability of radiation hardened FPGAs makes it alluring for space application. As an example, some systems on satellites has to work under reliability constraints, meaning that they have to be able to run in all conditions. In order to insure this system redundancy is often used, at the cost of extra weight and power. If instead a FPGA could be reconfigured to take over the responsibility in the case of system failure, reliable systems could be designed at a lower cost.

1.1.1 About Gomspace APS

Gomspace was established in 2007 by three Ph.D. engineers from Aalborg University who was involved in the startup of the AAU small satellite program. The company is developing solutions for nano-satellites, but have also been involved in other projects such as ship positioning and ESA projects[1]. The cubesat satellite marked segment is still not fully mature and is currently maintained mostly by universities around the world. But with cheep technology available and the relative low launch costs for small satellites the company hopes to attract customers that under normal circumstances not would have considered space technology as a solution to their problems. It is therefore Gomspace's strategy to be at the forefront of this development and in the long run profit from it. To push the development Gomspace is furthermore planning its own satellite launch in Q1 2011.

1.2 Problem Description

The proposal from Gomspace leaves no questions as to why a space GPS receiver is interesting to the company. Precise position determination in space would be a valuable asset to the company and many of the related issues in designing such a receiver are interesting in other contexts as well. Some examples are the satellite-to-satellite communication, Doppler-shift estimation and FPGA based receiver technology, which could be re-used in other projects.

However, the problem, seen from a project point of view, is that the proposal is very broad. To get an idea of the magnitude of the project, the Navigation department at Aalborg University was two years to design and implement a terrestrial FPGA-based GPS receiver. By including space into the equation, the scope of the current proposal appears. Needless to say, but this must be considered an unrealistic challenge even for an ambitious master student. Hence, the first task must be to confine the project into a more manageable size.

The problem, that makes this project special compared to most GPS projects, is the environmental factor. Satellites in low-earth orbit (LEO) orbit at velocity considerably larger than any application would on earth. Hence the Doppler-shift offset on the carrier would also be larger than what can be expected on earth. This poses a design constraint on the system and would have to be evaluated in order to design at functional receiver. Furthermore the absence of atmosphere means that the temperature of the receiver is more extreme than on earth, radiation levels are higher - a long with other problems.

Although all of the space factors are important when considering an implementation of any application that have to work in space, the evaluation of the factors with respect to GPS lies beyond the scope of the ASPI specialization. Hence the cynic approach would be to disregard all of the space related issues and thus reducing the project to a earth based receiver. In this case the value of the project, both scientifically and for Gomspace, is also reduced as the implemented receiver will be just another entry in the sequence of already existing FPGA GPS receivers. Hence, the only acceptable solution is a compromise, where the most important space related factors are included, but a blind eye is turned to some of those that can be considered less important. The downside to this solution is of cause that some effort will be used elsewhere than on the signal processing and implementation.

As space cannot be fully disregarded, the GPS receiver must be considered. In order to do this a high level description is necessary. Figure 1.2 shows the signal chain in a typical GPS receiver.

[1] `www.gomspace.com`

An *antenna* captures the GPS signals from the range of satellites visible to the receiver.

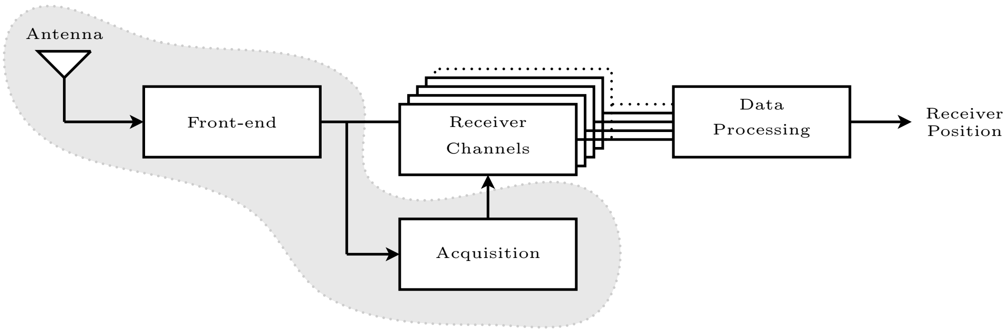


Figure 1.2: Illustration of a generic GPS receiver. The high-lighted area is the focus of the project.

This is down-converted, usually to an intermediate frequency, and sampled in the receiver *front-end*. In a cold start of the receiver, no information about the signals visible, their phase and/or Doppler shift are available. Hence a process known as acquisition is performed. Acquisition uses the spreading sequences[2], unique to each satellite, to identify the GPS signals present in the received signal, along with their phase and Doppler shifts. When a satellite is acquired a receiver channel is allocated to track the signal. In order to maintain a lock on a signal it must be tracked both in phase and frequency. Once a GPS signal is tracked it is possible to extract the navigation data and the receiver position can be calculated. In general the acquisition and tracking is referred to as the *signal processing* of the received signal and the interpretation of the navigation data is performed by the *data processing* block.

As the antenna, front-end and signal processing have to deal with the direct effects of space environment, such as additional Doppler shift and temperature, these are considered the most interesting both in relation to the Gomspace proposal, but also seen from a ASPI point of view. To narrow the project down even further only the acquisition part of the signal processing will be covered. The basis for this decision is both the limited time available, but also that acquisition is a precondition for a GPS receiver and its tracking loops to work at all. Hence, by designing and implementing an acquisition algorithm suitable for space, one of the important initial steps towards a fully functional receiver would have been taken.

In summery the problem statement is as follows:

> *Design, evaluate and implement a low power GPS acquisition algorithm in FPGA technology suitable for use in LEO.*

In order to accomplish this a series of problems must be evaluated. First of all the GPS system have to analyzed with respect to use in space. This involves evaluating the amount of Doppler shift and the adverse effects on the acquisition process. Also the effects of temperature on the system have to be evaluated. This is especially interesting with respect to the antenna, that might induce additional noise, when exposed to the extremes of space. All of these problems will be addressed in the analysis and design of the system.

[2]GPS is a spread spectrum CDMA system, but this will explained in detail later.

Delimitation

The project will not include analysis of radiation effects on the system. Also the project main focus will be on the civil branch of the GPS system, which is also known as Standard Positioning Service (SPS). The SPS is based on a set of spreading sequences known as coarse acquisition (C/A) codes, which means that C/A code acquisition is the aim of the project. The scope of the project will be narrowed continously through out the report, as more knowledge about the system becomes available.

1.3 The ASPI Paradigm

One of the great sources of confusion when people with insight in different subjects has to communicate, is that words and terms evoke different associations. One example, taken from this project, is the area of spread spectrum communication. In a spread spectrum receiver the input signal is correlated with a copy of the spreading sequence (pseudo-random noise codes) in order to obtain the data. To a communication engineer this would be referred to as Digital Matched Filtering (DMF), whereas it to a signal processing engineer would be interpreted as a simple correlation. In other branches of science it might even be recognized as a pattern recognition algorithm. The point is that if these engineers where to design a spread spectrum system, the first challenge would be to harmonies their point of view. Otherwise they would talk past each other and/or end up confused at a higher level.

In the realm of the ASPI specialization this is also a problem and one area in particular is prone to misunderstandings, which is that of abstraction models. In Very Large System Integration (VLSI) projects such as ASCI or FPGA design the complexity of the system is beyond human comprehension if all aspects are taken into account at once. Hence, different levels of abstractions are used to hide details that can be considered less important at a given point in the design process. E.g. if the theory on DMF is analyzed it has little importance whether it would eventually be implemented in an ASIC or FPGA. Hence, the designer abstract from these details. In order prepare the reader, the point of view, i.e. abstraction model, used in this project will be presented in the following.

In this project an abstraction model known as A^3 has been used, see Figure 1.3. A^3 is a hierarchical model where a project or problem is divided into three domains; application, algorithm and architecture. The application domain constitutes the highest level of abstraction and is where the overall functionality and constraints are considered. In order to formulate a specification for the system, the problem space in the application domain has to be explored. In this project the problem space exploration is a large part of the project, as the effects of space environment has to evaluated in relation to GPS.

Once a specification has been formulated the system has to be mapped to the algorithm domain, i.e. taken to a lower level of abstraction. As the specification can be mapped in many different ways, the design space in the algorithm domain is huge. E.g. a filter can be implemented using a multitude of methods. In order to chose a suitable solution the design space must be explored. Once this has been accomplished each solution is evaluated according to a set of objective, such as power, area or execution time. The solution with the best performance is chosen for the next mapping. In Figure 1.3 this solution is indicated by $s*$.

In the final step the chosen algorithm is mapped onto the architecture, lowering the abstraction level even further. Again this is an one-to-many mapping, as the algorithm can be implemented on many different hardware platforms, e.g. ASICs, FPGAs, DSPs ect. However, in this project the FPGA architecture has been chosen in advance, which means that it is not necessary to con-

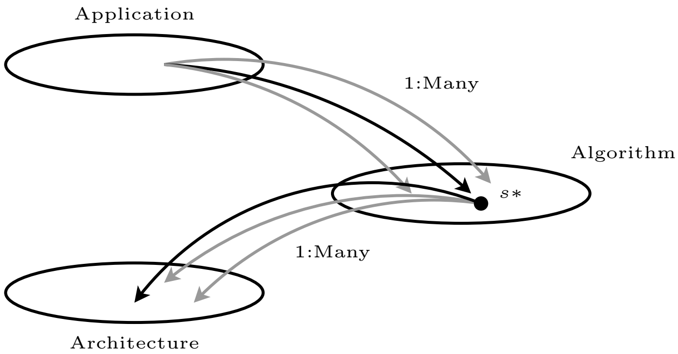


Figure 1.3: The A^3 high level abstraction model. $s*$ marks the optimal solution relative to a set of project specific objectives. Common objectives are power, area and execution time.

sider other candidates.

As the model has been used consistently through out the project, it is also reflected in the documentation. Hence, the first three part covers each domain in the model, whereby the abstraction level is lowered small steps at a time. In this way it should be is easy to follow the progress as long as the A^3 abstraction model is kept in mind.

Part I

Application Analysis

Chapter 2

GPS Signal

In this chapter the components and properties of the GPS signal are described. The focus in this chapter and further on is mainly on the civil signals, but to illustrate the complexity of the signal the non-civil components have also been include. The first section gives a detailed description of the GPS signal components. Next it is shown how the signals are multiplexed and modulated onto the carriers. Finally the properties of the signals are investigated. For the interested reader a historical background of GPS can be found in Appendix A.

2.1 System Overview

The GPS system consists of three components; the space segment, user segment and control segment. The space segment consists of 24 (at the moment 32) satellites. The satellites constellation consists of six different orbital planes inclined 55° to the equator [2]. In each plane a minimum of 4 GPS satellites are orbiting at an altitude of 20.000 to 21.000 km. This constellation ensures that at least 4-12 satellites are visible at any time around the globe [3]. The current constellation of satellites are known as Block IIF and are in operation until 2018-2020 [2]. Each satellite has on-board atomic clocks with a drift of only a few parts in 10^{-13} over a day [3]. The clocks are synchronized between the satellites and are the very heart of the GPS system. They are used to derive the GPS system time, which, as will be shown, is the basis of the GPS navigation, as well as the fundamental frequency used to generate the carrier waves.

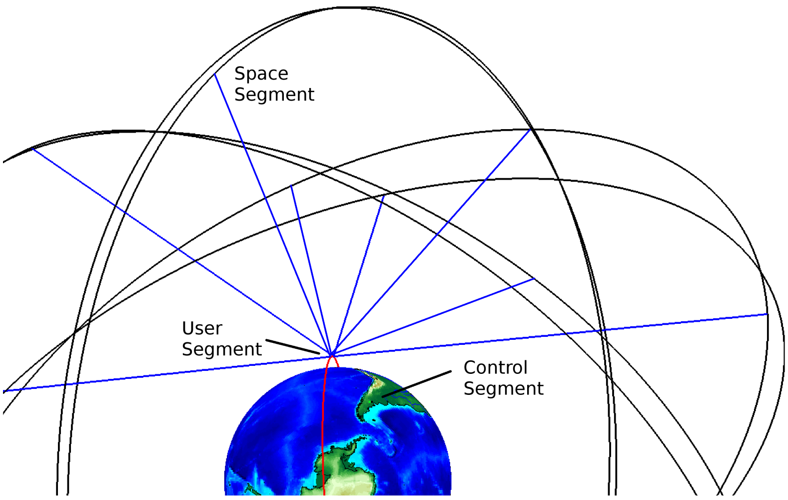


Figure 2.1: GPS system components.

The control segment consists of a number of monitor stations, which are equipped with precise atomic time standards. These station measures the position of the visible satellites and relays the information to a control stations that sends correction the GPS satellite when needed. Ground control can upload new data at every moment, but most of the time its done every 24 hours [4].

The user segment is divided into two categories; military and civil users. Military users are authorized by the DoD and can access the encrypted GPS signals, which can improve the accuracy considerably. The service provided to civil users is commonly referred to as Standard Positioning Service (SPS) and is described in [5]. In the current GPS system, civil users can only access the so-called Coarse Acquisition (C/A) codes [5], which will be discussed further in Chapter 2. In the modernized system additional signals will become available and the Block IIF satellites also have the ability to broadcast these new signals [2][6], and according so some sources they do[1]. This fact is however disputable and the author have not been able to find any official source stating that the signal is available.

2.1.1 Positioning

The positioning concept used by the GPS system is based on the propagation of radio waves through space. The distance ρ traveled by a radio wave is given by the propagation delay Δt multiplied with the speed of light c. For the kth satellite this is[1]

$$\rho^k = c \cdot \Delta t^k \quad [\mathrm{m}] \tag{2.1}$$

In order to calculate the propagation delay to the receiver. each satellite sends navigation data containing the exact GPS system time at which the data was transmitted. This is then compared with a synchronized clock in the receiver, whereby the delay and hence range to the satellite can be found. The navigation data from the satellites also contains information that can be used to calculate the satellites position. Using this information the position of the k satellite (X^k, Y^k, Z^k) relative to the center of the earth can be found at the epoch the data was send. Now, given the position and range to a satellite the position of the receiver (x, y, z) is given by

$$(\rho^k)^2 = (x - X^k)^2 + (y - Y^k)^2 + (z - Z^k)^2 \tag{2.2}$$

Equation 2.2 has three unknowns, hence one need the position and range to at least two more satellites, to find the receiver position. Assuming that these are available, Eqn. 2.2 can be solved and the receiver position found. This method is known as *trilateration*, as opposed to triangulation where the angles are used to find the position [7].

The problem with the method just described is that the clock at the receiver has to be very precise due to the very large value of the speed of light. E.g. 1 μs error translates into a range error of 300 m. This can be solved by fitting the receiver with a very precise clock. This solution is however very costly and instead the range error induced by the difference between the receiver and satellite clock is accepted as a fourth unknown into Eqn. 2.2

$$(\rho^k)^2 = (x - X^k)^2 + (y - Y^k)^2 + (z - Z^k)^2 + (c\,dt)^2 \tag{2.3}$$

where $c\,dt$ is the error in meters and dt is clock difference. This is, however, only possible as the GPS satellite clocks are synchronized with each other, which means that the clock error in the receiver is the same for all satellites. Given this new trilateration equation at least four satellites are needed to calculate the receiver position, as illustrated in Figure 2.2.

[1]The notation is not to be mistaken with the kth power

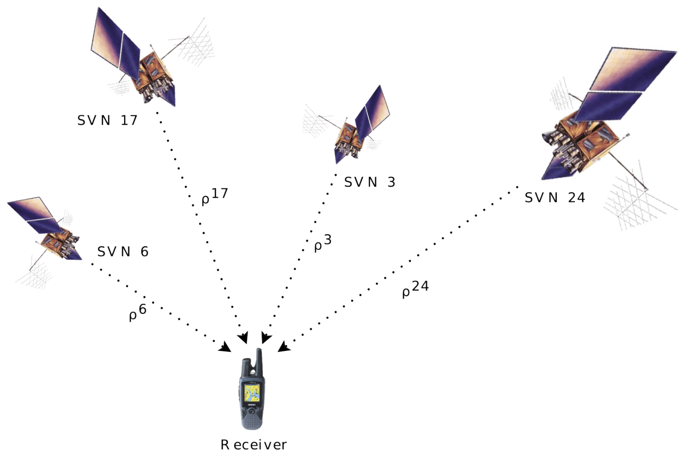


Figure 2.2: 4 satellite GPS positioning scenario. ρ^k is the range to the kth satellite.

2.2 Signal Components

As a result of the current modernization process, described in Appendix A, many sources are outdated or in conflict with the current state of the system. To make matters worse, many of the changes made as part of the modernization regards the GPS signals, which are about to be described. In order to abstract from the conflicting literature it has been chosen to describe the GPS signals as specified in the official GPS interface specification IS-GPS-200D [6], supplemented with others sources as secondary literature.

Carrier

The GPS signal consists of two RF links denoted L_1 and L_2. Each link has a carrier that is derived from a nominal frequency source f_0 on-board the GPS satellites of precisely 10.23 MHz seen from the receiver. That is, the nominal frequency on the SV is offset by a few miliHertz to compensate for relativistic effects [8, 6], so that f_0 at the receiver is 10.23 MHz. Each RF link carrier is an integer multiple of the nominal frequency [8], giving

$$f_{L1} = 154 f_0 = 1575.42 \text{ MHz}$$
$$f_{L2} = 120 f_0 = 1227.60 \text{ MHz}$$

Each carrier of L_1 and L_2 are modulated with one or more bit-trains, usually consisting of a *spreading sequence* and the *navigation data*. How this is accomplished is discussed shortly, but first the components of the bit-trains components are described:

Navigation Data

The navigation data contain information about the satellite clock, health, orbit data and other data like parity bits. The orbit data is also known as *ephemeris* data referring to the fact that it is only valid for a short period of time [8]. In order to keep the ephemeris up to date, the satellites update their orbit information every 2 hours.

Two set of navigation data are available as specified in Table 2.1 denoted NAV and CNAV respectively. The NAV data has a bit rate of 50 bps and a frame size of 1500 bits. One frame contains the information needed to calculate the position of satellite, the propagation delay and hence the receiver position. The same NAV frame is repeated every 30 s.

The CNAV navigation signal has been added as part of the GPS modernization. CNAV contains the same information as the NAV data but is normally more accurate [6]. CNAV frames are 300 bits long and have a period of 12 s. However, a total of 13 different message types are used to accommodate for the more precise data [6], i.e. 13 frames have to be send.

Name	Acronym	Shorthand Notation	Bit Rate [bps]	Bit Period [ms]	Frame Size [bits]	Frame Period [s]
Navigation Data	NAV	$D(t)$	$f_D = 50$	$T_D = 20$	1500	30
Civil Navigation Data	CNAV	$D_C(t)$	$f_{D_C} = 25$	$T_{D_C} = 40$	300	12

Table 2.1: Specification for the GPS navigation data.

It should be noted that the term “civil” is a bit misleading as both the signals are available as part of the SPS in the modernized system. At the moment only the NAV data is available [5].

Spreading Codes

Spreading sequences or codes are periodic pseudo-random noise (PRN) sequences with a noise-like waveform [9]. Spreading sequences are used to spread a narrowband signal, such as the navigation signals, over a larger frequency band. This method is called spread spectrum modulation, and is described in detail in Section 2.3.1.

The GPS system uses a wide range of spreading codes, which are listed in Table 2.2.

Spreading Code Name	Acronym	Shorthand Notation	Chip Rate [cps]	Chip Period [s]	Length [chips]	Period
Coarse Acquisition	C/A	C^k	f_C =1.023e6	T_C ≈977.5e-9	1023	1 ms
Precision	P(Y)	$P^k(Y)$	f_P =10.23e6	T_P ≈97.75e-9	≈ 2.35e14	7 days
L_2 Civil-Moderate	L2 CM	$L_2\ CM$	f_{CM} =511.5e3	T_{CM} ≈ 1.95e-6	10230	20 ms
L_2 Civil-Long	L2 CL	$L_2\ CL$	f_{CL} =511.5e3	T_{CL} ≈1.95e-6	767250	1.5 s

Table 2.2: Specification for spreading codes used in the GPS system.

The coarse acquisition code C^k is a *Gold* code sequence of 1.023 chips repeated every 1 ms, giving a chipping rate of 1.023 Mcps. Each satellite is designated with a unique C/A code - a pseudo-random number (PRN). The superscript k denotes the C/A code of the kth satellite. Being random the codes of different satellites do not correlate, which means that it is possible to distinguish between satellites by correlating with a local copy of the PRN code (this will be elaborated later on). The C/A codes are both used by the Standard Position Service (SPS) and the Precise Position Service (PPS) [10]. Initial SPS was based on C/A codes alone [8, p. 19], but with the modernization of the GPS system two new codes have been introduced, namely the L2 civil-moderate and civil-long code. These codes are time multiplexed together with the CNAV data to form the L2 civil signal, which has the same chipping rate, of 1.023 Mcps, as the C/A code [6, 11].

Precision codes P^k are approximately $2.35 \cdot 10^{14}$ length sequences with a chipping rate of 10.23 Mcps. The P-code are used to calculate the ionospheric delay which normally degrades the accuracy considerably [3, p. 92]. Currently the Anti-Spoofing is enabled which means that an unknown/secret W-code is used to encrypt the P-code to the Y-code [3], denoted $P^k(Y)$. Hence the P(Y) is only available for privileged users.

Combining the signals

The navigation data and spreading sequences are combined before being modulated onto the carrier. The combination is accomplish by feeding the signals to modulo-2 adders (denoted $\oplus$ and is essentially an exclusive OR). The basic principle is shown in Figure 2.3.

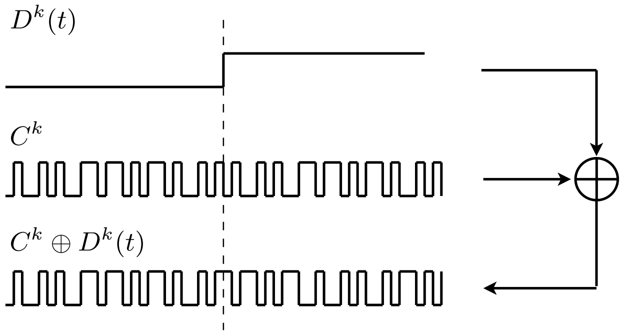


Figure 2.3: Illustration of module-2 addition of a coarse acquisition code and a data bit at a bit transition.

This yields a new bit stream which is then modulated onto the carrier. As there exists many combinations of the spreading sequences and navigation data, phase quadrature modulation is used, such that more signals can occupy the same channel. The in-phase and quadrature message signals for L_1 and L_2 are shown in Table 2.3.

L_1		L_2	
In-phase	Quadrature	In-phase	Quadrature
$P(Y) \oplus D(t)$	$C \oplus D(t)$	$P(Y) \oplus D(t)$ or $P(Y)$	$L_2\ CM \oplus D_C(t)$ with $L_2\ CL$ or $C \oplus D(t)$ or C

Table 2.3: Block IIF GPS signal configuration as given in the GPS interface specification IS-GPS-200D [6]. The configuration of L_2 is determined by ground control.

Link 2 has several options that is chosen from ground control. As the in-phase options uses encryption, the state is irrelevant. The quadrature phase has three states where all are accessible for civil users. However, it has not been possible to find the configuration that the current GPS satellites are using. The official SPS description [5] states that only the L_1 C/A signal is implemented.

Relation between signal components

In order to get a complete perspective of the GPS signal it is important to notice the relationship between the periods of the different signals. One chip of a C/A code consists of 1540 L_1 wave lengths and one NAV bit consists of a 20 C/A code sequences of 1023 chips. Figure 2.4 illustrates the relation for the C/A code and NAV data on L_1 .

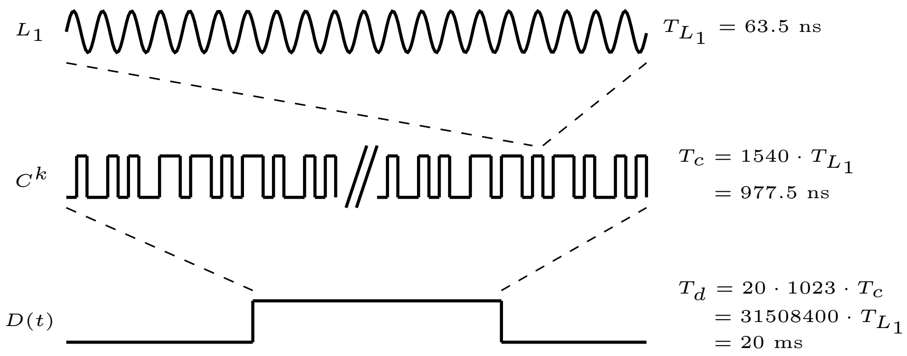


Figure 2.4: Relationship between the periods of the carrier, spreading sequences and data bits.

Another aspect is the distance traveled by the signal over one period. The wave length of L_1 is approx. 19 cm [11]. The chip period of a C/A code is approx. 1 μs which translates into 300 m. Hence in order to obtain precise pseudo-ranges, the propagation delay must be estimated to fractions of a chip period.

2.3 Signal Modulation

Link 1 and 2 are modulated using quadrature-carrier multiplexing (QAM). The in-phase and quadrature phase components are modulated by two separate bit-trains using binary phase shift keying (BPSK), see Figure 2.5. The first step in BPSK is to encode the bit stream which consists of 0's and 1's to -1's and 1's using a polar non-return-to-zero (NRZ) encoder. When multiplied with the in-phase or quadrature component, the phase is shifted 180° and the BPSK signal is created. Finally all the signals are combined before being bandpass filtered and amplified (not shown) [2].

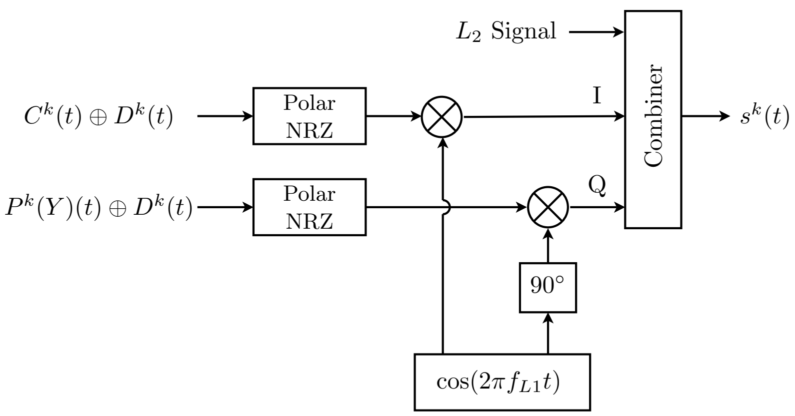


Figure 2.5: Illustration of the signal generated at the kth GPS satellite. The signal bit-trains are encoded using a polar non-return-to-zero encoder where after it is quadrature-carrier multiplexed onto the carrier.

Consequently, the combined signal for the transmitter illustrated in Figure 2.5 can be written as

$$s^k(t) = \sqrt{2P_{c,L1}}(C^k(t) \oplus D^k(t))\sin(2\pi f_{L1}t) + \\ \sqrt{2P_{p,L1}}(P^k(Y)(t) \oplus D^k(t))\cos(2\pi f_{L1}t) + L_2\,\text{Signal}, \quad (2.4)$$

where P_c and $P_{p,L1}$ the power of the C/A code signal. The in-phase terms containing the P(Y) code are attenuated 3 dB according to the C/A codes before being combined [8].

2.3.1 Spread-Spectrum modulation

The signal scheme described in the two previous sections is commonly known as spread spectrum modulation. According to Haykin [9] spread-spectrum modulation can be characterized as a transmission where 1) the bandwidth occupied by the data sequence is in excess of the minimum bandwidth needed to send it 2) where the spectrum spreading is accomplished before transmission by using a code that is independent of the data sequence. And where the same code is used in the receiver to despread the received signal, whereby the original data sequence is recovered.

From Table 2.3 on page 13, it can be seen that in the GPS signal narrow-band navigation signals are combined with a wide-band spreading sequences. Hence the data sequence are spread in frequency. To illustrate the signal spreading concept Figure 2.6 shows how the NAV data is spread with help of the C/A code.

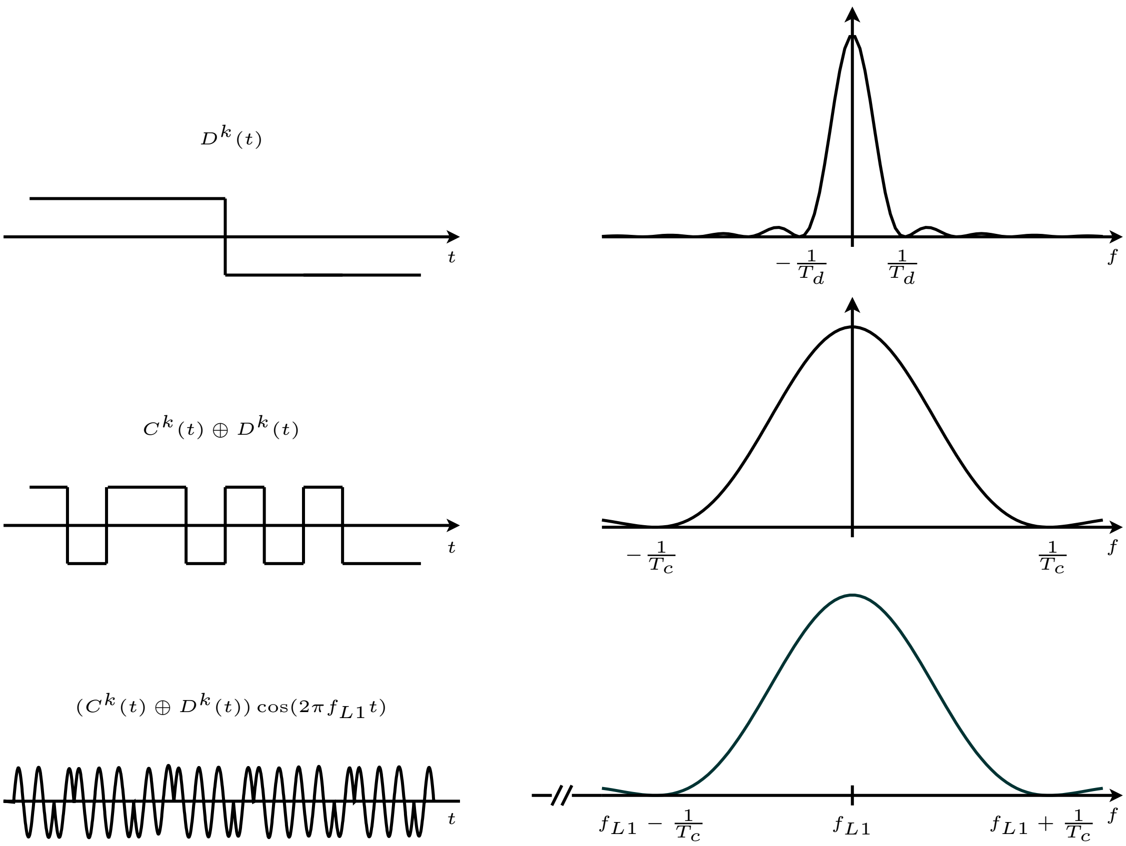


Figure 2.6: Illustration of the signal spreading. Top: The low bandwidth data signal. Middle: Data signal modulo-2 added with the spreading sequence. Bottom: Spread spectrum signal modulated with the carrier using binary phase shift keying. The figure is inspired by [12]

More precisely the spread-spectrum method used in GPS is known as binary phase shift keying direct sequence spread spectrum (BPSK DSSS [12]. Direct sequence refers methods where the spreading of the spectrum is accomplished by phase modulating the carrier [12], such as it is shown in Figure 2.5. Furthermore since each satellite is assigned with a unique spreading sequences, the system is referred to as code division multiple access (CDMA) [9, 12, 13].

2.4 Signal Power Level

The GPS signals described in the previous sections are transmitted towards the earth at different power level according to the quadrature phase and carrier frequency of the individual signal components. The effective isotropic radiated power (EIRP) of the different components are listed in Table 2.4.

	L_1		L_2	
	In-phase	Quadrature	In-phase	Quadrature
EIRP [dBW]	24.23	27.23	24.23	n/a
User Minimum Received Power [dBW]	-160.5	-157.5	-161.5	-160.0

Table 2.4: The effective isotropic radiation power (EIFP) transmitted and minimum received RF signal strength for Block IIF GPS satellites as stated in the GPS interface specification IS-GPS-200D [6] and [2]. The minimum received power is specified for a 3 dB linear polarized antenna.

As the GPS signal propagates through space it is prone to significant free space losses. At an approximately distance of 20.000 km the free loss factor is -182 dB or $5.73 \cdot 10^{-19}$ [12]. In addition the signal is further attenuated through the atmosphere of the earth. An often used estimate of the attenuation is 2 dB [2, 12, 14]. (This figure have been found to be very conservative [2], and much smaller in most cases. But this is a discussion beyond the scope of this section). By combining all the losses it is possible to find the minimum signal strength at the receiver as seen in Table 2.4. As it is evident the signal strength is very weak and as will be shown in Section 4.2 the signal is actually well below the receiver noise floor. This is widely seen as the Achilles' heal of the GPS system, as makes vulnerable to jamming [10].

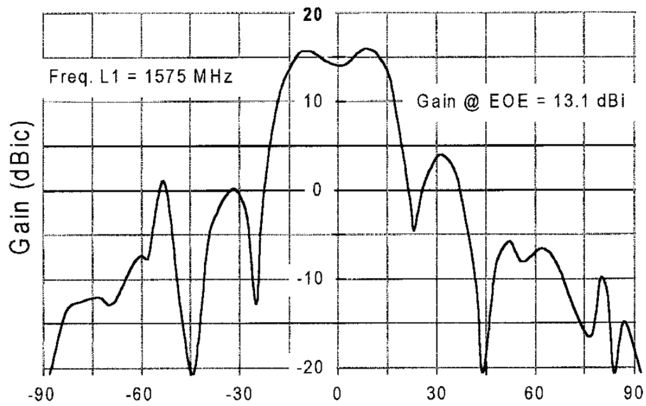


Figure 2.7: Antenna gain pattern for a Block IIF GPS satellite[2] (EOE = Edge-Of-Earth)

The signals are transmitted using a phased array consisting of 12 right hand circular polarized (RHCP) helix antennas [14, 2]. The resulting antenna pattern, which can be seen in Figure 2.8, results in a main beam width of approximately 46° for the L_1 signal. The beam form is designed with a dimple in the bore-sight of the beam pattern where the signal is attenuated -2.1 dB [14]. As illustrated if Figure 2.8 this makes the repudiation pattern follow the contour of the earth. The minimum received signal strength specified in Table 2.4 is at a user elevation angle of 5° or 90° [6]. That is when the satellite is at zenith or 5° above the horizon according the user. At other elevation angles the signal strength increases and a maximum is reached at 40° where it is approximately 2 dB above the nominal value.

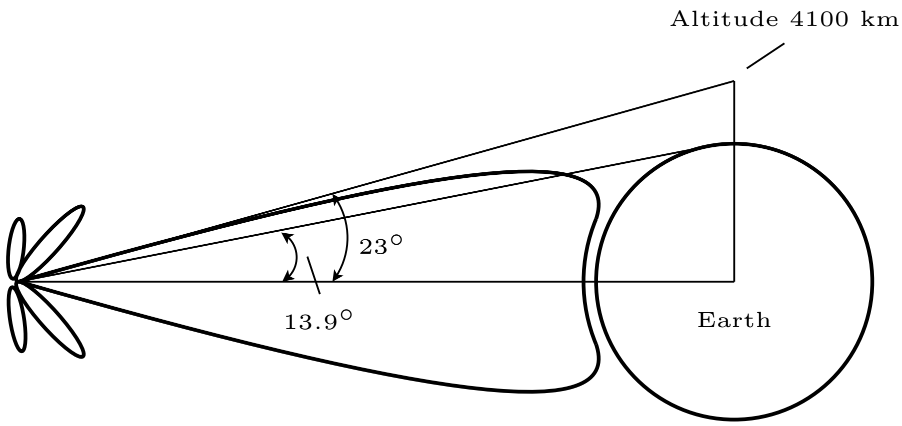


Figure 2.8: Inspired by Fig. 15 p. 235 and Fig. 16, 17 p. 96 in [14], but is updated according to the radiation pattern of the Block IIF satellites shown in Figure 2.7

With respect to GPS in space Figure 2.7 is interesting as it shows that the main beam of the GPS signal extents beyond the edge of the earth (EOE). This means that satellites in LEO orbits can receive GPS signals even if they are not in the direct path of the beam. However, it should be noted that the signal is attenuated considerably when the angle to the satellite (according to the bore-sight of the GPS SV antenna) is beyond 13°. From Figure 2.7 its seen that at 13°, equivalent to EOE, the signal is attenuated to approx. 21-13.1 = 7.9 dBi. This figure is further reduced to 21-(-5) dBi = 26 dBi at the beam width of the main lobe. In other words in these cases the minimum received signal strength at the satellite is below the minimum received signal strength specified in Table 2.4. The (new) expected minimum received signal strength is also dependent on the altitude of the satellite and the matter is complicated furthermore by the fact that the free space loss factor deteriorates as the distance increases. Hence these factors must also be considered if the receiver is to be operated beyond the intended limits of the GPS system.

Chapter 3

Space GPS

In this chapter the adverse effect of space flight are analyzed with special attention to the Doppler shift offset induced on the carrier signal. The goal is to achieve a better understanding of the nature and magnitude of Doppler shift that a GPS receiver would experience in LEO. As it has not been possible to find information or data on Doppler shift between the GPS satellites and a receiver in LEO, the data is obtained by using orbit propagation models. Two cases are used as surrogate for the GOMX mission, namely the Ørsted and AAU Cubesat. But first a current GPS receiver designed for small satellite mission is presented.

3.1 State of Space GPS

The idea of using GPS in space is by no means a new one. The first satellite that demonstrated GPS in space was the Landsat 4, launched in 1982, and between 1982 and 1999 GPS was used in more then 25 space missions [15]. One of the more noticeable missions is the International Space Station, which uses GPS for navigation, attitude determination, tracking of vehicles approaching the station and as a source of time for on-board operations [10]. Although there exist a long history of space GPS receivers, compared to terrestrial receivers, only little literature is available on the subject. The main body of research available has been published by Deutsch Zentrum für Luft- und Raumfahrt (DLR) [1, 16, 17, 18], whom is also the designer and retailer of a miniature space GPS receiver called Phoenix.

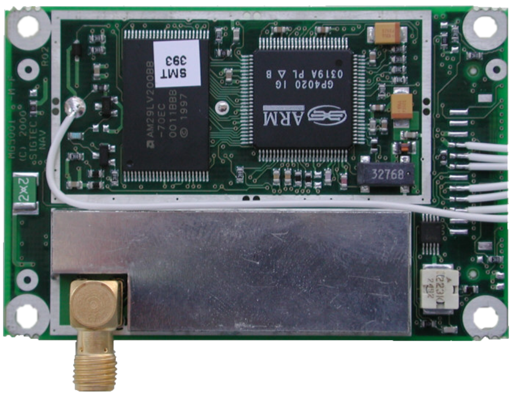


Figure 3.1: The DLR Phoenix receiver from [19]

The Phoenix receiver is a L_1 C/A receiver and is unique in the sense that it is the only available GPS receiver purposely designed for micro-satellites, e.g. the receiver has a power consumption below 0.8 W, a small footprint and the components have been tested in radiation environments equivalent to those in LEO [16, 19]. As an outset for the development, DLR used a prototype receiver called Orion GPS, which was made by the Mitel Corporation to promote the Mitel GPS GP2000 chipset [16]. One of the initial challenges DLR had to face was to reduce the acquisition and re-acquisition time. This was achieved by using a perturbation model, known as SGP4, to estimate the state (position and velocity) of the receiver. As the states of the GPS satellites can be found the same way, it is possible to calculate which satellites that are visible to the receiver and, hence, the phase and Doppler frequency of the C/A codes. In this 'warm' start of the receiver a time to first fix (TTFF) can be achieved within 1 min [19]. The TTFF is the time it takes to acquire *one* GPS signal and lock onto it. Without any orbit information or 'cold' start as its known, the receiver has a TTFF of 10 min.

3.2 GPS to LEO-Satellite Doppler Shift

A simplified view of the current case is depicted in Figure 3.2. As described earlier the GPS satellite constellation consists of six orbital planes inclined approximately 55° to the equatorial plane. Each plane has a minimum of 4 GPS satellites, orbiting at an altitude between 20.000 km and 21.000 km. In comparison, the GOMX satellite will be placed in low earth orbit, which is approximately between 160 and 2000 km [20, 1]. The inclination of the LEO orbit is unknown, but it is assumed that it will be a high-inclination orbit around 90°. This is based on previous Danish missions, such as the AAU Cubesat and Ørsted, which all use high inclination orbits to maximize the time window of passes over Denmark.

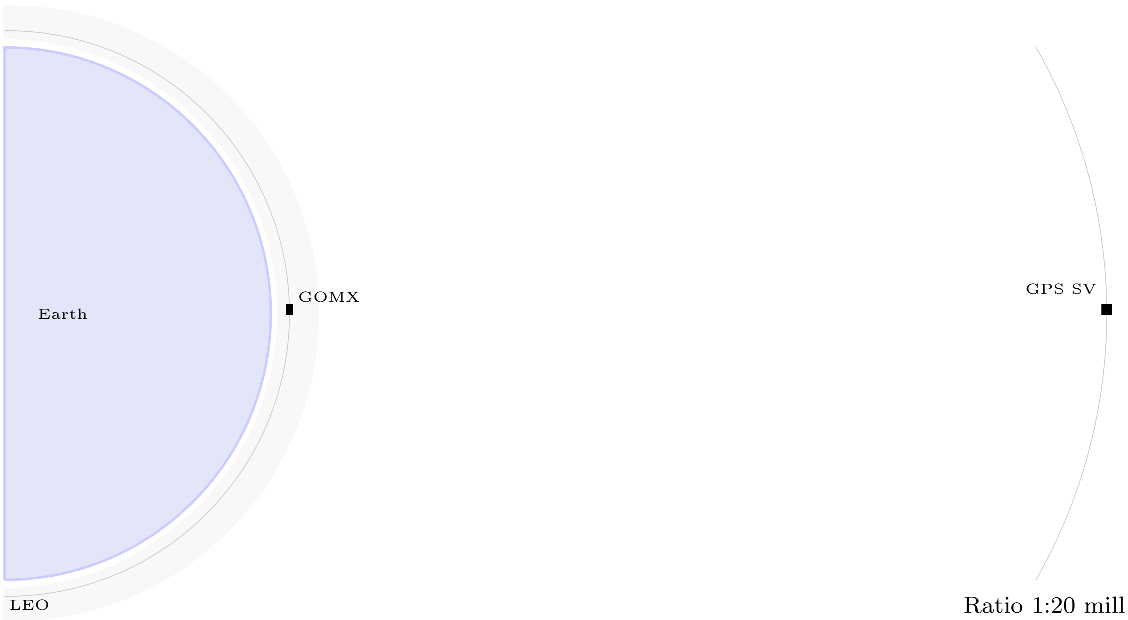


Figure 3.2: Illustrates the relation between the GOMX satellite at an altitude of 500 km, a GPS satellite 20.000 km and the earth. The proportions are in ratio 1 to 20 million.

In order to estimate the Doppler shift as experienced by the receiver, it is necessary to obtain the position and velocity of all satellites in the system. However, when thinking of a system that includes over 24 GPS satellites in six orbital planes and a LEO satellite in yet another orbital plane, all moving independently, the complex nature becomes clear. Examples do exist of LEO orbits being described in closed form as seen from a receiving antennas on earth [21]. But even for a seemingly simple case like this, the derivations are long and complex to say the least.

Instead two different approaches are taken to estimate the satellite-to-satellite doppler shift. First, a simple model is used to capture the maximum or worst case doppler shift between two objects in circular orbit around to earth. The second uses a perpurtation model known as SGP4 to estimate the position and velocity of the GPS satellites and the two dummy satellites in LEO. From this data the Doppler shift is calculated and presented.

Maximum Doppler shift

The general equation for the Doppler shift between a source and a receiver, or as in this case a GPS satellite and a receiver in LEO, is given by [22, p. 525]:

$$f_d = f_{L1}\left(\frac{c + v_{LEO}}{c - v_{GPS}}\right) \tag{3.1}$$

where c is the speed of light, f_{L1} the carrier frequency of L_1 , and v_{LEO} and v_{GPS} are the velocity of the two satellites. The velocity of satellites in earth orbit can be approximated as a function of the altitude by using the following equation [22, p. 400]:

$$v = \sqrt{\frac{GM}{r}} \tag{3.2}$$

where $G = 6.6726 \cdot 10^{-11}$ Nm^2 kg^{-2} is the gravitational constant, M the masse of the earth and r the earth center distance to the satellite. From Eqn. 3.2 it is clear that the satellite velocity decreases with distance. E.g, a satellite at 500 km would orbit with 7.62 km/s, whereas a GPS satellite at 20.000 km would have a velocity of approximately 3.9 km/s. In comparison a stationary terrestrial receiver would only move due to the rotation of the earth, which is around 0.46 km/s (assuming a circumference of 40.000 km and 24h per revolution).

At this point it would be tempting to insert the velocities into Eqn. 3.1. However, in 3.1 the velocities are line-of-sight, meaning that the source and receiver have to move in a straight line relative to each other. This is not the case when satellites are in different orbits, which complicates matters. However, under the assumption that the orbits are perfectly circular and in the same plane, the line-of-sight velocities have been derived, which can be found in Appendix C. Furthermore the maximum line-of-sight velocities was found, which, when inserted into the doppler equation, yields the following expression:

$$f_{d,max} = f_{L1}\left(\frac{c' + \frac{r_{GPS}}{\sqrt{r_{LEO}\left(r_{LEO}^2 + r_{GPS}^2\right)}}}{c' - \frac{r_{LEO}}{\sqrt{r_{GPS}\left(r_{LEO}^2 + r_{GPS}^2\right)}}}\right) \quad \text{where} \tag{3.3}$$

$$c' = \frac{c}{\sqrt{MG}} \approx 15 \text{ kg/Nms} \tag{3.4}$$

The result of Eqn. 3.3 is shown in Figure 3.3 for different LEO altitudes. As can be seen, the maximum doppler shift occurs at the lowest altitude, which is also expected as satellite velocity increases with lower altitude. According to the figure, the maximum satellite-to-satellite doppler shift is just under 45 kHz. This concurs with other literature where 50 kHz have been used [1]. The small discrepancy is probably due to the inclusion of clock error in the literature result, which induced up to 5 kHz extra Doppler shift depending on the quality of the clock [8]. Compared to terrestrial application, where most receivers are designed for a maximum Doppler shift of ± 5 kHz (excluding clock errors), the maximum Doppler shift in LEO is considerably larger. This poses a serious design constraint and the consequences are discussed further in Chapter 4.2.

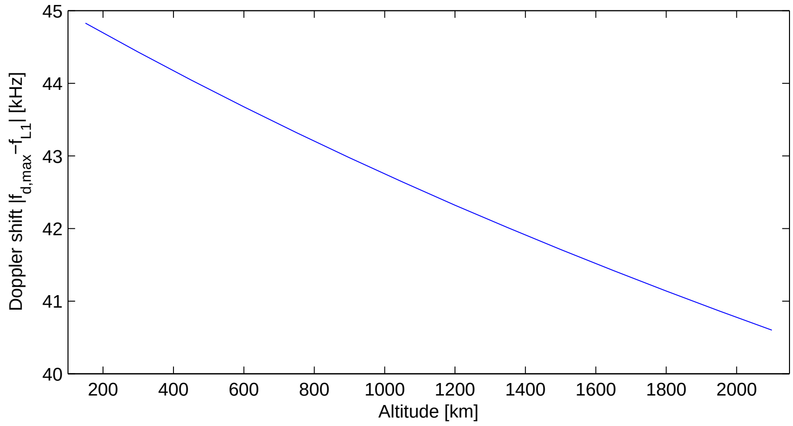
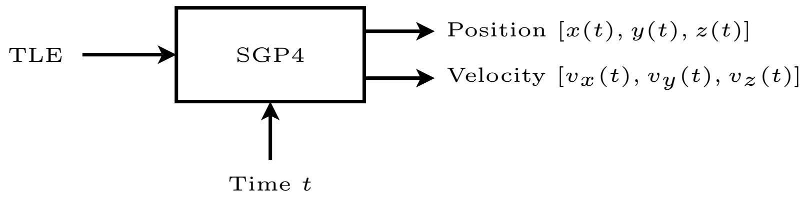


Figure 3.3: Maximum doppler shift on L_1 between a GPS satellite at 20.000 km and a receiver at different LEO altitudes.

Doppler Shift Statistics

So far the bounds of the Doppler shift have been found according to altitude, but little is still known about the dynamics of the Doppler shift. What would a typical satellite-to-satellite pass look like? At which rate does the Doppler shift change, when traveling at 7.6 km/s in LEO? To answer these questions along with others, the line-of-sight velocity between the receiver and GPS satellites has to be found, just as in the previous section. The precondition for this is knowing the state (position and velocity) of the system at a given epoch.




```
GPS BIIRM-5 (PRN 29)
1 32384U 07062A 09284.90253878 .00000026 00000-0 10000-3 0 3817
2 32384 55.0077 154.9802 0035218 284.4617 75.3188 2.00571599 13381
```

Figure 3.4: The input/output relation of the SGP4 orbit perturbation model and the NORAD two-line element (TLE) for the GPS satellite with pseudo-random number (PRN) 29.

In order to estimate the position and velocity in relation to time of satellites in orbit the NORAD SGP4 analytical orbit model has been chosen. The SGP4 model is a perturbation model, which is the general description for models used to estimate movement in gravitational fields [1]. The model provides the position and velocity of a satellite given the Julian time and date, and a NORAD two-line element (TLE), see Figure 3.4. The TLE contains data such as the orbit inclination, eccentricity of the orbit, revolutions per day, etc., which is used by the model to give an estimate of a satellites current position. The result is a set of earth-centered-earth-fixed (ECEF) coordinates (basically Cartesian coordinates with the z-axis parallel to the rotational axis of the earth and the x-axis at 0° longitude 0° latitude) of the satellite position and velocity, with an accuracy of 2 km and 2 m/s, respectively [1]. A trace of the SGP4 output is illustrated in Figure

3.5 for six GPS satellites and the Ørsted satellite, together with the line-of-sight paths.

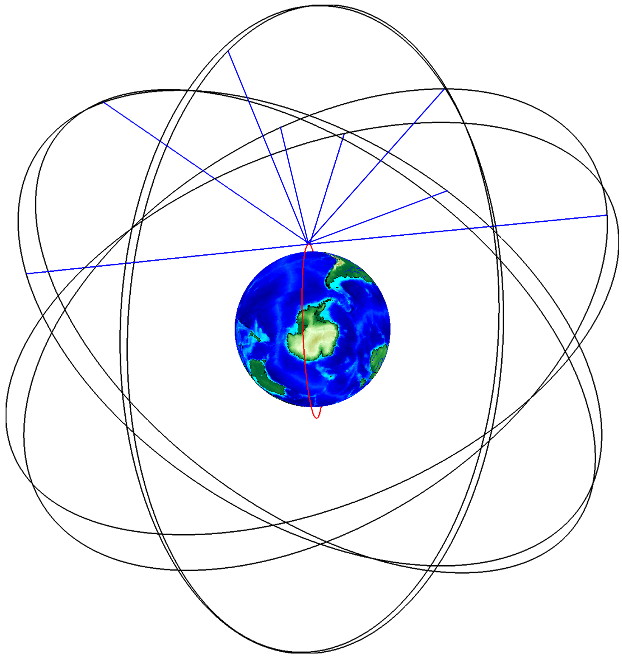


Figure 3.5: Illustration of the GPS orbits in relation with the orbit of the Ørsted satellite. The blue lines indicate the line-of-sight or signal path between the satellites.

In order to obtain information about the dynamics of the system, state data was gathered from 24 GPS satellites and the two dummy satellites over a period of 2 days at 10 s epochs. Following this the data was post-processed to 1) calculate the line-of-sight velocity and subsequently the doppler shift and 2) filter out data point where the GPS satellite is not visible to the receiver. The full details of the method used can be found in Appendix C.

Figure C.2 shows the distribution of the Doppler shift and its rate of change for the Ørsted and AAU Cubesat. The statistical approach has been chosen to get a broader view of the dynamics of the system, i.e. an idea of the general Doppler shift scenario. As can be seen, the Doppler shift is symmetrical distributed around 0 Hz and approximately uniform. This means that in practice a GPS receiver in LEO with an inclination similar to those of Ørsted and AAU Cubesat, would experience the range of doppler shift spanned by $[-f_{d,max}, f_{d,max}]$ with equal probability.

Regarding the maximum doppler shift, the altitude of Ørsted and Cubesat is approximately 700 km, which means that the maximum doppler shift according to Figure B.3 should between 43 kHz and 44 kHz. The result from the SGP4 simulation shows a maximum of 44.4 kHz for Ørsted and 43.7 kHz for AAU Cubesat, which means that the figure confirms the result of the previous section.

As for the Doppler shift rate of change, the dynamic nature of the system is very clear. The distribution has a mean of -29 Hz/s and can be compared to a shifted Beta distribution [23]. As it is evident, the drift is seldom above zero, meaning that the LEO satellites are always gaining on the GPS satellites. It also indicates that all signals starts out with a positive doppler shift, and then move towards the negative range, as this is the only way the entire spectrum can be covered. The Doppler shift drifting is bound to have implications on the acquisition process as stationarity assumptions are weakened, but this will be analyzed in detail later on.

In Figure C.3 the states of the system at epochs where a new satellite have just become visible to the receiver have been plotted. This is interesting as this is the situation where the signals

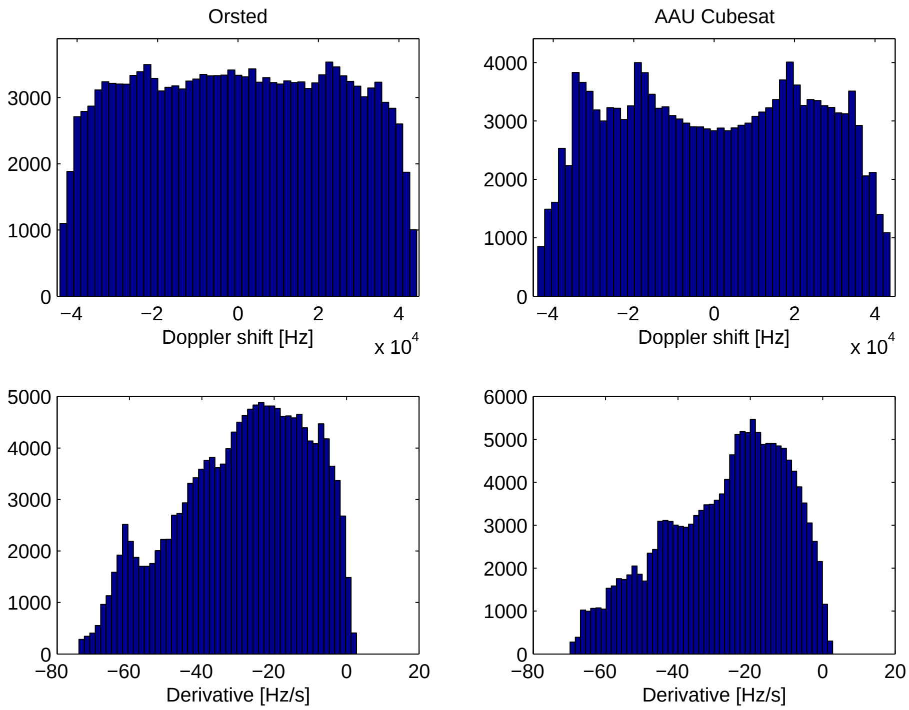


Figure 3.6: Distribution of the line-of-sight and rate of change doppler shift for the AAUSAT-II and Ørsted satellites according to 24 GPS satellites. The data have been collected over a 2 day period in 10 second intervals.

have to be acquired. As can be seen, the distribution can be approximated with a triangular distribution from 0 to $f_{d,max}$. This tells us that new signals, in the majority of cases, appear with a large positive Doppler shift. In the acquisition process this can be used to limit the number of frequencies that have to be searched, but this will be evaluated in detail in the next chapter.

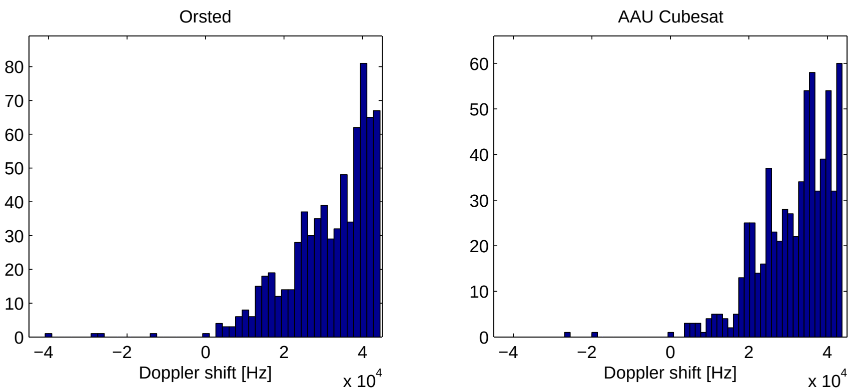


Figure 3.7: Distribution of the doppler shift at the epochs where a GPS satellite have just entered the sight of the receiving antenna.

3.3 Tumbling

According to Gomspace it is expected that the GOMX satellite would revolve 360°/orbit, meaning that the main payload, consisting of an antenna and camera, would be nadir-pointing at all time, i.e. pointing towards the ground. Similarly a GPS antenna, placed on zenith pointing surface of the satellite, would always point towards the GPS satellites. However, when considering the experiences from other cubesat missions, unintentional rotation or tumbling as its known is a common issue. One example is the AAUSAT-II that experienced severe tumbling which was in the order of a few hertz. Figure 3.8 shows the tumbling over time. The sudden dip in tumbling indicates the initiation of the de-tumbling of the satellite, which mitigates it considerably. However, the satellite never stabilizes totally. In the AAUSAT-II case the reason for this is a design flaw in the satellite, which makes the on-board computer restart due to overheating. As a consequence, the detumbling mechanism only works in short periods at a time, making it a gross example. But it goes to illustrate the tumbling issue.

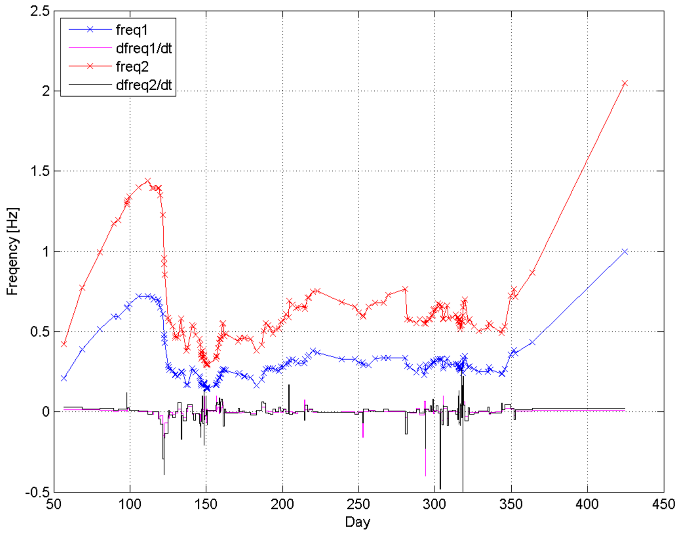


Figure 3.8: AAUSAT-II tumbling. The steep dampening around day 120 indicates the activation of the satellite detumbling. However, due to a unintended restart of the satellite computer, the de-tumbling only worked around 4-8 hours a day. Source: `www.space.aau.dk`

Another example is a test of the the Phoenix receiver on-board the PCsat mission which experienced slow tumbling [1]. In this situation a significant delay in TTFF of about 4 minutes was observed [1]. According to the DLR this was caused by the tumbling which lead to unfavorable antenna orientation and low SNR.

In summary thumbling is a realistic scenario for small satellites, and it would have an adverse effect on the GPS receiver performance. But in order to reduce the complexity of the project, confidence is given to Gomspace to design a good detumbling system, so the effects can be neglected.

Chapter 4

C/A Signal Acquisition

At this point the GPS signal has been descriped and the work environment of the receiver analyzed. Given this knowledge a basis has been estalished to describe the functionallity of the GPS receiver front-end and signal acquisition in relation to LEO operation. Also, the effects of the GPS signals dynamic behaviour on the acquisition process will be evaluated.

4.1 Front-end and antenna

Figure 4.1 shows the basic principle of a GPS receiver front-end. The GPS signals from the satellites visible to the receiver is received at the antenna and amplified using a low noise amplifier (LNA). Afterwards the signal is down-converted to either baseband or an intermediate frequency (IF) using a mixer and bandpass filter. At this point the signal is sampled to enable digital baseband or IF processing of the GPS signals. In most receivers there are a lot more steps involved, such as several stages of bandpass filtering, amplification and down-convertion, but Figure 4.1 shows the basic principles.

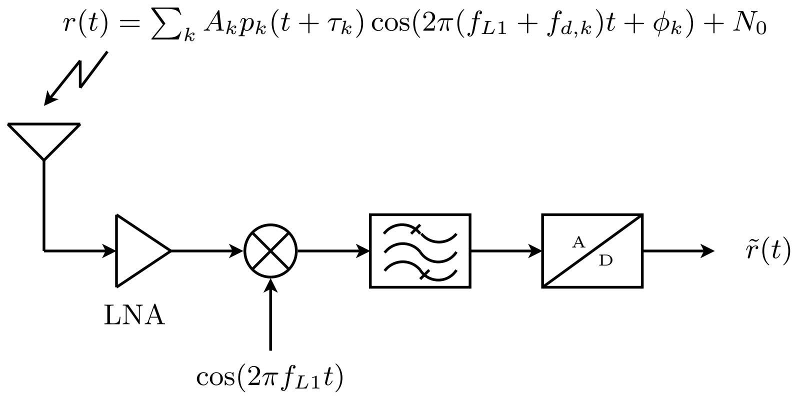


Figure 4.1: A GPS receiver front-end. The received signal is amplified using a Low-Noise Amplifier (LNA) and down-converted to baseband using a mixer and band-pass filter. Finally the signal is digitized for further baseband processing. The received signal $r(t)$ is represented as the sum of GPS signals plus noise.

As mentioned the received signal has components from all the satellites visible to the receiver. This is commonly known as a multiple access signal as it is possible to access one of multiple signals present on the same carrier. This is made possible by the fact that the binary codes $p_k(t)$ are near orthogonal to each other, i.e. they do not correlate with each other. Mathematically, the received signal can be represented as the sum of the GPS signals plus noise, which is given in Eqn. 4.1.

$$r(t) = \sum_{k \in S} A_k p_k(t + \tau_k) \cos(2\pi(f_{L1} + f_{d,k})t + \phi_k) + N_0 \tag{4.1}$$

where

S	is the set of visible GPS satellites
A_k	is the amplitude of the kth GPS signal
$p_k(t + \tau^k)$	is time shifted binary data and spreading code sequence of the kth satellite
f_{L1}	is the L_1 carrier frequency
$f_{d,k}$	is the doppler shift of the kth satellite
ϕ_k	is the carrier phase for the kth satellite
N_0	is noise

In Eqn. 4.1 each of the binary code sequences is delayed due to the propagation delay, but also because the code phase is unknown. That is, the C/A is 1023 chip long, but the receiver do not know which chip in the sequence is being received. The carrier phase ϕ_k describes the fraction of a L_1 wavelength that the signal is delayed by. Finally, due to the mobile nature of the system the doppler shift $f_{d,k}$ is included. After down-conversion the carrier is “striped’ of the signal, leaving:

$$\tilde{r}(t) = \sum_{k \in S} A_k p_k(t + \tau_k) \cos(2\pi f_{d,k} t + \phi_k) + N_0 \tag{4.2}$$

Hence, the output signal from the front-end is a series of spreading sequences phase modulated with independent Doppler frequencies plus noise.

Signal to noise ratio

As shown in Section 2.4 the GPS signal level for the L_1 C/A signal at the antenna is -157.5 dBW. This is a very weak signal and the noise induced by the antenna is usually much larger then GPS signal. To illustrate this, the noise power of an antenna can be calculated very easily using the relation

$$P_n = k \Delta f T_a$$

where k is Boltzmann’s constant, Δf the signal bandwidth and T_a the equivalent antenna temperature (not to be mistaken for the physical temperature - more on this in Appendix F). The bandwidth for the C/A signal is 2.034 MHz and a commonly used figure for the antenna temperature is 130 K [14][24]. The consequent noise power level is -144.4 dBW which translates into a signal-to-noise ratio, at the antenna, of

$$\text{SNR} = -157.5 - (-144.4) = -13.1 \text{ dBW}$$

Hence the GPS signal is well below the noise floor at the antenna. This is a fundamental feature of spread spectrum signaling, which has its origin in military communication where the signals have to remain hidden (i.e under the noise floor) from enemy forces [9]. From a signal processing perspective this might seem problematic, but as will be shown in the next section the SNR can be increased by baseband processing.

4.2 Acquisition

In this section the acquisition process will be analyzed based on the maximum likelihood (ML) estimator derived by Hurd et al. [25]. Furthermore, the available acquisition time is evaluated as well as how the acquisition results are affected by signal drifting.

The code phase τ and doppler frequency offset f_d related to a given spreading sequence $p(t)$ can be found by evaluating the maximum likelihood estimator, given by [25]:

$$L(\tau, f_d) = \frac{1}{N_0} \left| \int_{-T/2}^{T/2} \tilde{r}(t) p(t-\tau) e^{-j2\pi f_d t} dt \right|^2 \tag{4.3}$$

where N_0 is the noise power, $\tilde{r}(t)$ the down-converted signal as given in Eqn. 4.2 and T the integration period, which is usually set to the period of the spreading sequence. Most of the acquisition methods that can be found in the literature are approximations of Eqn. 4.3 implemented according some design metric usually time, computational complexity or probability of false alarm [26, 25, 27, 8].

Eqn. 4.3 can be interpreted as the (squared) correlation between the local copy of the spreading sequence $p(t)$ and the input signal $\tilde{r}(t)$ combined with a Fourier transformation over the range of possible Doppler shift frequencies. The correlation property becomes clear in the case where the received signal has a Doppler shift of f_d and contains a spreading sequence similar to $p(t)$, i.e. the received signal can be written as $\tilde{r}(t) = Ap(t)e^{j2\pi f_d t}$. Inserting into Eqn. 4.3 yields:

$$L(\tau, f_d) = \frac{1}{N_0} \left| \int_{-T/2}^{T/2} Ap(t) e^{j2\pi f_d t} p(t-\tau) e^{-j2\pi f_d t} dt \right|^2 \tag{4.4}$$

$$= \frac{A^2 T^2}{N_0} \left| \frac{1}{T} \int_{-T/2}^{T/2} p(t) p(t-\tau) dt \right|^2 \tag{4.5}$$

$$= \frac{A^2 T^2}{N_0} r_{pp}^2(-\tau) \tag{4.6}$$

Eqn. 4.5 is the SNR of signal multiplied with the squared continuous-time auto-correlation of $p(t)$. At the code phase τ where the received and local sequence are aligned, the correlation will produce a large peak, which indicates that a code is present at this coordinate in the (τ, f_d)-plane. The magnitude of the peak is determined by the so-called processing gain (PG), which, for DSSS systems, is the ratio between the integration period T and the chip period T_c of the spreading sequence [9].

$$\mathrm{PG} = \frac{T}{T_c} \tag{4.7}$$

If the codes are not aligned the correlation will not produce a peak, indicating that no code is present at the coordinate under consideration. Similarly, if no peak is produced over the entire (τ, f_d)-plane, no signal with the desired spreading sequence is present in the received signal. An example of the (τ, f_d)-plane, or search space as it is also known, is seen in Figure 4.2. In this case the peak generated by the ML estimator is very distinct. But with deteriorating SNR the peak gets smaller, which eventually makes it impossible to destinguish the peak from peaks generated by noise.

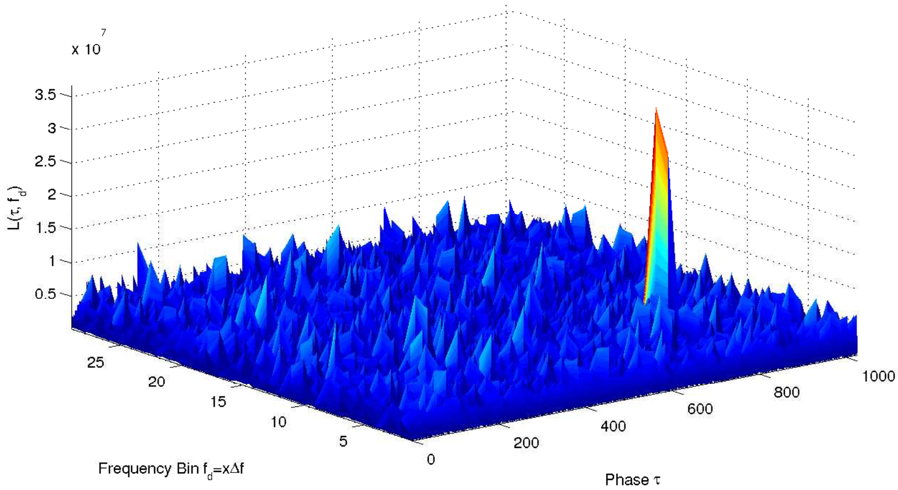


Figure 4.2: Example of the ML estimator output working on raw GPS data sampled from a professional GPS receiver. The plot has been generated using the framework provided in [8], which is based on an approximation of the ML estimator.

A common way to implement the ML estimator is shown in Figure 4.3. The received signal is mixed with a carrier that is offset by a small amount Δf_d and then correlated for all lags between $T/2$ and $T/2$ in steps of $\Delta\tau$. Then the carrier frequency is incremented by Δf_d and correlation performed again. This continues until the (τ-f_d) plane has been searched. Once the search space has been explored, the code phase τ and frequency f_d that yields the maximum output is the ML estimate of the phase and frequency related to the spreading sequence $p(t)$.

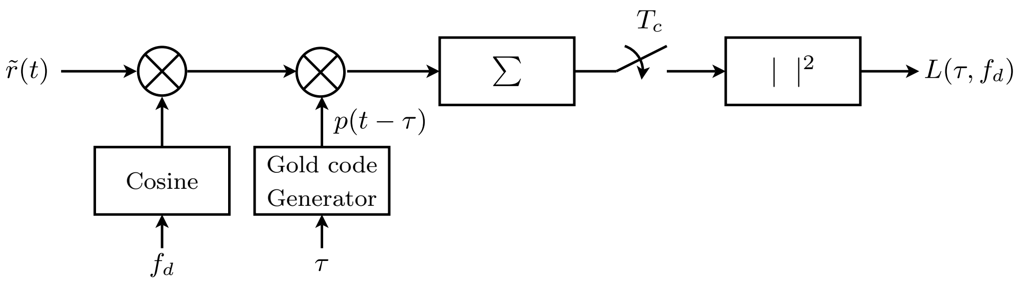


Figure 4.3: Serial search acquisition. Code lag τ and frequency offset f_d is incremented one at a time until the whole search space has been explored.

This method is known as serial search acquisition and is an approximated version of the ML estimator (AMLE). Now, as one could imagine the time it takes to acquire the signal, which depends on the range of the (τ-f_d) plane and frequency and phase step size, is potentially very large. And even more so if all code phases and frequencies have to be searched sequentially as in Fig. 4.3. Hence it is necessary to investigate the size of the space that has to be searched.

4.2.1 Search space Analysis

Instead of evaluating the ML estimator in Eqn. 4.3 for all possible coordinates in the (τ-f_d) plane, it is sectioned into a number of cells, where the size of each cell is determined by the frequency and phase step size, given by Δf_d and $\Delta\tau$. Each cell is then evaluated accordingly using Eqn. 4.3. The evaluation of a cell is referred to as a dwell and the time is takes to evaluate one cell the dwell time. If the range of the search space is given by (T_0, F_0), then the total number of cells that has be searched is given by:

$$M = \frac{T_0 F_0}{\Delta f_d \Delta\tau} \tag{4.8}$$

Hence the precondition to find the size of the search space is to 1) determine the sizes of the cells, and 2) determine the range of the (τ-f_d) plane. From Eqn. 4.8 it is also clear why it is of interest to divide the search space into larger cells. E.g. let $T_0 = 1000$ and $F_0 = 5$ kHz then, if the search space was to be evaluated at a 1 Hz resolution, M would be 5 million. If, on the other hand, $\Delta f_d = 500$ Hz then M would be reduced to 5000. Hence the search space can be reduced drastically by increasing the cell sizes. However, the increased cell size also affects the performance of the ML estimator, which has to be investigated.

In order to illustrate the effects[1], Eqn. 4.3 is evaluated when the SNR is high, which means that the down-converted signal can be represented as $\tilde{r}(t) = Ap(t)$, where A is the amplitude of the received signal. By inserting $\tilde{r}(t)$ into Eqn. 4.3, the ML estimator can be written as [25]:

$$L(\tau, f_d) = \frac{A^2T^2}{N_0}|\Phi(\tau, f_d)|^2 \qquad \text{where} \tag{4.9}$$

$$\Phi(\tau, f_d) = \frac{1}{T}\int_{-T/2}^{T/2} p(t)p(t-\tau)e^{-j2\pi f_d t}dt \tag{4.10}$$

By solving 4.10 for $\tau = 0$, the effects of changing doppler f_d can be evaluated separately as $p(t)p(t) = 1$. This yields

$$\Phi(0, f_d) = \frac{\sin(\pi f_d T)}{\pi f_d T} \tag{4.11}$$

Similarly, when $f_d = 0$ Eqn. 4.10 reduces to the correlation between the spreading codes. For the C/A codes, which are Gold codes with $n = 10$, the correlation is given by [8].:

$$\begin{aligned}\Phi(\tau, 0) &= \begin{cases}(2^n - 1)(1 - |\tau|/T_c) & \text{for } |\tau| \le T_c \\ \le 2^{(n+2)/2} + 1, & \text{otherwise.}\end{cases} \\ &= \begin{cases}1023(1 - |\tau|/T_c) & \text{for } |\tau| \le T_c \\ \le 65, & \text{otherwise.}\end{cases}\end{aligned} \tag{4.12}$$

The effects of changing τ and f_d in Eqn. 4.11 and 4.12 are illustrated in Figure 4.4. As expected the correlation produces a very large peak when the codes correlate, but is attenuated considerably elsewhere. This means that the code phase search steps $\Delta\tau$ have to be small and from Eqn. 4.12 one can see that $\Delta\tau$ has to be smaller than or equal to the chipping period T_c. This also concurs with receivers considered in literature, where correlation is performed at a sub-chip level (usually half chips resolution) [24].

[1] The investigation is a restatement of the work done by Hurd et al. in [25].

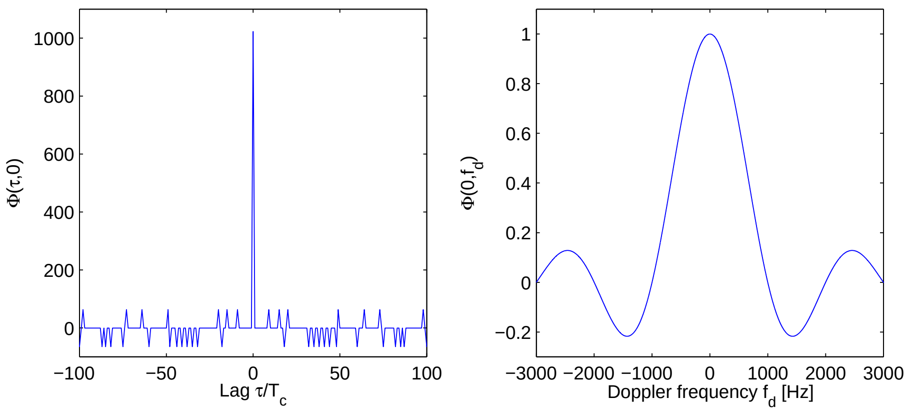


Figure 4.4: Shows the behavior of the ML estimator for zero frequency and zero phase cases. The correlation peak is clearly seen as well as the sinc behavior of the frequency.

Regarding the frequency, Fig. 4.4 shows that the ML estimator is affected less by changes to f_d. This means that, by accepting a small degradation in the output of the estimator, the frequency bin size can be increased considerably. In most receivers a degradation of -1 dB is accepted [24] which, by intersection in Eqn. 4.11, can be found to yield a frequency bin size of $\Delta f_d = 523$ Hz. This is usually rounded to 500 Hz, which seems to be the de facto frequency bin size for acquisition in GPS receivers [8][24].

All that is needed at this point is the range of the (τ-f_d) plane. T_0 is given as the Gold codes have a length of 1023. Regarding the range of the frequency, F_0, two cases have to be considered; namely terrestrial and space receivers. In terrestrial receivers the doppler shift caused by the relative movement between the receiver and GPS satellites can be assumed not to exceed ± 5 kHz [24][8]. Hence a 10 kHz search range will be used as reference. In space, the maximum Doppler shift was found to be approximately 45 kHz (see Section 3.2). But by taking into account that new satellites have a positive doppler shift only, the frequency search range can be reduced to the interval $[0, f_{d,max}]$. In most receivers half-chip correlation is used in the tracking loops[2]. Hence this will also be used as the outset, i.e. $\Delta\tau = 0.5$.

It is now possible to find the size of the search space, which is shown for the terrestrial and space receiver case in Table 4.1. Also the search time, assuming serial search acquisition and an integration period of 1 ms, is shown.

		Terrestrial	Space
F_0	[Hz]	10000	45000
T_0	[chips]	1023	1023
Δf_d	[Hz/cell]	500	500
$\Delta\tau$	[chips]	0.5	0.5
M	[cells]	40920	184140
Dwell time, T	[ms/cell]	1	1
Code Acq. Time	[s]	40.9	184.1

Table 4.1: Comparison between search space size and search time for terrestrial and space acquisition.

[2]Most tracking loops use a phase tracking method known as early-prompt-late, where the signal is correlated at three lags places 1 half chip apart and where the feedback is controlled by the outcome of these correlations.

The results show that it takes 40.9 s to acquire one satellite using serial search acquisition in terrestrial receivers. For the entire constellation of 32 GPS satellites this amount to 21.8 min. For LEO application one acquisition take 184.1 s, which amounts to 98.2 min for a full acquisition. This high acquisition time illustrates the sheer size of the acquisition search space. In order to reduce the acquisition time, different methods exists to reduce to search space. If all phase-frequency cells are searched the acquisition method is referred to as non-aided or cold-start acquisition. In aided or warm start acquisition, orbit models are used to give a rough estimate of the phase and frequency, which is the method used in the Phoenix receiver. This greatly reduces the search space and hence the acquisition time. E.g. the Phoenix receiver achieves a time-to-first-fix of just 1 min. using warm start.

However, alternative ways also exist to speed up the acquisition process. One such method is to decrease the dwell time [28, 29], whereby more cells can be searched in a shorter period. This also increases the frequency bin size, which means that the number of cells can be reduced even further. The problem is that the processing gain is decreased as well. This has widespread effects on the performance of the system, e.g. the link margin in the link budget has to be taken into account when designing and evaluating the system. The peak values in the acquisition output is smaller and thus closer to the noise peaks. Also tracking loops have to be able to cope with the less precise results given by the acquisition process. These are all new problems that have to be considered.

Another aspect is that the purpose of the coarse acquisition (C/A) code is to enable receivers to acquire the GPS signals - P(Y) codes included. So a code period of 1 ms is not a random choisce made by the designers, but more likely a result of carefull analysis of how long the integration time has to be in order to assure possible acquisition. This is also reflected by the fact that almost all receivers use a integration period of 1 ms and the derived 500 Hz size of the frequency bins.

All in all, reducing the dwell time is a cumbersome way to reduce the acquisition time as it induces many new problems. As such the method will not be considered further. The final option is to use more hardware to accelerate the process, which will be discussed in detail later.

4.2.2 Available Acquisition Time

In the ideal case navigation data starts being collected the epoch a GPS satellite signal appears in the received signal, which maximizes the utilization of the GPS signals. However, as it is evident from the previous section, some time must be allocated for acquisition. Hence, the outset is to find an optimal ratio between acquisition and data gathering time. A case is considered optimal if the utilization is maximized, while all satellites are found by the acquisition. If a satellite is not found, the utilization drops as no data is gathered. The assumption is that acquisition with few misses is more important than reducing acquisition time. This is based on the fact that the mean time window for GPS satellites is approximately 2500 s (see Appendix C), meaning that missed satellite results in loss of a lot of data, whereas extra acquisition is more affordable.

In order to find the available acquisition time in LEO, the statistical results from Section 3.2 will be used. From the analysis it is known that the signals at epochs where they have just become visible to the receiver are triangular distributed in the interval $[0, f_{d,max}]$ as given in Eqn. 4.13.

$$f_{X_0}(x) = \frac{2}{f_{d,max}^2}x, \quad 0 \leq x \leq f_{d,max} \tag{4.13}$$

If, as proposed, only this interval will be searched by the acquisition algorithm, then it cannot be avoided that some of the GPS signal will be missed, due to the constant Doppler shift drifting.

The reason being that they drift out of the search area. And as the signals drift in the negative direction, the exposed signals are those close to 0 Hz.

Given the triangular distribution, the percentage of signals that resides above a threshold limit x_0 can be found, as is illustrated in Figure D.1.

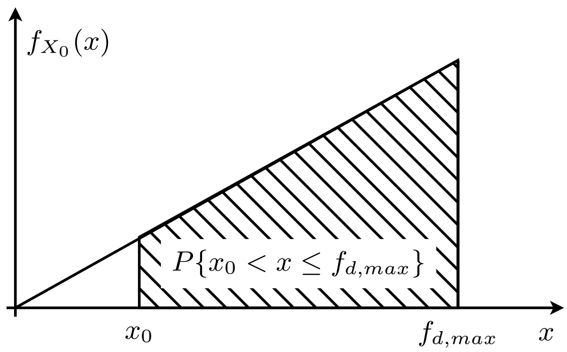


Figure 4.5: Triangular distribution of signals that have just become visible to the receiver.

$$x_0 = \sqrt{1 - P\{x_0 \leq x \leq f_{d,max}\}} f_{d,max} \tag{4.14}$$

At a maximum doppler shift of 45 kHz, this means that 99% of the acquirable signal lies between $x_0 = 4500$ Hz and 45 kHz, which can be found using Eqn. 4.14 where $P\{x_0 \leq x \leq f_{d,max}\} = 0.99$. By assuming that the drift between two epochs is independent and that the mean drift and variance is μ and σ^2, then it is shown in Appendix D that 99% of the satellites can be acquired at a 5% significance level, when the acquisition time does not exceed:

$$\frac{x_0 + n\mu}{\sqrt{n}\sigma} \leq 1.6449 \tag{4.15}$$

$$n \leq 143 \text{ s} \tag{4.16}$$

where the mean drift $\mu = -29$ Hz/s and the drift variance $\sigma^2 = 318$ (which is the drift parameters found in Section 3.2). An acquisition time of 143 s seems like a fair ratio between acquisition and data gathering time, when compared with the mean time window of 2500 s. Hence, this figure will be used further-on. However, the method presented does have its flaws, which means that the 143 s serves more as a guideline, than a clear-cut constraint. Especially the assumption of drift independence is weak due to the underlying periodic nature of the system. The consequence of this needs to be investigated, but is left for future projects to explore due to limited project time available.

4.2.3 Dynamic Behavior of Acquisition Results

Due to the drifting of the code phase and frequency of the GPS spreading sequences, the acquistion results are also going to drift. In order to evaluate the nature and severity of the acquisition result drifting a larger statistical analysis has been performed based on the drifting characteristics found in Section 3.2. The analysis can be found in Appendix E. The result is a set of code phase and frequency bounderies that specifies an area in the search space where the acquisition results can be found t seconds after the initial acquisition. Given these bounderies one is capable of predicting the path of the acquisition results at a 5% significance level, as long as the mean drift and variance is known. The bounderies are given by:

$$f_{min}(t) = t\mu - \sqrt{t}\sigma 1.6449 + (m-1)\Delta f \quad (4.17)$$
$$f_{max}(t) = t\mu + \sqrt{t}\sigma 1.6449 + m\Delta f \quad (4.18)$$
$$p_{min}(t) = p_0 + \frac{T_0}{f_{L1}T}\left\{\frac{\mu\, t^2}{2} - \frac{2\sigma 1.6449\, t^{3/2}}{3} + \Delta f\,(m-1)\,t\right\} \quad (4.19)$$
$$p_{max}(t) = p_0 + \frac{T_0}{f_{L1}T}\left\{\frac{\mu\, t^2}{2} + \frac{2\sigma 1.6449\, t^{3/2}}{3} + \Delta f\, m\, t\right\} \quad (4.20)$$

where m and p_0 are the initial acquisition frequency bin number and phase, T_0 the code length, T the dwell time and Δf the frequency bin size. Figure E.5 shown the predicted bounderies over a period of 150 s for acquisition results with different initial code frequencies but the same phase. The lines between the upper (green) and lower (blue) bound indicate the vector between the coordinates $[p_{min}(t), f_{min}(t)]$ and $[p_{max}(t), f_{max}(t)]$. The angle of the vector increases over time, indicating that number of possible phase values also increases.

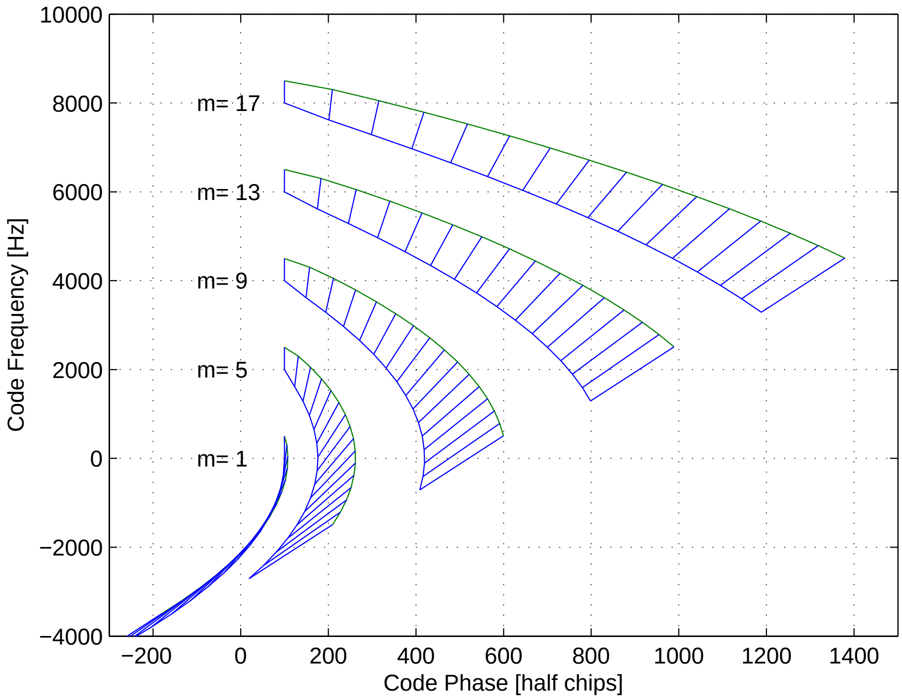


Figure 4.6: Frequency and phase prediction in 10 s steps for signals with an initial code phase of $p_0 = 100$, but with different initial frequency values. The prediction is based on a mean doppler shift drift of $\mu = -29$ Hz/s and variance of $\sigma^2 = 318$, and a bin size of $\Delta f = 500$ Hz

The figure gives a good impression of the dynamics effect of LEO and it shows that both the code phase and frequency change significantly over the acquisition time period found in the previous section. To illustrate the severity furhter, it was found that with a Doppler shift of 45 kHz (worst-case) the signal drifts half a chip in approximatelyt 17 ms (see Appendix E).

As most tracking loops are narrow both in phase and frequency this means that the time from acquisition to tracking has to be minimized. Otherwise the tracking loops is not able to lock onto the code. It is therefor proposed that tracking commences as soon as correlation peaks occur. In this way it is insured that the signals are not invalidated due to drifting.

Chapter 5
Summery

In the previous chapters the initial and very broad project description has been analyzed in order to establish a more profound understanding of the application and the problems that is related with the reception and acquisition of GPS signals in LEO.

Both analytical results and analysis of the Ørted and AAU Cubesat missions, showed that the Doppler shift induced on the L_1 carrier of the GPS signal, was significantly larger than what would be expected in terrestrial environments. E.g. the maximum Doppler shift in LEO is 45 kHz, compared to 5 kHz on earth. Furthermore statistical results showed that the Doppler shift drifts with mean of -29 Hz/s in the Ørsted and AAU Cubesat orbits and that the Doppler shift drift is only positive very few cases. The latter is because the orbit velocity in LEO is much larger then at the altitude of the GPS satellites, which means that a LEO receiver always gains on the GPS satellites. Satellite tumbling was also identified as a possible problem for small satellite in LEO and experiences from previous missions show that the GPS receiver performance is affected negatively by tumbling. However, in order to limited the extent of the project, it assumed that the receiver platform is stable and that the GPS antenna is zenith looking at all time.

A consequence of the Doppler shift is that the GPS spreading codes are phase modulated with the Doppler shift residual when sampled at the output of the GPS front-end. Similarly the phase of the spreading sequences are unknown due to the propagation delay from the transmitting GPS satellite to the receiver. In order to acquire the phase and Doppler frequency a maximum likelihood estimator was presented and analyses. This showed that due to the large Doppler shift the acquisition search space is larger in LEO. However, it was also found that at the moment where a new GPS satellite have just become visible to the receiver the Doppler is never, or in very few cases, negative. This means that it is only necessary to search the upper half of the search space, if only new satellites are to be acquired. However, in cases where the GPS receiver returns from a power-down all GPS signals still needs to be acquired, which means that the hole search space between ± 45 kHz has to be searched. Although this is realistic scenario in space, it was chosen to concentrate on the acquisition of new satellites as the main case for this project, to limit the project. That is acquisition has to be performed between 0 and 45 kHz for all phases.

Finally a commercial space GPS receiver called Phoenix from DLR was presented. This receiver has a power consumption below 1 W, small size, features fast warm start acquisition and is designed for sporadic power-downs. As such the receiver fulfills most of the Gomspace demands apart from the wish to implement the receiver in FPGA. Also in order to perform as specified the receiver relies on updated TLEs, which is not always available. Hence, the performance of the receiver might fluctuate over time and as such a cold start acquisition would be a more robust solution.

In order to limit the project the following chapters will deal with a cold-start acquisition algorithm based on the maximum likelihood estimator presented. Furthermore it is assumed that receiver is in continuous operation in orbit, which means that the search space can be halved.

Part II

Algorithm Analysis

Chapter 6

System Exploration

In nature space environment imposes more harsh constraints on the hardware solution compared to earth. Consequently, in order to design a reliable space GPS receiver, the environmental factors cannot be ignored. Also, the hardware solution chosen for the space environment is going to impose constraints on the rest of the receiver. Hence, within the scope of APSI this chapter can be seen as an exploration of the design constraints for the acquisition algorithm .

6.1 GPS Antenna

The receivers GPS antenna is the most exposed part of the system as it has to be placed on the outside of the satellite, in order to receive the faint signals from the GPS satellites. The antenna is assumed to be zenith-looking at all time, i.e. pointing away from the earth and towards the GPS satellites at all time. The zenith pointing surface of the satellite is rather small at approximately 10x10 cm. Most of this area is covered with solar-panels, which leaves little space for a GPS antenna. Consequently the antenna has to have a very small footprint. Furthermore, the surface temperature of the satellite fluctuates between approximately -30 to +85 °C. Hence, the antenna has to cope with both the extreme temperature values as well as the continuous temperature cycling.

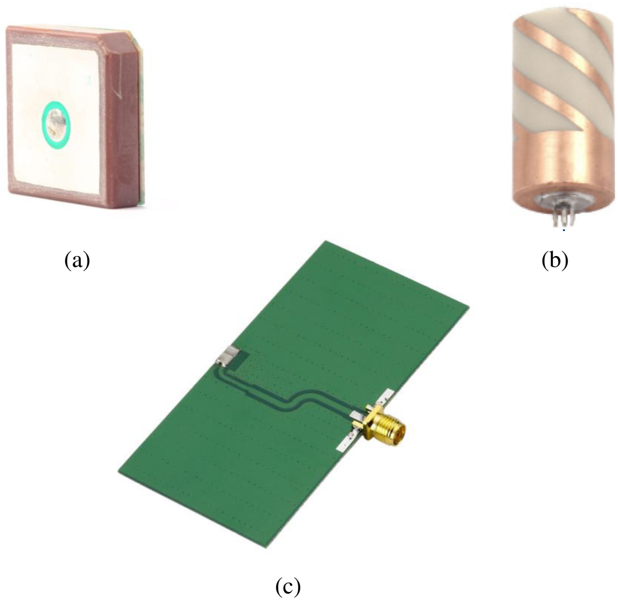


Figure 6.1: Different small size GPS L_1 antennas. (a) Taoglas SGP.12 patch antenna (b) Sarantel SL1300 Helix antenna (c) Taoglas GLA.1 loop antennas

Figure 6.1 shows three different types of small-footprint L_1 GPS antennas. The data for these along with other possible antennas are given in Table 6.1. The antennas have been chosen for their small size and weight. A full survey on existing GPS antennas made by GPS World can be found in the enclosed material on the CD. From this survey it is clear that most GPS antennas are quite large and heavy, which disqualifies them for this project.

Figure 6.1(a) is a 12x12x4 mm right hand polarized patch antenna from Taoglas' SGP antenna series. According to Taoglas the SGP series is specially designed to circumvent the cracking of the feed-point (terminals) which is a common problem for patch antennas in environments with high temperature and vibrations[1]. The SGP series is rated to temperatures between -40 °C and +105 °C and accelerations up to 20 G. This makes the SPG series an obvious candidate for a space GPS antenna, as it satisfy the demands posed in the introduction.

	Taoglas					Sarantel	
	SGP.12	SGP.15	SGP.18	GLA.1	GLA.2	SL1200	SL1300
Type	Patch	Patch	Patch	Loop	Loop	Helix	Helix
Dim. [mm]	12x12x4	15x15x4	18x18x4	5x1.3x0.6	10x3.2x4	10.1Øx17.8	7.5Øx12
Polarization	RHCP	RHCP	RHCP	Linear	Linear	RHCP	RHCP
Gain [dBi]	-2.0	1.0	1.0	2.5	1.34	-2.8	-5.0
Beamwidth [°]	n/a	n/a	n/a	n/a	n/a	135	>135
Bandwidth [MHz]	6	6	5	50	20	20	15
VSWR (max)	1.5	1.5	1.5	2	2	2.3	2.3
Temp. [°C]	-40 - 105	-40 - 105	-40 - 105	-40 - 105	-40 - 105	-40 - 85	-40 - 85
Weight [g]	6	8	12	1	1	n/a	3

Table 6.1: Comparison between different small-footprint passive GPS L_1 antennas. The gain for the patch and helix antennas are at zenith and the loop antennas its specified for as the peak gain. For Taoglas antennas the bandwidth is relative to a return loss less then -10 dB where the Sarantel antennas are -3dB.

Two other obvious antenna candidates are the Taoglas GLA.1 and GLA.2 loop antennas. They both have a very small footprint, high gain and a relatively low VSWR. However, the gain specified is peak gain and they also require a large ground plane as shown in Fig. 6.1(c). Further more the antennas are prone to a polarization loss of -3 dB as they are linear polarized. Figure 6.1(b) is helix antenna from Sarantel. It has been included to show the diversity of the available GPS antennas.

From the above the Taoglas SGP.12 passive patch antenna was chosen for further analysis. The choice is mainly based on its supposed sturdiness, small size and weight. The main goal of the analysis is to determined the noise temperature of the antenna when exposed to the space environment. One might question the necessity of such an analysis as references to antenna noise are abundant in GPS literature. The problem, however, is that all references to antenna noise found by this author, are for terrestrial applications and not space. Consequently, a thorough analysis on the antenna noise temperature has been conducted based on a 0 dBi zenith-looking patch antenna, which can be found in Appendix F. The results are summarized in Table 6.2. In general the analysis is very conservative, so the best and worst-case should be upper limits. As can be seen the the table the total antenna noise temperature is very dependent on whether the antenna is exposed to the sun or not. This is due to psychical temperature of the temperature and the detailed reasons for this can be found in the appendix. The antenna noise temperature is a

[1] http://www.gpsworld.com/consumer-oem/news/taoglas-unveils-patch-antennas-easy-assembly-9

bit lower compared with terrestrial applications, where 130 K often is used as an outset for GPS receivers [14, 24].

Temperature	Best Case (No sun) [K]	Worst Case (Sun) [K]
Brightness ($G = 0$ dBi)	3	4.6
Transmission Line (RF174)	6.5	8.9
Aperature ($e_{cd} = 0.8$)	58.3	95.8
Total	67.8	109.5

Table 6.2: Results of the antenna noise analysis for a 0 dBi zenith-looking patch antenna on a space vehicle.

Although the SGP.12 antenna would have been the preferred solution, due to practical reasons an active antenna from Taoglas have been used instead. One of these reasons is the need for a separate LNA when using passive antennas, which would complicate the solution. Also, the antenna have previously been used as a subject for investigation in relation to the AAU space program, which means that it have already been acquired. So to minimize the amount of work in relation to the receiver front-end this solution was chosen.

The active antenna is a Taoglas AP-10B active patch antenna. It is 10x10 mm and features a 2-stage LNA. Assuming that the antenna noise temperature is the same as found in the analysis the combined antenna noise temperature including LNA noise can found. The result is given in Table 6.3

		Value	Comment
Antenna Gain	G_0	-3 dB	
LNA Gain (min.)	G_1	23 dB	
NF	F_1		
- 25°C		1.4 dB	
- 85°C		1.8 dB	
Antenna Temp.	T_{ant}		From Tabel 6.2
- Earth		130 K	
- Space (WC)		110 K	
- Space (BC)		89 K	
LNA Temp.	T_{LNA}		By convention $T = 290(F - 1)$ [9]
- 25°C		110 K	
- 85°C		148 K	
Equivalent Temp.	T_e		$T_e = T_{ant} + T_{LNA}$
- Earth		240 K	re. 25°C
- Space (WC)		258 K	re. 85°C
- Space (BC)		199 K	re. 25°C

Table 6.3: Combined antenna noise temperature including LNA noise.

6.2 Front-end

Abundance of GPS receiver RF front-ends exists, but few have been designed or tested for use in space. However, there exist an obvious front-end candidate, namely the Zarlink GP2015. This front-end is used in the commercially available Phoenix space GPS receiver made by DLR[2]. DLR have conducted thorough test on the front-end with respect to radiation, vacuum and temperature

[2]Deutzche Luft und Räumfart zentrum

[1]. Hence, the front-end should fulfill all requirements for a space GPS receiver, which is why it has been chosen for this project as well.

The GP2015 is designed for the L_1 C/A signal [30]. The signal is down converted to an intermediate frequency of 4.309 MHz and sampled using a 2-bit sign/magnitude quantizer. On average the magnitude bit is set high 30% of the time, which is insured by an internal AGC. The sign and magnitude data are latched by the rising edge of a sample clock, which have to be provided from external hardware. This also means that the sample frequency can be freely chosen by the designer, by polling the data. The sample frequency is limited by a sample clock to sign/magnitude delay of 20 ns. This translates into a maximum sample frequency of 50 MHz.

The front-end does require some external components, such as filters and a clock. Designing these circuits and PCBs are well beyond the scope of this project, which is why a preassembled front-end have been acquired from GPSCreations.com, in cooperation with GomSpace. The acquired front-end is depicted in Fig. 6.2.

Figure 6.2: GPS501 RF front-end from GPS Creations.

The GP2015 front-end amplifies and filters the GPS signal in several stages. But as well as amplifying the signal, each of the stages will also contribute with noise. To account for this the gain of the first two stages as well as noise factor have been restated in Table 6.4.

		Value	Comment
Gain			(typical values)
- 1-stage	G_2	18 dB	
- 2-stage	G_3	27 dB	
NF			
- 1-stage	F_2	9 dB	
- 2-stage	F_3	n/a	
Front-end Temp.			
- 1-stage	T_{IF}	2014 K	$T_{IF} = 290(10^{F_2/10} - 1)$
- 2-stage		n/a	

Table 6.4: Gain and noise factor for the two first down-conversion stages in the GP2015 [30], as well as the equivalent noise temperature.

As can be seen the equivalent noise temperature is quite larger compared to that iof the antenna and LNA. However, as the antenna, LNA and front-end are in cascade Friis formula is invoked, which means that the contribution from T_{IF} is divided with the gain of the LNA. The overall noise and consequent signal-to-noise ratio will be considered in the next section.

6.3 Link Budget

In Table 6.5 the equivalent noise temperature at the terminals of the entire receiver have been calculated using Friis' formula [9, 31]. The contribution of the third stage of the front-end can be omitted, due to the high gain at this point. As can be seen the noise temperature in the worst-case scenario in space is only bit worse than when receiving the GPS signals on earth. But in the best-case scenario, the noise is considerably lower than on earth. In space all of these fluctuation can be lead back to the physical temperature of the system and thus the heat radiation from the sun and cold of space.

		Value	Comment
Eqv. receiver temp.	T_e		$T_e = T_{ant} + T_{LNA} + T_{IF}/G_1$
- Earth		250 K	
- Space (WC)		268 K	
- Space (BC)		209 K	
Signal Power	P_s	-157.5 dBW	[6]
Signal Bandwidth	B	2.046 MHz	
Noise Power	P_n		$P_n = 10\log(kBT_e)$
- Earth		-141.5 dBW	
- Space (WC)		-141.21 dBW	
- Space (BC)		-142.3 dBW	
Quatization noise (2-bit)	P_Q	0.88 dBW	[14][24]
Signal-to-noise ratio			$SNR = P_s - (P_n + P_Q)$ dB
- Earth		-16.86 dBW	
- Space (WC)		-17.17 dBW	
- Space (BC)		-16.09 dBW	

Table 6.5: Link budget for the receiver system, consisting of an AP-10B active patch antenna from Taoglas and the Zarlink GP2015 front-end.

The resulting signal-to-noise at the receiver terminals including quantization noise is -17.17 dBW in the worst-case. This figure concur with current literature, where similar figures have been used. I.e -19.1 dBW in [12] and -18 dBW in [24]. Hence, the noise level found in space is equivalent to those in terrestrial receivers or better.

Summary

In the previous the receiver antenna, LNA and front-end have been chosen and analyzed in order to evaluate the performance of the chosen solution. Care have been taken to distinguish between the different cases that the receiver have to work under. This also includes earth, as this is the only feasible test scenario for the system. The chosen antenna is not optimal in the sense that the LNA is integrated in the antenna, which in a space scenario would degrade the performance of the LNA as higher noise figure comes with higher temperature. But due to practical reasons the active antenna was chosen anyway.

The combined receiver front-end system is illustrated in Figure 6.3. In order to drive the LNA in the antenna a bias-tee have been included. Any noise contributions from this component have been ignored. The performance of the system have been summarized in Table 6.6.

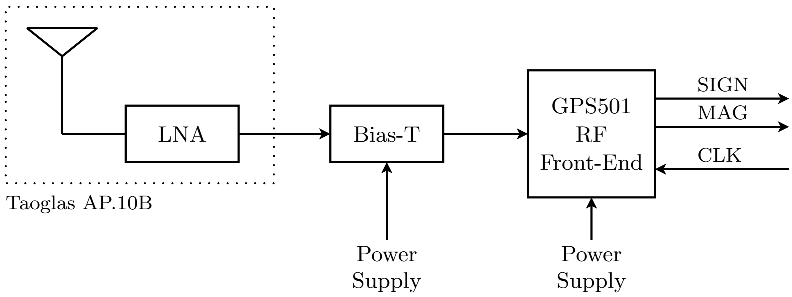


Figure 6.3: Overview of the hardware setup

Performance Figures	
IF	4.309 MHz
Bandwidth	2.046 MHz
SNR	
- Earth	-16.86 dBW
- Space (WC)	-17.17 dBW
- Space (BC)	-16.09 dBW
Quantization	2-bit Sign/Mag
Current	
- LNA (max)	13 mA
- Front-end (max)	101.5 mA
Total	114.5 mA

Table 6.6: Performance figures for the L_1 C/A receiver front-end.

In order to reduce the power consumption the supply voltage have to be chosen as low as possible. Fortunately, both the LNA and GP2015 are rated down to 2.7 V, which translates into a total power consumption of 309 mW or 31% of the total allowable power consumption. Hence, given the upper limit of 1 W the power constraint for the FPGA is 691 mW.

Chapter 7

Algorithm Exploration

When mapped to the algorithm domain, the design space of the ML estimator acquisition is huge, due to the countless ways that it can be realized. This poses a problem as it is impractical to search through all solutions to find the optimal one. Hence, a clever way to explore the design space must be considered. When such a method has been found, the algorithm for the ML estimator must be evaluated to minimize costs related to implementation, but without degrading performance. This process is the subject of the following chapter.

7.1 Design Space Exploration

In order to evaluate the characteristics of the design space, the work done by Matthias Grais from University of California has been used as an outset [32]. In Chapter 3 and 4 of his work a generic collection of methods for evaluating and exploring the design space have been compiled. That is, it is written without a case in mind, but describes some of the most important aspects that have to be considered when exploring the design space. By applying these considerations to the acquisition case, the design space and how it could be evaluated, should become clear.

In the application analysis, the acquisition process and effects of LEO was considered at a high level of abstraction using analytical and statistical models. This provided information about system time constraints, search space size and about the dynamic nature of the acquisition results.

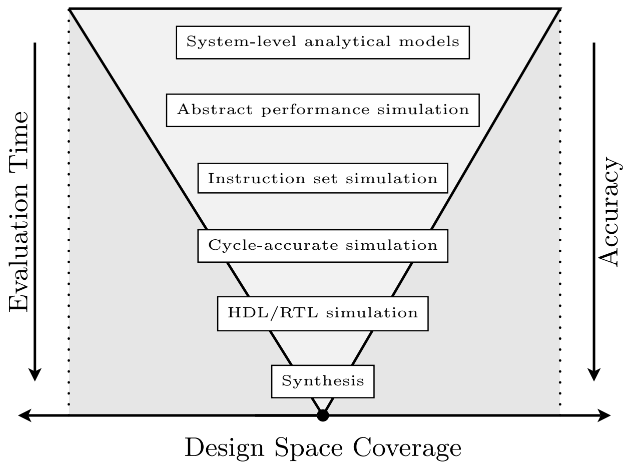


Figure 7.1: The design funnel model. Fine grain evaluation methods narrow the reachable design space, whereas coarse grain are less accurate but feature lower evaluation time and larger coverage [32].

When mapped to the algorithm domain, the acquisition process can be realized in a number of ways that adheres to e.g. the time constraint, but the question is which one to chose and how to evaluate the cost of possible solutions. Due to the many possible solutions the design space is huge and a low-level implementation of all solutions in order to evaluate their performance would be very time consuming. Hence, it is necessary to choose the right level of abstraction in order to achieve the desired coverage of the design space. The overall trade-off scenario is illustrated by the funnel model, as shown in Figure 7.1. In the following design space exploration (DSE) the level of abstraction will be chosen according to the problem at hand, but in general a high-level approach is chosen in order to limit the evaluation time (i.e. the time used to evaluate the search space). The trade-off is a less accurate picture of the search space, but this is the nature of VLSI design and must be accepted.

The next three sections, will cover the questions when can a solution be considered optimal, how is the cost of the implementation found and how is it insured that the design space have been covered.

7.1.1 Optimization Strategy

Before an optimization strategy can be considered, it is necessary to define when a solution can be considered optimal. In [32] this is achieved by introducing a concept called Pareto-optimality. Pareto optimal is based on two definitions:

Def. 1 - Pareto criterion for domination *Given two solutions A and B with values* $\{a_1, a_2, ..., a_k\}$ *and* $\{b_1, b_2, ..., b_k\}$*, and* k *objectives to be minimized, then solution A dominates B if and only if:*

$$\forall_{0 \leq i \leq k}\ i : a_i \leq b_i \qquad \text{and} \qquad \exists\ j : a_j < b_j$$

Def. 2 - Pareto-optimal solution *A solution is called Pareto-optimal if it is not dominated by any other solution.*

By Def. 1 a solution dominates another solution, if it is better in at least one objective and is at least the same in all others objectives. Any solutions that is not dominated by any other solution is considered Pareto-optimal (Def. 2). In Figure 7.2 a two-dimensional design space is shown with execution time and cost, both to be minimized. Solution 2 is clearly dominated by solution 1, which means that it is not Pareto-optimal. Solution 1 on the other hand is, as it is not dominated by any other solution. Solution 1 is, however, not the only Pareto-optimal solution, as solution 3 and 4 have better execution times, which means that they are not dominated.

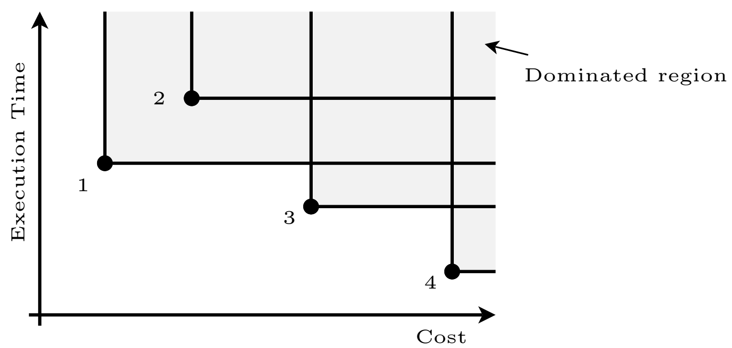


Figure 7.2: Illustration of a 2D design space with Pareto-optimal and dominated solutions.

The set of non-dominated solutions is also a set of Pareto-optimal solutions, all of which constitute reasonable design choices. The final choice must be subject to an extra set of constraints or preferences in order to make a final choice. Hypothetically, in the acquisition case solution 1 might be opted out because it does not meet the time constraint of 143 s, whereas solution 3 might be too costly and have a far better execution time than needed.

Returning to the issue of optimization strategy, three classifications exist [32]; *Decide and search*, *search and decide* and *decision during search*. The previous example represents a search and decide strategy, where the search space is evaluated according to a set of objectives, where after one of the solutions is chosen. During the search, the objectives are kept separate. So in the previous example, cost must indicate price (not cost function output) in order for it to be a search and decide strategy. As no precondition, such as cost functions, are used that could influence the search, the result is unbiased [32].

In a decide and search strategy some or all of the objectives are combined into a single cost function before the actual search is performed. Using this method the performance of an entire or part of systems is summarized in a single objective, which means that common optimization techniques such as simulated annealing can be used. The decision in this strategy is made when designing the cost function. A common example is a weighted sum of objectives, but more complex functions such as ratios or exponential could be used as well. The problem is that the method requires a good understanding of the characteristics of the search space (heuristics) in order to formulate good objective functions. If not, sub-optimal solutions are likely to occur and in worst cases certain regions of the search space is rendered unreachable [32].

The final method, decision during search, is a mixture between the first two. In this method a search and decide strategy could be used to provide some heuristics needed to design/formulate a cost function, that would be used in another optimization process. Or, in the other case around the results from a decide and search strategy could be used as an outset for a series of search and decide optimizations.

In this project it has been chosen to use a search and decide strategy - the knowledge about the nature of the search space is simply too small to formulate any meaningful cost function. Some general heuristics will however still be used to guide the searches that have to be performed. This will be subject for the next sections.

7.1.2 Cost Function and Design Metrics

Although the cost function will not be used to optimize the acquisition algorithm, it is stated in Eqn. 7.1 to highlight the connection between the total cost of the system implementation and the design metrics that will be considered.

$$\text{COST} = f(T, A, E, P_{fa}, ...) \tag{7.1}$$

Obvious any number and types of metrics can be included in the cost function, as long as they are relevant for the case in mind. In this case four primary metrics have been singled out based on the requirements stated in the summeary of the application analysis. That is, time T, area A, power E and performance P_{fa}. The definition of these metrics and related heuristics are presented in the following.

Time, T

Time T is the maximum time the algorithm must use to search through the acquisition search space. If 99% of the satellite are to be acquired then the acquisition time must not exceed 143 s, as given in Eqn. 7.2. Hence, the acquisition time is a constraint that the system must be designed according to. However, it should be noted that the constraint is not clear-cut, which is due to the uncertainties of the underlying methods used to generate the constraints. Also changes in the acquisition time have to be quite large in order for the percentage of acquired satellites to change considerably.

$$T_{99} \sim 143 \text{ s} \tag{7.2}$$

Power, E

As power is a limited source on-board the satellite, the power usage must be minimized. In the system exploration it was found that approximately 691 mW is available for the FPGA GPS receiver. Apart from signal acquisition this includes power used in tracking loops and data processing. Hence, the acquisition can only use a fraction of the available power. However, instead of estimating the minimum power usage and using this as a design constraint, a general goal of minimizing the power consumption will be used. The reason behind this choice is that it is very hard to estimate the total power consumption of a system at high abstraction level. Instead it is proposed to design and implement the algorithm with power in mind, and then measure the power consumption once it have been synthesized and mapped to the FPGA. In order to include power as a design metric the following heuristic is used:

$$E \propto C_L V^2 f \tag{7.3}$$

Eqn. 7.3 is based on the power that is dissipated in CMOS circuit due to switching [33], i.e. dynamic power. C_L is the load capacitance, V is the supply voltage and f is the switching frequency. Unfortunately the core voltage in FPGAs is fixed [34] meaning that the quadratic voltage term cannot be exploited. Likewise, the load capacitance is architecture specific. Hence, to minimize the power dissipation it is necessary to minimize the frequency of the system, i.e. the clock/sampling frequency.

The previous is based on the switching loss in a single gate. If the design involve a hundred gates, then the switching loss would be hundred times larger. Hence, power is also proportional with the number of gates in a design. If it is assumed that the number of gates per area is constant and the system has a common clock, then if an algorithm uses more area the additional power consumption is proportional to the increase in area. In other words

$$E \propto A \tag{7.4}$$

This means that the power is not only dependent on the clock frequency, but is also connected to the area usage. Consequently, the area must be minimized not only to make room for tracking loops and data processing, but also as it minimizes power.

Area, A

The area or size of the final implementation has to be minimized to reduce power and to make way for tracking loops which also is an integrate part of a complete GPS receiver. In order to reduce the area of the implementation two heuristics will be used **[Don't know whether heuristic is a good term to use??]**. The first is the bit width B. If the width changes then more area must be used for routing and functional units (FU) grow in size to accommodate for the extra bit width. Hence, it is assumed that a relation exists between the area used and bit width, i.e.:

$$A \propto B \tag{7.5}$$

Another factor that influences the area is the number of FUs used by the implementation. As computational complex solutions use more FUs they must take up more area. Hence, it is assumed that computational complex algorithms use more area, i.e.:

$$A \propto O(f(N)) \tag{7.6}$$

It must be noted that these assumption only hold at a high level of abstraction and direct proportionality would never be the case in the final implementation. E.g. the area covered by signal routing is determined by the EDA tools used, which again is determined by how the designer has chosen to describe the algorithm in HDL.

Performance, P_{fa}

In order to evaluate the acquisition process under different circumstances, such as different sampling frequencies or SNR, a performance measure is needed. Due to the stochastic nature of radio signals in noisy environments, sample average methods are then used to evaluate the receiver performance, e.g. bit error rates. The case of acquisition is no different and in [25] a closed form solution have been derived for the probability of false acquisition P_{fa} according to SNR. That is the probability that the highest peak in the acquisition search space is not the right one. The P_{fa} is given by:

$$\begin{aligned} P_{fa} &= 1 - \int_0^\infty \frac{2x}{\sigma^2} \exp\left(-\frac{A^2T^2 + x^2}{\sigma^2}\right) \\ &\quad \cdot I_0\left(\frac{2ATx}{\sigma^2}\right)\left[1 - \exp(-x^2/\sigma^2)\right]^{M-1} dx && (7.7) \\ &= f(SNR, M) && (7.8) \end{aligned}$$

where σ^2 is the noise variance, A the signal amplitude, T the integration period, M the size of the search space, I_0 the Bessel function of the 0th order and dx the integration variable. The problem with Eqn. 7.7 is that it only evaluates the performance according the two parameters. If effects of e.g. sampling frequency or quantization is to be evaluated, an alternative method must be used.

Instead it is proposed to use an abstract performance simulation based on sample average estimation, to evaluate the probability of false acquisition. The basic idea is to construct a signal generation function $G_k(\Psi)$ that emulates the down-converted signal plus AWGN from the front-end according to state vector Ψ. The state vector may include information like which satellites to include in the signal and their phase τ and Doppler frequency f_d , SNR, sampling frequency and any other parameter that is relevant to the case.

$$\tilde{r}_k(t : \Psi) = G_k(\Psi) \qquad \text{where} \qquad \Psi = \{SNR, f_s, \tau^*, f_d^*, ...\} \tag{7.9}$$

Once the test signal has been generated it is passed to the ML estimator in Eqn. 7.10 and the acquisition result is calculated.

$$L(\tau, f_d : \tilde{r}_k(t:\Psi)) = \frac{1}{N_0}\left|\int_{-T/2}^{T/2} \tilde{r}_k(t:\Psi)p(t-\tau)e^{-j2\pi f_d t}dt\right|^2 \tag{7.10}$$

Given that a code $p(t)$ with phase and frequency τ^* and f_d^* is present in the signal, then if the peak in the output of the acquisition result $L(\tau, f_d : \tilde{r}_k(t:\Psi))$ is located at the (τ^*, f_d^*) coordinate, the boolean variable $z_k(\Psi)$ is assigned with a 1. In other cases a zero is assigned, indicating false acquisition.

$$z_k(\Psi) = \begin{cases} 1 & \text{if } (\tau^*, f_d^*) = \underset{(\tau, f_d)\in S}{\arg\max}\, L(\tau, f_d : \tilde{r}_k(t:\Psi)) \\ 0 & \text{otherwise} \end{cases} \tag{7.11}$$

If the above is repeated K times and $\forall_{k \neq j}\ G_k(\Psi) \neq G_j(\Psi)$ then the probability of false acquisition is given by one minus the sample average of $z_k(\Psi)$:

$$P_{fa}(\Psi) = 1 - \frac{1}{K}\sum_{k=1}^{K} z_k(\Psi) \tag{7.12}$$

By using this method the performance of a given acquisition algorithm can be evaluated according to any parameter that can be emulated by the signal generated function $G_k(\Psi)$. The main problem with the method lies in the computational complexity of the evaluating a single design point, which is increased by a factor K. Also the generation of the test signal $G_k(\Psi)$ adds to the total computational complexity as well as it is prone to changes in the state vector Ψ. E.g. increasing the sampling frequency will also increase the computation time needed for each design point, due to the extra samples. This makes the metric unsuitable for iterative optimization methods such as simulated annealing, where $P_f a(\Psi)$ have to be evaluated until a stop criterion is meet, which may by after 100 or 1.000 iterations. But for evaluation of a single point or for sub-sampling the design space, the metric is very useful.

7.1.3 Covering the Design Space

All realizations of the acquisition algorithm are represented with a single point in the design space. In the current case the design space is four dimensional, spanned by execution time, area, power and performance. As a search and decide strategy has been chosen, a method to cover or search through the design space must be considered. One such method is exhaustive search where all points are evaluated and classified as either feasible or infeasible. But this is prohibited due to the computational complexity involved in estimating P_{fa}. Instead a combination of knowledge-based search and design space pruning will be used, which will be described next.

Having presented the design metrics it is possible to formulate a optimization problem for the project. Both power, area and probability of false acquisition are metrics that have to be minimized. But due to the time constraint, only solutions with an execution time of approximately 143 s are of interest. I.e.:

$$\begin{array}{ll} \text{minimize} & E, A, P_{fa} \\ \text{s.t} & \\ & T \sim 143 \text{ s} \end{array}$$

This means that the initial set of solutions can be reduced considerably. Now, as it was shown in the application analysis it is quite easy to estimate the execution time of an implementation once the size of the search space was established. So in order to establish the first point in the guided search through the design space, a solution that fulfills the execution time constraint will be used. The $P_{fa}(\Psi)$ of this solution is assumed to be low as the state vector Ψ at this point do not include parameters like quantization, which could affect the acquisition negatively. Similarly, the power and area usage is assumed to be high, as shown in Figure 7.3.

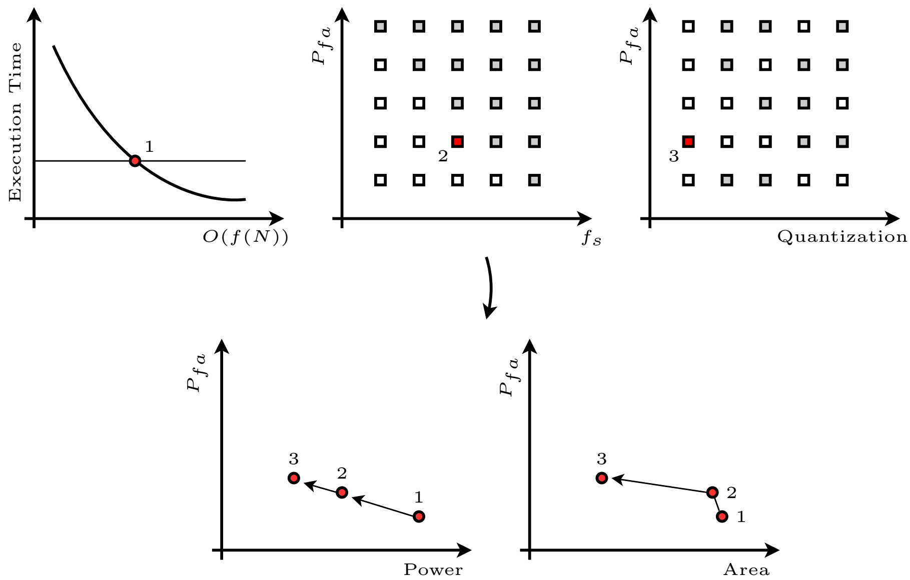


Figure 7.3: Guided-search. Top row shows the sequence of pruned design spaces (sub-spaces). Knowledge obtained from one sub-space is used as an outset in the next, whereby the search is guided through the design space, as shown in the lower row.

Next, the power will be minimized by introducing a seperate (sub)problem, where the sampling frequency is reduced and the performance of the acquisition process is observed. In order to reduce evaluation time the search space will be sub-sampled, indicated by the dots in Figure 7.3. The result is a set of Pareto-optimal solutions of which one is chosen as a new design point. Using this point, the effects of quantization are evaluated to minimize the bit width and hence area of the system.

The combined result of this method is that the power and area of the implementation is minimized while the performance is kept at an acceptable level. However, the method suffers some problems which could lead to sub-optimal solutions. E.g. the choice of the sampling frequency could guide the search around an optimal point. This is however a common problem which many methods suffers from [32] and it is also one of the reasons why many projects go through several iterations before they are finished. Unforeseeable problems appear and opportunities becomes clearer as the abstraction is lowered, which is why high-level simulations might have to be changed the second time around.

7.2 Acquistion Realization

So far the maximum likelihood estimator has only been considered in its continous-time form. In this section the estimator is mapped to the algorithm domain. That is, the discrete-time interpretation of the estimator will be presented along with architectures that can be used to implement the estimator. As the estimator is correlation-based the number of ways it can be implemented are many. Hence, two destinct methods have been singled out; one with (almost) full hardware sharing and another which uses a high degree of parallelism. These will serve as a reference point for the work to follow.

The estimator presented in Section 4.2 is given in Eqn. 7.13 and 7.14. By using the identity $e^{-j\theta} = \cos\theta - j\sin\theta$ expression 7.14 can be expanded as shown in 7.15.

$$
\begin{aligned}
L(\tau, f_d) &= \frac{A^2T^2}{N_0}\left|\Phi(\tau, f_d)\right|^2 && (7.13)\\
\Phi(\tau, f_d) &= \frac{1}{T}\int_{-T/2}^{T/2} \tilde{r}(t)p(t-\tau)e^{-j2\pi f_d t}dt && (7.14)\\
&= \frac{1}{T}\int_{-T/2}^{T/2} \tilde{r}(t)p(t-\tau)\cos(2\pi f_d t)dt - \\
&\quad \frac{j}{T}\int_{-T/2}^{T/2} \tilde{r}(t)p(t-\tau)\sin(2\pi f_d t)dt && (7.15)
\end{aligned}
$$

Apart from normalising Φ, Eqn. 7.13 does not serve any function other than scaling the result. Whether the output of the estimator is scaled correctly or not is not important, why the scaling terms can be dropped. Hence, the ML estimator can be written as:

$$
\begin{aligned}
L(\tau, f_d) = \left|\Phi(\tau, f_d)\right|^2 &= \left(\frac{1}{T}\int_{-T/2}^{T/2} \tilde{r}(t)p(t-\tau)\cos(2\pi f_d t)dt\right)^2 + \\
&\quad \left(\frac{1}{T}\int_{-T/2}^{T/2} \tilde{r}(t)p(t-\tau)\sin(2\pi f_d t)dt\right)^2 && (7.16)
\end{aligned}
$$

Consequently, the estimator can be considered as the correlation between the spreading sequences $p(t)$ and the received signal $\tilde{r}(t)$ mixed with in-phase and quadrature Doppler frequency components, respectively. Finally the correlation results are squared and summed together, to yield the acquisition output.

In order to transform the ML estimator to discrete-time the integration can be approximated with a summation over the integration period. Again the scaling is not considered important. The number of samples over the integration period is equal to the integration period T multiplied with the sampling frequency f_s, which means that the discrete-time Approximated ML Estimator (AMLE) is given by:

$$
\begin{aligned}
\tilde{L}(\tau, f_d) &= \left(\sum_{n=0}^{f_sT-1} \tilde{r}(n)p(n-\tau)\cos(2\pi f_d n)\right)^2 + \\
&\quad \left(\sum_{n=0}^{f_sT-1} \tilde{r}(n)p(n-\tau)\sin(2\pi f_d n)\right)^2 && (7.17)
\end{aligned}
$$

Figure 7.4 is a direct interpretation of the AMLE acquisition given in Eqn. 7.17. As only one cell can be evaluated at a time, the method is known as serial search acquisition. With one multiplication and two MACs per sample, the method uses a minimum number of FUs, e.i $O(3N)$ It is possible to reduce the number FUs furher if only one of the I-Q branches is considered, which has been evaluated in [26, 24] and is shown in Figure 4.3 in page 30. However, this makes the receiver vulnerable to Doppler, as the received signal after down-convertion, in most cases, still contains low frequency Doppler residuals. These residuals are modulated as cosines onto the spreading sequence in the received signal, meaning that signal "disappears" during zero-crossings. Consequently, the serial search acquisition as shown in Figure 7.4 will be the point of reference, when comparing computational complexity and acquistion time.

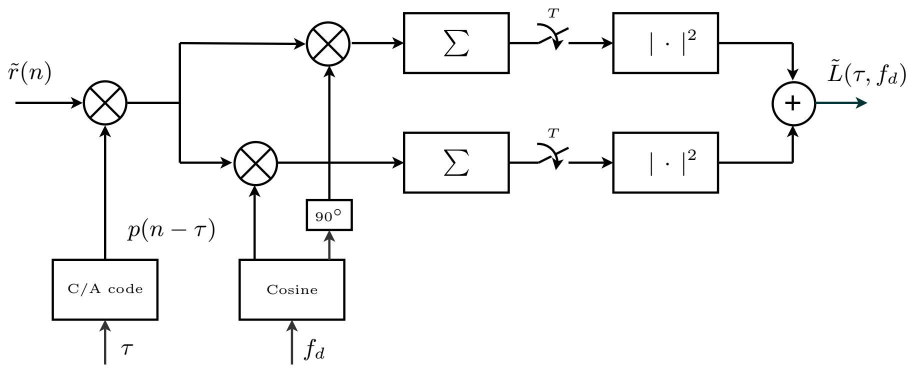


Figure 7.4: Direct implementation of the ML estimator, also known as the serial search acquisition architecture.

Another way to implement the estimator is by using frequency-domain correlation, as shown in Figure 7.5. This method was presented as a solution to the ML estimator in the original paper although for tracking loop implementation [25]. Similarly it has been used in the GPS software receiver made by the Navigation Center at AAU [35, 8]. The main advantage of this methods is that the correlation result for all lags is calculated in just one dwell period, which means that the acquisition time can be reduced considerably. The computational complexity is however somewhat larger. Assuming that FFT can be used, the complexity of one transformation is $O(NlogN)$. This has to be performed three time along with a $4N$ multiplications, giving a total computational complexity of $O(3NlogN + 4N)$

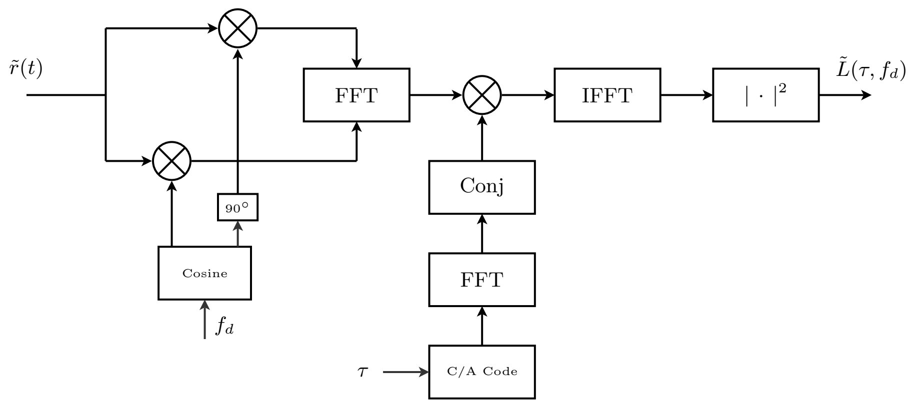


Figure 7.5: Acquisition using frequency-domain correlation as proposed in [25, 8]

7.2.1 Algorithm Speed-Up

In the application analysis it was found that 184.140 cells had to be searched per C/A code in approximately 143 s in order to acquire the majority of GPS satellites. With 32 unique C/A codes this amount to 5.891.200 cells. Using a serial search acquisition scheme this would take 5.891 s or 98 min, assuming a dwell time τ_d of 1 ms. Consequently a faster method must be found to reduce the acquisition time. But as discussed in the previous section, faster acquisition also means increased computational complexity and hence more area usage. Hence, the optimal solution must be a trade-off between acquisition time and area, where acquisition time have the highest priority initially.

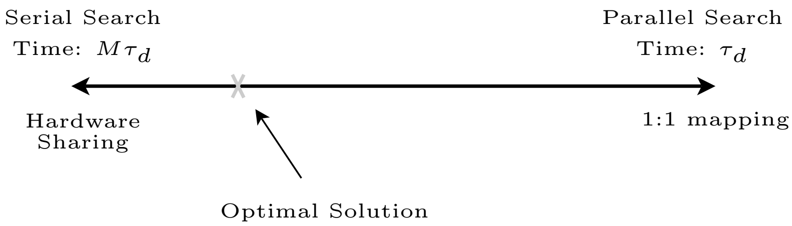


Figure 7.6: Time-area trade-off. Hardware sharing is slow but uses less area. Increase use of inherent parallism decrease acquisition time, but uses more area. The dwell time is given by τ_d and M is the number of cells to be searched.

In order to evaluate where the optimal point is on the trade-off axis in Figure 7.6, the time it takes to perform serial search acquisition on 5.891.200 cells have been used as the point of reference, which is the worst case in relation to acquisition time. By dividing the worst-case acquisition time until it is less then 143 s it is possible to evaluate the required speed-up factor. This factor inducates how fast the algorithm, used for acquisition in LEO, has to be compared to a single serial search acquisiton architecture. Figure 7.7 shows the acquisition time in relation to the speed-up factor. As can be seen it is possible to perform the acquisition in under 143 s if the serial search architecture is speed up with a factor of approximately 42.

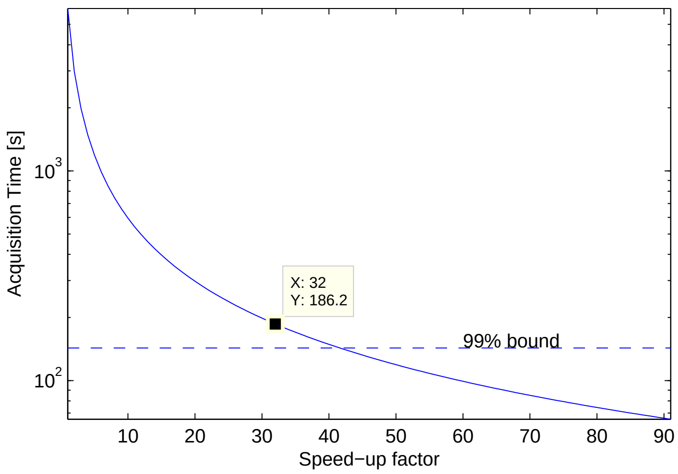


Figure 7.7: Algorithm speed-up relative to serial search acquisiton. The result is based on the data in Table 4.1 on page 32.

The reason that the coordinate have been high-lighted at 32 is to show the solution where all 32 C/A codes are correlated in parallel using serial search acquisition. This solution is aluring as it means that no control structures have to be used to keep track of which C/A codes have been correlated or not, which greatly simplifies the implementation. Also the extra acquisition time can be shown to be insignificant. By using the method presenten in the analysis of the available acquisition time in Section 4.2.2, an acquisition time of 186.2 s translates into acquisition of 98.3% of the satellites, as compared with 99% at 143 s. Hence, by acceptign a small degradation in the number of acquired satellites, a it is possible to reduce the pratical complexity of the implementation, which is why a speed-up factor of 32 will be used in the following. The mapping of the algorithm onto architecture and how the speed-up is achieved is reserved for the section on algorithm mapping (Section 9.1).

7.3 Sampling Frequency

In order to simulate the effects of reduced sampling frequency in relation to the performance of the acquisition process, a signal generating function $G(\Psi)$ have been designed in Matlab that emulates the signal from the front-end given a state vector Ψ. Instead of simulating the hole signal chain from GPS satellite to front-end output, the signal is constructed at a IF level and noise added. The reason for this is the high frequency of L_1, which means that gigabytes of data would be generated even for short simulation periods. Also the signal only consists of the BPSK modulated spreading sequences plus noise. This means that no navigation data is present, which under normal circumstances could interfere with the acquisition process (data bit transition induced a 180 degree phase shift on the spreading codes) and thus degrade performance.

For the evaluation of the design space a semi-static state vector have been used. That is, only the sampling frequency and SNR (along with noise) are changed between simulations to evaluate performance at a given design point. The spreading codes (PRN) in the signal and their code phase and frequency remain the same. The state vector used in the simulations is given by:

$$
\Psi = \{ \quad
\begin{array}{lcll}
PRN & = & \{2^*, 6, 13, 3\} & \text{[-]} \\
\tau & = & \{52^*, 7, 232, 1020\} & \text{[chips]} \\
f_d & = & \{0^*, 500, 8000, 20\} & \text{[Hz]} \\
f_{IF} & = & 4.309 & \text{[MHz]} \\
f_s & = & \{2f_{IF},\ 2f_{IF} + 1e6\ , 2f_{IF} + 2e6\ , 50e6\ \} & \text{[MHz]} \\
SNR & = & \{-40, ..., -15\} & \text{[dBW]} \\
t_{sim} & = & 1 & \text{[ms]} \\
Q & = & \text{Matlab precision} & \qquad \}
\end{array}
$$

The entries marked by asterixes are the phase, frequency and satellite number (PRN) of the spreading code used to test the outcome of the acquisition process. As can be seen the Doppler shift of the test code is set to zero, which can be considered a best-case scenario. Simulations run at the worst case, i.e. 250 Hz from the bin center, yielded very poor results, as the signal in around half the cases would be classified as belonging to the adjacent bin.

Once the GPS signal, as seen from the front-end, has been constructed it is passed to the acquisition process. In the previous it was found that serial search acquisition would serve as an addiquite acquisition method. In spirit of the guided-search strategy, it would have been best to apply this method to the generated test signals. Unfortunately, serial search acquisition performed in Matlab has a prohibitely high execution time, which means that frequency domain acquisition has to be used instead. In theory these methods should be equivivalent. But the resolution of the frequency domain correlation is much finer (lag output for each sample) whereas serial search normally only has a resoltion of half a chip. Also the path to the correlation result (direct vs.

transformation) are very different. The effects of this is hard to evaluate, so for now and in lag of better the assumption is that the two are equivivalent.

Due to all of these factors the simulation is expected to yield a biased and liberal performance figure of the design point under consideration. It is however assumed that the method is capable of provided a true picture of the design landscape, although a bit blury. In order to account for this a safe margin must be kept to the SNR threshold of -17 dBW found in the linkbudget. A more unbiased/accurate performance figure could be achieved if a true Monte Carlo simulation, where the phase, frequency and code values in the state vector is changed continously, was used. But the frequency vs. time domain correlation problem would still exist.

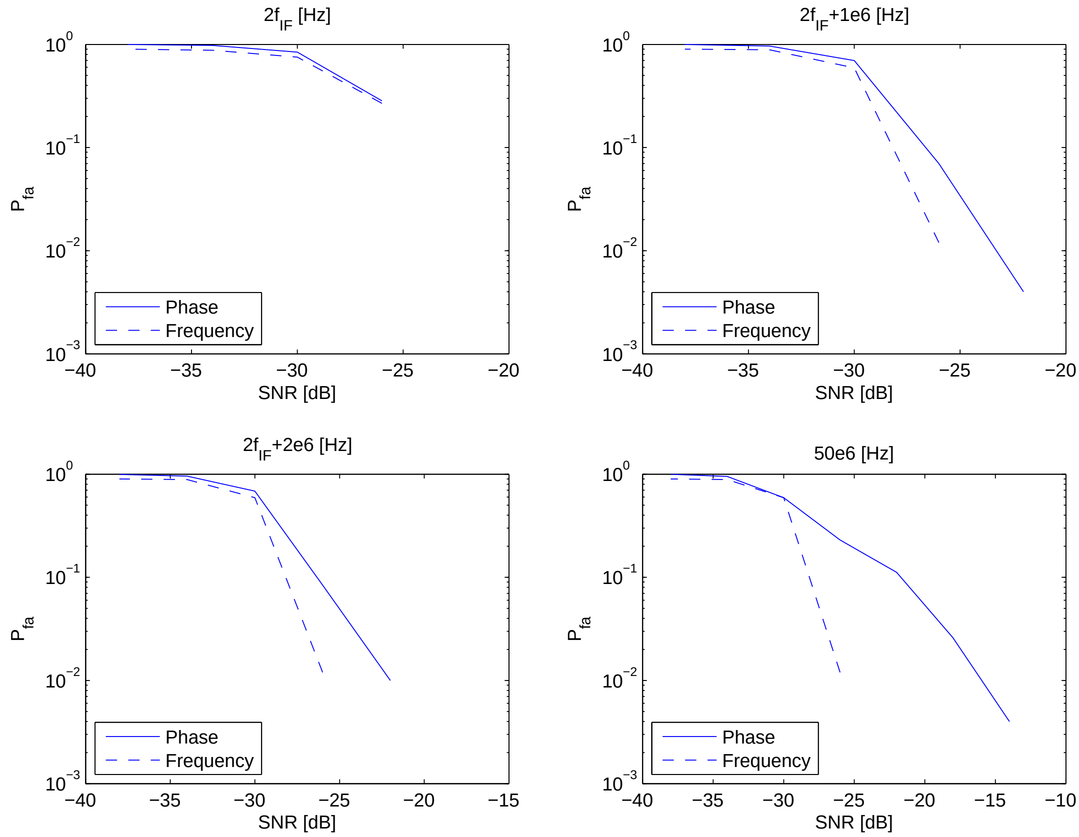


Figure 7.8: sdf

Figure 7.8 shows the performance in relation to the SNR of the four design points. The result is somewhat surprising as the system yeilds better performance at lower sampling frequency. In fact the $2f_{IF}$ sampling frequency dominates the other design point[1]. Although this is a good result from an design optimization point of view, it leaves behind a question of why this is the case? The GPS signal has a bandwidth of 2.046 MHz which means that the Nyquist sampling frequency is $2(f_{IF} + 1.023\text{ MHz})$ or 10.664 MHz. This means that parts of the signal is undersampled, effectively mirroring it into the negative frequency range. This means that some of the signal is lost. In most cases this scenario would lead to degraded performance figures, but not here. As no obvious explanation can be given, it is assumed that the flaw lies somewhere in the generation of the signal.

[1]Results, not shown, showed that below $2f_{IF}$ the system performance deterioated considerably.

7.4 Quantization

The scope of this section is to minimize the number of bits that is needed to represent the signal internally in the algortihm. As discussed earlier this minimized the area of the implementation and lowers the power consumption.

The fixed point number representation that will be used in the implementation is signed two's compliment. The output from the front-end is however a sign/magnitude (2-bit) signal, meaning that one bit respresents the sign of the sampled signal and the other the magnitude. However, no values are assigned to the signal. This means that it is up to the designer to chose how the sign/magnitude is interpreted in the system - in other words how should the sign/magnitude signal be mapped to two's compliment values.

Once the signal has been mapped it has to be down-converted, which is achieved by multiplying the signal with a local copy of the IF plus any Doppler shift offset. This posed two problems and the scenario is shown in Figure 7.9. First of all the IF has to be a quantized where an integer and fraction length has to be chosen.

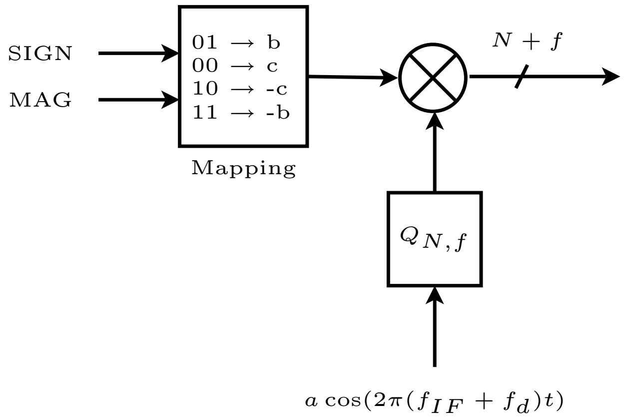


Figure 7.9: Quantization scenario. The sign/magnitude signal is mapped and multiplied with a local copy of the IF, which have been quantized according the chosen integer and fraction length given by N and f respectively. The resulting bit width of the signal is $N + f$.

Secondly, in order for the correlators not to run off, it has to be assured that the mean value of the signal is zero after the multiplication. This is potentially a problem as the positive bound is one value smaller that the lower bound in signed two's compliment. This means that positive signals saturates faster than negative signals, which would shift the mean if allowed. Hence, the system must be designed so that maximum output of the multiplied signals is less or equal to the upper bound in the signed two's compliment number representation. This statement is presented in Eqn. 7.18, where a is the amplitude of the IF, b the maximum value of the mapped input signal, N is number of integer bits and f is the fractional bits.

$$ab \leq \frac{2^{N-1} - 1}{2^f} \Rightarrow \tag{7.18}$$

$$b \leq \left\lfloor \frac{2^{N-1} - 1}{2^f\, a} \right\rfloor \tag{7.19}$$

$$c < b \tag{7.20}$$

By rearranging 7.18 b can be represented as a function of the IF amplitude and quantization parameters as done in 7.19. Also as b is the maximum value, c can be stated as all values below b, which is done in 7.20.

As many unique combinations of a, b, c, N and f exists, the design space is potentially very large. But for $N \leq 3$ the search space is limited to an extent where exhaustive search is possible, which can be seen in Table 7.1. If this space does not yields any feasible solutions it is possible to use combinatorial optimization like evolutionary computation and genetic algorithms to evaluate a larger design space (see [36] for futher references).

$[2,0]$			$[2,1]$			$[2,2]$			$[3,0]$			$[3,1]$		
a	b	c	a	b	c	a	b	c	a	b	c	a	b	c
1	1	0	1	0.5	0	1	0.25	0	3	1	0	3	0.5	0
			0.5	1	0	0.25	1	0	1	2	0	1	1	0
			0.5	1	0.5	0.25	1	0.25	1	2	1	1	1	0.5
						0.25	1	0.5	1	3	0	1	1.5	0
						0.25	1	0.75	1	3	1	1	1.5	0.5
									1	3	2	1	1.5	1
									2	1	0	0.5	3	0
												0.5	3	0.5
												0.5	3	1
												0.5	3	1.5
												0.5	3	2
												0.5	3	2.5
												1.5	1	0
												1.5	1	0.5

Table 7.1: Unique combination of a, b, c, N and f for $N \leq 3$ generated according to Eqn. 7.19 and 7.20. The bracket braises in the top indicate the integer and fraction length, i.e. $[N, f]$.

In order to evaluate each design in Table 7.1 the parameters was added to the state vector Ψ and testet indivually by using the signal generating function $G_k(\Psi)$. Furthermore the signal was generated at the optimal sampling frequency found in the previous section. In order to reduce the search space a single design point at -20 dBW SNR has been used. According to Figure 7.8 the P_{fa} at this point should be very low ($P_{fa} < 2 \cdot 10^{-3}$). Hence, feasible combinations should perform equally well.

In Table 7.2 the results from the exploration of the designs given in Table 7.1 are shown. As can be seen the performance of combinations where the magnitude bit is mapped to zero all have a high P_{fa}. This comes from the fact that only half the information in the magnitude bit is used, which basically means that the front-end quatization is reduced to 1.5 bit, instead of 2 bit.

In Table 7.2 the set of Pareto-optimal solutions are highlighted. The 2-bit solution has low performance, when compared with the other Pareto-optimal solutions, why this design choice will not be used. The remaining solutions all have a very low P_{fa}, which makes it hard to distringuish between their performance. To reveal the nature of these design points, their performance with respect to SBR have been evaluated using the same method as in the previous section.

First of all it is noticed that false acquisitions are induced at higher SNRs compared with the optimal unquantized solution, found in the evaluation of the sampling frequency (Fig. 7.8 on page 58). This degradation in performance is an expected consequence of quatization. Below -20 dBW the P_{fa} of the Pareto-optimal solutions is however still very low, and would all be acceptable designs. But as there is no visible different between the 3-bit solution and those with 4-bit, the 3-bit solution is chosen as it minimizes the area the most. This means that the design chosen would map the sign bit to the value 3 ($b = 3$) and the magnitude to 2 ($c = 2$). Similar the

[2, 0]		[2, 1]		[2, 2]		[3, 0]		[3, 1]	
τ	f_d	τ	f_d	τ	f_d	τ	f_d	τ	f_d
0.42	0.382	0.39	0.356	0.45	0.418	0.384	0.34	0.4	0.368
		0.428	0.386	0.428	0.386	0.428	0.386	0.37	0.33
		0.016	0.016	0.092	0.084	0.016	0.016	0.01	0.004
				0.012	0.012	0.418	0.396	0.388	0.362
				0	0	0.024	0.018	0.028	0.022
						0	0.004	0.002	0.002
						0.376	0.362	0.444	0.398
								0.13	0.124
								0.016	0.024
								0.014	0.01
								0	0
								0	0
								0.39	0.364
								0.008	0.008

Table 7.2: Probability for false phase and frequency acquisition at -20 dBW signal to noise ratio according to the amplitude combination given in Table 7.1. Red marks Pareto-optimal solutions. As three 4-bit points do not experience any false acquisitions, they are all considered Pareto-optimal solutions, althoug this conflicts somewhat with the definition.

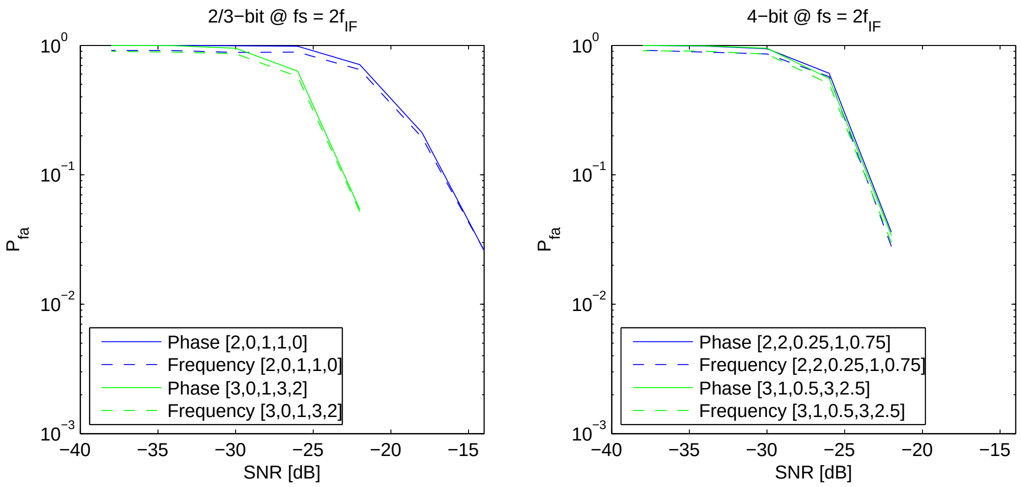


Figure 7.10: Probability of false acquisition for the Pareto-optimal set of designs. The bracket parameters are $[N, f, a, b, c]$

IF would have an amplitude of 1 ($a = 1$) and would be quantized to 3-bits ($N = 3$), without any fractional bits ($f = 0$).

Chapter 8
Summery

In the previous two chapters both the hardware and algorithmic aspects was explored with respect to a GPS signal acquisition in LEO.

The proposed hardware solution consists of a Taoglas 10x10mm patch antenna, which was chosen based on its small size, and the Zarlink GP2015 front-end. The GP2015 has a long history of space flight due to its use in the Pheonix space GPS receiver, which makes it an obvious candidate. The combined signal-to-noise of the front-end and antenna was evaluated for three cases; terrestrial, space with sun exposure and space without sun exposure. This showed that the SNR is slightly worse when the antenna is exposed to the sun compared to terrestrial applications. But where the noise in terrestrial based antennas is a due to a combination of many factors (e.g. sky/ground noise, antenna temperature ect.), the main source of noise in space is the high physical temperature of the antenna. The worst-case scenario the SNR of the received signal in LEO was found to be -17.2 dBW, and the best-case -16.1 dBW. Both figures are comparable with the signal levels in terrestrial GPS receivers, which means that from a SNR point of view receiver operation in space is no different from terrestrial operation. Regarding the power budget the front-end and antenna has a total power consumption of 310 mW, which leaves 690 mW for signal processing.

In the algorithm exploration different methods to explore the design space was considered. Eventually a combination of guided-search and design space pruning was chosen. The mapping of the ML estimator to the algorithm domain, showed that it can be implemented in a number of ways. The most simple method, but also slowest, was found to be the serial search algorithm, where the acquisition search space is searched sequentially. Using this method a speed-up factor of 32 will result in acquisition of 98.3% of satellites that enter the visible area of the receiving antenna. As the time constraint allowed some slag, this solution was chosen. Hence, the challenge in the mapping to the architecture domain is to design a acquisition process that is 32 times faster than a single serial search algorithm.

In order to reduce the power and area of the implementation the effects of sampling frequency and quantization was investigated. This showed that the sampling frequency could be lowered to 2 times the intermediate frequency without any increase in the probability of false acquisition. Similar, effects of quantization was investigated and several Pareto-optimal solutions was found. By evaluating these individually a 3-bit quantization was chosen, as it showed similar performance equivalent to quantizations using more bits.

Since, the design space exploration of the algorithm domain some details, previously overlooked, has become clear. This regards the sampling frequency of the system. The assumption in the previous part was that the GP2015 front-end would be able to sample data at frequencies up to 50 MHz. However, during the implementation of the FPGA system it became clear that the front-end is actually designed for use at a fixed frequency of 40/7 MHz or 5.714 MHz. In other words the IF is undersampled, which mirrors it to 1.425 MHz [30]. The consequence of this is that the sampling frequency and quantization results found in the previous chapter are invalid. Attempts where made to re-simulate the results, but undersampling the generated test signal was not successful. This, combined with project delivery dates lurking in the horizon, meant that it was chosen to implement the system using the new sampling frequency, but with the old quantization configuration. The functionality of this solution has not been confirmed and as such it is unknown whether the finial implementation will work. However, experiences from handling the ML estimator shows that it is quite robust. Also, the Phoenix receiver uses the same front-end at 5.714 MHz, which indicates that the solution is feasible.

Part III

Architecture

Chapter 9
Algorithm Mapping

Contrary to the design space exploration, which was very methodical, the mapping of the algorithm onto the architecture has been performed with less attention to methods and exploration of the design space. Instead the focus has been on the practical issues involved in mapping an algorithm onto a FPGA. This means that instead of dwelling with different estimator designs/architectures and comparing their cost, e.g. in relation to power and area, a design has been proposed and used as an outset for the implementation.

9.1 Algorithm Realization

The main outset for the algorithm mapping was to find a solution that would enable acquisition 32 times faster compared to serial search acquisition. Also as the number of GPS satellites is 32 the possibility of acquiring all GPS satellites simultaneously will be explored.

1st Proposal - Utilizing Code Balance

Apart from being orthogonal to each other the spreading codes p are also balanced, meaning that the symbol probability is the same, which is stated in Eqn 9.1.

$$P_p\{1\} = P_p\{-1\} = 0.5 \tag{9.1}$$

The consequence of this in relation to the acquisition process, is that all signals is multiplied with either one or minus one with a probability of 1/2. With 32 different codes this means that on average 16 is multiplied with one and 16 with minus one at a given point in time. So instead of multiplying all signal with their individual spreading sequence it is proposed to multiply all signals with one and minus one, and then route the signals according to the values of the spreading codes. The concept is illustrated in Figure 9.1. The advantage of this method is that instead of having dedicated multipliers for each channel they are replaced by only two, which is a reduction of 30.

The method does, however, have some major drawbacks. As the signals from the multipliers is broadcasted to the accumulators the number of signals to route is doubled, increasing the area of the implementation. Also, in the proposed design the result of the accumulators is ready simultaneously, meaning that the results either have to be squared in parallel or the results store for later processing. If all signals are squared in parallel a total of 2x32 multipliers is needed. Also the multipliers is only used once every 1 ms, which means that the utilization is very low. This can be mitigated by storing the results from the accumulators and pipelining the results trough a square and sum block, but this methods is expensive in memory. So by design the method has some issues and as such a second proposal was made.

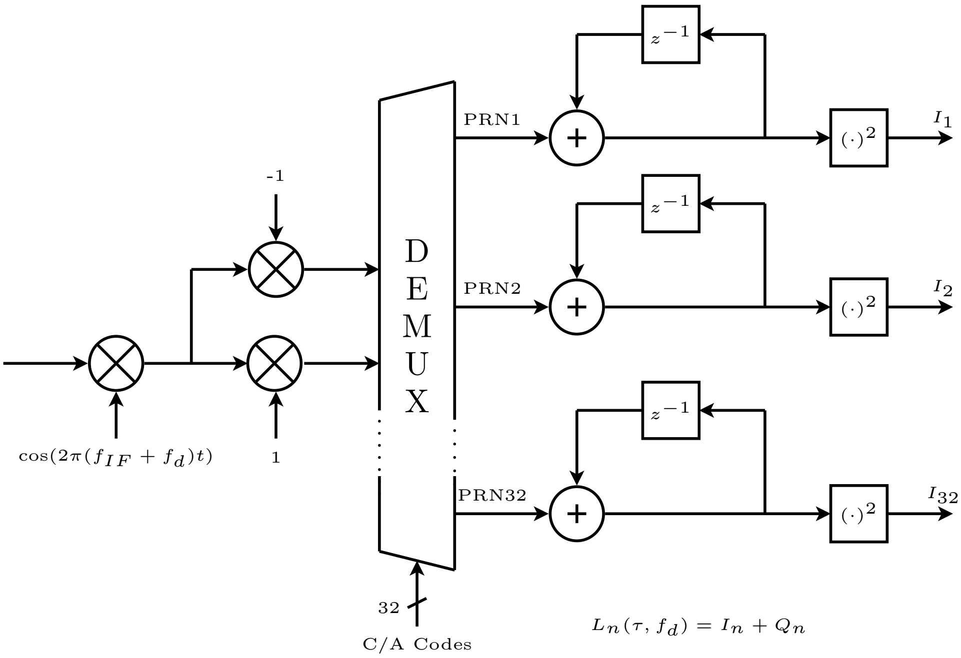


Figure 9.1: Hardware design that utilizes code balance to reduce the number of multipliers. Only the in-phase branch is shown. In order for the system to work a similar design must be implemented for the quadrature branch. The results from the are then added to obtain the estimator output.

2nd Proposal - Pipelining the design

In order to improve the previous proposed method, two issues have to be addressed. Firstly, the additional routing seems unnecessary, and a better method must exist. Secondly, a more clever way has to be introduced to handle the results from the accumulators.

The first problem can be solved by noticing that the effect of multiplying one and minus one to the signal is essentially the same as telling the accumulator to add or subtract. I.e.:

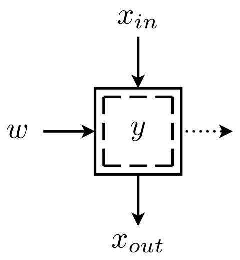
$$\sum_{n=0}^{f_sT-1} \tilde{r}(n)p(n-\tau)\cos(2\pi f_d n) = \sum_{n=0}^{f_sT-1} \tilde{r}(n)(-1)^w \cos(2\pi f_d n) \quad \text{where} \tag{9.2}$$

$$w = \frac{1-p(n)}{2} \quad \text{and} \quad p(n) \in \{1, -1\} \tag{9.3}$$

Hence, by controlling the “direction” of the accumulators using the spreading codes, the multipliers can be eliminated altogether from the design.

To handle the second problem inspiration was taken from the work done by Kung et al [37] on systolic arrays. A systolic array consists of cells which can perform very simple tasks, such as adding or multiplying results. The data in the system then moves from cell to cell each clock, just as the blood would in the systolic system in the body (thereby its name). A systolic cell with accumulator functionality is shown in Figure 9.2. The input data x_{in} is added or subtracted to the internal variable y according to value of the spreading code, in this case denoted w. The next clock the data input data is latched to the output which is connected to the input of another cell.

At the end of the integration period the cell is instructed to latch the result (indicated with the dotted line) and reset it self. By stagging the cells the input data x_{in} will move in a systolic manner through the cells. And by applying the spreading codes for each satellite as weights w to



$$x_{out} \leftarrow x_{in}$$
$$y \leftarrow y + (-1)^w x_{in}$$

Figure 9.2: Systolic cell with accumulator functionality.

the array of cells, serial search acquisition is performed for all satellites in parallel. The concept is shown in Figure 9.3, but with the addition that the code chips (weights) are delayed one chip. The consequence of this is that the results of the accumulators are also delayed according to each other, which means that the square and add block can be used to calculate the estimator output for all spreading codes.

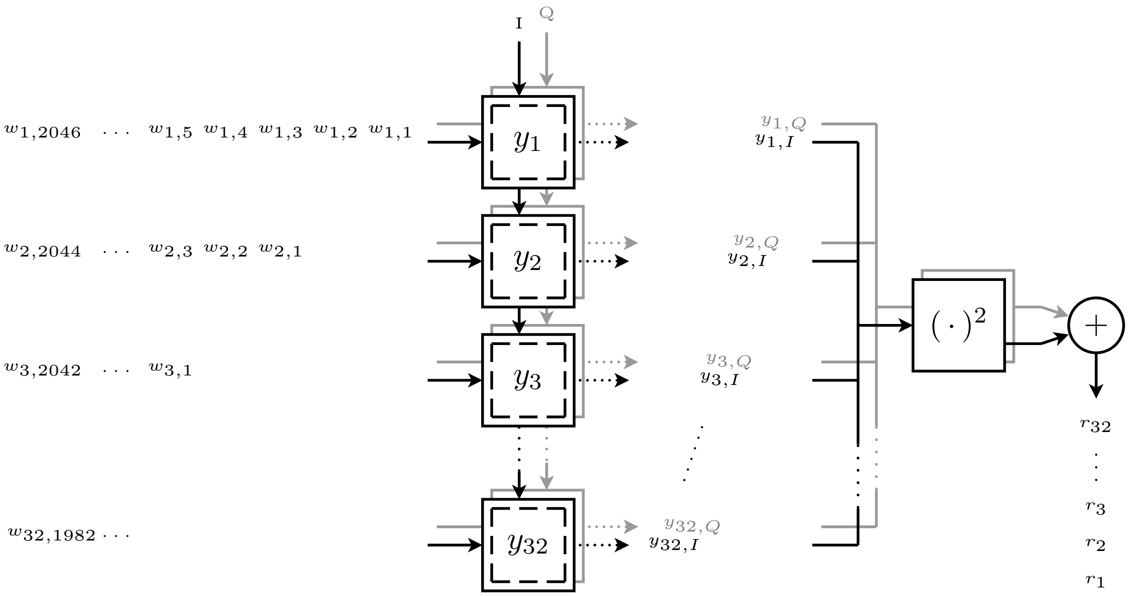


Figure 9.3: sdf

Chapter 10

FPGA Implementation

This chapter describes the experiences gained in the implementation of the proposed architecture given in Section 9.1 on a FPGA. Hence, the reader should not expect a detailed explanation of how each blocks in the system has been implemented. Instead an effort is made to describe some of the challenges one might face in FPGA implementation. For the interested reader the VHDL code can be found on the appended CD.

10.1 System Synthesis on FPGAs

In the realm of applied signal processing and implementation synthesis is used to describe the process by which steps are taken to lower levels of abstraction. Hence, in order to implement the proposed acquisition design on a FPGA it has to be synthesized down to a configuration file for the chosen FPGA hardware architecture.

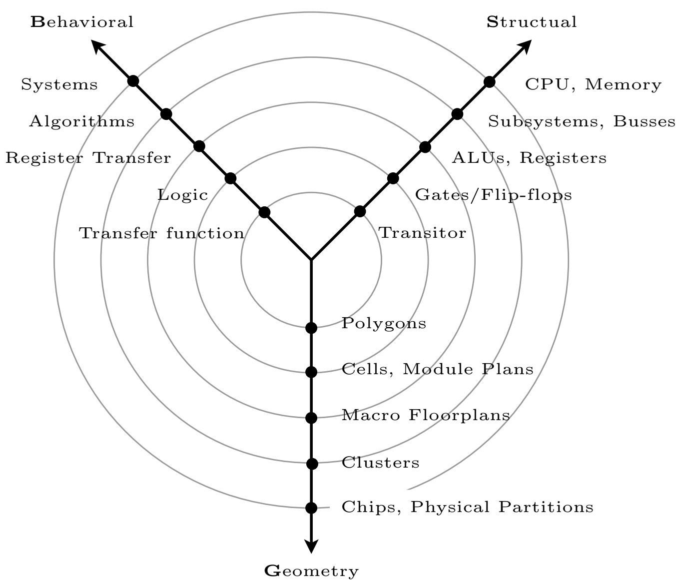


Figure 10.1: The Gajski Y-chart abstraction model [38].

The different abstraction layers by which a design can be described, is illustrated by the Gajski Y-chart in Figure 10.1. Each branch in the "Y" indicates a different domain: The behavioral branch describes the functionality of a system, whereas structural describes the basic building blocks and how they are interconnected. Finally, the geometry is a description of the physical layout of the system. The center of the Y can be considered the final implementation, where all aspects of the design has been taken into account.

At this point in the project the majority of aspects concerning the two outer most layers has been covered. That is, the behavior of the system and algorithm has been analyses and a structure was found in the previous section. Similar, the geometry is given by the chosen Xilinx development platform. As most of the low level geometry is fixed in FPGA technology, the main effort related to implementing the design on a FPGA is in describing the behavior and structure of the system in a hardware description language (HDL), which is the subject of the next section. Also, once the system has been described the design has to be taken to an even lower level of abstraction, which is done by using a synthesis tool. This process is covered in Section 10.1.2.

10.1.1 Hardware Description Language

Hardware description languages enables the designer to model the structure and behavior of a system in a standardized manner and in this project the VHDL language has been chosen. In VHDL the level of abstraction stretch from logic and gates to system and CPUs [38]. This means that the language can be used to describe high level as well as gate level functionality. One example is an adder which can be implemented by using a simple “+” sign, but if desired it can also be constructed using a series of full-adders. Hence, a large degree of freedom is available to the designer, when implementing a design.

When describing a system in VHDL one start with some simple building blocks which is combined to more complex structures. In Figure 10.2 this is illustrated for a 3-tap delay block, which consists of three D flip-flop registers. In a similar manner large blocks with complex functionality and structure can be combined at a higher level.

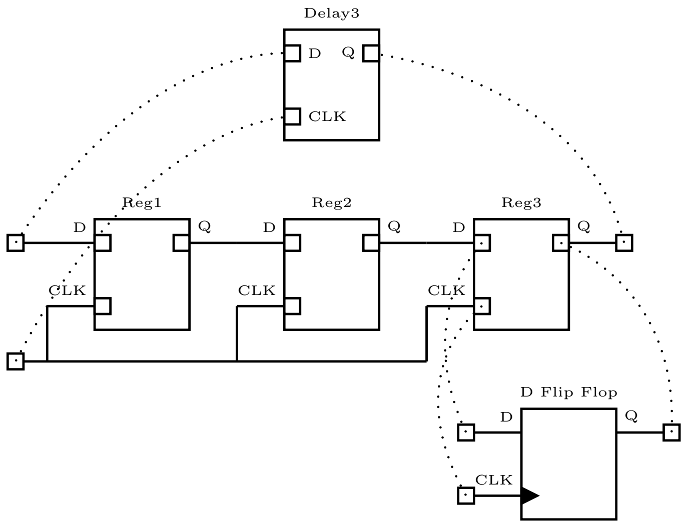


Figure 10.2: Illustration of the design hierarchy in HDL for a 3-tap delay block.

In practice the design of a high level block, such as the proposed acquisition algorithm, would start with the design of the blocks with the most simple functionality. Then as the correct behavior is confirmed blocks are combined to form a larger part of the system and eventually the hole system would be implemented. In order to insure that the behavior is as expected test benches are designed in each step. The test bench is a separate VHDL program that applies a stimuli to the unit under test (UUT). Finally, the test bench and UUT is simulated using a HDL simulator, such as Xilinx ISim or alternatively a combination of the open source VHDL compiler GHDL and GTKwave. If the behavior is different from the expected the code has to be reviewed until the simulation shows the desired behavior.

The problem with this method is that in large implementations the number of unique blocks is potentially large, which means that so is the number of test benched. Hence, in an implementation witting VHDL code to test the system is a natural part of the process. As such a large part of the code on the appended CD is test benches for smaller system blocks. A further consequence of the need for test benches and continuous testing is that the implementation of complex system is very time consuming. E.g. the time allocated for system implementation was three weeks, but it ended up taking in excess of six.

Design for Synthesis

In many ways the VHDL language allows many of the same structures one would find in a sequential programming language such as C or Assembly. E.g. loops, if-statements and counters can be used as one would in a normal program. In the initial phase of learning the language this is an advantage as the syntax and coding style is (almost) comparable with other computer languages. However, when the code has to be synthesized (i.e. mapped to hardware) most synthesizers would throw errors and warnings and quit, unless the code has been written with synthesis in mind. The main reason for this is that the code is written with the wrong mind-set. Most engineers learn program writing on von Neumann based computers which means that program is expected to executes in the same sequence as one would read the code - from top to bottom. In VHDL, however, the code within an architecture is executed concurrent, which means that all code is executed simultaneously. This problem is by some referred to as the von Neuman Syndrome [39].

Another aspect is that it is not all coding constructs, where an equivalent in hardware exist, which means that the code must be rewritten.

All in all it was found that learning the syntax of VHDL was relatively straight forward. The real challenge is writing synthesizable code. And in many cases it is necessary to have a profound understanding of how the synthesizer interprets the code in order to correct errors. Also the error codes can be somewhat mysterious at times, which makes it hard to find the real cause of the problem. So apart from testing the behavior of the system it is also necessary to perform synthesis to insure that the code can be implemented.

10.1.2 Xilinx ISE tool flow

In order to map the VHDL code onto the FPGA an Electronic Design Automation (EDA) tools has to be used. This is provided by the vendor of the FPGA chip, so in this case the Xilinx ISE tool has been used. Due to the proprietary nature of FPGA hardware no open toolchain is available, so the design flow in dictated by the tools provided by the vendors. In Figure 10.3 the design flow used in the Xilinx ISE is shown. First synthesis is performed on the VHDL code, which generates a low-level representation of the system [38]. Usually the target FPGA provided specialized features such as dedicated memory or adders. Code that fit these features are automatically identified by the synthesis process. The mapping tools allocates the resources on the FPGA specified in the low-level presentation provided by the synthesis tool. And the Route & Place routine connects the memory, logic and other blocks used in the design according to a user constraint file (.ucf). This file contains information such as which I/O pins to use and delay constraints in the system. The output of place and route is a database file (.ncd) that contains detailed information of how the the system should be implemented in the FPGA.

Finally the information in the database is used to generate a bit file that can be used to program the FPGA. The is transfered via JTAG either directly to the FPGA or to flash memory that the FPGA can boot from.

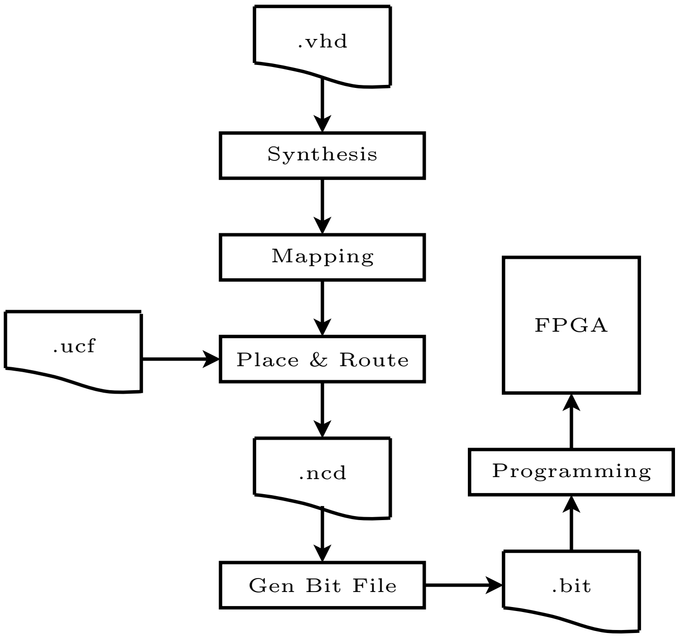


Figure 10.3: Xilinx ISE design flow. Inspired by [38].

Experiences shows that if the code passes the synthesis without any errors or serious warnings, the rest of the tools usually also succeed. For small systems the synthesis-to-bit-file time is relatively low (approximately 2 min). But for more complex system the time increases and for the final system a total time in excess of 15 min was not unlikely.

10.2 System Overview

Figure 10.4 shows a high level schematic of the implemented acquisition system. The outset was to create a self containing system, meaning that constrol as well as communication structures is part of the system.

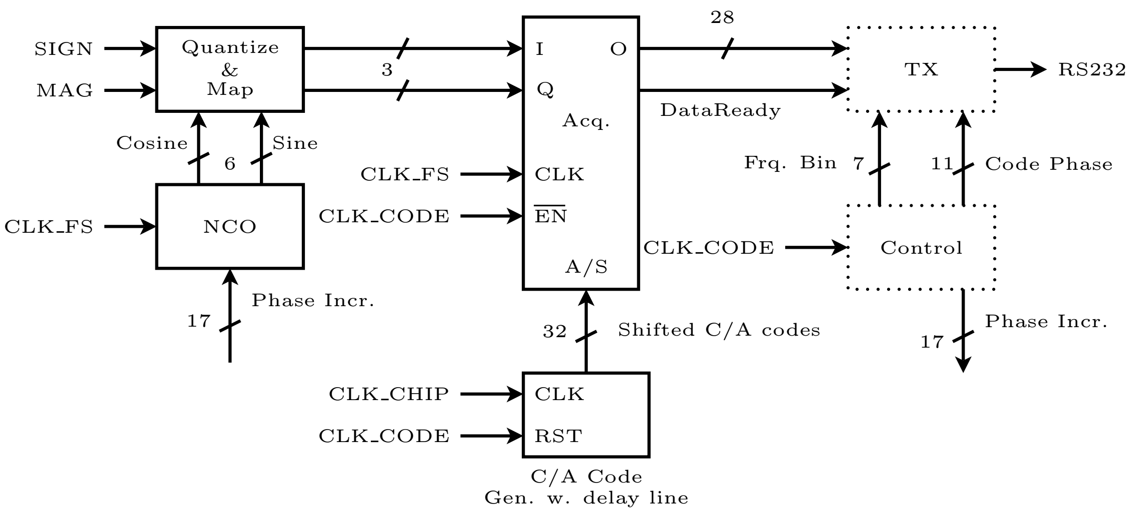


Figure 10.4: High level overview of the implemented system.

The system is driven by two clocks, namely the sampling freuqency (clk_fs) of 5.71 MHz and the double chip frequency (clk_chip) of 2.046 MHz. As there is no simple integer relation between these two clocks, the acquisition is asyncronious.

Most of the system is controlled by the code clock. This clock has a low period of 1 ms and a high period of half a chip. This means that the acqusition process is enabled 1 ms at a time, and that the integration is shifted one half chip for each code clock. In this manner the signal is correlated for different lags over time. When the code clock is high the acquisition block latches the accumulator data one by one to the output, O, and a data ready signal is toggled to indicate that data is available. And, apart from some simple control structures, the acquisition block is a one to one mapping of the proposed acquisition design presented in Section 9.1.

Once 2046 phases have been searched the phase increment value of the NCO is changed to another frequency search bin. This process continous in an endless loop, effectively searching through acquisition search space.

NCO

NCO or numerically controlled oscillator generates the sine and cosine signal necessary to down-convert the received signal from the intermediate frequency down to baseband. In short a NCO consists of a phase accumulator and a sine/cosine look-up table (LUT). The LUTs are used to translate the phase into the corresponding sine or cosine value, whereby the mixing signal is generated. The frequency of the signal is controlled by specifiying how much the phase should be incremented each clock.

In order to implement the NCO a DDS (Direct Digital Syntheziser) Compiler from the Xilinx ISE toolchain was used. Other solutions was also considered such as the NCOs found in the GHLibrary from `www.opencores.org`, but this solution required customization in order to work. So to reduce the workload the IP core from Xilinx was chosen.

Quantize & Map

This process maps the sign and magnitude signal from the GP2015 front-end according to the values found in Section 7.4. That is the sine and cosine signals are quantized to 3-bit, and scaled such that the amplitude of the signal is 1. Similar, the sign and magnitude values are mapped to either 3,2,-2 or -3 according to the state. The multiplication of the signals was implemented by using a look-up table.

C/A Code Generator

The C/A code generator has been designed according to the architecture specified in the GPS manual [6]. The Gold codes are generated using two linear feedback registers, G1 and G2. The feedback is fixed as shown in Figure 10.5. Each clock, the feedback polynomie is calculated and the registers shifted right. Once the registers has been shifted, the result from the feedback is saved in the first entry of the registers. In order to generate the C/A codes the signals from the G1 and G2 are combined in a specific manner. By selecting different phases in G2 the unique codes belonging to the individual GPS satellites can be calculated. The phase combinations that create the individual GPS signals can be found in the GPS manual [6]. As all C/A codes use the same feedback registers, G1 and G2 can be used to generate all C/A codes concurrently.

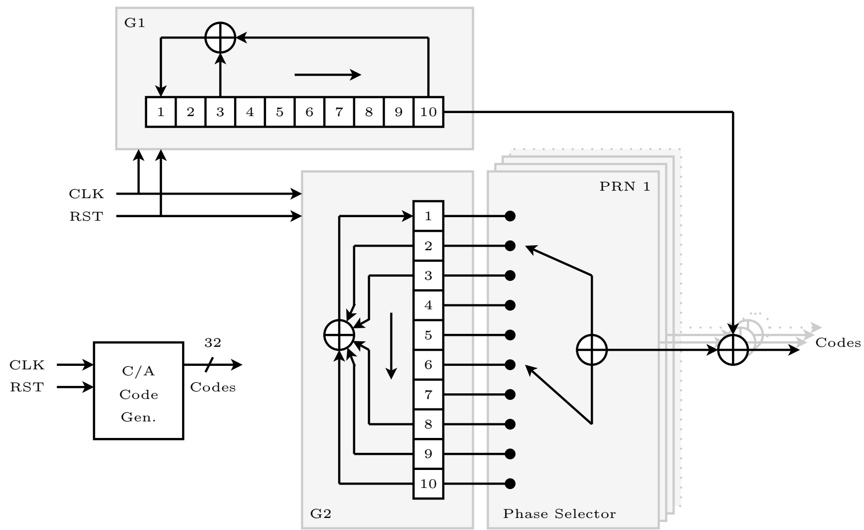


Figure 10.5: C/A code generator

Not shown in Figure 10.5 is the delay line that shifts the signal according to each other. This is importent in order to optain the pipelined result from the accumulators. The delay line has been implemented by using a series of registers as shown in Figure 10.2 on page 72.

Clocks

The system clocks are derived from the 40 MHz clock featured on the front-end. In order to generate the sampling frequency the reference clock has to be divided by 7, as shown in Figure 10.6. This is used internally in the system, but is also used to poll the data from the front-end. In order to generate the fundamental GPS frequency of 10.23 MHz, which all clock in a GPS receiver is derived from, a phase locked loop (PLL) was used. The ratio that yielded the lowest frequency error was found to be 11 to 43, which has an error of 256 Hz or 25 ppm.

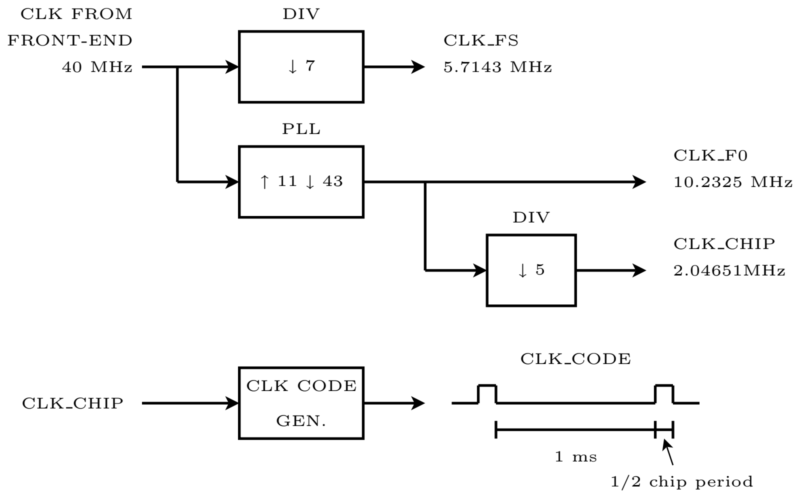


Figure 10.6: Generation of system clocks.

10.3 Results

The system described in the previous section was synthesized and implemented on a Xilinx Virtex5 ML507 development board together with the chosen front-end and antenna. Initially problems had to be overcome as the purchased front-end would not work at 3.3 V. The problem was identified to be a badly designed power-on reset circuit. But once this was fixed the front-end started working as expected. In order to verify the functionality of the implemented system, the internal control signals and clocks was routed to I/Os on the development board. Measurement results on these I/Os are summarized in Table 10.1. As can be seen all of the clocks worked as expected, indicating that the PLLs are working in the FPGA. Also, signals was measured on the data ready signal from the acquisition process. However, the signal showed sawtooth characteristics instead of the expected square wave, which was expected. This might indicate that something is not working in the acquisition block, but it could also be caused by other effect. E.g. the data ready signal is implemented as a tri-state bus. But during synthesis this was changed to internal pull-ups. This combined with any I/O capacitance would induce sawtooth waveforms. So whether acquisition process is running as expected is inconclusive.

Parameter	Status	Remark
Front-end		
- 40 MHz clk	Working	
- Data sampling	Working	
CLK_FS	Working	
CLK_CODE	Working	
CLK_CHIP	Working	
DataReady	Working	Sawtooth shaped
Acqusition	Inconclusive	
TX	Inconclusive	Sporadic bursts of random data

Table 10.1: Test results.

The communication showed some signs of the intended functionally as data was capture using a RS232 dongle under FPGA start-up. The data was however random and made no sense. A consequence of the failing communication is that it has not been possible to conclude whether the GPS signal acquisition works or not. The acquisition does however show some sign of life due to the data ready signal. So in order to conclude anything further reliable communication must be established, so data from the acquisition process can be analyzed. However, as the implementation time was longer than expected no time was left to improve on the communication.

Power/Area

In order to evaluate the power and area of the implementation, the power consumption and the number of used blocks in the FPGA is summarized in Table 10.2. The power results are estimates from the Xilinx XPower Analyzer which is a tool provided by Xilinx to analyze power consumption in their FPGAs.

Name	Value [mW]	Used	Utilization [%]
Clocks	57.8	11	-
Logic	1.3	4010	9
Signals	1	6181	-
IOs	33.1	10	1.4
PLLs	74	2	33.3
Total Quiescent Power	1460		
Total Dynamic Power	93.9		
Total Power	1555		

Table 10.2: Power results from Xilinx XPower Analyzer. The results are relative to a Virtex5 FPGA.

As can be seen the total power consumption of the implementation is above 1 W which was the limit stated by Gomspace. However, as is evident the majority of power is quiescent power, which is the power that the FPGA uses when no switching is performed. This is caused by effects such as leak currents which is architecture dependent. But this is somewhat expected as the Virtex5 has a reputation of being very power hungry, which is also evident from the high psychical temperature of chip (usually between 50-60°C). By changing the FPGA architecture it should be possible to reduce the quiescent power considerably. But this is a subject for further investigation.

The most interesting results in Table 10.2 are the dynamic power figures. The power minimization effort presented in the design space exploration was based on minimizing the dynamic power and the area of the implementation. As can be seen the clocks, PLLs and IOs, all of which switch often, consume the most power. In comparison the area of the implementation (indirectly indicated by the number of logic units and signals) only contribute with very little. Hence, the assumption that minimizing area also minimizes power is not evident from the results. Instead it would be more effective to minimize the number of clocks, PLLs and IOs used in the system.

Regarding the total power of the system, it was found that the front-end and antenna uses approximately 310 mW. Hence, 690 mW is available for the FPGA. As the dynamic power of the implementation is only 93.9 mW a headroom exists of almost 600 mW for quiescent power. This rules out the Virtex5, but other FPGA exists with lower quiescent power, which means it would be possible to implement the signal acquisition algorithm and use under 1 W of power.

Experiences

Al thought the final implementation did not work in all aspects several valuable experiences was drawn from the implementation effort. These are summarized in the following:

- **Modeling systems in VHDL**
  Describing large systems in VHDL is tedious work as the functionality of all blocks has to be checked continuously with test benches. Also it has to be insured that the blocks can be synthesized and whether any warning from the synthesis tool can be ignored or has to be taken care of, which adds to the implementation time. In some cases this means that even simple tasks turn out to be complex to solve. The problem is, however, most dominant if one is new to VHDL developing. Another aspect is if the systems of some reason has to be restructured or clock frequencies are changed. In this case the timing in the subsystems has to be review, which may mean that some blocks have to be redesigned.

- **Finite-State Machines**
  In order to control parts of a system it is often necessary to use finite-state machines. During the implementation it was found that Moore-type FSMs are easy to handle but often lag the flexibility needed in more complex control structures. To solve this, structures with conditional output is used. That is the output of the state machine is determined by its state as well as some input (also known as Mealy-type). This, however, results in more states, which complicates the design and makes it hard to manage as the code size grow fast. The paradox in this context is that when complex state machines are needed, the designer quickly reaches a point where it would be more easy to implement and use a micro controller. And this is definitely one of the lessons learned in implementing this system; finite-state machines are for simple tasks, where micro controllers, with their flexibility, offers a better choice if more complex behavior is needed.

- **Timing**
  As it was found, there is a reason why ASIC engineers talks about the timing being "closed" of a system. The term is used to indicate that the system works under a set of constraints such as a specific clock frequency. If the constraint is not meet the system will not perform as expected. A perfect example of this is midway through the implementation where the sample frequency issue, discussed earlier, was discovered. By changing the sampling frequency the system stopped working. The result was that all test benches had to be re-simulated with the new sampling frequency and the code changed in several places before the system again behaved as expected. This example illustrates the importance of making a thorough analysis in advance, such that constraint do not have to be changed during the design.

Part IV

Closure

Chapter 11

Conclusion

The outset of this project was to design, evaluate and implement a low power GPS acquisition algorithm in FPGA technology. In the following the result of this effort will be summarized.

In the first part to project the problem space of the project was explored. The GPS system and the ideal signal was described to form the basis for the evaluation of GPS signal acquisition in LEO. In order to evaluate the high dynamic environment, empiric data was obtained using the SGP4 perturbation model for 24 GPS satellites and the Ørsted and AAU Cubesat missions. This showed that Doppler shift on the L_1 carrier is approximately uniformly distributed between ± 45 kHz. Also, it was found that the Doppler shift has a mean drift of -29 Hz/s. With an outset in a maximum likelihood estimator-based signal acquisition algorithm the effects of Doppler shift was investigated. This showed that the search space that the acquisition algorithm has to cover was increased considerably in LEO compared to terrestrial applications. Also due to drifting of the GPS signal, the acquisition results is quickly invalidated. Based on the initial results constraints for an acquisition algorithm that would acquire 99% of the GPS signals under continuous receiver operation, was proposed.

In the second part the design space of the algorithm was explored using a combination of guide-search and design space pruning and the ML estimator was mapped to the algorithm domain. The mapping showed that a serial search acquisition with a speed-up factor of 32 would be able to acquire 98.3% of satellites that enter the visible area of the receiving antenna. In order to reduce the power and area of the implementation the effects of sampling frequency and quantization was investigated.

Finally, the algorithm was realized using a semi-systolic array of add/subtract controlled accumulators. The received GPS signal ripples through the accumulators, whereas spreading codes are used to control the add/subtract behavior, effectively correlating the spreading codes for all 32 GPS satellite with the received signal simultaneously. The architecture was modeled in VHDL and implemented on a Xilinx development board. If the quiescent power of the FPGA is disregarded, the final solution, including an active patch antenna and GPS front-end, has a power consumption of approximately 400 mW, which is below the limit imposed by Gomspace. Due to practical issues it was not possible to fully confirm the functionality of the implemented algorithm, but theory and offline simulations support that is would.

Regarding the problem statement, it can be concluded the challenges in relation to using GPS in space has been evaluated and that an algorithm has been designed that is able to cope with the high dynamic environment in LEO. Furthermore a low power algorithm has been implemented on a FPGA. However, the implementation is only partly successful as the functionality has not been confirmed. Also due to the limited project period not all issues have been addressed, which means that room for improvement exist. These areas are discussed in the following chapter.

Chapter 12
Perspectives

Although the basis for the work in this project is GPS, the scenario is relavent in many other applications as well. The system described is basically a satellite to satellite communications system, and the methods used to estimate and predict the Doppler shift would be applicible in similar scenarios as well. Also spread spectrum communication is heavily used in satellite communication as a method to reduce the transmitting power. In all these cases signal acquisition is a part of the system. Finally the principle behind GPS are to some degree similar to those used in the russian Glonass or the Chinese COMPASS GNSS systems.

As for this project not all aspect has been addresed due to time constraints on the project. The problems can be categorized in short term, where a quick fix can be provided and long term where more profound changes are needed.

Short Term

As it has been documented through out the project, several smaller problems appeared during the project. One was the overlooked sampling frequency which to a large extent invalidated the results of the design space exploration. Hence an obvious short term improvement is to get the simulation working with undersampling and redo the design space exploration using the same method. As the sampling frequency is fixed only the quantization has to be reevaluated. Furthermore as the timing in the system is not changed, it would be relatively easy to implement new quantization values.

One of the other problems, was the FPGA communication. The underlying reason for this is almost certainly due to faults in the implementation. Modeling the communications in VHDL was very cumbersome and faults was found continuously during the implementation in this block. In order solve the communication problem it is suggested to implement a microprocessor on the FPGA to handle communication.

Long Term

In the method used, all spreading sequences are correlated in parallel. This is smart as long as no signal is tracked by the receiver, but when a signal is tracked there is no reason why the acquisition module tries to acquire the signal anymore. As 8 satellites is visible on average to the receiver, this means that 1/4 of the acquisition architecture is not utilized. Hence, it would be beneficial to implement a more flexible acquisition algorithm. One example would be an architecture that correlate 32 cells, e.g. 4 by 8, in the acquisition search space according to a single code and then let a microcontroller control area to search. In this manner more intelligent search strategies can be used. E.g. given that a code is present in the signal it is possible to compare the peak value with the noise values or apply some simple pattern recognition algorithms to recognize correlation peaks.

Finally, the quiescent power in the Virtex5 FPGA was found to be prohibitively high, which means that another FPGA has to be considered. The obvious reason is that the power budget is to small on the satellite to accommodate for a power hungry FPGA, the second reason is that the heat from the FPGA has to be dissipated. This is easier said than done in space, due to the lag of atmosphere.

Bibliography

[1] Sunny Leung, Oliver Montenbruck, and Bob Bruninga. Hot Start of GPS Receivers for LEO Microsatellites. *NAVITEC*, December 2001. 1st ESA Workshop on Satellite Navigation User Equipment Technologies NAVITEX'2001, ESTEC Noordwijk.

[2] Steven C. Fisher and Kamran Chassemi. GPS IIF - The Next Generation. *Proceedings of the IEEE*, 87(1), jan 1999. Invited Paper.

[3] Bernard Hofmaann-Wellenhof, Herbert Lichtenegger, and James Collins. *GPS - Theory and Practice*. Springer-Verlag Wien New York, 5 edition, 2001.

[4] Rob Conley. GPS Operational Performance Assessment for 2006. Technical report, Overlook Systems Technologies, Inc, feb 2007. `http://gps.afspc.af.mil/gpsoc/documents/2006_GPS_Performance_Report_V3.pdf`.

[5] Department of Defence and NAVSTAR GPS. *Global Positioning System Standard Positioning Service Performance Standard*, 4 edition, sep 2008. `http://gps.afspc.af.mil/gpsoc/documents/Signed_SPS_PS_(Sept_30_2008).pdf`.

[6] Navstar GPS Joint Program Office. *Navstar Global Positioning System - Interface Specification*, rev. d edition, December 2004. IS-GPS-200D.

[7] Gilbert Strang and Kai Borre. *Linear Algebra, Geodesy, and GPS*. Wellesley-Cambridge Press, 1997.

[8] Kai Borre, Dennis M. Akos, Nicolaj Bertelsen, Peter Rinder, and Søren Holdt Jensen. *A Software-Defined GPS and Galileo Receiver - A Single-Frequency Approach*. Birkhäuser, 2007.

[9] Simon Haykin. *Communication Systems*. John Wiley & Sons, Inc, 4 edition, 2001.

[10] Per Enge and Pratap Misra. Special issue on global positioning system. *Proceedings of the IEEE*, 87(1):3,15, January 1999.

[11] Per Enge. GPS Modernization: Capabilities of the New Civil Signals. *Invited Paper for the Austrailian International Aerospace Congress*, August 2003.

[12] Michael S. Braasch and A.J. Van Dierendonck. GPS Receiver Architectures and Measurements. *Proceedings of the IEEE*, 87(1):48,64, jan 1999.

[13] Mosa Ali Abu-Rgheff. *Introduction to CDMA Wireless Communication*. Elsevier, 2007.

[14] Bradford W. Parkinson, James J. Spiker Jr., A.J. Van Dierendonck, P. Axelrad, et al. Progress in astronautics and aeronautics. In Bradford W. and Jr. James J. Spiker Parkinson, editor, *Global Positioning System: Theory and Applications*, volume 163 of *Volume 1*. American Institute of Aeronautics and Astronautics, Inc, 1996.

[15] Penina Axelrad and Charles P. Behre. Satellite Attitude Determination Based on GPS Signal-to-Noise Ratio. *Proceedings of the IEEE*, 87(1), jan 1999.

[16] Oliver Montenbruck, Markus Markgraf, Sunny Leung, and Eberhard Gill. A GPS Receiver for Space Applications. *ION-GPS Conference, Salt Lake City*, Sep 2001.

[17] M. Markgraf, S. Vaccaro, I. Kalnins, P. de. Maagt, and O. Amend. The Asolant/Rubin-5 Tehcnology Demonstration Mission - System Description and First Flight Results. *The 4S Symposium*, September 2006.

[18] Oliver Montenbruck, Ben Nortier, and Sias Mostert. A Miniature GPS Receiver for Precise Orbit Determination of the Sunsat 2004 Micro-Satellite. *IOT NTM*, January 2004.

[19] Deutche Luft und Raumfart. Pheonix - Spaceborn GPS Receiver. Technical report, DLR, 2010. `http://www.weblab.dlr.de/rbrt/GpsNav/Phoenix/Phoenix_DataSheet_v11.pdf`.

[20] Wikipedia. Low Earth Orbit. Internet, `http://en.wikipedia.org/wiki/Low_Earth_orbit`, May 2010.

[21] Irfan Ali, Naofal Al-Dhahir, and John E. Hershey. Doppler Characterization for LEO Satellites. *IEEE Transactions on Communications*, 46(3), March 1998.

[22] Jr John W. Jewett and Raymond A. Serway. *Physics for Scientists and Engineers*. Thomson Brooks/Cole, 6 edition, 2004.

[23] Micheal P. McLaughlin. *A Compendium of Common Probability Distributions*. Appendix A of Regress+. 3 edition, 1993-2001. mpmcl@mitre.org.

[24] Frank van Diggelen. *A-GPS: Assisted GPS, GNSS, and SBAS*. Artech House, 2007.

[25] W.J. Hurd, J.I. Statman, and V.A Vilnrotter. High Dynamic GPS Receiver Using Maximum Likelihood Estimation and Frequency Tracking. *IEEE Transactions on Aerospace and Electronic Systems*, AES-23(4), jul 1987.

[26] Gordon J.R. Povey. *Wireless Personal Communications*. Kluwer Academic Publishers, 1998. Chap: Spread Spectrum PN Code Acquisition using Hybrid Correlator Achitectures.

[27] Unjeng Cheng, W.J. Hurd, and J.I. Statman. Spread-Spectrum Code Acquisition in the Presence of Doppler Shift and Data Modulation. *IEEE Transactions on Communications*, 38(2), feb 1990.

[28] M.K. Sust, R.F. Kaufmann, F. Molitor, and G.A. Björnström. Rapid Acquisition Concept for Voice Activated CDMA Communication. *Global Telecommunications Conference, 1990, and Exhibition. 'Communications: Connecting the Future', GLOBECOM '90., IEEE*, 3, 1990.

[29] R.A. Stirling-Gallacher, A.P. Hulbert, and Gordon J.R. Povey. A Fast Acquisition Technique for a Direct Sequence Spread Spectrum Signal in the Presence of a large Doppler Shift. *IEEE*, 1996.

[30] Zarlink Semiconductor. GP2015 - GPS Receiver RF Front End. Technical Report Issue 3.1, Zarlink Semiconductor Inc., feb 2002.

[31] Constantine A. Balanis. *Antenna Theory - Analysis and Design*. Wiley-Interscience, 3 edition, 2005.

[32] Matthias Gries. Methods for Evaluating and Covering the Design Space during Early Design Development. *Integration, the VLSI Journal*, may 2004.

[33] Anantha P. Chandrakasan and Robert W. Brodersen. Minimizing Power Consumption in Digital CMOS Circuits. *Proceedings of the IEEE*, 83(4), apr 1995.

[34] Xilinx. Virtex-5 FPGA Data Sheet. Technical Report v5.3, may 2010. DS202.

[35] Nicolaj Bertelsen, Kai Borre, and Peter Rinder. The GPS Software Receiver at Aalborg University. Technical report, Danish GPS Center, Aalborg Univerity.

[36] Kenneth A. De Jong. *Evolutionary Computation - A Unified Approach*. The MIT Press, 2006.

[37] H.T Kung. Why Systolic Architectures? *IEEE*, jan 1982.

[38] Peter J. Ashenden. *The Studentøs Guide To VHDL*. Morgan Kaufmann, Elsevier, 2 edition, 2008.

[39] Reiner Hartenstein. The von Neumann Syndrome. *Stamatis Vassiliadis Memorial Symposium*, sep 2007.

[40] U.S Coast Guard Navigation Center. U.S. Policy Statement Regarding GPS Availability, March 21, 2003. Internet. `http://www.navcen.uscg.gov/GPS/`.

[41] Sheldon M. Ross. *Introduction to Probability and Statistics for Engineers and Scienctists*. Elsevier, 3 edition, 2004.

[42] Hans Ebert and Poul Raskmark. *Grundlæggende Transmissionledningsteori*. Aalborg Universitet, 3 edition, aug 2008.

[43] Unión Internacional de Telecomunicaciones. *Radio Noise*. ITU, aug 2007. Recommendation ITU-R P.372-9.

[44] Thomas A. Milligan. *Modern Antenna Design*. Wiley, 2 edition, 2005.

[45] Melvin M. Weiner. *Monopole Antennas*. CRC Press, 2003.

[46] Keith R. Carver and James W. Mink. Microstrip antenna technology. *IEEE Transactions on Antennas and Propagation*, AP-29(1), jan 1981.

[47] David M. Pozar. Microstrip antenna. *Proceedings of the IEEE*, 80(1), jan 1992.

[48] J. R. James. *Handbook of Microstrip Antennas*, volume 2. The Institution of Engineering and Technology, jun 1988.

[49] E. Alboni and M. Cerretelli. Microstrip patch antenna for gps applications. Internet, `http://www.taconic-add.com/pdf/technicalarticles--microstrip_patch_antenna_for_gps.pdf`, 2002.

[50] X. L. Bao, G. Ruvio, M. J. Ammann, and M. John. A novel gps patch antenna on a fractal hi-impedance surface substrate. *IEEE Antennas and Wireless Propagation Letters*, 5, 2006.

Part V

Appendix

Appendix A
Global Positioning System History

The development of the Global Positioning System (GPS) was initiated in the early seventies by the U.S Department of Defense (DoD) to replace the older TRANSIT system, which suffered from low accuracy and high update rates [3]. The system was primarily intended to provide precise estimation of position, velocity and time around the globe for the U.S military. Civil use of the system was only considered as a secondary objective and the performance was purposely degraded on the basis of national security [10]. Initially the accuracy of the civil system was intended to be around 400-500 m [10][3]. But as civil systems at the time had an accuracy of 100 m the DoD decided that the standard positioning service (SPS) provided to civilians should have the same accuracy. However, early field test showed that the precision was in the order of 15 to 40 m [3]. As such, the DoD decided to impose a dilution of precision method measure known as *selective availability* (SA), which degraded the system accuracy to 100 m horizontally and 156 m in height [3]. SA was activated in 1990, five years before the GPS system was announced fully operational, and have been so until a Presidential Decision Directive in 1996 dictated that SA did not serve it original purpose [10]. However, SA was first turned off by the year 2000[1] enabling more accurate civil positioning.

The GPS system was primarily designed to serve a predicted number of 40.000 military users[11]. In the beginning, the number of civilian users was limited by the high price of receivers. But as the price of ICs went down throughout the nineties, and hence the price of GPS receivers, the number of users started to grow drastically. By 2003 there was an estimated 20 mill. civilian users and the industry was shipping 100.000 civil receivers per month [11]. Today GPS receivers have become common households objects and are incorporated in a multitude of applications and have by far outnumbered the military receivers. As such the GPS system has come to play an important role in the information infrastructure around the world. Obviously the U.S government is aware of this and have in response stated “We believe we can ensure that GPS continues to be available as an invaluable global utility at all times, while at the same time, protecting U.S. and coalition security requirements.” [40]. They go on to ensure that they would “provide the best possible service to civil [...] users worldwide. This is as true in times of conflict as it is in times of peace”. However it should be noted that the U.S government do not promise anything. But considering the many users today it must be considered unlikely that changes degrading the system would be made in the foreseeable future.

GPS modernization

Over the last decade the GPS system has undergone a major modernization[2], which has introduced a lot of changes and improvements to the system, especially for civil users. The improved system, generally referred to as GPS III, has in particular introduced new signal components; both for military and civil users, which will improve accuracy. However, the new system is not

[1] `http://en.wikipedia.org/wiki/Global_Positioning_System`
[2] `http://www.navcen.uscg.gov/gps/modernization/default.htm`

expected to be fully implemented until 2013[3] or 2015 [11]. Hence GPS is in a state of transition, where some of the new signals are available and some not.

[3] http://en.wikipedia.org/wiki/GPS_modernization

Appendix B

Maximum Doppler shift

This appendix gives an estimation of the maximum doppler shift that a LEO satellite would experience in orbit. In a stationary terrestrial GPS receiver there are two sources to the doppler shift. That is, the relative movement between the user and GPS satellite and any clock bias in the receiver. In such application the doppler shift is usually assumed to never exceed ± 5 kHz and ± 10 kHz including clock bias [8]. However, a satellite in LEO orbits with a considerable larger speed than any ground application would move, hence the doppler shift must also be expected to be larger.

B.1 Doppler shift between two moving objects

The doppler shift between a receiver and source moving on a line with speeds v_r and v_s is given by

$$f_d = f_0 \left(\frac{c + v_r}{c - v_s} \right) , \tag{B.1}$$

[22, p. 525]

where f_0 is the frequency of the source signal and f_d is that of the received doppler shifted signal. In B.1 the receiver and source are expected to move towards each other when v_r and v_s are positive and away when negative. That is, the doppler shift becomes larger when object move towards each other and smaller in the opposite case.

However, objects orbiting in space, e.g. satellites, seldom move on a straight line manner, which means that B.1 can not be use directly. Instead one have to find the rate of change in distance between the objects, given their position and velocity vector, as illustrated in Figure B.1

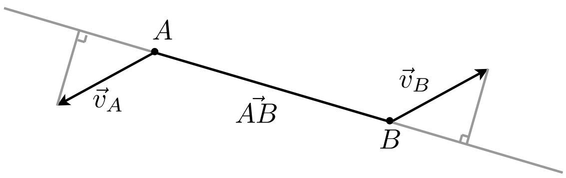


Figure B.1:

Now, let A represent the receiver and B the source, with velocity vectors $\vec{v}_A$ and $\vec{v}_B$. Then the line-of-sight (LOS) vector between A and B can be described by

$$\vec{AB} = \vec{b} - \vec{a} \, , \tag{B.2}$$

where $\vec{a}$ and $\vec{b}$ are the vectors to A and B relative to the earth center. In order to obtain the line-of-sight velocity between A and B the velocity vectors are projected onto $\vec{AB}$. This is given by the wellknown relations

$$|\vec{v}_{A,LOS}| = \frac{\vec{AB} \cdot \vec{v}_A}{|\vec{AB}|} = v_r \tag{B.3}$$

$$|\vec{v}_{B,LOS}| = \frac{\vec{AB} \cdot \vec{v}_B}{|\vec{AB}|} = v_s \,, \tag{B.4}$$

where $(\cdot)$ is the dot-product. Using the above equations together with B.1 it is possible to calculate the doppler shift between two objects in orbit given their earth-centered-earth-fixed (ECEF) coordinates and velocity vectors.

B.2 Maximum doppler shift

In order to estimate the maximum doppler shift that the satellite receiver would experience it is necessary to know the ECEF coordinates of the satellites as well as their velocity. This can be obtained in several ways. One way is to use the two-line elements (TLE) provided by NORAD. This would provide real life orbit coordinates from satellites orbiting the earth. Although precise, the orbit data obtained from the TLEs would only provide the maximum doppler shift between the two satellites chosen, which most likely would be a local maximum. In order to obtain a global maximum the coordinates are found by assuming that the satellites are in circular orbits and in the same orbital plane. The scenario is illustrated in Figure B.2. These assumption makes it easy to compute the satellite position and velocity using simple trigonometric and would provide a global maximum doppler shift as they are in the same plane.

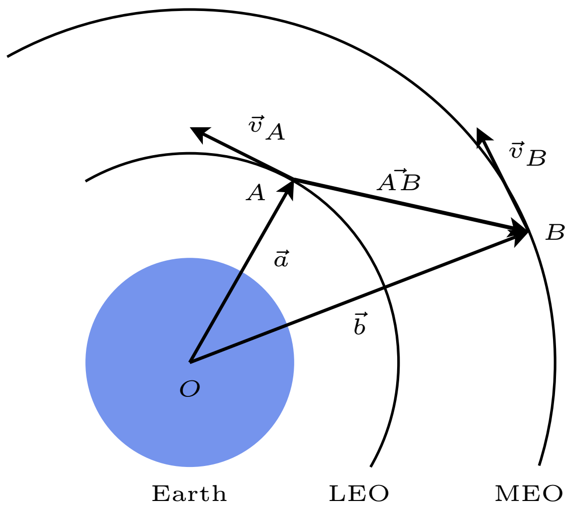


Figure B.2: Illustration of the satellite orbits and vectors associated assuming perfectly circular orbits.

From Figure B.2 it can be seen that point A and B, representing the “Gomspace” and GPS satellite respectively, can be described by the following vector

$$\begin{bmatrix} \vec{a} & \vec{b} \end{bmatrix} = \begin{bmatrix} \cos(\theta_A) & \cos(\theta_B) \\ \sin(\theta_A) & \sin(\theta_B) \end{bmatrix} \begin{bmatrix} r_A & 0 \\ 0 & r_B \end{bmatrix} . \tag{B.5}$$

Hence the line of sight vector between A and B is given by

$$\vec{AB} = \begin{bmatrix} r_B\cos(\theta_B) - r_A\cos(\theta_A) \\ r_B\sin(\theta_B) - r_A\sin(\theta_A) \end{bmatrix}. \tag{B.6}$$

The velocity vectors are almost similar to B.5 except that the rotation matrix is offset by 90° and multiplied with the absolute speed of the satellites. This yields

$$\begin{bmatrix} \vec{v}_A & \vec{v}_B \end{bmatrix} = \begin{bmatrix} \sin(\theta_A) & \sin(\theta_B) \\ -\cos(\theta_A) & -\cos(\theta_B) \end{bmatrix} \begin{bmatrix} v_A & 0 \\ 0 & v_B \end{bmatrix}. \tag{B.7}$$

As the range to the GPS satellites are given and the Gomspace satellite is in LEO ($r_A = R_E + \{150...2000\}$ km), the only unknowns are the speed of the satellites. Fortunately the speed of a satellite in circular orbit can be found by equaling the gravitational force on the satellite with Newton's second law. This gives

$$v = \sqrt{\frac{GM}{r}}, \tag{B.8}$$

[22, p. 400]

where $G = 6.6726 \cdot 10^{-11}$ Nm^2 kg^{-2} is the gravitational constant, M the mass of the earth and r the earth center distance to the satellite. Hence the relation between the range to satellite A and B and the their velocities is given by

$$\begin{bmatrix} v_A \\ v_B \end{bmatrix} = \sqrt{GM} \begin{bmatrix} \frac{1}{\sqrt{r_A}} \\ \frac{1}{\sqrt{r_B}} \end{bmatrix} \tag{B.9}$$

Dividing with the ranges gives the angular velocity

$$\begin{bmatrix} \omega_A \\ \omega_B \end{bmatrix} = \sqrt{GM} \begin{bmatrix} \frac{1}{\sqrt{r_A^3}} \\ \frac{1}{\sqrt{r_B^3}} \end{bmatrix} \tag{B.10}$$

Using the equation above and the relation $\omega = \frac{\Delta\theta}{\Delta t}$ it is possible to evaluate the satellites positions and velocities at different epochs using Eqn. B.5 and B.7 and the range to each satellite.

In the previous section the positions and velocities of the Gomspace and GPS satellite were found given the angle and range to the satellites. In order to find the maximum doppler shift it is necessary to evaluate the doppler shift equation and hence the projected velocities given in Eqn. B.3 and B.4. The largest doppler shift occurs when the receiver and source speed are at their maximum, i.e. moving towards each other. I the present case the receiver and source speed are denoted $v_A = |\vec{v}_{A,LOS}|$ and $v_B = |\vec{v}_{B,LOS}|$, respectively.

When expanded, the numinator of B.3 and B.4 gives

$$\vec{AB} \cdot \vec{v}_A = v_A\sin(\theta_A)\left(r_B\cos(\theta_B) - r_A\cos(\theta_A)\right) - v_A\cos(\theta_A)\left(r_B\sin(\theta_B) - r_A\sin(\theta_A)\right) \tag{B.11}$$

$$\vec{AB} \cdot \vec{v}_B = v_B\sin(\theta_B)\left(r_B\cos(\theta_B) - r_A\cos(\theta_A)\right) - v_B\cos(\theta_B)\left(r_B\sin(\theta_B) - r_A\sin(\theta_A)\right). \tag{B.12}$$

Similar the denominator is

$$|\vec{AB}| = \sqrt{(r_B\cos(\theta_B) - r_A\cos(\theta_A))^2 + (r_B\sin(\theta_B) - r_A\sin(\theta_A))^2}. \tag{B.13}$$

From the above equations it can be seen that the vector products have a maximum at $\theta_A = \frac{\pi}{2}$ and $\theta_B = 0$, i.e. when $\vec{a}$ and $\vec{b}$ are perpendicular to each other. Hence substituting the angles into the vector products and the norm yields

$$\vec{AB} \cdot \vec{v}_A|_{\theta_A=\pi/2,\theta_B=0} = r_B v_A \tag{B.14}$$

$$\vec{AB} \cdot \vec{v}_B|_{\theta_A=\pi/2,\theta_B=0} = r_A v_B \tag{B.15}$$

$$|\vec{AB}|_{\theta_A=\pi/2,\theta_B=0} = \sqrt{r_A^2 + r_B^2}. \tag{B.16}$$

Substituting into the doppler equation B.1 we get

$$f_{d,max} = f_c \left(\frac{c + \frac{r_B v_A}{\sqrt{r_A^2+r_B^2}}}{c - \frac{r_A v_B}{\sqrt{r_A^2+r_B^2}}} \right). \tag{B.17}$$

Hence the maximum doppler shift between two objects in circular orbit around the earth is given as a function of the altitude and speed of the objects. By using the relationship between the altitude and speed $v = \sqrt{\frac{GM}{r}}$, Eqn. B.17 can be further reduced so that it only depends on the altitude

$$f_{d,max} = f_c \left(\frac{c + \frac{r_B}{\sqrt{r_A}}\sqrt{\frac{GM}{r_A^2+r_B^2}}}{c - \frac{r_A}{\sqrt{r_B}}\sqrt{\frac{GM}{r_A^2+r_B^2}}} \right). \tag{B.18}$$

LEO extents from approximately 150 km to around 2.000 km. GPS satellites orbits at an altitude of approx. 20.500 km. Using this information the maximum doppler shift have been plotted over the span of the LEO orbit; see Figure B.3.

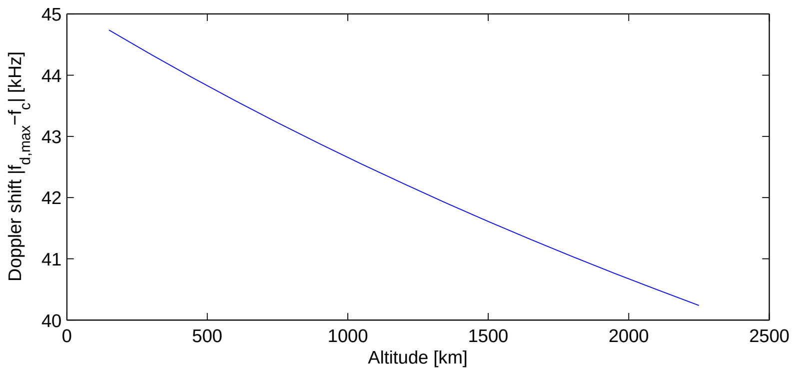


Figure B.3: Maximum doppler shift on L_1 for GPS applications at different LEO altitudes.

As it is clear from Figure B.3 the maximum doppler shift occurs when in low orbit around the earth. This is because the orbit speed according to B.8 is larger at low altitude. This also means that LEO is the most demanding regarding the demands posed on the receiver. In terrestrial receivers it is normally assumed that the doppler shift never exceeds ± 10 kHz including clock bias. A space GPS receiver would have to be designed for doppler shifts of ± 45 kHz plus clock bias.

Appendix C
Doppler shift estimation

C.1 Doppler shift distribution

Due to the constant movement between the GPS satellites and the GPS receiver, the doppler shift would change constantly during an orbit. Finding the distribution of the doppler shifts could provide valuable information about the dynamic environment that the receiver has to work in. This information is essential in the design process of the receiver and two cases are of special interest. That is the distribution of the doppler shift and its derivative at 1) all epochs where the GPS satellites are visible to the receiver and 2) the epochs where a GPS satellite have just become visible to the receiver. The first case will yield information of the general dynamics of the doppler shift, i.e. how fast does the doppler shift change? what is the maximum doppler shift for the given orbit ect. This information is very helpful when designing the mechanism that have to track the carrier frequency to keep the receiver in sync with the satellites. In the second case, knowing the distribution could be used to design an efficient search strategy for the signal acquisition.

Method

Although several satellites have been equipped with GPS receivers no raw data on the doppler shift, known to this author, have been published or made available. Instead the data can be obtained by using the orbital elements that describe the orbits of satellites in space. The orbital elements are provided by the NORAD and are freely available on the internet in a format known as two-line-elements (TLE). The TLE for the GPS satellite identified with PRN 29 is given in the following.

```
 GPS BIIRM-5 (PRN 29)
1 32384U 07062A    09284.90253878  .00000026  00000-0  10000-3 0  3817
2 32384  55.0077 154.9802 0035218 284.4617  75.3188  2.00571599 13381
```

The precise meaning of each entry is too comprehensive to describe in this section as it would require a detailed description of how orbits are described. But to give an example the second entry in line two (55.0077) is the inclination in degrees of the satellite orbit according to the equatorial plane. The TLE can be converted to ECEF coordinates and velocities at a given epoch using a model known as SPG4. A reference model is available in FORTRAN but for this simulation a Matlab implementation used by the AAU space program have been used. Using the model the coordinates and velocities of two LEO satellites and 24 of the GPS satellites have been found at 1 second epochs over a 2 day period. Subsequently the data where used to calculated the doppler shift at each epoch. For the simulation the AAUSAT-II and Ørsted satellites where used as reference.

In the simulation it is assumed that the receiving antenna is half-isotropic (half-sphere with 0 dB gain) and that it is zenith looking at all time i.e. pointing out into space. To take this into account the doppler data is only collected when the angle θ_A in Fig. C.1 is between 0 and π

or when the GPS satellite is in line-of-sight with the antenna. The angle is calculated using the relation given in Eqn. C.1.

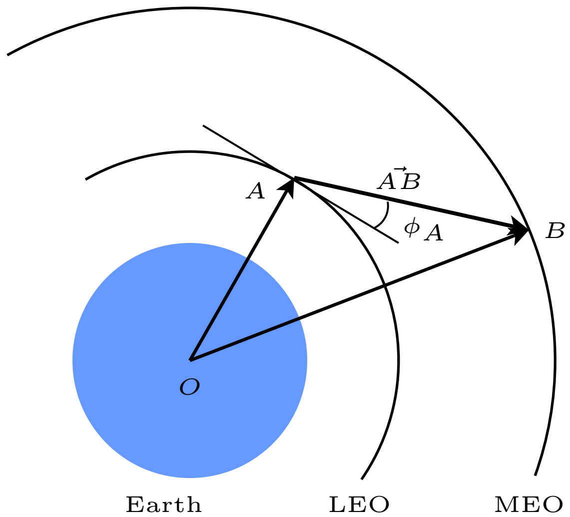


Figure C.1: Illustration of the simulation scenario. The data is only collected when the angle θ_A between A and B is in the interval $[0, \pi]$.

$$\phi_A = \cos^{-1}\left(\frac{\vec{a}\cdot\vec{b}}{|\vec{a}||\vec{b}|}\right) - \frac{\pi}{2} \tag{C.1}$$

Results

Figure C.2 shows the distribution of the doppler shift offset and its rate of change on the L_1 carrier for the Ørsted and AAU Cubesat LEO satellites. The distribution of the doppler shift is symmetric around zero. Both distributions are limited at just over 40 kHz (43 kHz to be precise), which confirms the earlier predictions for the maximum doppler shift. The distribution can to some degree be described as uniform if a limited interval near zero is considered. Overall the distribution would best be described by a semi-circular distribution with zero mean [23].

A semi-circular pdf is described by the following relation

$$\mathrm{pdf}(f_d) = \frac{2}{\pi f_{d,max}}\sqrt{1-\left(\frac{f_d}{f_{d,max}}\right)^2} \tag{C.2}$$

and the variance is given by

$$\sigma^2 = \frac{(f_{d,max})^2}{4} \tag{C.3}$$

The distribution of the derivative is skewed in the positive direction and only in very few cases is the derivative negative. In other words given the doppler shift at one epoch it would most likely be between 0 and 4 Hz larger the next i.e. increasing. This makes sense as the velocity of LEO satellites is much grater than that of the GPS satellites - meaning that they are constantly gaining on the GPS satellites.

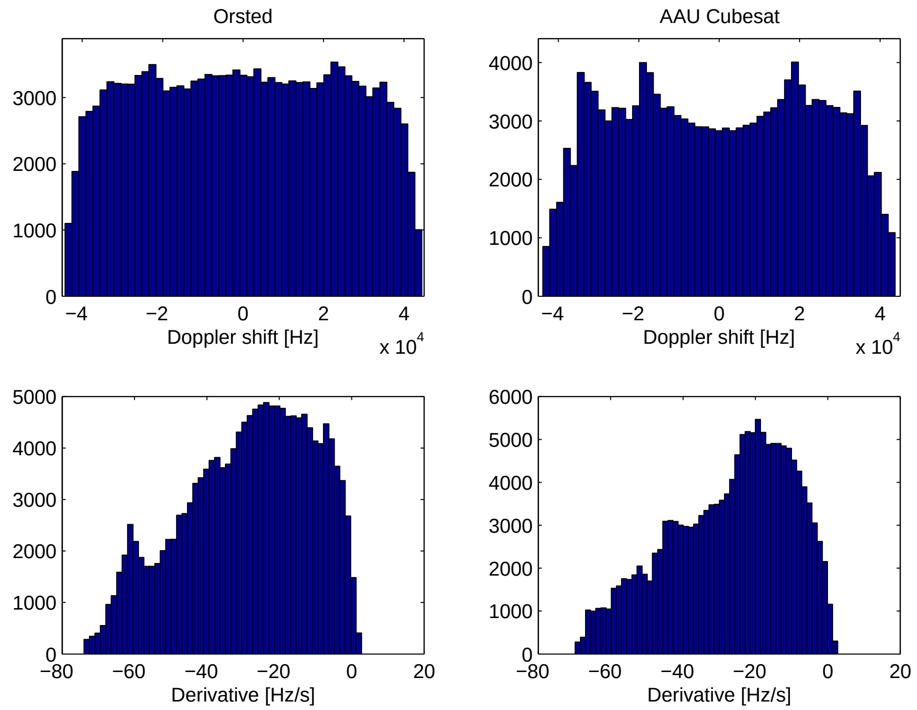


Figure C.2: Distribution of the line-of-sight and rate of change doppler shift for the AAUSAT-II and Ørsted satellites according to 24 GPS satellites. The data have been collected over a 2 day period in 10 second intervals.

		Ørsted	AAU Cubesat
μ	[Hz]	46	36.5
σ^2	[Hz^2]	5.83e8	5.9e8
$[f_{d,min}, f_{d,max}]$	[Hz]	[-44294 , 44449]	[-43873 , 43740]
μ'	[Hz/s]	-28.81	-27.12
$\sigma^{2'}$	[Hz^2/s^2]	318.2	282.3

Table C.1:

Figure C.3 shows the doppler shift the moments when a new GPS satellite have become visible to the receiver. The distribution is clearly skewed and can be approximated as a triangular distribution. The derivative is also skewed, but in the positive direction. In general Fig. C.3 shows that the doppler shift of signals from "new" GPS satellites is strongly biased towards the minimum received doppler shift. Also the probability that a doppler shift is larger at one epoch to the next is high. In the majority of cases signal drifting can be assumed to be negative. Only in 0.56% and 0.31% of the case is the drift positive, for the Ørsted and AAU Cubesat respectively.

Analysis error

The above analysis is prone to gross errors if the scenario is not compatible to xthat of the Orsted and AAU CubeSat. That is, the orbits have to correlate with that of these missions. Particular the inclination of the orbits affect the distribution of the doppler shift. And in general each orbit is associated with its own set of statistics, which is also evident from the above.

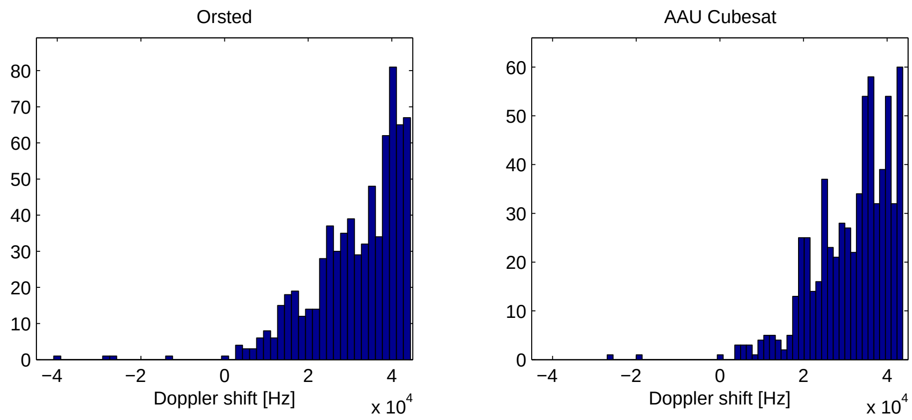


Figure C.3: Distribution of the doppler shift and its derivative at the epochs where a GPS satellite have just entered the sight of the receiving antenna.

C.2 Time window

Due to the shadowing effect of the earth and beam form of the receiving antenna the GPS satellite signals continues to move between a state where they are visible to the receiver and not. The window of time where a GPS satellite is in sight it is possible to acquire and track the signal. Hence the performance w.r.t time of the receiver is dependent on the size of this window. A short window leaves less time to acquire the signals as some time also must be allocated to track and gather navigation data. Also the time between the appearance of new GPS satellites is of interest as it gives an indication of how often the acquisition have to run.

Method

The time window data is collected using the same framework described in the Section C.1. That is, the length of the periods where each GPS satellite is visible to the receiver is in sight. Afterwards the distribution of the data is found for the length of the window and time between new GPS satellites, respectively.

Results

As can be seen from Fig C.4 the time period that a satellite is visible to a receiver is in most cases over 2000 s or 33 min. As each navigation data frame is 30 s long there is more than enough time to gather data from the satellites. However, time for acquisition also have to allocated which reduces the window size. One thing, a bit surprising, is that a few of the time windows exceed the revolution time for the satellites. E.g. the AAU CubeSat have a revolution of approximately 6084 s. Hence a few of the GPS satellites have an orbit that at some epochs makes them visible to the satellite for more than one revolution.

Fig. C.4 shows the distribution of the interval between new satellites becoming visible to the receiver. This result is interesting as it states that the interval between new satellites becoming visible is on average rather small.

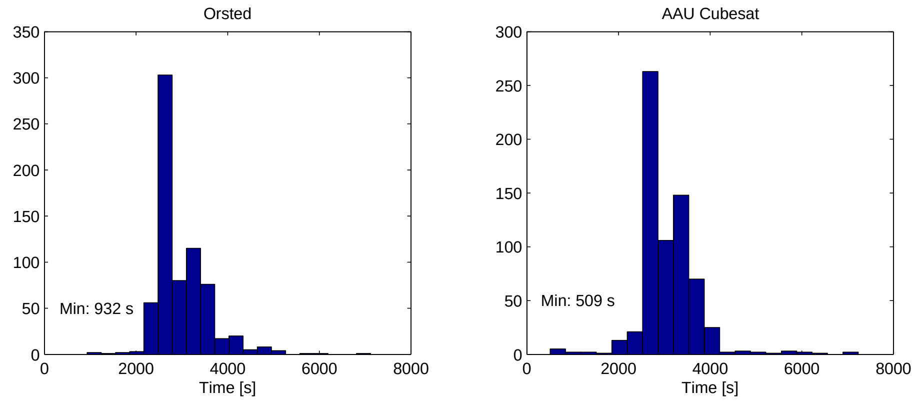


Figure C.4: Distribution of the visibility period length.

		Ørsted	AAU Cubesat
μ	[s]	2269	2412
σ^2	[s^2]	2.45e5	2.9e5
Kurtosis		9.8	10.9
Range	[s]	[463 , 5153]	[545 5638]

Table C.2: Window length statistics.

Appendix D

Search Range Analysis

Acquisition of a signal is only necessary from the time it appears in the received signal to it has been acquired, whereafter a channel is allocated to track the signal. From Appendix C.1 we know that the inteval between new or non-acquired GPS satellites is 12 seconds on average. Furthermore signals that have just become visible to the receiver can be approximated by a right triangular distribution between $[0, f_{d,max}]$. Hence, it is possible to acquire all GPS signals by searching half the range of possible doppler shifts on a regular basis. The problem is that, due to the relative motion between the receiver and satellites, the doppler shift drifts over time. This means that as time progresses some, otherwise acquirable, signals drift out of the range searched by the acquisition process. Hence, the question is how long time the acquisition process have to acquire the signals, according to some acceptable level of missed satellites?

Let Y_n denote the r.v that describes the doppler shift of an acquirable signal after n seconds, X_0 the initial doppler shift of the signal and $X_{1...n}$ the random drift at second $\{1...n\}$ from the initial doppler shift. Then Y_n can be expressed as:

$$Y_n = X_0 + S_n \quad \text{where} \quad S_n = X_1 + X_2 + ... + X_n \tag{D.1}$$

and S_n is the total drift after n seconds. As found in Appendix C.1, the initial doppler shift X_0 can be approximated by a right triangular distributed random variable, described by the following pdf:

$$f_{X_0}(x) \quad = \quad \frac{2}{f_{d,max}^2} x, \quad 0 \leq x \leq f_{d,max} \tag{D.2}$$

Furthermore the drift was found to be Beta-like distributed r.v. with mean $\mu = -29$ Hz/s and variance $\sigma^2 = 318$. If it is assumed that the drift between two epochs are independant[1] the distribution of S_n can be found by invoking the central limit theorem, which states that the distribution of the sum of n independaent and identically distributed random variables is approximately normal with $S_n \sim \mathcal{N}(n\mu, n\sigma^2)$ [41, p. 204].

$$f_{S_n}(x) \quad = \quad \frac{1}{\sqrt{2\pi n}\sigma} \exp\left\{\frac{-(x-n\mu)^2}{2n\sigma^2}\right\} \tag{D.3}$$

At this point the pdf of Y_n can be found by convoluting $f_{X_0}(x)$ and $f_{S_n}(x)$ as given in Eqn. D.4. Given $f_{Y_n}(x)$ it is possible to evaluate the level of missed satellites according to the acquisition time n. Unfortunately, it is not trivial to calculate the solution of D.4 as it involves integrating Eqn. D.3.

[1] The drift between two epochs is not independent due to underlying periodic nature of the system, but this is a general weekness of the statistical appoach.

$$f_{Y_n}(x) = f_{X_0}(x) * f_{S_n}(x) \tag{D.4}$$

To circumvent this a constant threshold value x_0 is chosen from $[0, f_{d,max}]$, whereby Eqn. D.1 is transformed into a linear combination between a constant and a normal r.v. E.i.:

$$Y_n = x_0 + S_n \tag{D.5}$$

In this case Y_n is also normal, but with $Y_n \sim \mathcal{N}(x_0 + n\mu, n\sigma^2)$ [41, p. 170]. However, by doing this it is accepted that some of the signals are missed due to drifting. Lets start out by saying that one miss in a hundred is acceptable, then $P\{x_0 \leq x \leq f_{d,max}\} = 0.99$ in Figure D.1.

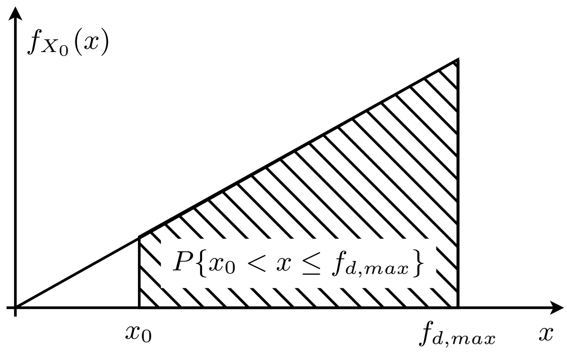


Figure D.1: sadf

Given the triangular distribution of Eqn. D.2 the marked area in Figure D.1, equivivalent to $P\{x_0 \leq x \leq f_{d,max}\}$, can be derived:

$$\begin{aligned} P\{x_0 \leq x \leq f_{d,max}\} &= 1 - P\{x < x_0\} && \text{(D.6)} \\ &= 1 - \int_0^{x_0} f_{X_0}(x)dx && \text{(D.7)} \\ &= 1 - \frac{2}{f_{d,max}^2}\left[\frac{1}{2}x^2\right]_0^{x_0} && \text{(D.8)} \\ &= 1 - \frac{x_0^2}{f_{d,max}^2} && \text{(D.9)} \end{aligned}$$

Rearranging gives

$$x_0 = \sqrt{1 - P\{x_0 \leq x \leq f_{d,max}\}} f_{d,max} \tag{D.10}$$

It is now possible to find x_0 for which we know that 99% of the acquirable satellites are within the interval $[x_0, f_{d,max}]$.

$$x_0 = \sqrt{1 - 0.99} \cdot 45000 \text{ Hz} = 4500 \text{ Hz}$$

Hence the probability density function for the doppler shift according to time can be approximated by

$$f_{Y_n}(x) = \frac{1}{\sqrt{2\pi n\sigma}} \exp\left\{\frac{-(x - x_0 - n\mu)^2}{2n\sigma^2}\right\} \tag{D.11}$$

where $\mu = -29$ Hz/s, $\sigma^2 = 318$ and $x_0 = 4500$ Hz. The effects on D.11 as time progresses is illustrated in Figure D.2. As $\mu < 0$ the mean of the distribution is shifted in the negative direction for each time step. At the same time the variance is also increased which means that the distribution is spread out.

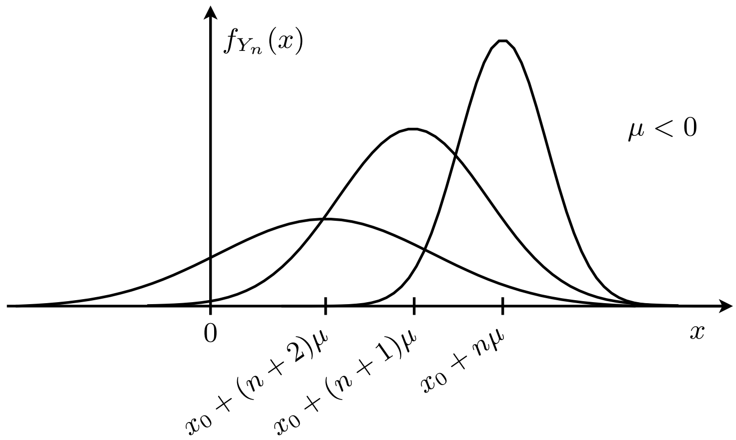


Figure D.2: Sketch of how the distribution of Y changes as time progresses

As time progresses the distribution will eventually be shifted and spread to an extent where some of the distribution streches beyond the search range of the acquisition process. In this case it cannot be guaranteed that 99% of the acquirable signals are found. Instead a one-sided confidence interval can be specified for the case. So lets find n in the case 99% of the acquirable signals are found with 95% confidence. The standard normal random variable is given by:

$$Z = \frac{Y_n - x_0 - n\mu}{\sqrt{n}\sigma} \tag{D.12}$$

Hence the probability that Y_n is larger than 0 and thus within the range of the acquisition process can be expresses as

$$\begin{aligned} P\{Y_n \geq 0\} &= P\left\{\frac{Y_n - x_0 - n\mu}{\sqrt{n}\sigma} \geq \frac{0 - x_0 - n\mu}{\sqrt{n}\sigma}\right\} && \text{(D.13)} \\ &= P\left\{Z \leq \frac{x_0 + n\mu}{\sqrt{n}\sigma}\right\} && \text{(D.14)} \end{aligned}$$

Using the `normpdf` in MATLAB it can be shown that at 95% confidence level $P\{Z \leq 1.6449\} = 0.95$. Hence the right term in D.14 has to be smaller than or equal to 1.6449.

$$\begin{aligned} \frac{x_0 + n\mu}{\sqrt{n}\sigma} &\leq 1.6449 && \text{(D.15)} \\ n &\leq 143 \text{ s} && \text{(D.16)} \end{aligned}$$

Hence, in order to acquire 99% of the satellites with 95% confidence the acquisition time must be smaller than or equal to 143 s or 2 min and 23 s.

Appendix E

Dynamic Behavior of Acquisition Results

This appendix adresses the issues related to how the frequency and phase of an acquired C/A code is influences by the dynamic nature of a system consisting of GPS satellites and a receiver in LEO. So far it has been found that due to the relative movement between the receiver and GPS satellites doppler shift is induced on the signal and that the doppler shift drift over time. This implies that the frequency of the acquire also drifts, possibly making it invalid over time. Similar the change in distance between GPS satellites and the receiver, must also change the phase of the code. The purpose is to evaluate the extent of these changes and, if possible, derive suitable predictors that could increase the time the acquired signals are valid.

E.1 Frequency Drifting

To evaluate the drifting effects on the acquisition results, lets consider the general case where a signal with doppler shift x_0 have been acquired. However, as the search range have been divided into smaller intervals of Δf, the signal is assigned to a specific frequency bin B_m instead of its precise value. For right-sided acquisition the frequency bins are given by:

$$B_m \in [(m-1)\Delta f, m\Delta f] \quad \text{where} \quad m = \{1, 2, .., f_{d,max}/\Delta f\} \tag{E.1}$$

The distribution of $B = B_1 \cup B_2 \cup ... \cup B_{f_{d,max}/\Delta f}$ is triangular as it constitutes the initial population, but as the range of each B_m is small compared with the entire search range, B_m can be considered appoximately uniform. Hence, given that m is provided by the acquisition process, it can be assumed that x_0 belongs to B_m and is equal likely to take any value in the interval $[(m-1)\Delta f, m\Delta f]$.

Given $x_0 \in B_m$ and m at one epoch, it is certain that as time progresses $x_0 \notin B_m$ due to doppler shift drifting. Futhermore, as the mean of the drifting is negative m will decrease, or in other words x_0 will shift to one of the adjacent $B_{m-1}, B_{m-2}, ...$ frequency bins. This is problematic as the tracking loops expect the signal to be in the vicinity of the frequency bin center. Hence, it would be interesting to know how long the time can be allowed to progress without invalidating the acquisition results. Secondly, it would be interesting to investigate whether it is possible to predict which of the frequency bins $B_{m-1}, B_{m-2}, ...$, x_0 belongs to given that the signal was allowed to drift.

To answer the first question lets consider the bin range given in Eqn E.1. As it is possible to acquire a signal with a doppler shift equal to the lower bound, by the next epoch the signal will not be valid anymore due to drifting. Hence, no matter how fast the bin is evaluated, some misses will occour. If some misses, say 10%, are allowed, the case is the same as in the previous section whereby the equations can be reused. Lets find x_0 in Eqn. D.14 according to $P\{\text{miss}\} = 0.1$ and where Δf have been chosen to 500 Hz:

$$
\begin{aligned}
x_0 &= P\{\text{miss}\}\Delta f && \text{(E.2)}\\
&= 50 \text{ Hz} && \text{(E.3)}
\end{aligned}
$$

Using Eqn. D.15 the 95% confidence interval can be found:

$$
\begin{aligned}
\frac{x_0 + n\mu}{\sqrt{n}\sigma} &\leq 1.6449 && \text{(E.4)}\\
n &\leq 0.812 \text{ s} && \text{(E.5)}
\end{aligned}
$$

Hence, if 10% of the signals are allowed to drift outside the search bin the evaluation time must be less that 821 ms.

To answer the second question the signal is allowed to drift into the adjacent frequency bins as shown in Fig. E.1. As will be shown, the consequence of this is that the bounds will be shifted and spread out in frequency. This imposes new demands on the frequency tracking loops in terms of extra bandwidth. But if the time that the acquisition results are valid is improved, it would be worth considering.

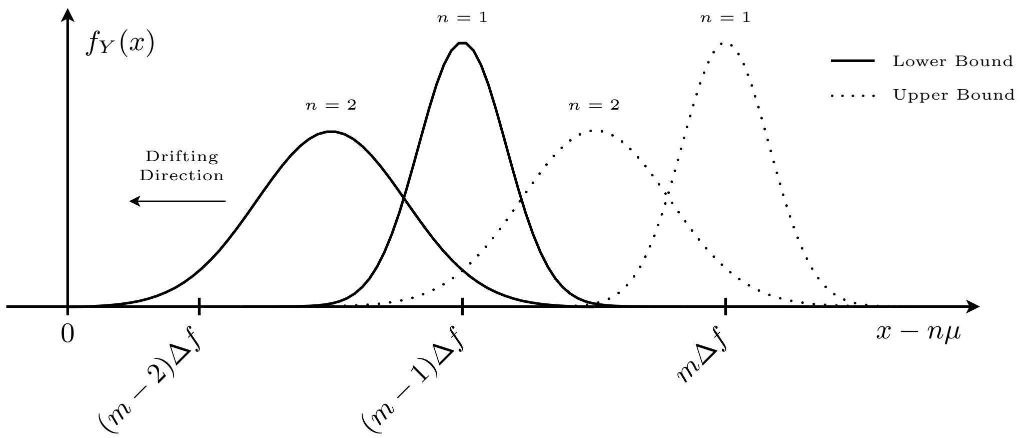


Figure E.1: Illustration of how the bounderies of the frequency bins drift according to time.

Given bin m then we know that the initial value x_0 is bounded by $[(m-1)\Delta f, m\Delta f]$. To account for the inevitable bins shifts as time progresses Eqn. E.1 is recast so B_{m-l}, where l is the number of shifted bins. Consequently, the lower and upper bound for the lth adjacent bin are given by:

$$
\begin{aligned}
B_{m-l} &\in [a_{m-l}, b_{m-l}] \quad \text{where} && \text{(E.6)}\\
a_{m-l} &= (m-l-1)\Delta f && \text{(E.7)}\\
b_{m-l} &= (m-l)\Delta f && \text{(E.8)}
\end{aligned}
$$

As $x_0 \in [a_{m-0}, b_{m-0}]$ it would be tempting to assume that the initial value just would shift l bins as time progresses, i.e. $x_l \in B_{m-l}$. The problem with this assumption is that the signal is not only shifted, but also *spead out* over time. This means, that as time progresses, the initial value would belong to a set of possible bins $B_{m-1} \cup B_{m-2} \cup ...$ and not just one. Consequently, the problem is to find the range of possible bins as time progresses. In a more formal manner this can be described as finding l_{min} and l_{max} in:

$$x_0(n) \in \bigcup_{l=l_{max}}^{l_{min}} B_{m-l} \tag{E.9}$$

where l_{min} and l_{max} are changing over time. To solve this problem lets start by considering the drift of the non-shifted lower bound a_{m-0}. In the previous section is it was found that drift with regard to time was given by:

$$Y_n \sim \mathcal{N}(x_0 + n\mu, n\sigma^2) \tag{E.10}$$

where x_0 is the doppler shift initial value, n is the time in seconds, μ the mean drift in Hertz per second and σ^2 the drift variance. Hence, the drift of the lower bound is a normal r.v. given by

$$Y_{n,m,0} \sim \mathcal{N}\left(a_{m-0} + n\mu, n\sigma^2\right) \tag{E.11}$$

Likewise, the standard normal variable for the lower bound is:

$$Z = \frac{Y_{n,m,0} - a_{m-0} - n\mu}{\sqrt{n}\sigma} \tag{E.12}$$

Hence, the probability that the lower bound a_{m-0} is larger than or equal to the lower bound of a shifted bin a_{m-l}, as time progresses, is given by:

$$\begin{aligned}
P\{Y_{n,m,0} \geq a_{m-l})\} &= P\left\{Z \geq \frac{a_{m-l} - a_{m-0} - n\mu}{\sqrt{n}\sigma}\right\} && \text{(E.13)}\\
&= P\left\{Z \geq \frac{(m-l-1)\Delta f - (m-1)\Delta f - n\mu}{\sqrt{n}\sigma}\right\} && \text{(E.14)}\\
&= P\left\{Z \geq \frac{-l\Delta f - n\mu}{\sqrt{n}\sigma}\right\} && \text{(E.15)}\\
&= P\left\{Z \leq \frac{l\Delta f + n\mu}{\sqrt{n}\sigma}\right\} && \text{(E.16)}
\end{aligned}$$

Consequently, the function that expresses that the signal is confined by the lower bound of the lth adjacent bin with 95% one-sided confidence, is given by:

$$\frac{l\Delta f + n\mu}{\sqrt{n}\sigma} \leq 1.6449 \tag{E.17}$$

Using the same procedure the drift of the upper bound can be found. Like the lower bound, the upper bound drift is given by the normal r.v. $Y_{n,m,0} \sim \mathcal{N}\left(b_{m-0} + n\mu, n\sigma^2\right)$, and the standard normal variable is:

$$Z = \frac{Y_{n,m,0} - b_{m-0} - n\mu}{\sqrt{n}\sigma} \tag{E.18}$$

Hence, the probability that the upper bound b_{m-0} is less than a new shifted bound b_{m-l} is given by

$$
\begin{aligned}
P\{Y_{n,m,0} \leq b_{m-l}\} &= P\left\{Z \leq \frac{b_{m-l} - b_{m-0} - n\mu}{\sqrt{n}\sigma}\right\} && \text{(E.19)}\\
&= P\left\{Z \leq \frac{(m-l)\Delta f - m\Delta f - n\mu}{\sqrt{n}\sigma}\right\} && \text{(E.20)}\\
&= P\left\{Z \leq \frac{-l\Delta f - n\mu}{\sqrt{n}\sigma}\right\} && \text{(E.21)}
\end{aligned}
$$

Again, the 95% confidence interval is given by:

$$\frac{-l\Delta f - n\mu}{\sqrt{n}\sigma} \leq 1.6449 \qquad \text{(E.22)}$$

As the bounds of the set of bins that the initial signal x_0 covers after n seconds is confined by Eqn. E.22 and E.17, the bins shift value l can be assinged to l_{max} and l_{min} for the lower and upper bound, respectively. Inserting l_{max} and l_{min}, and rearranging, yields:

$$
\begin{aligned}
l_{max} &\leq \frac{\sqrt{n}\sigma 1.6449 - n\mu}{\Delta f} && \text{(E.23)}\\
l_{min} &\geq \frac{-\sqrt{n}\sigma 1.6449 - n\mu}{\Delta f} && \text{(E.24)}
\end{aligned}
$$

This shows that the limits in Eqn. E.9, and hence the number of bins that the signal covers, are dependent on time. This means that, given $x_0 \in B_m$, the bound of B_m will change over time. By inserting the value of l_{max} and l_{min} in Eqn. E.6, the bins size now dependents of time n instead of the bin shifts l. The new result for the time dependant bin size bounderies is given by:

$$
\begin{aligned}
B_m(n) &\in [a_m(n), b_m(n)] \quad \text{where} && \text{(E.25)}\\
a_m(n) &= \left(m - \frac{\sqrt{n}\sigma 1.6449 - n\mu}{\Delta f} - 1\right)\Delta f && \text{(E.26)}\\
&= (m-1)\Delta f - \sqrt{n}\sigma 1.6449 + n\mu && \text{(E.27)}\\
b_m(n) &= \left(m - \frac{-\sqrt{n}\sigma 1.6449 - n\mu}{\Delta f}\right)\Delta f && \text{(E.28)}\\
&= m\Delta f + \sqrt{n}\sigma 1.6449 + n\mu && \text{(E.29)}
\end{aligned}
$$

Hence, given m it is now possible to predict the interval of x_0 as time progresses by using the knowledge that $x_0(n) \in B_m(n)$ in Eqn. E.25. The result of Eqn. E.25 is confirmed by the fact that as n goes to zero, $B_m(n)$ is reduced to the defaul bin size given in Eqn. E.1, which was the outset of this investigation.

From Eqn. E.25 it is also possible to evaluate the integer number of bins that the signal covers over time. This is given by the difference between the maximum and minimum bounds of the signal divided by the bin size. This result is ceiled to give the integer number of bins.

$$
\begin{aligned}
\#\text{Bins}(n) &= \left\lceil \frac{b_m(n) - a_m(n)}{\Delta f} \right\rceil && \text{(E.30)}\\
&= \left\lceil \frac{2\sqrt{n}\sigma 1.6449}{\Delta f} + 1 \right\rceil && \text{(E.31)}
\end{aligned}
$$

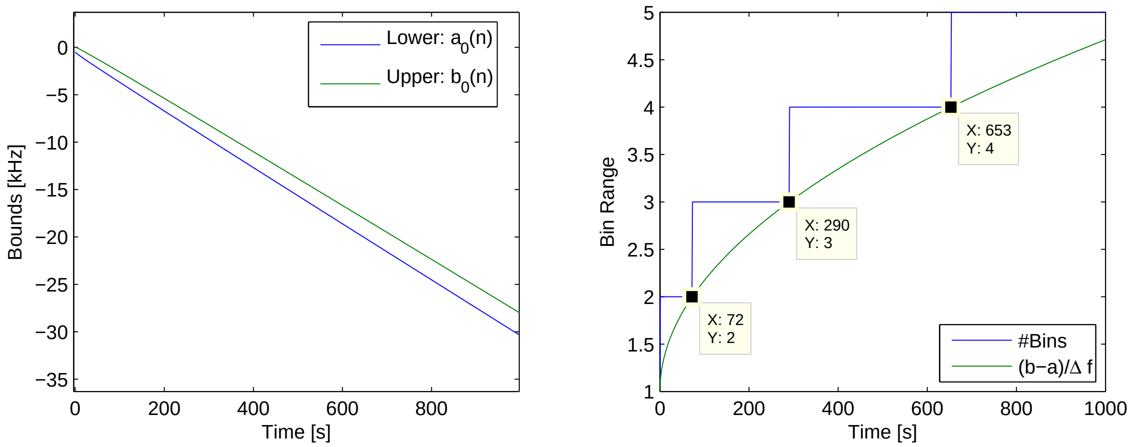


Figure E.2: Left: The upper and lower bounderies as time progresses. Right: Real and integer number of bins that the signal covers over time. All results are relative to $m = 0$ and $\Delta f = 500$ Hz

As can be seen from Figure E.2 the second n is larger than zero, the initial signal x_0 will reside in two bins. This comes from the assumption that the x_0 is uniformly distributed, whereby the lower bound of the initial bin is a possible value of x_0. Given that the signal drifts, the next epoch x_0 will most likely reside in the adjacent bin, although by a small margin. Figure E.2 also shows that if the initial signal is allowed to drift and reside in a two bin inteval, the signal bounds can be predicted until the 72th second from the time of the initial acquisition. Similar three bin interval would allow for 290 s prediction time.

E.2 Code Drifting

Until now the dynamic effects related to low earth orbit have only been considered with respect to frequency acquisition. Hence, the question remains as to how the code phase is affected. First of all it is clear that the code phase must be affected. This comes from the fact that the phase is used to estimate the ranges (pseudoranges) to the GPS satellites and consequently the receiver position. When in orbit the distance to the GPS satellites constantly change which, apart from inducing doppler shift, also must change the code phase at the receiver due to the change in position.

The easist way to evaluate the extent of the phase change is to consider a chip period in terms of the number of L_1 wavelength N with and without doppler. Over a duration of one second the total number of L_1 wavelengths is given by:

$$\begin{aligned} N &= f_{L1} \quad \text{[wavelengths/s]} && \text{(E.32)} \\ N' &= f_{L1} + f_d \quad \text{[wavelengths/s]} && \text{(E.33)} \\ \Delta N &= N' - N = f_d \quad \text{[wavelengths/s]} && \text{(E.34)} \end{aligned}$$

where N and N' are with and with doppler, respectively, and ΔN the difference in wavelengths between the two cases. The last term can be interpredet as the surplus of wavelengths generated by the doppler shift or to be a bit more precise the relative motion between the GPS satellites and receiver. Given the number of L_1 wavelengths per chip it is possible to evaluate the number of extra chips that is induced per second due to doppler:

$$\Delta_{chip} = \frac{f_d}{N_{chip}} \quad \text{[chips/s]} \tag{E.35}$$

Where N_{chip} is the number of L_1 wavelengths over one chip period. Futhermore, by inversion we find the time it takes to drift one chip at a given doppler frequency.

$$\tau_{chip} = \frac{N_{chip}}{f_d} \quad \text{[s/chip]} \tag{E.36}$$

In order to optain the number of chips that a code, with initial phase of p_0, have drifted t seconds after the initial acquisition, Eqn. E.35 is integrated over the time period, yeilding:

$$p(t) = p_0 + \frac{f_d}{N_{chip}} \int_0^t dn \tag{E.37}$$

However, as was shown in the previous section the doppler shift changes over time, which means that the doppler shift have to be included in the integration in Eqn. E.37. Also, in Eqn. E.25 on page 114 it was shown that doppler shift resides between two bounds, namely $a(n)$ and $b(n)$. By inserting the bounds in Eqn. E.37 and solving the integration, a phase interval with respect to time can be found, i.e. the interval where the phase can be expected after t seconds.

$$\begin{aligned} p_{min}(t) &= p_0 + \int_0^t \frac{a(n)}{N_{chip}} dn \\ &= p_0 + \frac{1}{N_{chip}} \left\{ \frac{\mu\, t^2}{2} - \frac{2\sigma 1.6449\, t^{3/2}}{3} + \Delta f\,(m-1)\, t \right\} \end{aligned} \tag{E.38}$$

$$\begin{aligned} p_{max}(t) &= p_0 + \int_0^t \frac{b(n)}{N_{chip}} dn \\ &= p_0 + \frac{1}{N_{chip}} \left\{ \frac{\mu\, t^2}{2} + \frac{2\sigma 1.6449\, t^{3/2}}{3} + \Delta f\, m\, t \right\} \end{aligned} \tag{E.39}$$

Using the p_{min} and p_{max} the code phase interval can be found, just as it was in the frequency case.

$$\text{\#Phases(t)} = \lceil p_{max}(t) - p_{min}(t) \rceil \tag{E.40}$$

$$= \left\lceil \frac{1}{N_{chip}} \left\{ \frac{4\sigma 1.6449\, t^{3/2}}{3} + \Delta f\, t \right\} \right\rceil \tag{E.41}$$

$$= \left\lceil \frac{2.193\, \sigma\, t^{3/2} + \Delta f\, t}{N_{chip}} \right\rceil \tag{E.42}$$

The behaviour of p_{min}, p_{max} and #Phases(t) is shown in Figure E.3. Most notisable is the rapid increase of the interval size of possible code phase. This means that the result of the prediction given in E.38 and E.39 quickly deteriorates. The practical implecations is that the tracking loop bandwidth (w.r.t code phase) has to be increased or that the evaluation time of the acquisition result has to be is minimized.

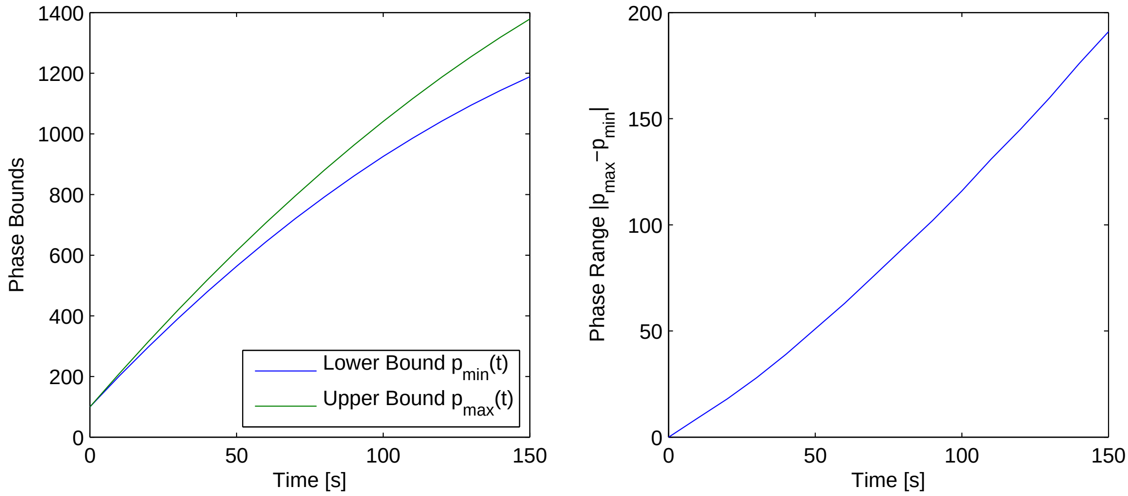


Figure E.3: Phase drift for at code with $p_0 = 100$ and a mean doppler drift of $\mu = -29$ Hz/s and variance of $\sigma^2 = 318$. The bin size is $\Delta f = 500$ Hz and $m = 17$, which is equal to a code doppler frequency of 85000 Hz. The right figure shows the interval size of possible code phases as time progresses.

As a closing remark a small example is given to further illustrate the severity of the code drift. In most situations phase acquisition is done with a resolution of half a chip period. The number of L_1 wavelengths over one half chip period is given by:

$$N_{chip} = 0.5 \cdot f_{L1} T_c = 0.5 \cdot \underbrace{154 \cdot f_0}_{f_{L1}} \cdot \underbrace{\frac{10}{f_0}}_{T_c} \tag{E.43}$$

$$= 0.5 \cdot 154 \cdot 10 = 770 \quad \text{[wavelengths/half-chip]} \tag{E.44}$$

Inserting this result into Eqn. E.36 together with the worst case doppler shift of 45 kHz, the drift time of one half chip can be found:

$$\tau_{half-chip} = \frac{770}{45000} = 17.1 \quad \text{[ms/half-chip]} \tag{E.45}$$

Hence, in the worst case scenario the code phase would have drifted half a chip in 17.1 ms.

E.3 Conclusion

In the previous two sections the dynamic behaviour of the acquisition results have been analysed with respect to frequency and phase. The dynamic behaviour is caused by the doppler shift and the doppler shift drift, which changes the frequency and code phase over time. As a consequence once a signal have been acquired the result “drifts“ as time pases and makes the result invalid. As most tracking loops are narrowband both in phase and frequency, the acquisition result would have drifted too far for the tracking loop to lock onto it, unless the result is passed to the loop immitiately after it is found. It was found that if the tracking loop frequency bandwidth is 500 Hz, then on average 10% percent of the acquisition results are invalid if 821 ms passes from acquisition to tracking loop initiation. Similar, if the correlator is 3 half chips wide[1], then in the worst case the signal would have drifted one chip in $2 \cdot 17.1$ ms, invalidating the result.

In order mitigate the effects of drifting, frequency and phase predictors have been derived based on the statistics. The result is two sets of predictors, given in Eqn. E.46-E.49, that estimate the upper and lower bounds at a 5% significans level of the frequency and phase according to time.

$$f_{min}(t) = t\mu - \sqrt{t}\sigma 1.6449 + (m-1)\Delta f \tag{E.46}$$

$$f_{max}(t) = t\mu + \sqrt{t}\sigma 1.6449 + m\Delta f \tag{E.47}$$

$$p_{min}(t) = p_0 + \frac{1}{N_{chip}}\left\{\frac{\mu\, t^2}{2} - \frac{2\sigma 1.6449\, t^{3/2}}{3} + \Delta f\,(m-1)\,t\right\} \tag{E.48}$$

$$p_{max}(t) = p_0 + \frac{1}{N_{chip}}\left\{\frac{\mu\, t^2}{2} + \frac{2\sigma 1.6449\, t^{3/2}}{3} + \Delta f\, m\, t\right\} \tag{E.49}$$

When a signal is acquired it is labelled with a frequency bin number m and a phase value p_0. Using the estimators it is possible to estimate a region in the code frequency-phase search space where an acquired signal is after t seconds, based on the initial values and the statistics of the doppler shift drift (μ and σ^2). The concept is illustrated in Figure E.4. As can be seen the region consists of the rectangular area spaned by the two coordinates $[p_{min}(t), f_{min}(t)]$ and $[p_{max}(t), f_{max}(t)]$

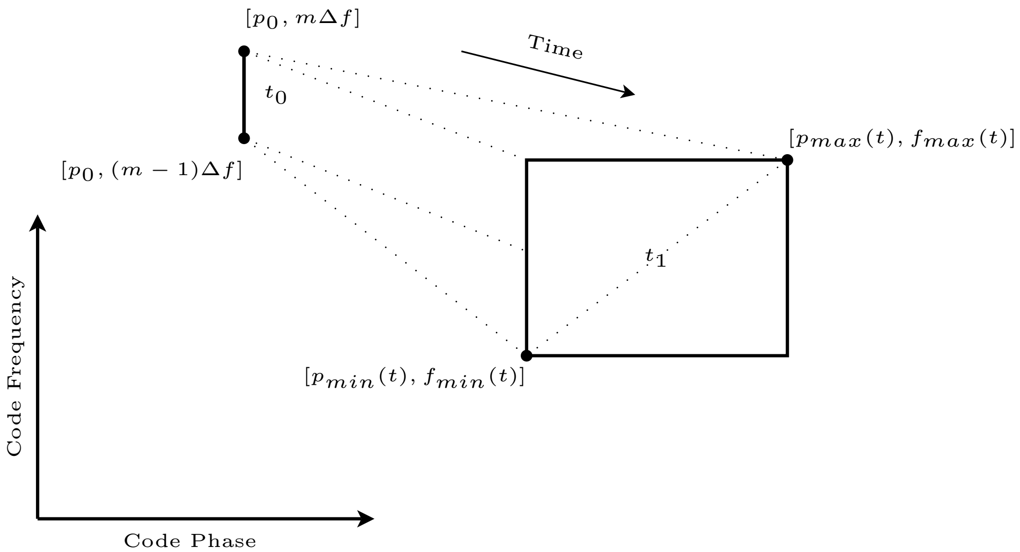


Figure E.4: Illustration of the acquisition result and the predicted area of the result t seconds after the initial acquisition.

In order to illustrate the dynamic nature of the system, the predicted path of codes belonging to different (initial) frequency bins have been plottet in Figure E.5. The path is illustrated for time steps of 10 s, and the lines between the upper (green) and lower (blue) bounds corresponds to the diagonal of the retangle illustrated in Figure E.4. The depicted time period is 150 s, which is approximately the time available if 99% of the GPS signals are to be found at a 5% significance

[1]Most tracking loops use a phase tracking method knwon as early-prompt-late, where the signal is correlated at three lags places 1 half chip apart and where the feedback is controlled by the outcome of these correlations.

level in LEO. As it is clear both phase and frequency drift considerably over this period of time. And what is worse the region containing the valid signal grows quickly, making the prediction unsuitable for narrowband tracking loops.

To give an example, lets consider a tracking loop with bandwidth of 1 kHz and correlators 5 half chips wide (allowing a maximum slip of three half chips), and under the same condition as in Figure E.5. Then according to Eqn.E.31 the range of the frequency prediction is not larger than 1 kHz until after the 72th second. On the other hand the phase range predictor given in Eqn. E.42 show that the phase range becomes larger than 3 half chips after just 4 s. Hence, prediction of the phase is the limiting factor as the frequency prediction allows for much longer prediction time without the need to change the bandwidth of the tracking loop. On contrary the width of the tracking loops correlators has to be increased even with minor increases in prediction time.

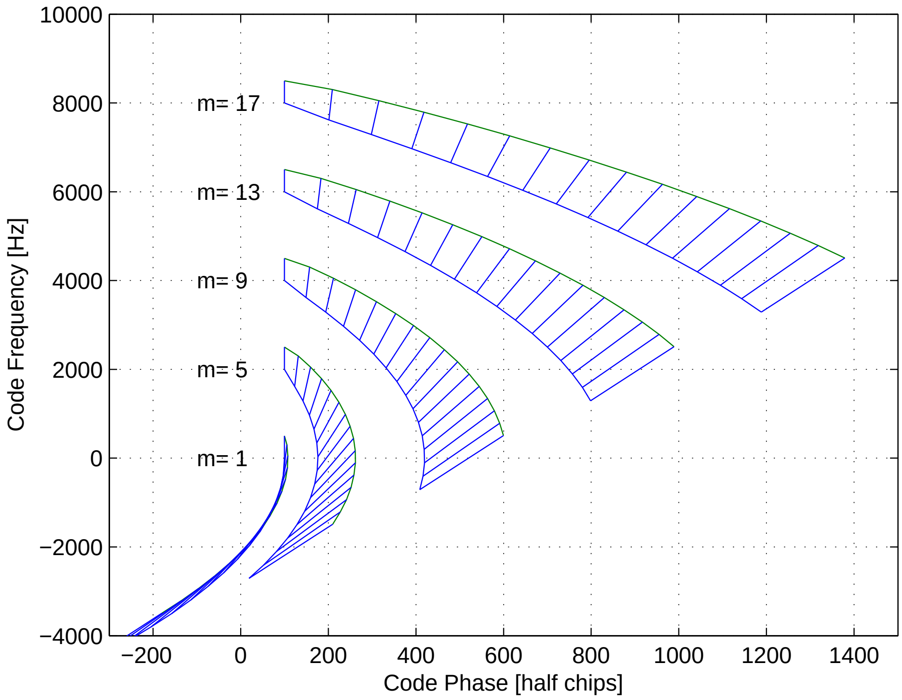


Figure E.5: Frequency and phase prediction in 10 s steps for signals with an initial code phase of $p_0 = 100$, but with different initial frequency values. The prediction is based on a mean doppler shift drift of $\mu = -29$ Hz/s and variance of $\sigma^2 = 318$, and a bin size of $\Delta f = 500$ Hz

Appendix F
Antenna Noise

This appendix contains an in-depth analysis of the noise induced in the antenna due to environmental factors such as the physical temperature of the antenna and noise from the sun. One might question the necessity of such an analysis as many references already exists to antenna noise in literature. However, all the references to antenna noise found by this author, are for terrestrial applications and not space. Hence, to complete the analysis of GPS in space, antenna noise must be considered in the work environment of the antenna.

Every object with a physical temperature above absolute zero radiates energy [31]. In electronics this energy is being considered thermal noise and is given by the relation

$$P = kT\Delta f \tag{F.1}$$

where P is the power of the noise, k Boltzmann's constant, T the temperature in Kelvin and Δf the bandwidth of the signal of interest. As with any other antenna, the GPS antenna is also prone to this noise and it has to accounted for when designing a receiver. The subject is even more pressing as the GPS antenna have to work in a hostile environment with high temperatures. Hence the scope of this section is to find an estimate of the antenna temperature and consequently the amount of noise power it induces. The main source is the Antenna Theory book by Balanis [31].

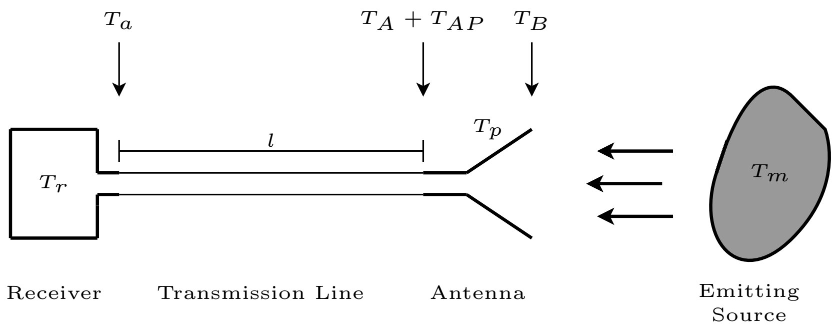


Figure F.1: Figure is inspired by [31]

Figure F.1 illustrates the scenario used to calculate the system noise power. The emitting source is the object at which the antenna points. In terrestrial applications this is normally the sky or ground, but on a satellite the antenna is more likely pointing towards the sun, moon or into deep space. As these source also has an temperature, commonly known as brightness temperature $T_B(\theta, \phi)$, above zero they will radiate energy towards the antenna. The brightness temperature emitted by different sources is intercepted by the antenna and appears on the terminal as an antenna temperature T_A. The relation between the brightness temperature and antenna temperature is given by

$$T_A = \frac{\int_0^{2\pi}\int_0^{\pi} T_B(\theta,\phi)G(\theta,\phi)\sin(\theta)\,d\theta\,d\phi}{\int_0^{2\pi}\int_0^{\pi} G(\theta,\phi)\sin(\theta)\,d\theta\,d\phi} \tag{F.2}$$

where $G(\theta,\phi)$ is the gain pattern of the antenna. The value of the antenna temperature in different circumstances will be discussed in detail shortly. The physical temperature of the antenna T_p also contributes to the total antenna temperature. The, aperture temperature T_{AP} is given by

$$T_{AP} = \left(\frac{1}{e_A} - 1\right) T_p \tag{F.3}$$

where e_A is the thermal efficiency of the antenna. The thermal efficiency of the antenna is the ratio between energy input and output. That is, if any of the incoming signal is dissipated in the antenna as heat the efficiency will drop.

The total antenna temperature T_a at the receiver is given by the following summation

$$T_a = T_A\,e^{-2\alpha l} + T_{AP}\,e^{-2\alpha l} + T_0(1 - e^{-2\alpha l}) \tag{F.4}$$

where T_0 is the physical transmission line temperature, α is transmission line attenuation coefficient and l the length of the transmission line. In the following each of the terms in Eqn. F.4 will be evaluated.

Transmission line temperature

The attenuation imposed by the transmission line in Eqn. F.4, as well as the rest of the terms, is a function of the attenuation factor and the cable length to the receiver. The attenuation factor depends on many factors [42] and to avoid any larger discussion on transmission line theory the attenuation factor different coaxial cables at 1500 MHz are listed in Table F.1 for cables made by RFS[1].

Type	Dia. [mm]	Attenuation [dB/100 m]	Attenuation α [Np/m]
RF174	2.7	114	131 E - 3
RG58	5	80.4	92 E - 3
RG223	5.4	61.9	71 E - 3

Table F.1: Attenuation factor for different RF cable types at 1500 MHz.

On a cubesat 5 mm cables would most likely be opted out due to their inflexibility and bulky nature, unless there is a very good reason to use them. As such the RF174 cable will be used in further calculations.

Next, the physical temperature of the transmission line has to be found. As the temperature is different from end to end of the line it must be assumed that there exist a temperature gradient along the line. The maximum gradient is given when the satellite is exposed to the sun where the surface temperature T_p is around 110 °C and the receiver T_r is 30 °C. The minimum is when the antenna is at -40 °C and the the receiver 0 °C. In order to account for the gradient we assume that the transmission line temperature T_0 can be approximated with the mean temperature of T_r and

[1] `http://www2.rfsworld.com/RFS_Edition4/pdfs/RGFLEX_Braided_Cable_47-54.pdf`

T_p. It is hard to say if this is a fair assumption. But considering the alternative where T_0 would be set to either T_r or T_p, which most definitely would be biased, the mean of the two seems like a fair compromise.

Hence the transmission line temperature at the two extremes is

$$T_{0,max} = 273.15\text{ K} + \frac{110°\text{C} + 30°\text{C}}{2} = 343.15\text{ K}$$

$$T_{0,min} = 273.15\text{ K} + \frac{-40°\text{C} + 0°\text{C}}{2} = 253.15\text{ K}$$

Using the attenuation factor for the RF174 cable it is now possible to find the equivalent transmission line temperature in Eqn. F.4. This have been illustrated in Figure F.2 for different line lengths.

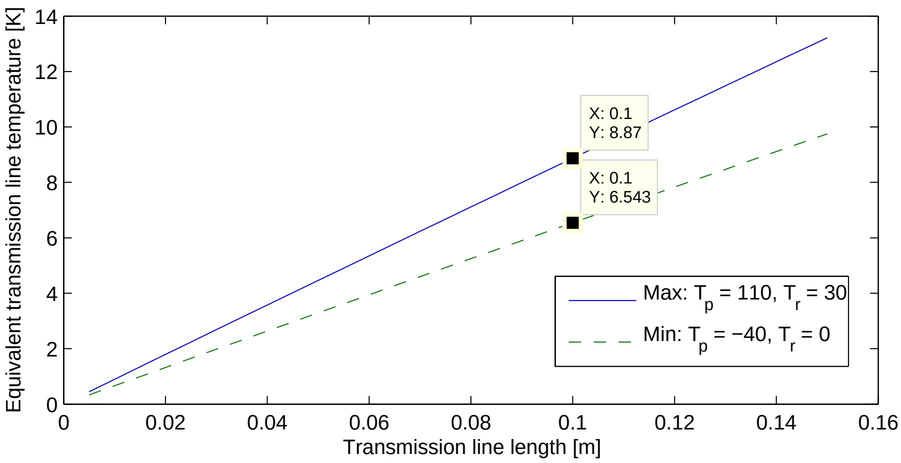


Figure F.2: Equivalent transmission line temperature for a RF174 coaxial cable at 1500 MHz with respect to length

Assuming a worst case scenario where the transmission line would stretch from one side of the satellite to the other the line length would be 10 cm. As illustrated in Figure F.2 the temperature contribution from the transmission line at 10 cm would be 8.9 K for a warm satellite and 6.5 K for a cold

Brightness temperature

It is expected that the GPS antenna on the satellite points in the direction of the GPS satellites at all time, i.e out into space. This means that there exists four emitting sources that can affect the antenna temperature. That is the sun, moon, cosmic and galactic noise. Figure F.3 shows the brightness temperature for these sources as recommended by the Internation Telecommincations Union.

It is evident that the sun by far is the biggest emitting source at around 150.000 K at the carrier frequency of $L_1 = 1,573$ GHz. This is followed by the moon at 300 K and background noise (cosmic and galactic noise) at 3 K. In order to evaluate the temperature contribution from each the sources it is necessary to integrate according to Eqn. F.2. The calculation can be simplified by using solid angles Ω instead of the azimuth and elevation angles, θ and ϕ. Using the relation $d\Omega = \sin\theta \, d\theta \, d\phi$ and assuming that the receiving antenna is "half isotropic" Eqn. F.2 becomes

$$T_A = \frac{1}{2\pi} \oiint_{2\pi} T_B(\Omega) d\Omega$$

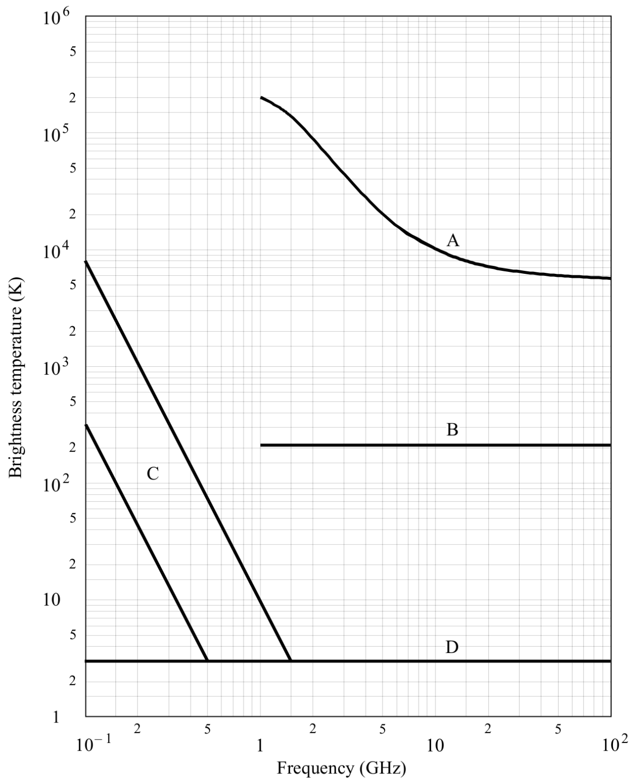


Figure F.3: Brightness temperature of extra-terrestial noise sources as recommended in ITU-R P.372-9 [43] on radio noise. A) Quiet sun B) Moon C) Range of galatic noise D) Cosmic background.

The term half isotropic is used to describe an antenna that have 0 dB gain in one half of the elevation plane and no gain in the other. This is used as an approximation to the patch antennas described earlier.

Next, the solid angle to the sun and moon according to the receiver has to be calculated. Figure F.4 shows the relation between the satellite and the blackbody noise sources (i.e. sun, moon and cosmic background). The distance from the satellite to the moon and sun is approximately given by $r_{sun} = 150\text{E}6$ km and $r_{moon} = 350\text{E}3$ km. Compared to these distances the altitude of the satellite is so small that it can be neglected.

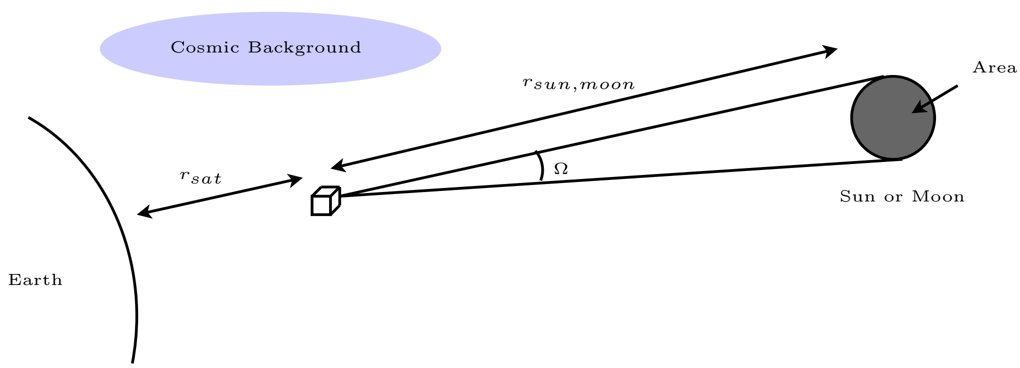


Figure F.4: Extra-terrestrial blackbody noise sources in relation to a satellite orbiting the earth.

The area of the sun and moon can be found to $A_{sun} = 1.52\text{E}\,12$ m^2 and $A_{moon} = 9.47\text{E}\,6$ m^2. Combined with the distances the solid angles can be found

$$\begin{aligned}\Omega_{sun} &= \frac{A_{sun}}{r_{sun}^2} = 67.6\text{E-}6 \text{ sr} \\ \Omega_{moon} &= \frac{A_{moon}}{r_{moon}^2} = 77.4\text{E-}6 \text{ sr}\end{aligned}$$

It is now possible to calculate the antenna temperature due to the different sources by integrating over the solid angles. Lets consider three cases: 1) Sun and background $T_{A,Sun}$ 2) Moon and background $T_{A,Moon}$ and 3) Background only $T_{A,Cosmic}$

$$\begin{aligned}T_{A,Sun} &= \frac{1}{2\pi}\left[\oiint_{\Omega_{sun}} T_{m,sun}\, d\Omega + \oiint_{2\pi-\Omega_{sun}} T_{m,cosmic}\, d\Omega\right] \\ &= \frac{1}{2\pi}\left[67.6\text{E-}6 \text{ sr} \cdot 150.000 \text{ K} + (2\pi - 67.6\text{E-}6 \text{ sr}) \cdot 3 \text{ K}\right] \\ &= 4.6 \text{ K} \\ T_{A,Moon} &= 3 \text{ K} \\ T_{A,Cosmic} &= 3 \text{ K}\end{aligned}$$

Aperature temperature

As will be shown the equivalent noise temperature T_{AP} (which we denote aperture temperature) generated due to the physical temperature of the antenna is an important factor when calculating the total antenna temperature. According to Balanis [31] the aperture temperature is given by

$$T_{AP} = \left(\frac{1}{e_A} - 1\right) T_p \tag{F.5}$$

where $0 < e_A < 1$ is the thermal efficiency of the antenna. Thermal efficiency is a term taken from thermo dynamics where it describes the efficiency of heat engines [22, p. 670]. In a heat engine any input energy that is not converted to work is considered loss and degrades the efficiency. The same holds for antennas; if any of the received energy is lost the efficiency drops. However, Balanis fails to describe the connection between thermal efficiency and other antenna parameters. A search through literature have yielded that the inverse of the thermal efficiency ($1/e_A$) is also known as the loss factor L of the antenna [44, p. 32]. The antenna loss factor is given by [45]

$$L = \frac{R_L + R_r}{R_r} = \frac{1}{e_{cd}}$$

where R_L is the ohmic loss resistance and R_r the radiation resistance of the antenna. This can be found to be the inverse of e_{cd} which is the radiation efficiency of the antenna [31]. Hence the term thermal efficiency used by Balanis is the same as the radiation efficiency of the antenna. This also makes sense as the loss resistance accounts for the fraction of the received energy that is dissipated as heat in the antenna, instead being converted to energy at output terminals.

Now that this have been sorted it should be possible to find the aperture temperature. However the radiation efficiency is very seldom specified for antennas, which is also why it is not stated in Tabel 6.1. Carver [46] gives a general estimate of 95-99% radiation efficiency for common patch antennas. But more precisely it has been shown that the efficiency of an patch antenna is a function of the electric length (the ratio between substrate height and wavelength of signal h/λ) and permitivity of the substrate [47] [31]. For $h \leqslant 0.05\lambda_0$ the efficiency of a patch antenna is given by the following relation [48, p. 416]

$$\eta = \frac{1}{1+q} \tag{F.6}$$

where

$$q = \pi^2 \frac{(\varepsilon_r - 1)^3 \frac{h}{\lambda}}{\frac{2}{3}\varepsilon_r^2(\varepsilon_r - 1) + \frac{4}{15}\varepsilon_r} \tag{F.7}$$

The efficiency for a patch antenna at different relative permitivitties ε_r is plotted in Figure F.5 using Eqn. F.6 and F.6.

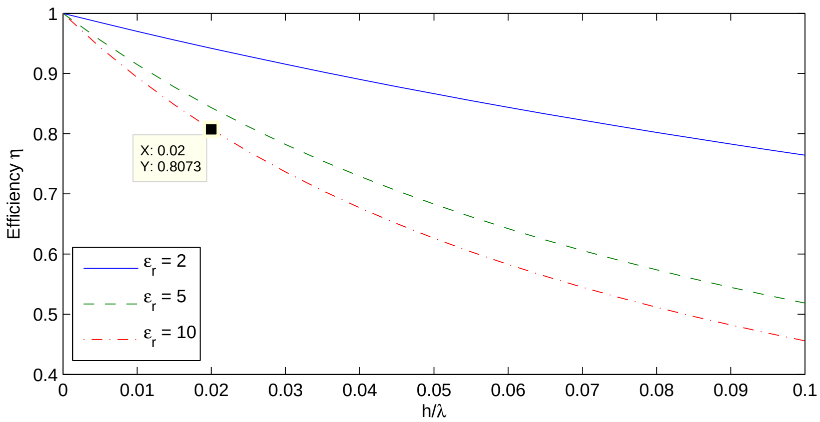


Figure F.5: Patch antenna radiation efficiency according to electric length of the substrate and relative permitivitty.

The substrate hight for all of Taoglas' patch antennas are 4 mm. With a L_1 wavelength λ_{L1} of 19 cm, the electric length is 0.021. Hence the radiation efficiency can be found from Figure F.5 given the relative permitivitty ε_r of the substrate. Unfortunately this is a parameter that most producers keep to them selves including Taoglas. However, in work done by Alboni and Cerretelli [49] on GPS patch antennas the permitivitty of the substrate used is 4.3 and other work [50] shows substrate permitivitty as high as 10. As such worst-case relative permitivitty of 10 is assumed. According to Figure F.5 this gives an efficiency of 80%. Hence the aperture temperature when the satellite surface temperature is at it maximum of 110°C is

$$T_{AP,MAX} = \left(\frac{1}{0.80} - 1\right)(273.15 + 110) = 95.8 \text{ K}$$

and the minimum

$$T_{AP,MIN} = \left(\frac{1}{0.80} - 1\right)(273.15 - 40) = 58.3 \text{ K}$$

An important thing to notice is that the aperture temperature increases dramatically if the efficiency decreases. E.g. if the efficiency drops to 70% the maximum temperature would suddenly be 164 K. Also compared to the brightness and transmission line temperatures the contribution made by the aperture is very large. This means that an important factor when choosing an antenna for applications where a good SNR is needed, is that the antenna has a good radiation efficiency.

Summery

In the previous the different sources that contribute to the total antenna temperature have been scrutinized. It have been shown that temperature contribution of the sun is very limited although it have a brightness temperature of 150.000 K. The transmission line contribution was larger then that of the sun but still relatively small compared to the contribution of the antenna itself, which is a direct consequence of the "inefficiency" of the antenna.

Temperature	Best Case [No sun]	Worst Case [Sun]
Brightness	3	4.6
Transmission Line (RF174)	6.5	8.9
Aperature ($e_{cd} = 0.8$)	58.3	95.8
Total	67.8	109.5
Noise (BW = 2.046 MHz)	-147.8 dBW	-145.1 dBW

Table F.2:

Table F.2 shows the combined antenna temperature of a patch antenna with a 80% efficiency. As it is evident the total noise temperature is heavily affected on the temperature of the satellite. As a result when the satellite is exposed to the sun the equivalent noise temperature is 61% higher then when in the earth shadow. But with this said the worst-case noise temperature is still better then 130 K which is a commonly used antenna temperature for terrestrial GPS applications [14] [24]. Others are even more conservative with antenna temperatures of 290 K [8] and 500 K [12].

As a final remark it should be noted that the figures presented are very prone to changes in radiation efficiency of the antenna. So to get as good a SNR as possible antennas with a good efficiency should be used.