Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing

Multi-System GNSS Receiver Software

Semester Project

Gianna Paulin

pauling@ethz.ch

Distributed Computing Group
Computer Engineering and Networks Laboratory
ETH Zurich

Supervisors:
Manuel Eichelberger, Dr. Pascal Bissig
Prof. Dr. Roger Wattenhofer

June 26, 2017

Acknowledgements

First of all I would like to thank my supervisor Manuel Eichelberger for his advice and support. I was very glad to be able to discuss the problems I encountered with him as he always gave me very good inputs and ideas. Additionally, I would like to thank my supervisor Dr. Pascal Bissig for his support and wish him all the best to the completion of his PhD.

Abstract

In this project, a novel GPS receiver is extended with processing capabilities for another GNSS, namely Galileo. This improves SNR and therefore the accuracy of the positioning. This report presents the Galileo signal modulation schemes and other concepts important for the implementation. Results show that the implementation is able to decode the Galileo signals correctly.

Contents

Chapter 1

Introduction

Low-power devices for Internet of Things (IoT) are limited in energy and computation power but should still be able to provide various communication methods and accurate measurements, such as determining the precise position of a device. The Distributed Computing Group has developed a low-power algorithm for a Global Positioning System (GPS) receiver using an approach very different from common receivers. In their algorithm the actual receiver only records the GPS signal and sends the recorded samples to a server, which then does the processing to get the final position. Additionally, the server does not extract the required navigation data from the received signal but instead downloads the data from the global data center. As a result the time a receiver has to be up and running is reduced and therefore enables a more efficient usage of the available energy.

GPS is a Global Navigation Satellite Systems (GNSS) under non-civilian American Control. Currently, there are only two operational alternatives to GPS. There is the Russian Global Navigation Satellite System (GLONASS) and the European Galileo navigation satellite system. GLONASS is equally to GPS under non-civilian control which makes Galileo an exception as it is being operated by civilians. Additionally, Galileo is able to determine the user's position with greater precision than what is offered by the other available systems (cf. Section 2.2)[1].

Simultaneously using multiple GNSS should achieve a greater accuracy and a better noise tolerance than using one alone [2]. As Galileo and GPS have signals of the same centre frequency, the same data can be processed by the combined GPS and Galileo implementation which results in a better performance. Therefore the existing GPS implementation is extended with Galileo.

Chapter 2

Background

Chapter 2 gives an overview of Global Navigation Satellite Systems (GNSS) and focuses on the specifications of the Galileo E1 and GPS L1 signals. Additionally, the classical positioning algorithm and the Collective Detection algorithm used by the implementation are introduced.

2.1 Global Navigation Satellite Systems (GNSS)

A satellite navigation system uses satellites which continuously transmit time signals along a line of sight by radio. With the help of those signals, receivers are able to determine their location, which includes longitude, latitude and altitude, to a high precision. Therefore such systems can be used for providing position, navigation, tracking and time synchronization. [1]

Satellite navigation systems can cover only a certain region or can cover the whole world. Some example for regional satellite systems are the Quasi-Zenith Satellite System (QZSS) that covers the region in and around Japan or the Navigation with Indian Constellation (NAVIC) for India. A satellite navigation system with global coverage is called Global Navigation Satellite System (GNSS). Currently there are three GNSS in operation and one in development. The operational GNSS include the American Global Positioning System (GPS), the Russian Global Navigation Satellite System (GLONASS) and the younger European Galileo navigation system. The Chinese BeiDou navigation satellite system has two separate satellite constellations. The system currently in operation only covers the region of China and the second one, currently in development, is planned to cover the whole world. [1]

As the topic of this project is about implementing a Galileo software-defined receiver to an already existing GPS implementation, this report is focusing on the Galileo navigation system and is giving a short overview of the differences between Galileo and GPS.

2.1.1 GNSS Frequency Bands

Frequency bands can be used by multiple services and users concurrently. Additionally, the same frequencies can be allocated for different purposes in different countries. The International Telecommunications Union (ITU) of the United Nations is responsible for coordinating the allocation of different frequency bands, such as the Radio Navigation Satellite System (RNSS) band which is part of the radio-frequency band. GNSS, hence GPS and Galileo belong to the RNSS. As visible in Figure 2.1 their frequency bands overlap partially with the spectrum for Aeronautical Radio Navigation Services (ARNS). The ARNS belong to the safety-of-life services which means that their frequency spectrum is protected against harmful interferences. Therefore signals within the ARNS bands are more robust than those only within the RNSS bands. [3][4]

The different frequency bands which are used by Galileo and GPS are listed in the two Tables 2.1 and 2.2 and are illustrated in Figure 2.1.

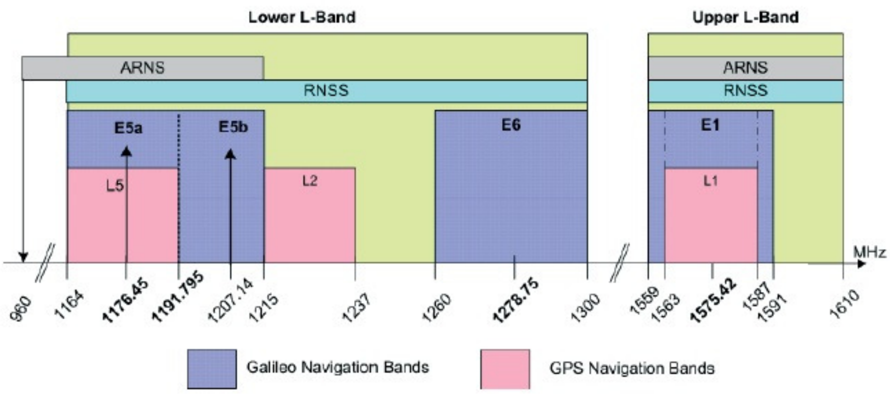


Figure 2.1: GPS and Galileo Frequency Plan. [4]

Galileo Signal	Carrier Frequency MHz	Receiver Reference Bandwidth MHz
E1	1575.420	24.552
E6	1278.750	40.920
E5	1191.795	51.150
E5a	1176.450	20.460
E5b	1207.140	20.460

Table 2.1: Galileo Signal Carrier Frequencies and Receiver Reference Bandwidths. [4]

The two Tables 2.1 and 2.2 show that the GPS L1 band and the Galileo E1 frequency band can be received over the same carrier frequency 1575.420 MHz.

GPS Signal	Carrier Frequency MHz	Receiver Reference Bandwidth MHz
L1	1575.420	24.000
L2	1227.600	24.00
L5	1176.450	24.000

Table 2.2: GPS Signal Carrier Frequencies and Receiver Reference Bandwidths. [5]

The preexisting GPS implementation uses the L1 band and therefore the new Galileo implementation uses the Galileo E1 frequency band for simultaneously receiving and process both GNSS signals. The following parts will focus on the GPS L1 and Galileo E1 signals.

As the preexisting GPS implementation uses the L1 band and the new Galileo implementation uses the Galileo E1 frequency band for simultaneously receiving and process both GNSS signals. The following parts will therefore focus on the GPS L1 and Galileo E1 signals.

2.1.2 GNSS Signals

The GNSS signals are continuously transmitted in multiple frequencies and are composed of the following main signal components: [3]

Carrier Signal $g_T(t)$

A sinusoidal signal at a given frequency inside the radio frequency range.

Navigation Data $d(u)$

$d(u)$ is a navigation data message symbol with the period T_b.

A binary-coded message containing the satellite's ephemeris data, clock bias parameters, almanac data, satellite health status, and other information.

Spreading Code i

$c_i(l)$ is the l$^{\text{th}}$ chip [1] of spreading code i with length L_c and chipping period $T_c = \frac{T_b}{N_c L_c}$.

Sequences of 0s and 1s which are unique for each satellite and are called Pseudo-Random Noise (PRN) sequences or PRN codes. They allow the

[1] A chip is a rectangular pulse of +1 or -1 amplitude which can be multiplied by a data sequence and/or by a carrier waveform. Chips are therefore the bit waveform produced by a code generator and are renamed to distinguish them from message bits. [6]

receiver to determine the travel time of radio signal from satellite to receiver.

A GNSS space vehicle i transmits a generic GNSS complex base band signal as described in Equation 2.1 where the previously listed definitions are used.

$$s_T(t) = \sqrt{P_T} \sum_{u=-\infty}^{\infty} d(u)p(t - uT_{b_I}) \tag{2.1}$$

P_T is the transmitting power and $p(t)$ is defined in Equation 2.2.

$$p(t) = \sum_{k=0}^{N_c-1} q(t - kT_{PRN}) \tag{2.2}$$

N_c is the number of repetitions of a full code word that spans a bit period. Therefore the code word period is $T_{PRN} = \frac{T_b}{N_c}$. The function $q(t)$ is defined in Equation 2.3 where the transmitting subcarrier $g_T(t)$ is considered energy-normalized for notation clarity. [7]

$$q(t) = \sum_{l=0}^{L_c-1} c_i(l)g_T(t - lT_c) \tag{2.3}$$

2.2 Galileo

The European Union (EU) is currently creating the Galileo GNSS through the European Space Agency (ESA) and the European GNSS Agency (GSA). This European GNSS is the only GNSS fully under civilian control. The Galileo space segment will once finished include 30 satellites whereof six satellites are used as backup. They are arranged in three orbital planes, each containing eight equally spaced operational satellites and two spare satellites. [8]

Currently there are 18 Galileo satellites online, whereof 12 are usable while the other six satellites are being tested or not usable. [9]

Table 2.3 shows what position accuracies can be achieved by using a single frequency Galileo Open Service.

Comparison with GPS

In contrast to Galileo the Global Positioning System (GPS) is operated by the United States Air Force (USAF) and is owned by the United States Government (USG). The United States is guaranteeing an availability of at least 24

		Galileo OS	GPS SPS
Accuracy	Horizontal:	15 m	< 100 m
(95 %)	Vertical:	35 m	< 156 m

Table 2.3: Galileo Open Service and GPS Standard Positioning Service Accuracy for single frequency (SF). [10] [11]

operational GPS satellites 95% of the time. This constellation with 24 satellites ensures that at least four satellites are visible from any point on the earth. Meanwhile, the constellation is improved with 27 operational satellites, which improves the coverage in most parts of the world. [12]

Table 2.3 shows that Galileo is expected to have a better single frequency accuracy. When combining both signals, even better performance can be achieved. [2]

2.2.1 Galileo Signal

Once fully operational Galileo will provide four high-performance services worldwide: [13]

- **Open Service (OS)**: free of charge, available to civilian users, positioning and timing services
- **Commercial Service (CS)**: provides additional navigation signal and added-value services in a different frequency band, can be encrypted in order to control the service access
- **Public Regulated Service (PRS)**: restricted to government-authorized users, for applications with a high requirement of service continuity
- **Search and Rescue Service (SAR)**: used for COSPAS-SARSAT, an international satellite-based search and rescue distress alert detection system

As already mentioned the Galileo implementation in this work uses the OS E1 band. Therefore, in the following an overview of the Galileo E1 signals is given.

Galileo E1 Signal

The Galileo E1 band is centered at $f_{GalE1} = 1575.420\,\text{MHz}$ and has a reference bandwidth of 24.5520 MHz. Figure 2.2 shows the Galileo E1 signal plan. [4]

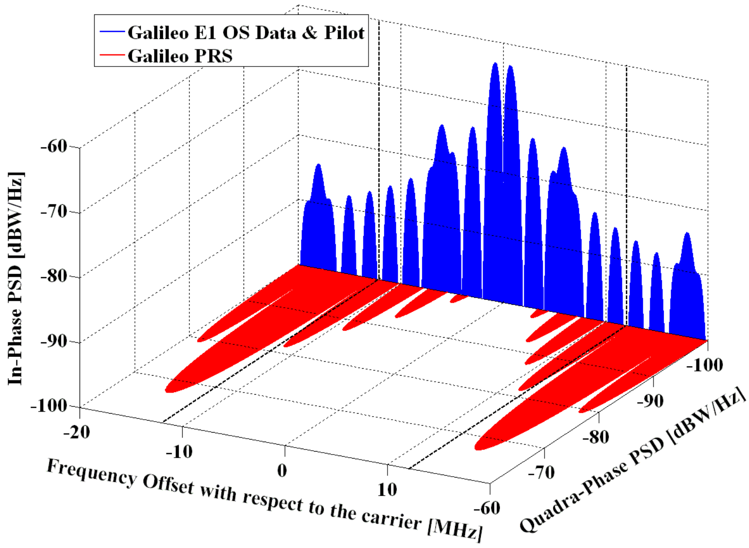


Figure 2.2: Galileo E1 Signal Plan. [14]

BOC Modulation The BOC modulation is a combination of a sine-phased subcarrier signal of frequency f_s with a code sequence signal $c(t)$ of frequency $f_c = \frac{1}{T_c}$ where T_c is the code period. Equation 2.4 defines the sine-phased BOC modulation signal with $c(t)$ defined in Equation 2.5 and subcarrier $sc(t)$ defined in Equation 2.6. c_k is the code sequence waveform and $h(t)$ is the Non Return to Zero (NRZ) code with value 1 over the time $[0, T_c]$. [15]

$$\text{BOC}(f_s, f_c) = c(t) \cdot sc(t) \tag{2.4}$$

$$c(t) = \sum_k c_k h(t - kT_c) \tag{2.5}$$

$$sc(t) = \text{sign}\big(\sin(2\pi f_s t)\big) \tag{2.6}$$

The BOC modulation is often shortened to BOC(m, n) where m and n are defined in Equation 2.7 and 2.8.

$$f_s = m \cdot 1.023\,\text{MHz} \tag{2.7}$$

$$f_c = n \cdot 1.023\,\text{MHz} \tag{2.8}$$

Equation 2.9 defines Φ which is the number of half periods of the subcarrier that fit in a single code chip as visible in Figure 2.3.

$$\Phi = 2\frac{f_s}{f_c} = 2\frac{m}{n} \tag{2.9}$$

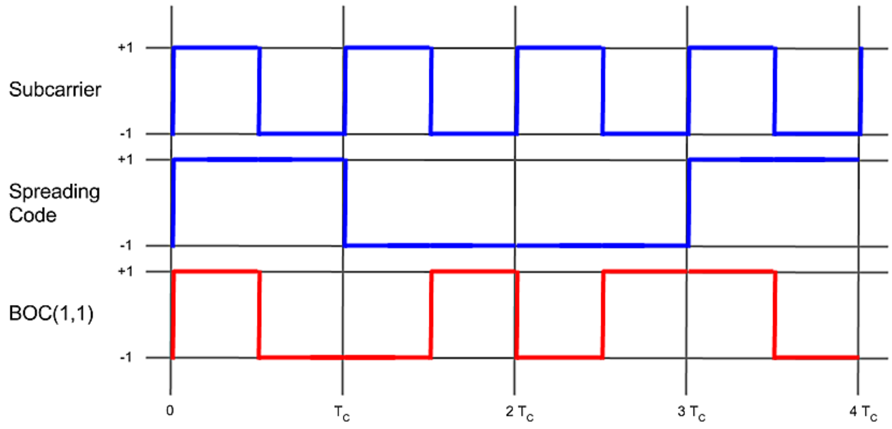


Figure 2.3: BOC(1, 1) Signal is the Multiplication of the Subcarrier with the PRN Code Sequence.

CBOC Modulation The E1 OS Signal is defined in Equation 2.10 with $\alpha = \sqrt{\frac{10}{11}}$ and $\beta = \sqrt{\frac{1}{11}}$. The E1 signal has two OS signal channels, the data channel B and the pilot channel C. Both channels use so-called Composite Binary Offset Carrier (CBOC) modulations. The CBOC modulation of the E1 OS signal is illustrated in Figure 2.4 and the resulting channel signals are plotted in Figure 2.5. [7][4]

$$S_T^{(GalE1)}(t) = \frac{1}{\sqrt{2}}\Big(e_{E1B}(t)\big(\alpha sc_A(t) + \beta sc_B(t)\big) - e_{E1C}(t)\big(\alpha sc_A(t) - \beta sc_B(t)\big)\Big) \tag{2.10}$$

A CBOC modulation combines two BOC modulations, namely BOC(1, 1) and BOC(6, 1). This means that both channels have the same two subcarriers $sc_A(t)$ and $sc_B(t)$ with the frequencies $f_{s,E1A} = 1 \cdot 1.023\,\text{MHz}$ and $f_{s,E1B} = 6 \cdot 1.023\,\text{MHz} = 6.138\,\text{MHz}$. The subcarriers $sc_A(t)$ and $sc_B(t)$ are defined in Equations 2.11 and 2.12. [16]

$$sc_A(t) = \text{sign}\left(\sin\left(2\pi f_{s,E1A} t\right)\right) \tag{2.11}$$

$$sc_B(t) = \text{sign}\left(\sin\left(2\pi f_{s,E1B} t\right)\right) \tag{2.12}$$

The data channel message $e_{E1B}(t)$ and the pilot data-less channel message $e_{E1C}(t)$ us so called primary, secondary and tiered codes. Therefore they are defined in Section 2.2.2.

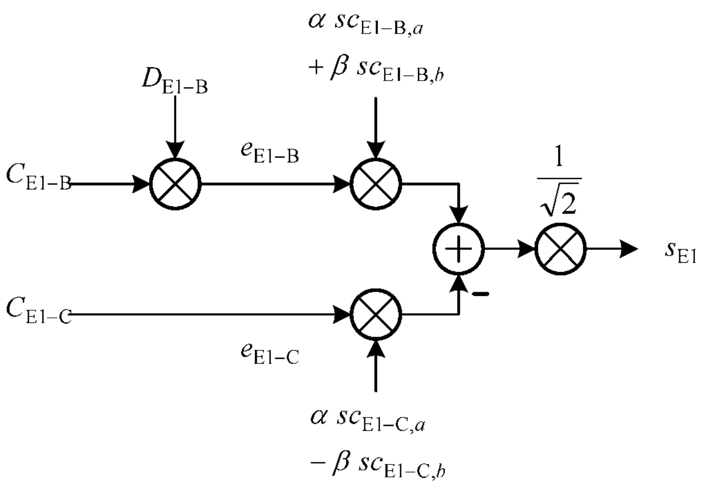
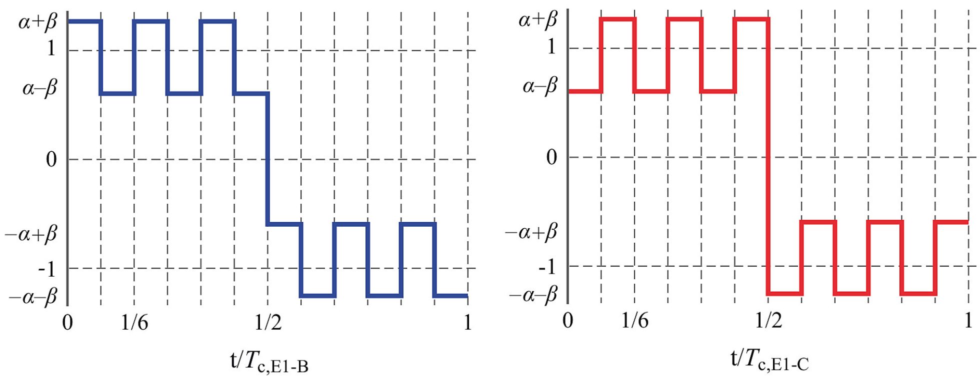


Figure 2.4: Galileo E1 Modulation Scheme. [4]



(a) Data Signal.

(b) Pilot Signal.

Figure 2.5: CBOC Signals for the B and C Channel of the Galileo E1 Band. [4]

Comparison with GPS

Similar to Galileo, GPS provides multiple services:

- **Precise Positioning Service (PPS)**: available primarily to the military of the United States and its allies
- **Standard Positioning Service (SPS)**: available to civilian users

Common receivers use the signals of the GPS L1 band, which is centered at $f_{GalE1} = 1575.420\,\text{MHz}$. Figure 2.6 shows the GPS L1 signal plan. The L1 band contains three signals in use and one additional currently in development as listed below. [7] [17]

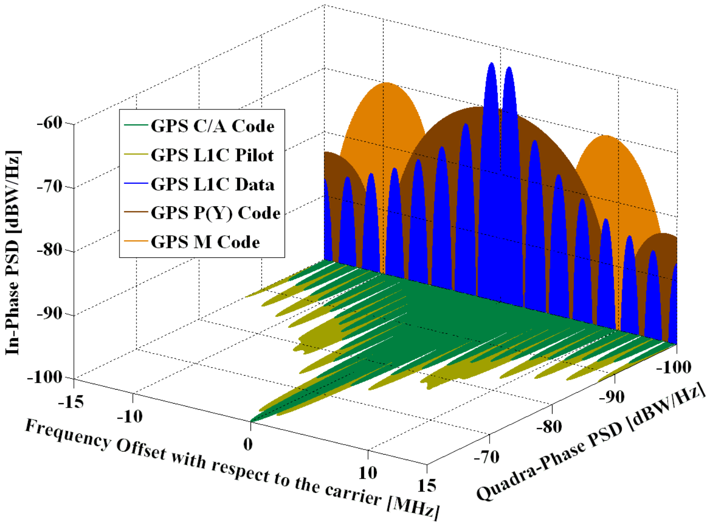


Figure 2.6: GPS L1 Signal Plan. [17]

- **Coarse/Acquisition (C/A) Code** (SPS)

  The C/A code signal is nowadays the most important signal for common applications and uses so-called Gold codes.

- **P(Y) Code** (SPS, PPS)

  The Precision code signal has an additional Anti-Spoofing (A/S) mode of operation where the signals are encrypted and the P-Code is replaced with the Y-Code. With enabling this mode the service is changing from SPS to PPS.

- **M-Code** (PPS)

  This code is exclusively for military use and provides more robust signal acquisition compared to P(Y) while offering better security and better jamming resistance.

- **L1C** (SPS)

  The L1C signal is a new civil signal in development. Similar to Galileo E1 OS, the signal will have a pilot and a data channel and uses a CBOC modulation of BOC(1, 1) and BOC(6, 1).

The complex base band GPS signal transmitted by the space vehicles uses Binary Phase Shift Keying (BPSK) and is defined in Equation 2.13.

$$S_T^{(GPSL1)}(t) = e_{L1I}(t) + je_{L1Q}(t) \tag{2.13}$$

The functions $e_{L1I}(t)$ and $e_{L1Q}(t)$ are defined in Equations 2.14 and in 2.15. The preexisting implementation uses only the GPS L1 C/A code signal and therefore the Q channel is the important one.

$$e_{L1I}(t) = \sum_{l=-\infty}^{\infty} D_{NAV}\Big[[l]_{204600}\Big] \oplus C_{P(Y)}\Big[|l|_{L_{P(Y)}}\Big] p(t - lT_{c,P(Y)}) \tag{2.14}$$

$$e_{L1Q}(t) = \sum_{l=-\infty}^{\infty} D_{NAV}\Big[[l]_{20460}\Big] \oplus C_{C/A}\Big[|l|_{1023}\Big] p(t - lT_{c,C/A}) \tag{2.15}$$

D_{NAV} is the navigation message bit sequence with a rate of 50 b/s, $C_{P(Y)}$ is the P-Code sequence with $T_{c,P(Y)} = \frac{1}{10.23}$µs and $L_{P(Y)} = 6.1871 \times 10^{12}$ while $C_{C/A}$ is the Gold Code sequence with $T_{c,C/A} = \frac{1}{1.023}$µs. The signal $p(t)$ is a rectangular pulse centered at $t = 0$ and the duration of a chip-period. [7]

Figure 2.7 illustrates the Q channel BPSK. The main difference to the Galileo E1C signal is that the GPS E1 C/A signal contains navigation data while the Galileo signal is data-less. Additionally, the spreading codes are different as described in the next Section 2.2.2.

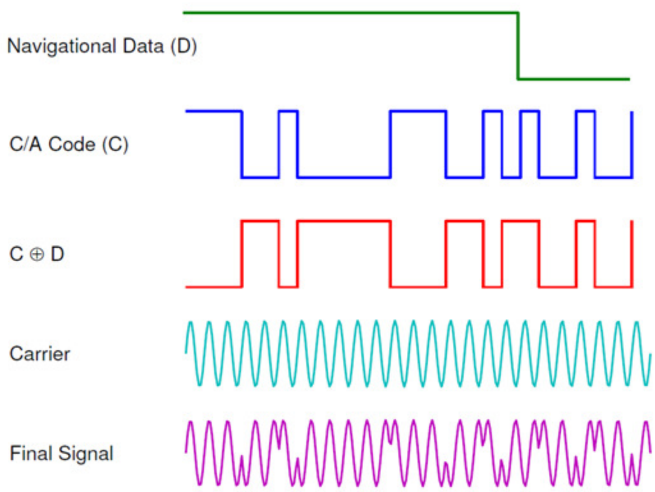


Figure 2.7: GPS L1 C/A Signal Modulation Scheme. [18]

2.2.2 Galileo Spreading Codes

The Galileo spreading codes are so called tiered codes consisting of a primary and a secondary code. These codes are so-called pseudo-random memory code sequences, which means they are defined and can directly being stored in the memory. There are 50 defined primary codes where each one belongs to a specific satellite and one unique secondary code which is reused for each satellite. [4]

As described in Section 2.2.1, the Galileo E1 signal has two channels which use the codes in a different way. The code lengths for both channels are listed in Table 2.4.

	Tiered/Gold Code Length	Tiered /Gold Code Period	Primary Code Length	Secondary Code Length
Galileo E1B	4092	4 ms	4092	N/A
Galileo E1C	102 300	100 ms	4092	25
GPS L1 C/A	1023	1 ms	N/A	N/A

Table 2.4: Code Lengths for Galileo E1 Band. [4] [17]

The data channel message $e_{E1B}(t)$ is defined in Equation 2.16 and uses a primary code C_{E1B_p} combined with an I/NAV type of navigation message and no secondary code. The primary code has a chipping rate of $T_{c,E1B_p} = \frac{1}{f_{c,E1B_p}} = \frac{1}{1 \cdot 1.023}$µs, the symbol $\oplus$ is the exclusive-or operation and $[l]_L$ stands for the integer part of $\frac{l}{L}$.

$$e_{E1B}(t) = \sum_{l=-\infty}^{\infty} D_{I/NAV}\left[[l]_{4092}\right] \oplus C_{E1B_p}\left[[l]_{4092}\right] p(t - lT_{c,E1B}) \tag{2.16}$$

The implementation uses the algorithm approach described in Section 2.3.2. As this implementation does not need to extract the data from the signal but downloads them from a server, the data signal is being ignored and the pilot signal is used for the new Galileo implementation.

The pilot (data-less) channel message $e_{E1C}(t)$ is also called tiered code and is defined in Equation 2.17 where $|m|_L$ stands for $m \pmod L$. This tiered code uses a primary code C_{E1C_p} with a chipping rate of $T_{c,E1C_p} = \frac{1}{f_{c,E1C_p}} = \frac{1}{1 \cdot 1.023}$µs and the unique secondary code C_{E1C_s} with the chipping rate of $T_{c,E1C_s} = 4$ ms.

$$e_{E1C}(t) = \sum_{m=-\infty}^{\infty} C_{E1C_s}\left[|m|_{25}\right] \oplus \sum_{l=1}^{4092} C_{E1C_p}[l] p(t - mT_{c,E1C_s} - lT_{c,E1C_p}) \tag{2.17}$$

Figure 2.8 explains the tiered code in a simpler way. In the figure the primary code has the length N and a chip rate of f_c while the secondary code has the length N_s and a chip rate of $f_{cs} = \frac{f_c}{N}$. For the tiered code all the bits of the primary code are exclusive-ored with one single bit of the secondary code. Therefore the primary code is repeated N_s times while the generated tiered code has the length $N_s \cdot N = 4092 \cdot 25 = 102\,300$. [4]

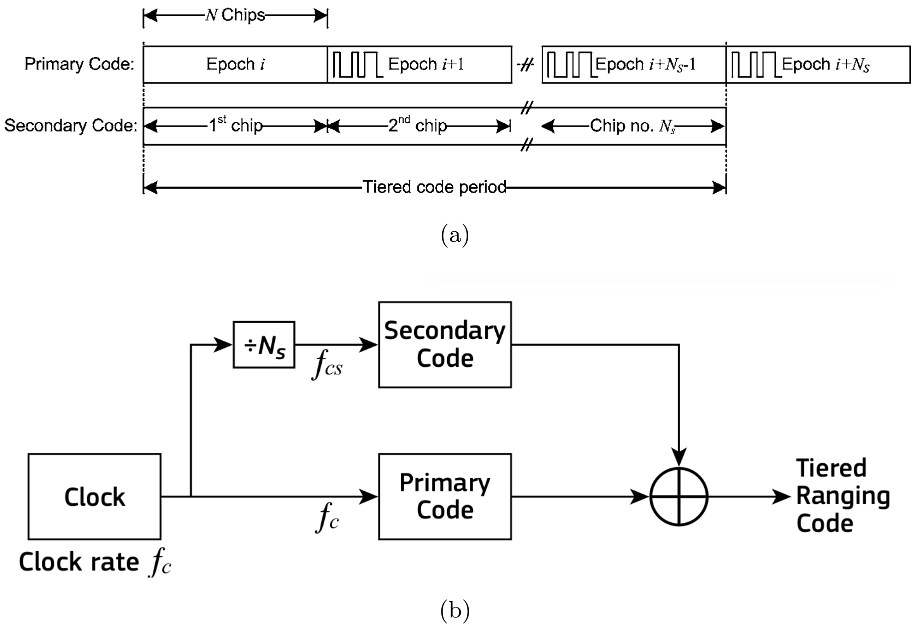


Figure 2.8: Graphical Illustration of the Generation of Galileo Tiered Codes. [4]

Comparison to GPS

GPS L1 C/A uses so-called pseudo-random Gold codes of length $L_{CA} = 1023$. Gold codes have a limited small cross-correlation with each other. This makes them ideal for satellite PRN codes as multiple satellites are broadcasting their signals within the same frequency range.

The codes are generated using two shift registers G_1 and G_2 of length $L_g = 10$. The shift registers are first initialized with 1s, then a feedback is computed. The feedback function for G_1 is defined in Equation 2.18 and the feedback of G_2 in Equation 2.19.

$$fb_{G_1} = \big(G_1[2] + G_1[9]\big) \pmod 2 \tag{2.18}$$

$$fb_{G_2} = \big(G_2[1] + G_2[2] + G_2[5] + G_2[7] + G_2[8] + G_2[9]\big) \pmod 2 \tag{2.19}$$

The fb_{G_1} is then inserted from the left (at index 0) of the shift register G_1 while all entries of the register are shifted to the right. The same happens with

fb_{G_2} and G_2. After this shift the output is computed as in Equation 2.20. The indexes k and l are defined by the space vehicle ID for which the Gold code is being generated and differ for each space vehicle ID.

$$o = \big(G_1[9]] + (G_2[k] + G_2[l])\big) \pmod 2 \tag{2.20}$$

The steps of feedback, shift and output computation is repeated $L_{CA} = 1023$ times. The output o of each round i is then the i-th bit of the Gold Code $C_{C/A}$ for a certain satellite.[19]

These codes can be generated on the go, or they could be precalculated and stored in the memory. This is one difference to the Galileo codes which cannot be generated. Additionally, the gold codes are shorter than the Galileo codes. A general code length comparison can be found in Table 2.4.

2.2.3 Galileo Navigation Data

The Galileo navigation data contains the following data:

- **Ephemeris data**: needed to compute the position of the satellite
- **Time and clock correction parameters**: needed to compute the pseudo-range
- **Service parameters**: needed to identify the set of navigation data, satellites and to determine the signal health
- **Almanac parameters**: needed to determine all satellite positions in the whole constellation with a reduced accuracy

More specific information and definitions can be found in the European GNSS (Galileo) Open Service Signal In Space Interface Control Document [4].

As Galileo uses mostly the same models as GPS there are only a few differences between their navigation data.

2.3 GNSS Algorithm

In this section, the classic approach to determine a position from a signal sample is shortly defined. Additionally, the actually implemented algorithm developed at the Distributed Computing Group is explained.

2.3.1 Classic Approach

Traditional receivers first perform an acquisition to determine which satellites are visible from the receiver's position. Therefore, a predetermined pseudo random noise (PRN) sequence of each possible satellite is correlated with the incoming signal. The line-of-sight velocity of the satellite causes a Doppler shift where the frequency can shift up to 10 kHz. Additionally, the receiver time is not synchronized to the satellites and the signal propagation time is unknown, which results in not knowing which part of the PRN has to be correlated with the incoming signal. Therefore, the acquisition algorithm has to search for each satellite over all possible code phases and frequencies. [20]

After finding enough satellites, the tracking phase is started in which the navigation data is extracted from the incoming signal.

2.3.2 Collective Detection

The Distributed Computing Group developed a different algorithm for determining a receiver's position. In this approach the receiver only records the signal and sends the samples to a server. The server processes the samples and sends the computed position back to the receiver.

The server uses a non classic approach for the position determination. In a first step it downloads the required GNSS data from a global data center for International GNSS Service (IGS), the NASA's Crustal Dynamics Data Information System (CDDIS)[21]. CDDIS summarizes all GNSS data and provides them in files of the RINEX format. The RINEX format is a receiver and platform independent format to exchange satellite navigation system data. A RINEX file contains all navigation data for satellites of exactly one GNSS (i.e. only GPS or only Galileo). The Navigation data are summarized in Section 2.2.3.

The Ephemeris data are used to compute the satellite position, satellite speed and the propagation delay of the signal from the satellite to the receiver. The Doppler shift can be derived from the satellite speed. The local generated signal can be resampled to the correct Doppler frequency and therefore the code phase can then retrieve by correlating the incoming GNSS signal with the local generated signal. In the end the server has all needed information without extracting any data from the recorded GNSS signal. As a result, the local generated signal only has to contain the data-less pilot signal for the newly added Galileo implementation part.

Once the code phase and correct frequency are known, the likelihood function is computed over a set of points in space and time. The position with the best likelihood result is then defined to be the receiver's location. The existing implementation uses a likelihood function in two ways. In the first method the likelihood is computed for every point in a defined range around a reference position. The second method, called branch and bound allows to compute the likelihood computation of a bigger area than in the first method.

Chapter 3

Implementation

In this chapter the different steps of the actual implementation of the Galileo receiver are described.

3.1 Navigation Data

As described in Section 2.3.2, the first step is to download all navigation data for GPS and Galileo from the server of NASA's Crustal Dynamics Data Information System (CDDIS). These downloaded RINEX files contain navigation data of multiple satellites of a certain system (GPS, Galileo, ...).

3.1.1 Download

The International GNSS Service (IGS) is a voluntary federation of over 200 self-funding institutions in more than 100 countries with the mission of ensure open access, high-quality GNSS data, products and services such as positioning, navigation and timing that benefit science and society. The IGS has a global network of permanent GNSS receivers operating 24 hours a day and sends their daily data to the CDDIS database which are made available to the public in form of the RINEX files. [22]

The RINEX files relevant for GPS and Galileo are stored gzip compressed and are sorted in directories of the structure *year/day/gnss_system/rinex_files*. The data for GPS satellites are nicely summarized into one single file per day while the Galileo satellite data are split up into multiple files per day, according to their permanent receiver location. Not each station is sending data for each day and some stations are summarizing the data of multiple days into one file. Therefore the number of available Galileo files per day can vary. In general, only one file per day is downloaded for the GPS data while for the Galileo data up to 40 files per day need to be downloaded.

Therefore, the Galileo navigation data download process is extended to download not only one specific file, but all contained in the folder which are fulfilling a certain name format. In a first step a file is downloaded by using the libcurl library [23], containing a list of all files in the directory. After receiving the list, one file after the other is downloaded to a temporary folder, uncompressed and copied to its final folder.

3.1.2 RINEX Parser

RINEX stands for Receiver Independent Exchange Format and is a data format for receiver and system independent storing and exchanging of satellite navigation data. There exist multiple different versions of the RINEX format. The files for GPS for example are available on the CDDIS server in the RINEX Version 2 (RINEX V2) while the files used for Galileo are stored in RINEX Version 3 (RINEX V3) files. Therefore, the preexisting GPS implementation had only an RINEX V2 parser which means that a new RINEX V3 parser is added for the Galileo implementation. The parser is implemented according to the RINEX The Receiver Independent Exchange Format Version 3.03 Documentation in [24].

As mentioned before there are much more files to be parsed than for GPS and therefore the Galileo implementation has to search the best model out of much more models than the GPS implementation. Therefore the RINEX parser checks if the received data are valid and if they contain the correct I/NAV data type which is needed for the Galileo E1 signals. If the data do not pass those checks they are discarded and the parser continues with the next data block.

3.2 Generation of Local Galileo Signal

Following the algorithm in Section 2.3.2, the incoming signal has to be correlated with a local generated signal with the correct Doppler shift frequency in order to get all visible satellites and their code phase.

Sections 2.2.1 and 2.3.2 provide a declaration why the data-less pilot signal E1C can be used alone to compute the correlation. Therefore, the B channel of Equation 2.10 can be ignored and the local signal $S_T^{(GalLocal)}(t)$ looks like in Equation 3.1 with $e_{E1C}(t)$ being the tiered code as defined in Equation 2.17 and Section 2.2.2.

$$S_T^{(GalLocal)}(t) = -\frac{1}{\sqrt{2}} e_{E1C}(t)\big(\alpha sc_A(t) - \beta sc_B(t)\big) \tag{3.1}$$

The correlation process is finished by taking the absolute value of the correlation vector. Additionally, the multiplication by a constant factor has no impact on the code phase but scales the overall correlation vector. Therefore Equation 3.1 can be simplified to Equation 3.2.

$$S_T^{(GalLocal)}(t) = e_{E1C}(t)\big(\alpha sc_A(t) - \beta sc_B(t)\big) \tag{3.2}$$

This signal is still a CBOC modulation combining BOC(1, 1) and BOC(6, 1). The two subcarriers $sc_A(t)$ and $sc_B(t)$ are differently weighted by α and β. As $\alpha = \sqrt{\frac{10}{11}} > \sqrt{\frac{1}{11}} = \beta$ the carrier $sc_A(t)$ is contributing much more to the final correlation than the faster $sc_B(t)$ which is only responsible for the small ripples as visible in Figure 2.5b. Therefore, the local generated signal can be simplified from a CBOC modulation signal to a single BOC(1, 1) modulation signal as defined in Equation 3.3.

$$S_T^{(GalLocal)}(t) = e_{E1C}(t) \cdot sc_A(t) \tag{3.3}$$

Section 2.3.2 states that the the code phase is the only thing needed to be retrieved by correlating the incoming signal with the local generated one. As the primary code is the shortest periodically repeating code of Galileo it is enough to correlate over the length of one primary code instead over the whole tiered code. This reduces the time slot over which the correlation has to be computed by a factor of 25 as visible from the Table 2.4.

As the primary code is a pseudo-random memory code, it is stored as an array of booleans. For the BOC(1, 1) modulation the code needs to be converted from the logic level to the signal level according to Table 3.1.

Logic Level	Signal Level
1	−1.0
0	1.0

Table 3.1: Logic Level to Signal Level Conversion. [4]

BOC(1, 1) means according to the definition in Section 2.2.1 and Equations 2.7 and 2.8 that $m = n = 1$ and therefore $\Phi = 2$ according to Equation 2.9. This implies that exactly two half periods of the subcarrier fit in a single code chip as visible in Figure 2.3. Therefore the code signal is sampled into an array of twice the length of the primary code with a sampling rate of $\Phi \cdot f_c = 2 \cdot f_c$.

Finally, the BOC signal is generated by multiplying the subcarrier with the primary code as illustrated in Figure 2.3. In the very end this code signal is being resampled to the correct frequency which takes the precalculated Doppler shift into account.

3.3 Combined Correlation of Galileo and GPS

The final implementation is supposed to support GPS and Galileo. Therefore, the general function that computes the correlation is split up into one function generating the Galileo correlations from the samples and another function generating the GPS correlations from the same received samples.

In general, the implementation is streaming in some recorded signal samples. It can be defined how long the time slots should be that are used for a correlation. This correlation time is then split up into multiple correlation windows.

3.3.1 Galileo Correlation

As the shortest repeated code in Galileo is the primary code with a period of 4 ms, the correlation time should be at least that period or a multiple of it. If it is a multiple of it the correlation time is split up into windows each exactly containing samples of 4 ms. The correlation is then computed over each of those windows and in the end summed up in order to get a sharper correlation peak. The correlation length is therefore exactly 4092.

The correlation itself is a circular cross correlation and is implemented like the preexisting GPS correlation using the Parallel Code Phase Search Acquisition algorithm. [20]

3.3.2 GPS Correlation

The gold codes of GPS have a period of only 1 ms. Therefore the correlation time is split up into windows of the length of 1 ms. Therefore, GPS always has $\frac{T_{C,Gal}}{T_{C,GPS}} = 4$ times more windows than Galileo. The correlation of these windows again are summed up for getting a sharper correlation peak. Additionally, the correlation length 1023 is 4 times shorter than the Galileo correlation length. For storing all correlations inside the same correlation map and for guaranteeing a proper working of the likelihood function the GPS correlation is padded with 0s to the length of a Galileo correlation. As the code phase is determined by getting the peak of the correlation, this zero-padding has no impact on the code phase.

Chapter 4

Evaluation

This chapter summarizes how the recordings were done and gives a final evaluation of the project.

4.1 Recordings

Four recordings were available for comparison and evaluation. The first recording file is an older one recorded on the 12^{th} of February, 2016 around 4pm. Unfortunately, this recording has no clean signal of Galileo satellites. This file was therefore used for verifying that the GPS part of the implementation is still working as intended.

The other three files were recorded on different days and different locations with an USRP B200. All those recordings contain good signals of both Galileo and GPS satellites and were therefore used to fully verify the new implementation. The settings of the USRP B200 and other informations about the recordings can be found in Table 4.1. The measurement named Glanzenberg is the only one where the recording was done with a bandwidth of 3 MHz instead of 8 MHz.

4.2 Correlation

As described in Section 2.3.2 the two important information needed to determine the correct position are the Doppler frequency and the code phase. The frequency is needed for generating the correct local signal which is correlated with the incoming one. The code phase is then the sample shift at which the correlation has its peek. Therefore having a correlation with a high peek shows that the signals are correctly decoded and the implemented system is working.

The correlation time can be at least 4 ms or a multiple of it. The correlations for the recording ETZ are plotted for a correlation time of 4 ms and 12 ms. Figure 4.2 shows some of the GPS correlations while Figure 4.1 shows all Galileo correlation for 4 ms. Additionally Figure 4.4 shows some of the GPS correlations

Name	Glanzenberg	ETZ	Fluntern
Location Coord.	47.3996, 8.4266	47.3774, 8.5529	47.3821, 8.5728
Heigth	385m a.s.l.	551m a.s.l.	609m a.s.l.
Date	June 5, 2017	June 7, 2017	June 20, 2017
UTC time	14:50:58.355	09:49:22.286	08:13:02.537
Max. Time Error	0.277 s	0.813 s	0.018 s
Type	Byte	Byte	Byte
Core frequency	1575.420 MHz	1575.420 MHz	1575.420 MHz
Sampling rate	8 MHz	8 MHz	8 MHz
Bandwidth	3 MHz	8 MHz	8 MHz
Gain	73 dB	73 dB	73 dB
Duration	60 s	60 s	60 s

Table 4.1: Recording Settings for USRP B200 and Informations about Time and Location of the Recording.

while Figure 4.3 shows all Galileo correlation for 12 ms. Those figures were generated by taking into account all satellites with an eleveation of at least 10 degrees.

As described in Section 3.3 the correlations in those figures show higher peaks for the 12 ms correlation time than those for 4 ms. The figures show that the height of the peaks vary with each measurement and each recorded satellite signal. There are even some correlations which do not show a peak at all which means that no good signal of those satellites was recorded. Additionally, it is visible that the GPS correlations are padded with 0s to the length of the galileo correlation as mentioned in Section 3.3.2.

As multiple Galileo correlations show a clear high peak, it shows the combined GPS and Galileo implementation works correctly.

4.3 Precision

Unfortunately, the project was build on a code base which is currently broken. This means that it is impossible to calculate the likelihood function using GPS or Galileo. This problem was found very late because the code is working for only the older measurement. Therefore it was not possible to get any results on how the precision of the computed position is changing when using GPS alone, Galileo alone or both systems combined.

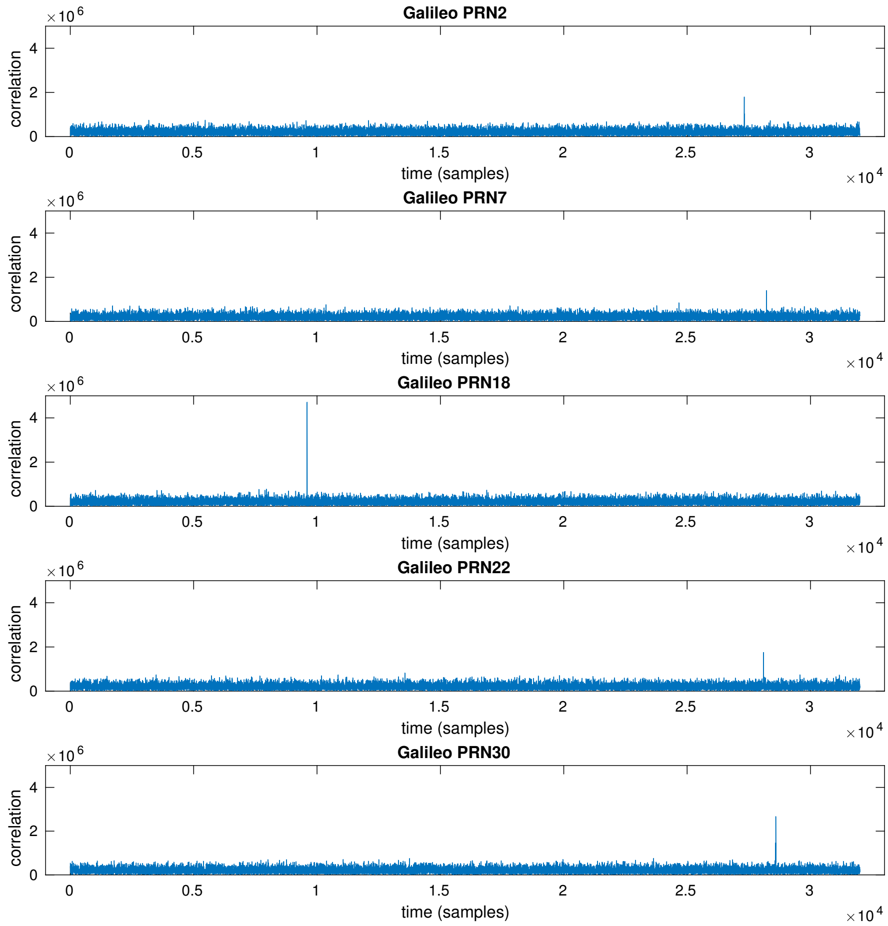


Figure 4.1: All Galileo Correlations with a 4 ms Correlation Window of the Recording named ETZ.

4.4 Performance

This section compares shortly the CPU performance of the GPS only implementation from the semesterproject of Manuel Eggimann, the GPS part of the combined implementation, the Galileo part of the combined implementation and the combined GPS and Galileo Implementation. Those tests were done with the recording ETZ for a correlation window of 4ms and were performed on the following system:

- Dual-Core Intel i7-5500U CPU @ 2.40GHz
- 8GB DDR3L-1600Mhz

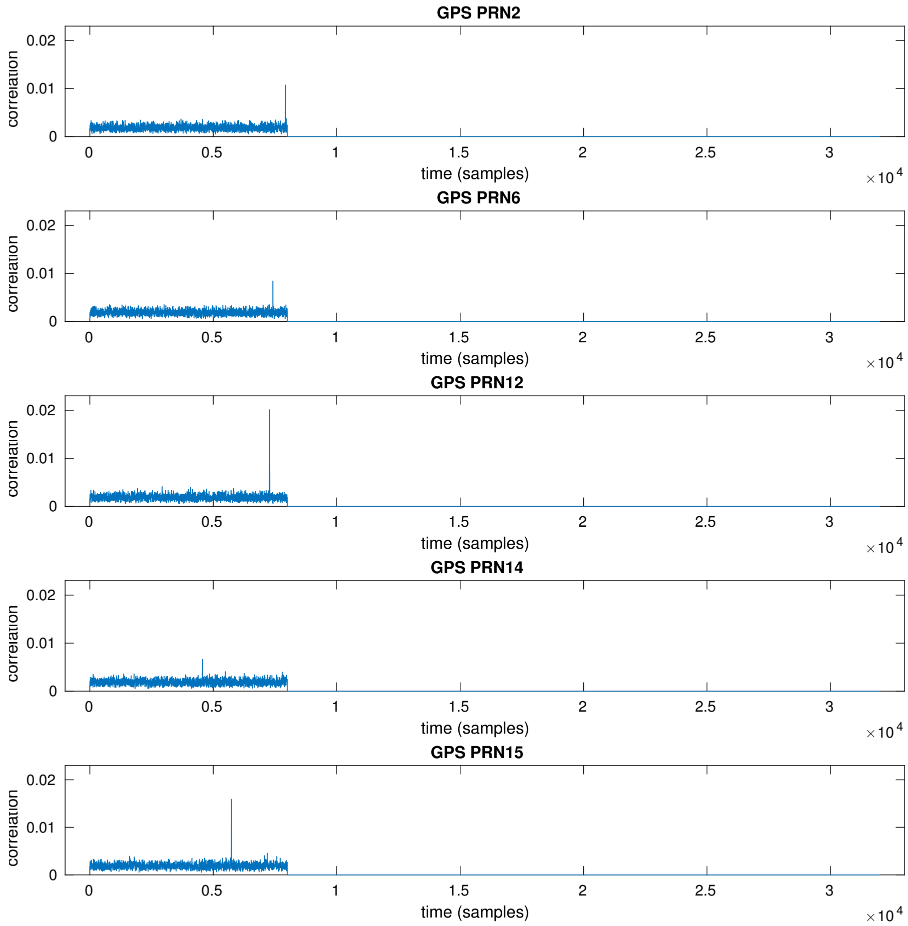


Figure 4.2: Some GPS Correlations with a 4 ms Correlation Window of the Recording named ETZ.

The first performance parameter is the time needed for downloading the newest data while the second parameter is the time needed for parsing the navigation data. The last parameter is the time needed to compute all correlations. This parameter is split in two, the first time is an averaged result of 10 correlation computation times while the second time is the worst case computation time of those 10 measurements. The results can be found in Table 4.2.

The first interesting comparison is between the GPS-only implementation and the GPS part of the Galileo and GPS implementation. Their results are comparable which is as expected. The navigation data download time is much higher for the Galileo system than for the GPS system. This makes sense because as mentioned in Section 3.1.1 there are much more files needed to be downloaded than for GPS. The same is valid for the navigation data parsing because more

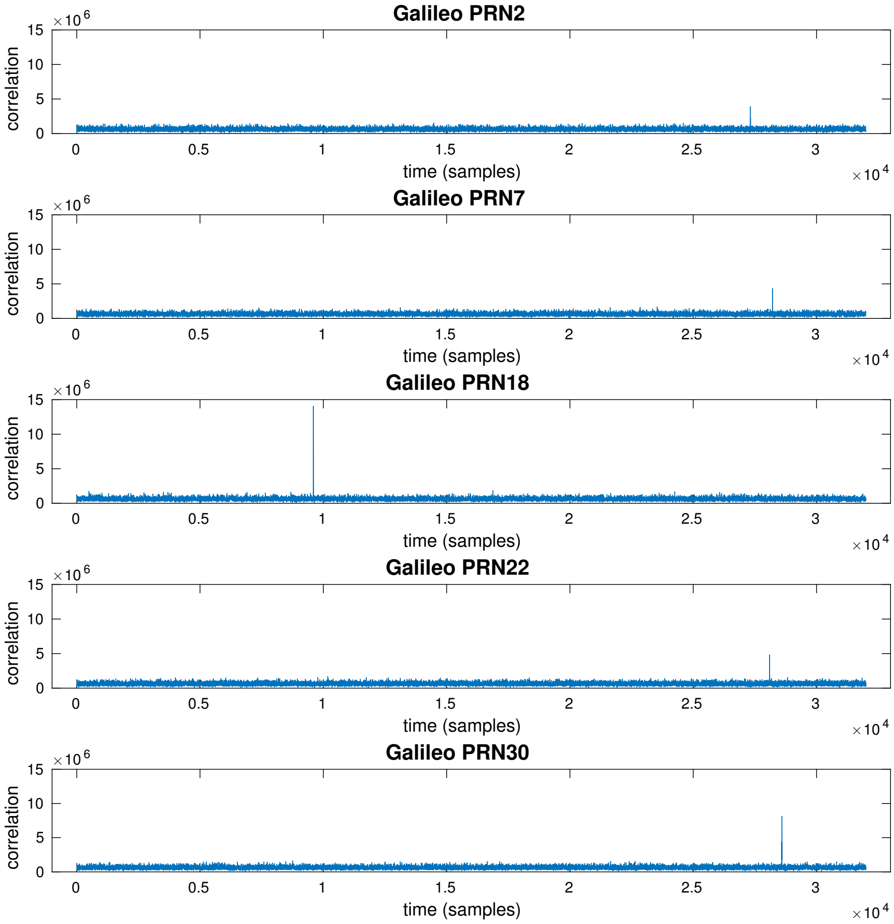


Figure 4.3: All Galileo Correlations with a 12 ms Correlation Window of the Recording named ETZ.

downloaded files result in more files that are needed to be parsed. The correlation computation is also correctly scalable as the correlation length for Galileo is 4 times higher than for GPS as described in Section 3.3.2. Therefore the complexity for the circular cross correlation should raise by a factor of $4 \cdot 4 = 16$ per satellite. As the correlation is computed for 9 GPS satellites and 5 Galileo satellites the ratio per satellite is $\frac{0.4944\,\mathrm{s}}{\frac{0.0624\,\mathrm{s}}{9} * 5} = 14.26$ which comes close to the expected factor of 16.

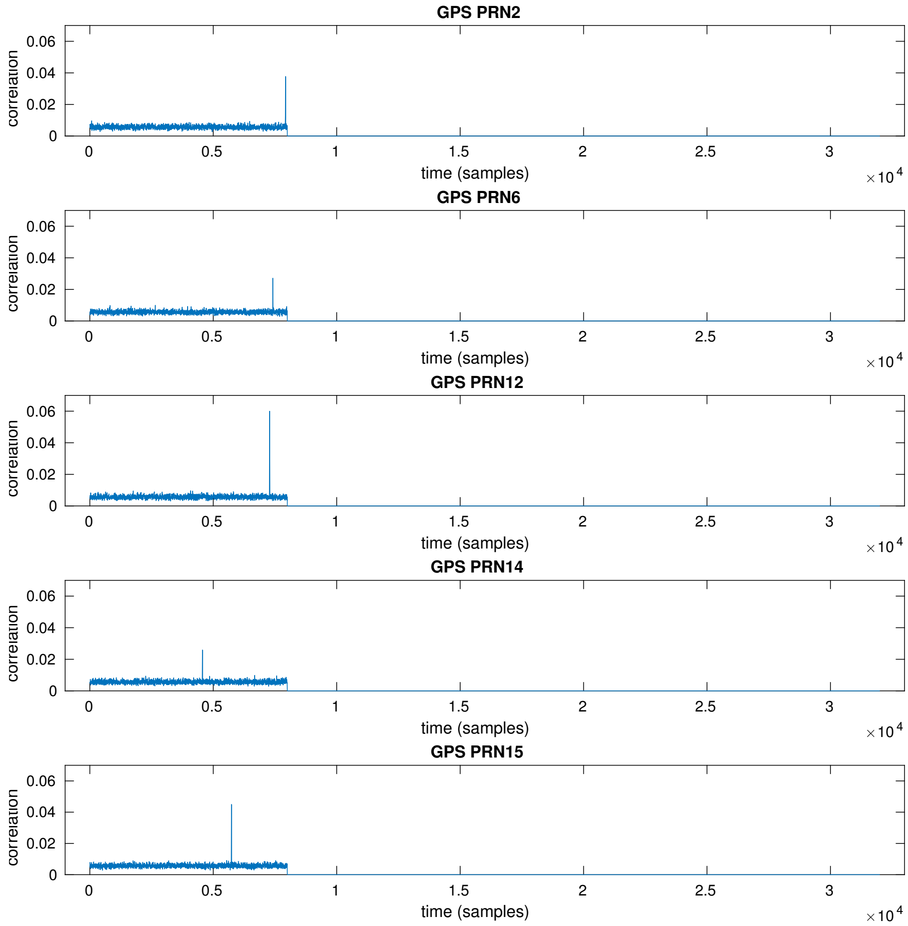


Figure 4.4: Some GPS Correlations with a 12 ms Correlation Window of the Recording named ETZ.

		GPS old	GPS new	Galileo new	GPS + Galileo
Navigation Data Download		4.0862 s	6.8371 s	66.7815 s	73.6186 s
Navigation Data Parsing		3.7620 s	3.3942 s	94.7042 s	98.0984 s
Correlation Computation	*Averaged*	0.0870 s	0.0624 s	0.4944 s	0.5568 s
	Worst Case	0.1051 s	0.0645 s	0.5655 s	0.6190 s

Table 4.2: Performance Comparison between GPS only, Galileo only and the Combined GPS and Galileo Implementation.

Chapter 5

Future Work

As Galileo and GPS are not the only GNSS the implementation could be extended by another GNSS as GLONASS or BeiDou. The more GNSS are evaluated the more satellites are available and the more accurate the position can be computed [2]. However, different frequencies are used by these signals.

The current implementation also has room for improvements as for example find a way to improve the memory needed by the high number of Galileo RINEX files. A way could be to store all files compressed or to summarize all files of a day and merge them into one file without repetitions.

Bibliography

[1] *Satellite Navigation*, Jun. 2017. [Online]. Available: https://en.wikipedia.org/wiki/Satellite_navigation

[2] X. Li, X. Zhang, X. Ren, M. Fritsche, J. Wickert, and H. Schuh, *Precise positioning with current multi-constellation Global Navigation Satellite Systems: GPS, GLONASS, Galileo and BeiDo*, Feb. 2015. [Online]. Available: http://dx.doi.org/10.1038/srep08328

[3] *GNSS Signal*, University of Catalunia, Spain, 2011. [Online]. Available: http://www.navipedia.net/index.php/GNSS_signal

[4] *European GNSS (Galileo) Open Service Signal In Space Interface Control Document*, European Union, Nov. 2015, issue 1.2.

[5] *GNSS Facts*, NavtechGPS, 2017. [Online]. Available: http://www.navtechgps.com/gnss_facts/

[6] *Chip (CDMA)*, Apr. 2017. [Online]. Available: https://en.wikipedia.org/wiki/Chip_(CDMA)

[7] *GNSS-SDR: GNSS Signals*, 2017. [Online]. Available: http://gnss-sdr.org/docs/tutorials/gnss-signals/

[8] *Galileo System*, European GNSS Agency, Jun. 2017. [Online]. Available: https://www.gsc-europa.eu/galileo-gsc-overview/system

[9] *Galileo Constellation Information*, European GNSS Agency, Jun. 2017. [Online]. Available: https://www.gsc-europa.eu/system-status/Constellation-Information

[10] *Galileo Global Navigation Satellite System Mission High Level Definition*, European Commission, European Space Agency, Apr. 2001, issue 2.0.

[11] *GPS Performances*, Navipedia, European Space Agency, 2011. [Online]. Available: http://www.navipedia.net/index.php/GPS_Performances

[12] *GPS Space Segment*, National Coordination Office for Space-Based Positioning, Navigation, and Timing, Jun. 2017. [Online]. Available: http://www.gps.gov/systems/gps/space/

[13] *Services*, European GNSS Agency, Jun. 2017. [Online]. Available: https://www.gsc-europa.eu/galileo-gsc-overview/services

[14] J. A. A. Rodriguez, *Galileo Signal Plan*, University FAF Munich, Germany, 2011. [Online]. Available: http://www.navipedia.net/index.php/Galileo_Signal_Plan

[15] J. A. A. Rodriguez, *Binary Offset Carrier*, University FAF Munich, Germany, 2011. [Online]. Available: http://www.navipedia.net/index.php/Binary_Offset_Carrier_(BOC)

[16] O. Julien, C. Macabiau, J.-L. Issler, and L. Ries, *Galileo E1 OS/SoL acquisition, tracking and data demodulation performances for Civil Aviation*, NAVITEC 2010, 5th ESA Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals, Dec. 2010. [Online]. Available: https://hal-enac.archives-ouvertes.fr/hal-01022204

[17] J. A. Rodriguez, *GPS Signal Plan*, University FAF Munich, Germany, 2011. [Online]. Available: http://www.navipedia.net/index.php/GPS_Signal_Plan

[18] J. Liu, B. Priyantha, T. Hart, H. S. Ramos, A. A. Loureiro, and Q. Wang, "Energy efficient gps sensing with cloud offloading," in *Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems*. ACM, 2012, pp. 85–98.

[19] N. Bergey, *The GPS PRN (Gold Codes)*, Sep. 2014. [Online]. Available: https://natronics.github.io/blag/2014/gps-prn/

[20] *Acquisition (GPS and Galileo Receiver)*, The-Crankshaft Publishing. [Online]. Available: http://what-when-how.com/a-software-defined-gps-and-galileo-receiver/acquisition-gps-and-galileo-receiver-part-1/

[21] *Background of the CDDIS*, Crustal Dynamics Data Information System (CDDIS), Feb. 2014. [Online]. Available: https://cddis.nasa.gov/About/Background.html

[22] *IGS About*, International GNSS Service (IGS). [Online]. Available: http://www.igs.org/about

[23] D. Stenberg *et al.*, "Curl (github library)," 1999-2017. [Online]. Available: https://github.com/curl/curl

[24] R.-S. IGS, "Rinex-the receiver independent exchange format (version 3.03)," 2015.

Appendix A

Debug Informations

This appendix chapter gives informations that are useful for working with the code of this GPS and Galileo implementation.

A.1 Debug: Separate GPS and Galileo Processing

This project is about extending an existing GPS implementation with Galileo. As mentioned in the main part of the report, there are three main parts which are implemented for both GNSS. Those three parts are each split into one function doing the part for GPS and another function doing the part for Galileo. Therefore the code can easily be changed to completely disable one of the systems. The informations about which function-call lines need to be commented out and where to find them are listed in the following, whereby the filepath is starting in the directory *path/to/gnss_me/*.

- **Download Navigation Data**
  In file: *./code/backend/receiver/controller/ShiftMatchingController.cc*
  - fetch_assistance_gps_files_from_server(config)
  - fetch_assistance_galileo_files_from_server(config)
- **Parse Navigation Data**
  In file: *./code/backend/receiver/controller/ShiftMatchingController.cc*
  - parse_and_load_all_gps_data(config)
  - parse_and_load_all_galileo_data(config)
- **Compute Correlations**
  In file: *./code/backend/receiver/shift_matching/ShiftMatcher.cc*
  - generateGpsCorrelationMagnitude(*job)
  - generateGalileoCorrelationMagnitude(*job)

Appendix B

Task Description

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing

Prof. R. Wattenhofer

SA:

Multi-System GNSS Receiver Software

Low-power devices for the Internet of Things (IoT) have very little energy and computation power available. Still, they should provide a lot of functionality to enable "smart" applications. For instance, one would always like to know where such a device is located. We try to tackle this problem using GPS, but with an approach very different from current receivers.

At our group, we developed an algorithm for a low-power GPS receiver. The receiver only samples the signal and then sends the samples to a server, which then does the processing to get the position.

The goal of this project is to implement a GNSS receiver with the capability to receive and use signals from satellites belonging to different GNSS, such as GPS, Galileo and GLONASS. Receiving signals from many satellites instead of only a few allows for greater accuracy and noise tolerance. You will implement these extensions in our existing GPS implementation, and test it with a software defined radio receiver.

Requirements: Creative thinking and familiarity with signal processing are advantageous to successfully work on this topic. The student(s) should be able to work independently!

Interested? Please contact us for more details!

Contact

- Pascal Bissig: pascal.bissig@tik.ee.ethz.ch, ETZ G95
- Manuel Eichelberger: manuel.eichelberger@tik.ee.ethz.ch, ETZ G97

Detailed Project Outline

We denote the following primary tasks mandatory (on the right side you find a rough estimate for the time that we allocate to the respective task):

- Literature research ($\star$)
- Implement a Galileo receiver ($\star\star\star\star$)
- Compare the positioning performance against that of the GPS receiver ($\star\star$)
- Integrate the two receivers to a multi-system GNSS receiver ($\star\star\star$)
- Characterize and evaluate the new multi-system GNSS receiver ($\star\star$)
- Write a report ($\star\star$)
- Present your findings ($\star$)

Extensions

Apart from these requirements, we can think of plenty of ways to extend the project with cool features. Of course, you may add your own ideas to this non-exhaustive enumeration:

- Implement demodulation for additional GPS signals (e.g. L1C or L5)
- Implementation of a third GNSS (GLONASS or COMPASS)
- Investigate possibility of multi-frequency GNSS reception using a software-defined radio