Linköping University

Institute of Technology - Intelligent Transport Systems and Logistics

Low-cost implementation of Differential GPS using Arduino

Master thesis

Martin Svatoň

13. September 2016

Supervisor: David Gundlegård
Examiner: Carl Henrik Häll

LINKÖPING UNIVERSITY

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

Fakulta dopravní

Ústav letecké dopravy

Nízkonákladavoá realizace Diferenciální GPS pomocí systému Arduino

Low-cost implementation of Differential GPS using Arduino

Diplomová práce

Studijní program: Technika a technologie v dopravě a spojích
Studijní obor: Inteligentní dopravní systémy

Vedoucí práce: Ing. Petr Bureš, Ph.D.
David Gundlegård, MSc

Martin Svatoň

CZECH TECHNICAL UNIVERSITY IN PRAGUE

Faculty of Transportation Sciences
Dean's office
Konviktská 20, 110 00 Prague 1, Czech Republic

K620 .... **Department of Transport Telematics**

MASTER'S THESIS ASSIGNMENT

(PROJECT, WORK OF ART)

Student's name and surname (including degrees):

Bc. Martin Svatoň

Code of study programme code and study field of the student:

N 3710 – IS – Intelligent Transport Systems

Theme title (in Czech): **Nízkonákladavoá realizace Diferenciální GPS pomocí systému Arduino**

Theme title (in English): Low cost implementation of Differential GPS using Arduino

Guides for elaboration

During the elaboration of the master's thesis follow the outline below:

- Introduction
- Analysis of satellites based navigation systems principles
- Analysis of available devices for more accurate positioning
- Design of the new concept and components selection
- Putting the new concept into operation
- Testing and applications
- Conclusion

Graphical work range: follows instructions from supervisor

Accompanying report length: at least 55 pages of text (including images, figures and tables, which are part of the report)

Bibliography: TAYLOR, George a Geoff BLEWITT. Intelligent positioning: GIS-GPS unification. Hoboken, NJ: John Wiley, c2006, xvii, 181 p., ISBN 0470850035.

HRDINA, Zdeněk, PÁNEK, Petr, VEJRAŽKA, František. Radiové určování polohy : Družicový systém GPS. 1. dopl. vyd. Praha : Vydavatelství ČVUT, 1999. 259 s. ISBN 80-01-01386-3.

Master's thesis supervisor: **Ing. Petr Bureš, Ph.D.**
David Gundlegård,MSc

Date of master's thesis assignment: **April 22, 2015**
(date of the first assignment of this work, that has be minimum of 10 months before the deadline of the theses submission based on the standard duration of the study)

Date of master's thesis submission: **November 30, 2016**
a) date of first anticipated submission of the thesis based on the standard study duration and the recommended study time schedule
b) in case of postponing the submission of the thesis, next submission date results from the recommended time schedule

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
Fakulta dopravní
L.S.
-3-

....
doc. Ing. Pavel Hrubeš, Ph.D.
head of the Department
of Transport Telematics

....
prof. Dr. Ing. Miroslav Svítek, dr. h. c.
dean of the faculty

I confirm assumption of master's thesis assignment.

Svatoň
....
Bc. Martin Svatoň
Student's name and signature

Prague ....June 1, 2016

Acknowledgment

This thesis was carried out with the Department of Science and Technology at the Linköping University, and Department of Transport Telematics at Czech Technical University in Prague as a final part of the Double Degree study program Intelligent Transport Systems. I would like to thanks to my examiner Carl Henrik Häll (LIU), supervisors David Gundlegård(LIU) and Petr Bureš (CTU) for introductory lectures in GPS positioning and their contributions during the thesis progress.

Special thanks to Donghwan Yoon from Sejong University for providing materials and description about the position-domain projection algorithm.

Declaration

I have no relevant reasons against using this schoolwork in the sense of § 60 of Act No121/2000 concerning the authorial law.

I declare that I accomplished my final thesis by myself and I named all the sources I used in accordance with the guideline about the ethical rules during preparation of University final thesis.

In Norrköping 13/09/16

Svatoň

Martin Svatoň

Abstract

In today's technical dimensions, there are various autonomous gadgets as drones or autonomous lawnmowers, which are widely used by the public to facilitate everyday life. Drones become more and more popular due to mobility and autonomous air operations. The continuous development leads to finding of diverse application for drones such as delivery service or filming.

These autonomous devices are used to navigate themselves and operate without human intervention. It predominantly uses a GPS as a source of position information, which is provided by GPS receiver attached to the device. However, these measurements are not accurate enough to estimate the exact position, which is one of the major requirements for autonomous operation. The main objective of this thesis is to propose the solution for improving the positioning accuracy, which is not expensive and easy to implement.

The Arduino DGPS solution implemented in this thesis proposes the algorithms for position accuracy improvements in two ways. The first mode uses a range residual computed by the receiver to estimate pseudorange corrections (PRCs) and corresponding position correction. The second mode works as an advanced Satellite-Based Augmentation System (SBAS) correction repeater, which uses the SBAS correction to acquire the position correction. Both solutions are implemented to the Arduino and user can choose, which correction method will be used.

Keywords: Differential GPS, Arduino, Low-cost implementation

Abstrakt

V dnešní přetechnizované době je využívano mnoha autonomních zařízení, jako jsou autonomní sekačky či drony, které jsou široce využívány v soukromém sektoru. Drony jsou velmi populární díky své mobilitě a schopnosti autonomních letů. Postupný vývoj těchto zařízení vedl k objevu mnoha aplikací, jako například filmování z ptačí perspektivy nebo autonomní doručování.

Tyto autonomní zařízení se samy navigují a fungují bez uživatelských intervencí. Pro navigaci se převážně využívá systému GPS. Běžně dostupné GPS příjmače však poskytují poziční data s určitou chybou, které je třeba eliminovat pro přesnou navigaci. Hlavním cílem této práce je návrh řešení zpřesňování GPS pozice, které je levné a snadno použitelné.

Navrhované řešení Arduino DGPS využívá dvou principů zpřesňování GPS měření. První mód počítá korekce pseudovzdáleností na základě rozdílu sledované a vypočtené pseudovzdálenosti (range residuals). Druhý mód pracuje na principu opakovače SBAS korekcí, kde se přijímané korekce SBAS aplikují na měřenou pozici, čímž dojde ke zpřesnění GPS pozice. Obě rešení jsou implementována do systému Arduino a uživatel si může vybrat jednu z korekčních metod.

Klíčová slova: Diferenční GPS, Arduino, Nízkonákladová implementace

Table of Contents

List of Tables

List of Figures

List of Abbreviations

BPSK	-	Binary-Phase Shift Keying
CDMA	-	Code Division Multiple Access
CEP	-	Circular Error Probability
DGPS	-	Differential Global Positioning System
DOP	-	Dilution of Precision
DRMS	-	Distance Root Mean Squares
ECEF	-	Earth – Centered Earth-Fixed (coordinates system)
ECI	-	Earth-Centered Inertial (coordinates system)
ENU	-	East North Up (coordinates system)
GALILEO	-	European Global Satellite Navigation System
GLONASS	-	Global Navigation Sputink System
GNSS	-	Global Navigation Satellite System
GPS	-	Global Positioning System
GRI	-	Group Repetition Interval
LEO	-	Low Earth Orbit
LLH	-	Longitude Latitude Height (coordinates system)
LOS	-	Line – Of - Sight
NDB	-	Non – Directional Beacon
PRC	-	Pseudorange Correction
PRN	-	Pseudo-Random Noise
RS	-	Reference Station
RTK	-	Real Time Kinematics
SA	-	Selective Availability
SBAS	-	Satellite Based Augmentation System
SV	-	Space Vehicle
VOR	-	VHF – Omnidirectional Radio Range
ToA	-	Time of Arrival

1 Introduction

1.1 Background

The Global Positioning System (GPS) is today widely used in the public sector for ordinary navigation. Some fields such as traffic or geodesy are using GPS as a primary source of information. The GPS is also used in applications related to newly developing autonomous devices such as lawnmowers or drones. These devices also mainly use GPS as a primary source for navigation. However, the GPS provides an only limited level of accuracy and precision. The autonomous ground-moving vehicles require a high level of accuracy to be able to follow a defined path and do not overpass a defined area. In addition, the flying objects require the same accuracy in vertical domain.

The GPS position augmentation is a method to increase the object position accuracy in terms of removing and decreasing errors caused by the clock biases and signal propagation errors. The GPS position augmentation methods (such as Differential GPS (DGPS), Real Time Kinematics (RTK)) are widely used in professional devices, which provide excellent position accuracy improvements. However, these devices are difficult to implement in private gadgets because of the price.

Due to high costs of professional devices, this thesis proposes a method, which can improve the positioning accuracy even with cheap hardware and easy and understandable algorithms.

The Arduino DGPS device is developed by using several modules, which are available and cheap (the GPS module uBlox 6m and Wi-Fi module ESP8266). The Reference Station (RS) is based on Arduino Due, which processes the data from rover station and computes the corrections. The rover is based on small Arduino Nano, which only collects the necessary data for correction computation, sends them to the RS, and applies received corrected position data to the output. The Arduino DGPS is designed to serve only one rover station by using the dedicated communication channel.

1.2 Aim

The Aim of the thesis is to develop algorithms, which provide the function of Differential GPS suitable for Arduino. Implement these algorithms and prove the functionality.

1.3 Objectives

The algorithms development is fulfilled by analyzing the current situation of satellite-based positioning systems and possible augmentation methods, which are suitable for Arduino implementation. The algorithms are designed to achieve the simplicity and preserve the basic functionalities of general positioning augmentation principals.

To preserve the functionality, the algorithms are secured by integrity monitoring, which adjusts the system setting according to input data.

Arduino Differential GPS accuracy is tested by static and dynamic measurements; which results prove the algorithms effectiveness.

1.4 Scope

The thesis includes a theory chapter, which describes only the functionality of the GPS system since the Arduino DGPS system does not consider the others navigation systems (GALILEO, GLONAS).

The position correction algorithms are based on principals of position domain and code range domain DGPS. The work does not include the implementation of phase range domain DGPS.

The thesis does not include the practical usage of the proposed devices. It is one of the tasks for the future work.

1.5 Related work

The thesis continues the research performed by (Kim, 2013), which defines the primary idea of low-cost DGPS implementation. The Kim's research paper provides the correction solution by using online services and Satellite-Based Augmentation System (SBAS). This thesis proposes the other possible solution based on the pseudorange residuals and SBAS with implementation to the Arduino.

1.6 Outline

Chapter 2 focuses on the theory and historical base of the GPS navigation, its structure and positioning algorithms, used data frames, and reference systems

Chapter 3 describes the positioning augmentation methods, which improves the positioning accuracy

Chapter 4 comprises the desired solution of the low-cost DGPS; define a necessary receiver setting, propose the data transmission and algorithm implementation to the Arduino.

Chapter 5 proposes the hardware design of the Reference Station (RS) and rover station.

Chapter 6 evaluates the measurements and estimates the level of correction for both modes. The chapter also includes the power consumption of the system.

Chapter 7 discusses the results, which were gathered during the design part and measurement (Chapters 4, 5, 6). This part also suggests the future work.

Chapter 8 summarizes the performed study and provides the results overview.

2 Global Navigation Satellite System

Satellite navigation is well known and globally spread positioning service, which founds its use in a variety of applications. The first chapter briefly describes the beginning stages of the satellite-based navigation, predecessors, and detailed description current GNSS in operation with emphasis on Global Positioning System. The methodology of positioning, specific errors that influence the system precision and methods of position augmentation, are also discussed in next Sections.

2.1 GNSS History

It the first Section, History of Global Navigation Satellite Systems will be introduced. GNSS are the successor of ground-based navigation systems such as LORAN, DECCA. The first satellite-based navigation system came in 1960 and primarily used for military purposes. Later, the satellite-based navigation systems became spread over the civil operations, where found its application until today.

2.1.1 Radio based navigation systems

Radio based navigation systems had the main boom in the 20th century. The sense of radio navigation became important during the Second World War. However, fragments of radio based navigation date back to 1907. During this year, the German company Schaller invented a complimentary dot – dash-guiding path. (Bauer, 2004) The Scheller's invention creates a base for future development. The principle is adopted in many applications, which are still in operation (VHF Omnidirectional Radio Range (VOR), Non-Directional Beacon (NDB)). Radio based navigation is used in hand with satellite navigation to create a robust and accurate navigation system.

2.1.1.1 DASH - DOT

The dash-dot system uses two identical crossed antenna loops. Each loop transmits different dash-dot sequence. One uses sequence (**.** -) and the other (**-** **.**), which refer to Morse letters A and N. This principle is known as the A/N system. (Bauer, 2004) For example, in air navigation, the aircraft will hear one of the letters while flying in the area that is covered by one of the antennae. The signal strength will increase by flying towards the antenna and conversely. The A/N system creates a simple audio-guided path for pilots.

2.1.1.2 LORAN

The Long Range Navigation (LORAN) is a big follower of Radio based navigation systems. Development began in 1940 by U. S. National Defense Research Committee (NDRC). The goal of the project was to design a long-range, precise aircraft navigation system. The original concept had accuracy requirements 1000ft and range of 200miles. The LORAN's development continues after the WWII. There were several versions of LORAN with extended range and improved accuracy (LORAN-B, LORAN-C). LORAN-C offered range up to 2500 miles and accuracy up to hundreds of feet. (Bailey, 1962) During the 70's it became popular due to the introduction of semiconductors, which led to the price drop and complexity decrease. LORAN-C was mainly used in marine and air navigation.

Each LORAN base station had one master unit, which was used to transmit the triggering impulses and one or more slave stations. Principally, the LORAN system was based on hyperbolic navigation principle. The master unit sends a set of pulses. After a defined delay, the corresponding slave unit transmits its set of pulses. The receiver has to get a signal from at least two stations to achieve a surface fix. The receiver identifies the signal pulses from the master station and then, it waits for the set of pulses from the slave unit. The time difference between pulses from master and slave station defines a position on the hyperbolic curve, where the time difference is constant. When the receiver gets a signal from three stations, the full position fix is available. It is necessary to know the position of the base station and the distance between master and slave unit. The set of pulses characterizes the station by the period between the pulses from the master station (GRI). Receivers have the database of all stations positions and all corresponding information.

Currently, the LORAN navigation system is discontinued. The development of new satellite system (GPS), forces the LORAN into seclusion. However, the development is not over because the idea to split the satellite navigation and surface navigation systems together and create a robust system is still there (eLoran). Both navigation systems have different properties and weaknesses. For example, the surface navigation system can cover hidden areas, where the GPS connection was difficult to establish because of the high-frequency spectrum. On the other hand, surface navigation systems have limited coverage in compare to GNSS and low-frequency spectrum requires large antennas and consumes an enormous amount of energy.

2.1.2 Doppler-based navigation systems

The Doppler based systems were primarily designed for marine defense network, especially for newly discovered atomic submarines. USA and USSR initiate the development of the new satellites systems independently in 60's. The main goal was to cover remote areas, where the former navigation principles were unavailable (Earth's poles).

The Doppler-based navigation system uses the frequency shift to calculate a relative position on the earth surface. A detailed description of the Doppler position measurement principle is described in Section 2.5.4.

2.1.2.1 TRANSIT

TRANSIT or NAVSAT (Navy Navigation Satellite System) was the first satellite system put into operation by US NAVY in 1964. The TRANSIT consisted of six satellites spread over the polar orbits. The primary usage of the TRANSIT was only for Naval and military purposes. Nevertheless, it became available commercially for civil vessels and planes since 1967.

Since there were only six satellites and they were on the LEO orbits, it required a wide radiation pattern. The satellites were used to cover the defined area on the earth surface, so the user could see only one satellite in time. The orbit period of each satellite is 107 minutes, so the service was not time-continues. The continuity improvement by the polar orbits results in a shorter average waiting time. (Forssell, 2008) Particular satellites were available for up to 30 minutes (depended on the current constellation) and then user had to wait 4-8 hours for another satellite. Each satellite transmits two carriers on 400 MHz and 150 MHz. The second carrier was used to eliminate the errors from the ionosphere.

The receivers were used to measure the Doppler shift of the satellite frequency, which was caused by the satellite movement. The Doppler shift measurement and satellite parameters (ephemeris) are critical values that are used to define a position on the ground. In the years of TRANSIT, there was no computer to depict and calculate the position. A newly developed receiver unit (called AN/UYK-1) was specialized for signal decoding and position calculation. It was using the principle of bit's overflow and sum. Nevertheless, the whole decoding process took up to 16 minutes. Thanks to this calculation interval, the movement of the ship had to be considered. (Stansell, 1978) The system was applied to a range of services such as vessel's localization, hydro and geodetical research, or accurate time service.

Various factors influenced the accuracy of the TRANSIT system. There are two major types of errors in position estimation. "Error in the measurement themselves caused by receiver velocity, noise, propagation variations, and errors in the translation of measurement results into position." (Forssell, 2008) The single channel receiver unit had a high inaccuracy due to ionospheric and tropospheric effects (up to 100m). Usage of dual channel receiver eliminates this vice. The accuracy reached the 18-35m using the dual channel receivers. (Forssell, 2008). However, the dual channel receivers had its largest error from the satellite orbits uncertainties because of the satellite orbit prediction. This error could be reduced by translocation or by long-term evaluation.

The TRANSIT was the trailblazer of the satellite-based navigation systems. It provided higher coverage and accuracy than its predecessor LORAN. It became widely used on many commercial and private vessels, even if it did not allow the continuous positioning service. TRANSIT ceased in 1996. It was in continual operation for 32 years. Now the TRANSIT is replaced by the GPS, which surmounts the TRANSIT in many ways.

2.1.3 Current GNSS situation

In 1973, the US government established the project NAVSTAR – GPS, which started development of the new satellite navigation system. The GPS removes drawbacks of the obsolete TRANSIT by offering higher precision and unlimited signal availability. GPS navigation system is now in full operation, and it is described in Section 2.2. Along the GPS development, Russian Federation started its project of satellite navigation called GLONASS. Later, in 1999 European Union initiated its project GALILEO. It is the first project in the chain, which is not developed primarily for military purposes.

2.2 GPS

The GPS (Global Positioning system) also called NAVSTAR is the most common satellite navigation system today. It is used in a variety of applications from mobile devices up to precise navigation systems used in civil and military aviation.

The GPS development is divided into several phases.

1. (1973 – 1979) In the first phase, the basic principal of satellite positioning was investigated and proved by the release of four spacecraft (Block-I), which provide 3D navigation for specified time of a day. Later, eleven spacecraft were launched. Each of them was equipped with three atomic clocks.
2. (1979 – 1985) The second phase represents the development and launch of 28 satellites (Block-II). Each of them was equipped with four atomic clocks, and it can operate up to 14 days without correction from the control center. (Slavíček, 2007). During the second phase, the development of the control segments begun.
3. (1985-1994) Development and launching of Block-II and Block IIA satellites have progressively replaced the Block-I satellites and in 1995, the system reached the FOC (Full Operational Capability). The GPS was composed by 24 operational satellites (Block II and BlockIIA), which assured that anytime and anywhere on the earth, at least, four satellites were visible, and the 3D navigational fix was possible to establish.

2.2.1 System composition

The GPS is composed of three segments similarly as the TRANSIT. The Cosmic segment consists of the spacecraft, Control segment consists of control stations located at the different places on the earth, and finally, the user segment formed by the range of different users and receivers.

2.2.1.1 Cosmic Segment

The satellite development and modernization are still in progress, and now there are 32 satellites in six orbits with an inclination of 55°. The nominal altitude is 20 183km, and one orbit takes a half of sidereal day (11hours and 58 minutes)

- 12 – satellites Block IIR
- 7 – satellites Block IIR(M)
- 12 – satellites Block IIF

Block IIF is currently the most modern satellite generation available. The satellites have extended life span (up to 12 years) and provide the third civil signal L5 for Safety-of-Life applications. They have a better shielding from the cosmic rays and all of them are equipped with advanced rubidium atomic clock. From the generation of Block IIR, the satellites can communicate with each other, which makes easier debugging in case of some failure. The last satellite was launched in February 2016.

Since the spacecraft have a limited lifetime and to satisfy the requirements of the new age, there is a new generation of GPS satellites called GPSIII in development, which launch is expected during the year 2016. (U.S. Air Force, n.d.)

2.2.1.2 Control segment

The GPS control segment consists of several stations, which continuously monitor the satellites, adjust the orbital trajectory; update the navigation data, and so on. “The current operational control segment includes a master control station, a backup master control station, 12 command and monitoring antennas and 16 monitoring sites" (U.S.A.F, n.d.).

Master Control Station (MCS) is located in Colorado US. It receives and evaluates the information from the monitoring stations spread over the world. Master control is responsible for generating and uploading the Navigation messages to the satellites

Monitoring Stations are responsible for surveillance of the satellites in their view and transmit the acquired data to the MCS. It also monitors the current atmospheric and ionospheric behavior and forwards them to the MCS.

Ground Antennas can communicate with the satellite in full duplex. Ground antennas are responsible for sending and receiving commands and data to/from the satellite.

2.2.1.3 User Segment

Receivers create the user segment, which can be very varied according to its purpose. The receiver's antenna is tuned to receive one of the carrier (L1, L2) or both in case dual band receiver. The advanced receiver can have the Safe-of-Live option, which requires the transmission on L5.

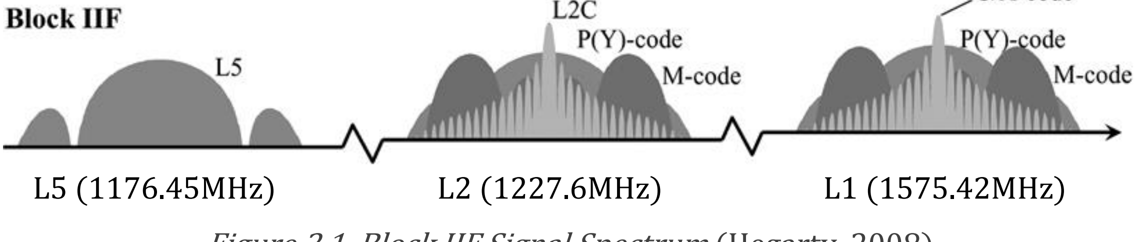
2.2.2 GPS signals

The latest GPS satellites transmit on two carriers frequency band L1 (1 575.42MHz) and L2 (1 227.6MHz). These carriers are modulated by code sequences (BPSK), which provide a satellite clock reading and orbital parameters (navigation message) to the receiver.

Table 1. Block IIF Codes Overview

Band	Code	Availability	Code frequency
L1	**C/A** (Coarse/Acquisition)	Civil	1.023 MHZ
L1/L2	**P(Y)** (Precise)	Authorized	10.23MHz
L2	**L2CM**	Civil	511.5kHz
L2	**L2CL**	Civil	511.5kHz
L1/L2	**M** (Military)	Authorized	5.115 MHz

L2CM/CL is an innovative signal available for civilian users. L2C contain two distinct PRNs unlike the L1 C/A (CL civil-long and CM civil-moderate). CM is modulated by the navigation message whereas; CL is not modulated (codeless). L2C allows dual band receiver effectively remove the ionospheric and tropospheric refraction in code ranging domain. However, this signal is carried only by current Block IIF, and the ability to receive these subcarriers requires special hardware. Simply stated, low-cost receivers are not able to access L2 frequency band let alone L2CM code. The most of the low-cost receivers are a single band, which means that support only L1 band and C/A code ranging.



L5 (1176.45MHz) L2 (1227.6MHz) L1 (1575.42MHz)

Figure 2.1. Block IIF Signal Spectrum (Hegarty, 2008)

The PRN (Pseudo Random Noise) sequence characterizes each code. It is a random code sequence with a unique pattern for each satellite. PRN allows the satellites transmit the different information on the same frequency (CDMA). It helps to identify the right message from the signal spread spectrum and pick up the needed data broadcasted by particular SVs at the receiver.

2.2.3 Code measurement determination

The GPS satellite transmits L1 carrier modulated by C/A PRN. The receiver side is used to demodulate the L1 carrier, so only coded useful signal remains. The receiver knows the PRN sequence for all SVs, which enables them to generate the same signal pattern for particular SVs. The signal produced by the receiver is not synchronized with the GPS time because the reference time is generated by the cheap crystal clock and the received signal from the satellite is shifted from the GPS time by its transmission time and satellite clock bias. The receiver continually splits the generated PRN and received PRNs. To synchronize both signals, the receiver shifts its generated signal by a defined time interval and comparison of the both signals results in the correlation coefficient. When the correlation coefficient is equal to 1, the synchronization was established, which results in subtracting the both (generated and received C/A PRN). If the both C/A PRN are eliminated, the receiver will provide only C/A PRN carrier modulated by the navigation message. This signal is simply demodulated, and raw navigation message is now ready for next processing.

From the navigation message and C/A PRN is possible to determine the time of transmission of the C/A PRN sequence. The difference between the time of transmission and time when the C/A sequence was received is called ToA (Time of Arrival). When the ToA is multiplied by the propagation speed (speed of light), it results into Pseudorange (Section2.5.1).

The minimal resolution of ToA is equal to the period of the C/A carrier (1.023MHz), which is approximately 1us.

2.2.4 GPS access regulation

The GPS navigation system was primarily developed for military purposes. When the GPS was released to the civil sector, it was decided to apply the limitation to the public due to national security reasons. The most common features are Selective Availability and Anti-Spoofing.

2.2.4.1 Selective Availability SA

Selective Availability is a process of applying errors into GPS signals and degradation of positioning accuracy. SA errors are randomly generated, so the prediction and mitigation are almost impossible. There are two general methods of applying the signal fluctuations. (Rapant, 2002)

- Introduce the SA error into the ephemeris, which causes the error in satellite position estimation and it makes difficult to cancel the satellite clock bias.
- Introduce the SA error into the satellite clock frequency

The SA could be suppressed in the receiver when the authorized user had the special code, which mitigates the error caused by the SA.

The second method (influence of the satellite clock) could be effectively removed by using the differential GPS (DGPS). As stated in Equation (3.8) the satellite clock bias is completely removed.

In May 2000 was decided to turn the SA off and the positioning error was improved from 100m to 20-30m horizontally and from 150 to 10-45 m vertically. (The Department of Energy, 1995)

2.2.4.2 Anti-Spoofing

In the case of war or other exceptional events, the US can use the method of anti-spoofing. It is due to reason; if attacker broadcasts a fake GPS signal to confuse the authority. To prevent this type of attack, the encryption is added to the P-code (Anti-spoofing is related only to P-code). While the P-code is encrypted, it is called Y-code. The encryption disables the possibility of replacing the GPS signal by the fake one. Encrypted signal is also unavailable for authorized users, who are allowed to use P-code.

2.3 GPS DATA

This Section describes the data protocols used in different stages in the GPS and DGPS navigation.

2.3.1 RAW Navigation messages

The raw navigation message means the protocol used for communication between satellites and receivers. Each satellite transmits on some defined carrier (L1, L2), which is modulated by the useful signal. These useful data are called the Navigation Message (NAV). The NAV is divided and transmitted in 25 frames, where each frame carries five subframes. Transmission of the whole L1 NAV takes 12.5 minutes with 50bps rate. (Subirana, GPS Navigation Message, 2011) .

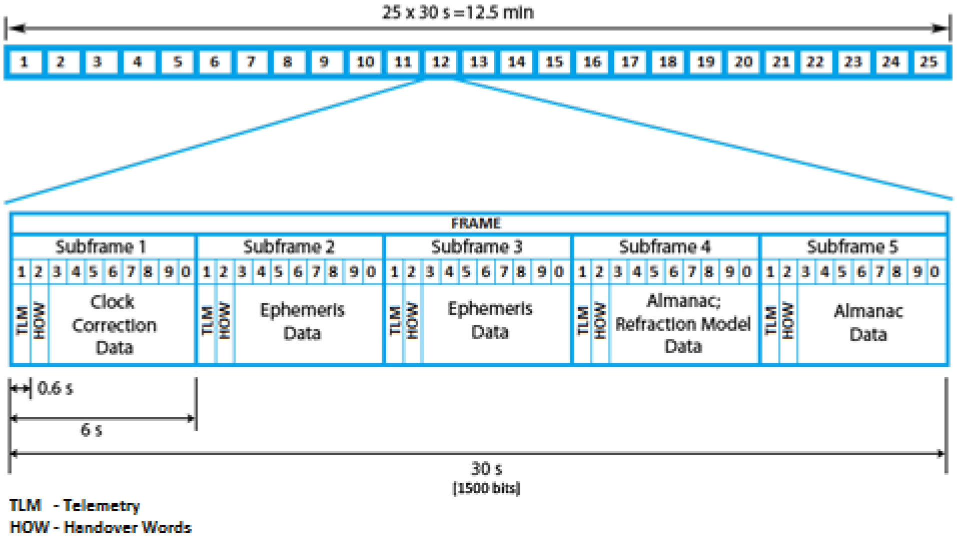
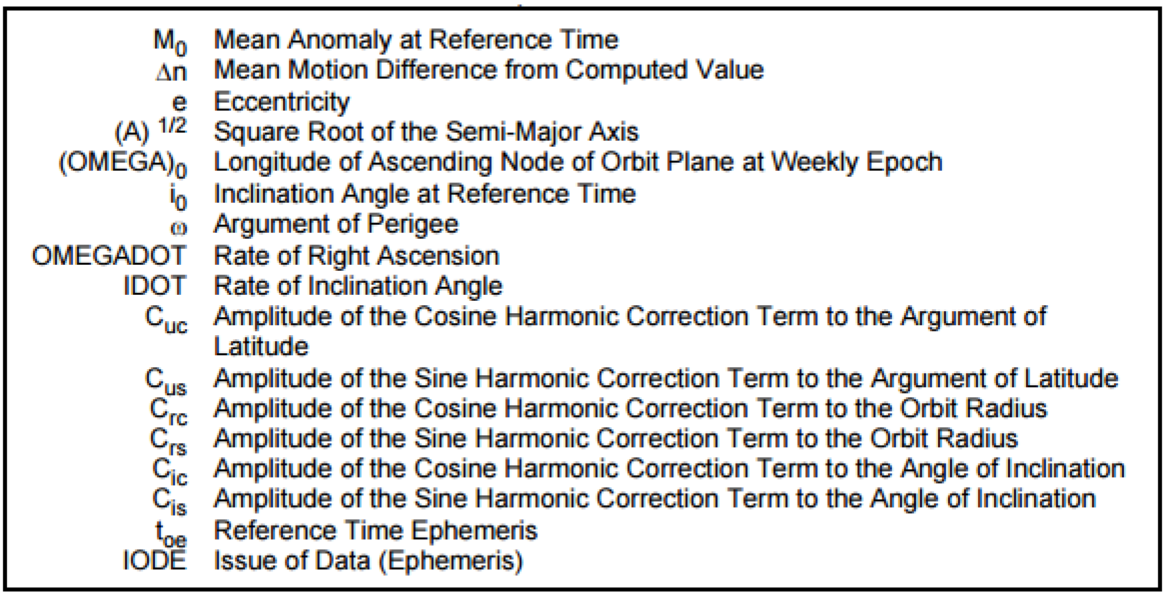


Figure 2.2. Structure of the L1 NAV subframe (Instruments, 2016)

Figure 2.2 shows the structure of the navigation message subframe. Each subframe contains the TLM and HOW words, which are used for subframe identification.

2.3.1.1 Ephemeris

Ephemeris data are used to estimate the satellite position concerning orbital parameters, which are carried in the second and third sub-frame of the navigation message. It includes the six traditional Keplerian parameters and perturbation parameters. Figure 2.3 sorts all the ephemeris parameter with description.



M_0	Mean Anomaly at Reference Time
Δn	Mean Motion Difference from Computed Value
e	Eccentricity
$(A)^{1/2}$	Square Root of the Semi-Major Axis
$(OMEGA)_0$	Longitude of Ascending Node of Orbit Plane at Weekly Epoch
i_0	Inclination Angle at Reference Time
ω	Argument of Perigee
OMEGADOT	Rate of Right Ascension
IDOT	Rate of Inclination Angle
C_{uc}	Amplitude of the Cosine Harmonic Correction Term to the Argument of Latitude
C_{us}	Amplitude of the Sine Harmonic Correction Term to the Argument of Latitude
C_{rc}	Amplitude of the Cosine Harmonic Correction Term to the Orbit Radius
C_{rs}	Amplitude of the Sine Harmonic Correction Term to the Orbit Radius
C_{ic}	Amplitude of the Cosine Harmonic Correction Term to the Angle of Inclination
C_{is}	Amplitude of the Sine Harmonic Correction Term to the Angle of Inclination
t_{oe}	Reference Time Ephemeris
IODE	Issue of Data (Ephemeris)

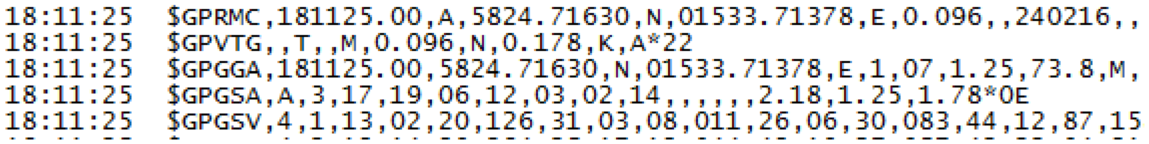
Figure 2.3. Ephemeris parameters (NAVSTAR, 1995)

These parameters are used for satellite position determination, which is needed to know for estimation of LOS vector and terrestrial position (Section 4.2.1.1.).

2.3.2 NMEA

Today, there are plenty of different receiver types, developed by various companies. To achieve the compatibility of devices and applications using the GPS data, the unified protocol needs to be used. This what the NMEA is. It is an unified protocol, which provides GPS information in human readable (ASCII) format.

Each NMEA sentence begins with *"$"*. It identifies the beginning of each sentence. This is followed by a code word, which identifies the navigational system and sentence type. The following figure shows the format of the NMEA0183.



```
18:11:25  $GPRMC,181125.00,A,5824.71630,N,01533.71378,E,0.096,,240216,,
18:11:25  $GPVTG,,T,,M,0.096,N,0.178,K,A*22
18:11:25  $GPGGA,181125.00,5824.71630,N,01533.71378,E,1,07,1.25,73.8,M,
18:11:25  $GPGSA,A,3,17,19,06,12,03,02,14,,,,,,2.18,1.25,1.78*0E
18:11:25  $GPGSV,4,1,13,02,20,126,31,03,08,011,26,06,30,083,44,12,87,15
```

Figure 2.4. NMEA example captured by U-center

The first two characters identify the navigation system *"GP"* states for GPS and e.g. *"GL"* states for GLONASS. The next three characters are used for sentence type ident. Bellow, the most common sentences are described.

- RMC – (Recommended minimum specific) contains the basic position coordinates, speed (SOG), course (COG), etc.
- GGA – (Global Positioning System Fix Data) are used for basic navigation purposes and time synchronization
- GSA – (GPS DOP and active satellites) shows the satellites used for navigation solution

The character *"*"* followed by control checksum terminates each sentence.

2.3.3 RTCM

RTCM or Radio Technical Commission for Maritime Services defines standards for correction data transmission used by the DGPS since 1985 (Hofmann-Wellenhof, 2001). This standard is widely used by commercial devices, which are used mainly for marine navigation. Of course, this is not only one standard used for PRC data transmission, but many receiver manufacturers develop its own data formats.

The current RTCM standard for DGPS is RTCM10402.3 ver. 2.3. It provides correction data divided into several message types. Each message begins with header and terminates with six parity bits.

Since the main idea of the RTCM is to broadcast the correction data to all connected stations, it will not be used in this project as described in Chapter 4.

2.3.4 UBX

UBlox receivers use the UBX frames. It uses the binary representation of the broad range of data types (position, time, used satellites).

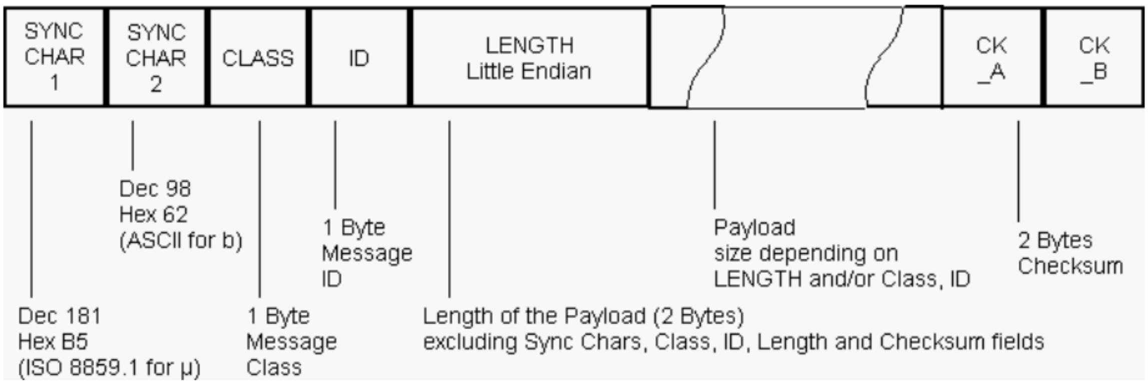


Figure 2.5. UBX packet format (uBlox, 2013)

Each binary sentence begins with two bytes, which identifies the UBX sentence (0xB5, 0x62). After that, another byte denotes the message class (navigational, configuration, monitoring) and the fourth byte denotes the message ID. This is followed by a payload, which begins with length and the whole frame ends with two checksum bytes.

2.3.5 RINEX

The next in line is the Receiver Independent Exchange Format. It was developed due many different kinds of receiver data formats (such as UBX or NMEA) to preserve a compatibility between software applications.

The RINEX is an ASCII formatted data, and the latest version 3.01 consists of three different files. Each file includes a header, which carries the RINEX version and global definition of the contained data followed by the main content. (Gurtner, 2007):

- The Observation file – contains raw observables such as pseudorange or Doppler shift and
- The Navigation file – is used to carry the data from GPS NAV such as ephemeris, satellite time correction coefficients
- The meteorological archive – is connected to specifically observed site and provides a weather information for atmospheric error modeling (temperature, wind)

For GPS data processing, the RINEX data structure is the most favorable. A wide range of software processing tools uses RINEX, even the professional ones.

2.4 Reference system

To be able to locate the receiver on the Earth, the unified reference system has to be used for all participants. The reference model specifies parameters, which are used for next computation. All GNSS uses two types of reference systems.

- Space – fixed
  - Also called inertial, is used for describing the satellite motion (Ephemeris)
- Earth – fixed
  - Also known as terrestrial, is used to represents the observer position

Position in the earth's system can be expressed by the Cartesian coordinates (X, Y, Z). There is a convention, which defines the axes of the Cartesian Coordinates System to achieve unity of all terrestrial (ECEF) and Inertial (ECI) reference frames.

ECEF:

- The origin of the coordinate system is equal to the Earth geocenter.
- X axis is associated with the mean Greenwich or IERS Meridian [1]and The Equator
- Z axis is related to the axis of the Earth rotation
- Y axis is orthogonal to X and Z

ECI:

- Z axis is the same as for the terrestrial reference system
- X axis points towards the vernal equinox (ascending node of the Sun)
- Y axis is orthogonal to both X and Z

The difference between the Terrestrial and Space-fixed is that Space-fixed reference does not move with the earth rotation, whereas the terrestrial reference system is fixed to the earth meridian. Thus, it will change its position with the earth.

There are many reference frames used by different GNSS system. The GPS uses the WGS84, GLONASS PZ-90, and GALILEO uses the GTRF terrestrial reference frame.

2.4.1 WGS84

World Geodetic System 1984 is 3-dimensional, terrestrial, global reference frame utilized by the GPS. It is based on a set of constant variable, which defines the Earth-size, shape, and motion. The earth center of mass defines the origin, and it uses the IERS meridian for defining the X-axis.

The form of the Earth defines the reference ellipsoid, which is defined by following parameters.

[1] IERS meridian is the international reference prime meridian (IRM) established by International Earth Rotation and Reference Systems Service.

Table 2. WGS84 ellipsoid parameters (Hofmann-Wellenhof, 2001)

Parameter	Description
a = 6 378 137.0m	Semi-major axis of the ellipsoid
f = 1/298.257 223 563	Flattening of the ellipsoid
ω_e = 7 292 115 x 10^{-11} rad/s	Angular velocity of the earth
μ = 3 986 004.418 10^8 m^3/s^2	Earth's gravitational constant

The WGS84 currently uses the Earth gravitational model EGM96 revised in 2004, which define a nominal sea level (geoid) (CDDIS-NASA). For a position computation, Cartesian coordinates are used. However, for navigational purposes, the geodetical (LLH) set of coordinates is more representative and human readable. It provides a direct view of the actual position on the earth surface (Latitude, Longitude) and Height. GPS defines the height as a perpendicular height above the ellipsoid (HAE). In most cases of navigation application, the height is represented by the altitude, which is a height above the geoid (MSL). The difference between HAE and MSL results in the geoid separation *N*. The values of geoid separation *N* ranges from -105 to 85m (NGA, n.d.).

$$N = HAE - MSL \tag{2.1}$$

The fidelity of the estimated position concerning the used reference frame will always be limited since the real Earth shape will never fit the simplified mathematical representation. Simultaneously, the Earth shape change at the time. Due to this, the reference frames have to be periodically updated. One of the limiting factors is the sea level change, which can be up to 0.7mm per year (NAP, 2010). The fact of the Earth's environmental changes led to the definition of two parameters.

- Stability – refers to the ability of prediction of the frame parameters
- Drift – refers to relative diversions between different frames

2.4.2 Positioning quality

The estimated position in the reference plane will generate specific systematic or random errors. Two terms define the representation of the mistakes.

2.4.2.1 Precision

Defines how much the particular measurement differs from each other. The precision is determined by the standard deviation σ, which is defined for each direction in vertical and horizontal plane.

$$\sigma_{1D} = \sqrt{\frac{\sum_{i=1}^{n}(x_i - x_t)^2}{n}} \tag{2.2}$$

Where x_t represents a sample mean, and 1D denotes the measurement in one dimension.

2.4.2.2 Accuracy

Defines how the position measurement differs from the reference (true) position. The positioning accuracy can also be described by the standard deviation, at this time with respect to true position x_t (2.2). The accuracy can be defined by different standards that were developed for different purposes of mapping. Accuracy is often related to some level of confidence. The confidence level defines that a percentage of measured points are laying in the circle (sphere), which radius is defined by the maximal deviation.

The deviations from each direction can be merged using the root mean squares (DRMS). Distance Root Mean Squares defines a single number, which represents the accuracy in two or three-dimensional space.

$$DRMS = \sqrt{\sigma_x{}^2 + \sigma_y{}^2} \tag{2.3}$$

2D accuracy can also be described by Circular Error Probability (CEP). CEP concept defines the level of confidence to 50%.

$$CEP = 0.62\sigma_x + 0.56\sigma_y \tag{2.4}$$

The Circular Error Probability is the most common parameter, which is presented by the GPS receiver's developers because of the low confidence level, which results in small error values.

Table 3. Accuracy standards and confidence levels

Accuracy standard	Confidence level
$1\sigma_{1D}$	68.27%
$2\sigma_{1D}$	95.45%
$3\sigma_{1D}$	99.73%
$DRMS_{2D}$	65%
$2DRMS_{2D}$	95%
$3DRMS_{2D}$	97.5%
CEP	50%

2.5 Positioning method

In this Section, the core range positioning method will be described together with methodologies of range measurement.

2.5.1 Pseudorange Code measurement

The code measurement represents the range measurement between satellite and receiver based on the code-word frequency.

When the C/A code word is received, the receiver proceeds to calculate the ToA between current receiver time (t_R) and satellite time carried by C/A code word (t_S). The spacecraft uses an atomic clock, so the time is set very precisely, whereas the receiver is commonly equipped with cheap crystal clock. The GPS uses its own time to achieve homogeneity of the whole GPS navigation system. Each satellite and receiver clock can slightly differ from the GPS time. This difference is denoted as satellite clock bias (δ_S) and receiver clock bias (δ_R).

$$\Delta t = t_r - t_s = \Delta t(GPS) + \Delta\delta \tag{2.5}$$

Where,

$$\Delta\delta = \delta_R - \delta_S \tag{2.6}$$

The satellite clock correction is included in the navigation message. It is automatically applied to the range measurement. Thus,

$$\Delta\delta = \delta_R \tag{2.7}$$

The propagation of a signal in space is defined by the speed of light *c*. Multiplication of the time difference and speed of light gives the range between the satellite and the receiver. This range is denoted as pseudorange (R).

$$R = c\,\Delta t = c\,\Delta t(GPS) + c\,\Delta\delta \tag{2.8}$$

The precision of the pseudorange measurement depends on the frequency of the ranging code. As mentioned, the C/A code frequency is about 1 MHz, which corresponds to a wavelength of approximately 300m. It was proved that precision is equivalent to 1-2% of the wavelength (Rapant, 2002). Thus, the wavelength of the C/A code corresponds to some 3m error. For the P-code, the frequency is 10.23MHz, which correspond to 30m long wavelength and 0.3m error. These values involve only the computation influence. The environmental errors such as ionospheric, tropospheric are not considered. See a summary of the code and phase method in Table 4.

2.5.2 Pseudorange Phase measurement

The measurement of the carrier phase of the GPS signal can be used to pseudorange estimation. It works on a different principle than range code measurement. The carrier phase measurement is based on counting the number of wavelengths between the satellite and receiver. The complete wavelength cycle is denoted as *(N).* The number of wave cycles between satellite and receiver is hard to estimate. For this reason, the (*N*) is called ambiguity. A total number of wavelengths between satellite and receiver is not composed only of the whole cycles, but the final cycle is only the fraction of the total length. This partial length, the receiver can estimate very precisely. The principal of the phase measurement is shown in Figure 2.6.

There are several methods how to estimate the ambiguities e.g. (Teunissen's LAMBDA, on the fly (OTF)). Once the initial value of ambiguity is determined, the receiver can continually follow the changes of the phase shift, which is projected to the position. The ambiguity number represent an initial reference from which, the whole wavelengths plus the fractional part is added or subtracted according to the position change. Mathematical representation is described by (Hofmann-Wellenhof, 2001).

$$\phi_r^j(t) = \frac{1}{\lambda}\rho_r^j(t) + N_r^j + f^j\Delta\rho_{RS}^j(t_0) \tag{2.9}$$

Where $\phi_r^j(t)$ characterize cycles of the measured carrier phase between the receiver r and satellite *j*. λ is the wavelength, and f^j is the carrier frequency. The ambiguity is denoted as N_r^j defined by integer numbers. As same as in the code range model, the $\rho_r^j(t)$ represents the geometric distance receiver – satellite and $\Delta\rho_{RS}^j(t_0)$ represents combined receiver and satellite clock bias.

The carrier frequency of the GPS L1 is 1575.42MHz corresponds to about 19cm long wavelength (Hofmann-Wellenhof, 2001). Comparing with code-based measurement, the wavelength between the code words is much longer (300m). If the rule that precision is equivalent to 1-2% of the wavelength is applied, the accuracy of the phase measurement is on millimeter level. This method is mainly used in geodetic applications, where the precise position measurement is required.

The drawback of this solution is when the signal is interrupted e.g. (passage under the bridge). The measurement has to start again from the beginning because the previous ambiguity is lost due to phase jump. This phase jump is also called “cycle slip.”

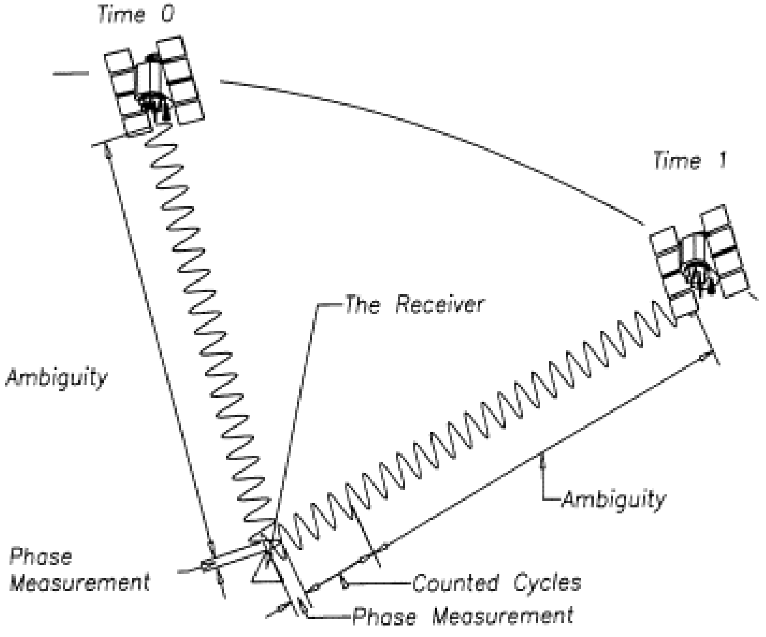


Figure 2.6 Measurement of Carrier Phase (NPTEL, n.d.)

The position calculation is based on the pseudoranges and known satellite position. Therefore, the resolution of the pseudorange measurement will influence the positioning error. The summary of the pseudorange measurement methods and their resolutions are shown in Table 4.

Table 4. Code and phase PR measurement

	Frequency [MHz]	Wavelength	Resolution (1%λ)
Code PR	C/A 1.023	≅ 293 m	≅ 2.93 m
	P 10.23	≅ 29 m	≅ 0.29 m
Phase PR	L1 1 575.42	≅ 19 cm	≅ 1.9 mm

Positioning error now comprises only the calculation precision, but each observation technique is additionally affected by other undesirable effects (Section 2.6).

2.5.3 Lateration

Lateration is a method of position calculation based on the range observables. While measuring the range, the method to get the position is called Circular lateration. The system of computation relies on solving j non-linear equations (2.10), where j denotes the number of satellite, which range is observed. To obtain a full 3D fix (X, Y, Z), at least, three ranges must be observed. This method is called trilateration. However, the GPS range measurement brings some difficulties with a receiver clock bias (described in Section 2.5.1) and other errors described in Section 2.6 thus; the range is denoted as Pseudorange (Equation (2.8).

The receiver clock bias is the next variable, so another equation (range observable) has to be added to solve the GPS positioning problem.

$$R^j(x,y,z) = c\,\Delta t^j(GPS) + c\,\Delta\delta = \rho^j + \Delta\rho_r^j + \Delta\rho^j + \Delta\rho_r \tag{2.10}$$

Where,

$$\rho^j = \sqrt{(X^j - x)^2 + (Y^j - y)^2 + (Z^j - z)^2} \tag{2.11}$$

- $X^j\ Y^j\ Z^j$ – are ECEF coordinates of the j-th Satellite
- $x\ , y, z$ – are ECEF position coordinates to be solved
- ρ^j – is a geometrical range between the receiver and j-th satellite
- $\Delta\rho_r^j$ – is a range bias, which depends on satellite position (ephemeris, ionospheric and tropospheric refraction effects)
- $\Delta\rho^j$ – is a satellite clock error, which is equivalent to $-c\ \Delta\delta_s$.
- $\Delta\rho_r$ – is a receiver r clock bias plus addition receiver based range error sources (multipath). It is equivalent to $c\ \Delta\delta_r + M$

The position can be solved iteratively by using the least squares method. To use least squares, the non-linear terms in equation (2.10) must be linearized. It is done by substituting the unknown receiver position with random position approximation (X_{tr}, Y_{tr}, Z_{tr}) and a correction vector $(\Delta x_{tr}, \Delta y_{tr}, \Delta z_{tr})$, which iteratively adjust the receiver position (Equation (2.12) until there is no change between the iterations (in most cases, at least three iterations are required). The *tr* denotes the number of iteration.

$$\begin{aligned} x_{tr+1} &= x_{tr} + \Delta x_{tr} \\ y_{tr+1} &= y_{tr} + \Delta y_{tr} \\ z_{tr+1} &= z_{tr} + \Delta z_{tr} \end{aligned} \tag{2.12}$$

Now the range equation can be expressed by the Taylor expansion. For the position fix, only first order terms are used.

$$R^j(x_{tr+1}, y_{tr+1}, z_{tr+1}) = R^j(x_{tr}, y_{tr}, z_{tr}) + \frac{\partial R^j(x_{tr}, y_{tr}, z_{tr})}{\partial x_{tr}}\Delta x_{tr} + \frac{\partial R^j(x_{tr}, y_{tr}, z_{tr})}{\partial y_{tr}}\Delta y_{tr} + \frac{\partial R^j(x_{tr}, y_{tr}, z_{tr})}{\partial z_{tr}}\Delta z_{tr}$$

Where partial derivatives are equal to

$$\begin{aligned} \frac{\partial R^j(x_{tr}, y_{tr}, z_{tr})}{\partial x_{tr}} &= -\frac{X^j - x_{tr}}{\rho_{tr}^i} = a_j \\ \frac{\partial R^j(x_{tr}, y_{tr}, z_{tr})}{\partial y_{tr}} &= -\frac{Y^j - y_{tr}}{\rho_{tr}^i} = b_j \\ \frac{\partial R^j(x_{tr}, y_{tr}, z_{tr})}{\partial z_{tr}} &= -\frac{Z^j - z_{tr}}{\rho_{tr}^i} = c_j \end{aligned} \tag{2.13}$$

And $\rho_{i,tr}$ is a geometrical range with respect to actual position approximation

$$\rho_{tr}^j = \sqrt{(X^j - x_{tr})^2 + (Y^j - y_{tr})^2 + (Z^j - z_{tr})^2}$$

The resulting linearized pseudorange equation is

$$R^j_{tr} = \rho^j_{r,tr} + \Delta\rho^j_{r,tr} - c\,\Delta\delta_{s,tr} + c\,\Delta\delta_{r,tr} + M_{tr} + a_j\Delta x_{tr} + b_j\Delta y_{tr} + c_j\Delta z_{tr} \tag{2.14}$$

In matrix notation, the range equation can be rewritten as follows:

$$\begin{bmatrix} -\frac{X^j - x_{tr}}{\rho^i_{tr}} & -\frac{Y^j - y_{tr}}{\rho^i_{tr}} & -\frac{Z^j - z_{tr}}{\rho^i_{tr}} & 1 \end{bmatrix} \begin{bmatrix} \Delta x_{tr} \\ \Delta y_{tr} \\ \Delta z_{tr} \\ c\,\Delta\delta_{r,tr} \end{bmatrix} = R^j_{tr} - \rho^j_{r,tr} - \Delta\rho^j_{r,tr} + c\,\Delta\delta_{s,tr} - M_{tr} \tag{2.15}$$

The yellow part of the equation is often called as a design or observation matrix *H*. The green part is a resulting correction vector $\overrightarrow{\Delta \boldsymbol{E}_{tr}}$. The latter part of the equation is a pseudorange difference ***b*** between iterations, which corresponds to position shift. To get the finite error vector, the Equation (2.15) is formed according to the least squares rule.

$$\overrightarrow{\Delta E_{tr}} = \left(H^T H\right)^{-1} H^T \vec{b} \tag{2.16}$$

The correction vector is then summed with the previous position estimate (Eq. (2.12)), which forms the correct position after a few iterations.

2.5.4 Doppler Shift positioning

Position can be computed by measuring the Doppler Effect. Satellite traveling in space transmits on the frequency L1. Signal received on the ground f_r will be shifted due to Doppler Effect. This change is differenced with local reference frequency f_0. This comparison happens in defined time intervals (time marks [s]), which are generated by satellites.

$$\Delta f = f_0 - f_r \tag{2.17}$$

Every time mark initiates the Δf periods counting, which can be expressed as an integral over the time interval, when the time marks was received $t_i + \Delta_i$ (Δ_i represents the duration of signal travel from satellite to receiver).

$$N_i = \int_{t_i+\Delta_i}^{t_{i+1}+\Delta_{i+1}} \Delta f \,\mathrm{dt} \tag{2.18}$$

When assumed, the period *T* between each transmitted time marks is the same as the period of received time marks. The position can be computed as follows. (Madron, 2009)

$$N_i = TF + \frac{f_0}{c}\left[\sqrt{(X_{i+1}-x)^2 + (Y_{i+1}-y)^2 + (Z_{i+1}-z)^2} - \sqrt{(X_i-x)^2 + (Y_i-y)^2 + (Z_i-z)^2}\right] \tag{2.19}$$

Where *F* is a difference between satellite and received frequency $F = f_{L1} - f_r$ and (X, Y, Z) are coordinates of the satellite, which are derived from ephemeris.

To be able to solve the 3D position fix (x, y, z), at least three measurements (N_i, N_{i+1}, N_{i+2}) are required.

2.6 GPS Errors

Several errors affect the pseudorange measurement. The major limiting factor is selective availability (SA) described at the beginning of this chapter. To estimate the pseudorange, it is important to know the time when the message was sent from the satellite. This information is stored in navigation message as a part of the C/At code. The transmission of the information to the user device is always contaminated by various errors, which have different influences.

2.6.1 Clock errors

The satellite uses an atomic clock, which is very precise. However, the atomic clock can slightly differ from the GPS time. This error is calculated by the GPS – ground segment and uplinked to the satellites. The clock error is then transmitted to the user segment as a part of the navigation message in the form of three coefficients. These coefficients are applied to the polynomial model, which describes the satellite clock state at the transmission time Δt_{sv}.

$$\Delta t_{sv} = a_{f0} + a_{f1}(t - t_{0c}) + a_{f2}(t - t_{0c})^2 + \Delta t_r \tag{2.20}$$

$$t = t_{sv} - \Delta t_{sv} \tag{2.21}$$

The coefficients sent by the satellites are $a_{f0}, a_{f1,} a_{f2}$, GPS system time is denoted as t, and t_{0c} denotes the reference time given by the last epoch. The Δt_r is a relativistic correction since the spacecraft is located in lower gravitational space. Acquired satellite clock correction is than subtracted from the original transmission time, which gives the current GPS time. Equations (2.20) and (2.21) are coupled thus the sensitivity of t_{sv} and t is negligible. This negligible sensitivity will allow the user to approximate t by t_{sv} in equation (2.20) (Dunn, 2013). The difference between actual satellite time and time in navigation message can be up to 10ns (few meters in range). (Forssell, 2008)

2.6.2 Ionospheric Errors

Errors caused by the ionosphere are due to different signal propagation than in the vacuum. The signal is delayed while passing the ionosphere layer. The size of the error varies during the daytime since the lower ionospheric layers disappear at night. It can cause error up to 100m in range (Forssell, 2008).

For standalone GPS with single frequency band (most of low-cost GPS receivers), the ionospheric model proposed by J. Klobuchar is used to reduce the ionospheric error. It is an empirical approach, which uses the polynomial coefficients transmitted in navigation message so-called Klobuchar's alpha and beta. These coefficients are computed from a global, empirical model and updated every ten days (Forssell, 2008). It is possible to estimate the electron content fluctuation in the ionosphere layers, and the delay of the signal passed through. The efficiency of the ionospheric error reduction is about 50% RMS. (Subirana, Klobuchar Ionospheric Model)

2.6.3 Tropospheric errors

Tropospheric errors have main influence when the elevation[2] of the satellite is low. It is because the tropospheric layer is located only 60km above the ground surface. The error caused by troposphere can be up to 30m in range and highly depends on current weather situation (pressure, humidity, temperature, and more). Delays are divided into two categories (hydrostatic and wet). Hydrostatic is caused by the dry gasses in the troposphere, and the wet component is composed of water vapor and condensed water in the form of clouds (Zornoza). These values are quite easy to measure and predict. Based on the meteorological data the index of refraction is estimated, which is used by empirical models to estimate the tropospheric propagation delay. The tropospheric environment has an advantage. It is non-dispersive. It means that it is not frequency dependent. Thus, the L1 and L2, phase and code measurements are affected by the same delay.

2.6.4 Ephemeris Error

Represent the position error of the actual satellite position and position information broadcasted in navigation messages. Ephemeris represents the orbital parameters, which is used for satellite position estimation. There are three different types of these orbital data.

Almanac, which contains orbital data for all satellites SVs. Almanac is valid for several months, and it does not include any precise values. It is mainly used for the primary acquisition of the satellites at power-up or for plotting the satellite's visibility charts.

Broadcast Ephemerides are data send by the satellites. Each satellite transmits its ephemeris data. Broadcast ephemerides are used for real-time satellite position estimation in the receivers. The data are valid for half an hour and transmitted by the satellite every 30 sec. (Section 2.3.1.1)

Precise Ephemerides are provided by Naval Surface Warfare Center. They are based on the observation and future prediction algorithms. They are available for post-processing with time delay about two weeks. The data may be obtained upon request, and they are free of charge.

Table 5. Ephemerides error summary (Hofmann-Wellenhof, 2001)

Ephemerides	Uncertainty
Almanac	≅ 2 kilometers
Broadcast	≅ 1meter
Precise	≅ 0.05 – 0.2 meters

2.6.5 Multipath

Multipath refers to receiving the same signal in several paths. It is caused by reflection from the buildings, water planes and so on. If the wall reflects the signal, for example, the receiver will receive this signal with specific time delay and phase offset. In other words, the receiver becomes confused due to multiple signals carrying the same data (NovaTel, 2000). Multipath effects are very hard to mitigate because it can happen randomly and always have a different geometry.

[2] Elevation – angle between horizon plane and satellite direction vector with origin at the user location

More advanced receivers can reduce the multipath effect by accepting only the shortest path since the direct path should be the shortest one. However, this functionality is far away from low-cost receivers.

Another approach how to reduce the multipath is to use multi-band receivers and combine both (L1 and L2) code and phase measurement.

The effective solution to minimize the multipath effect is to place the receiver away from the reflective objects or define the higher elevation mask, which neglects the low elevated satellites.

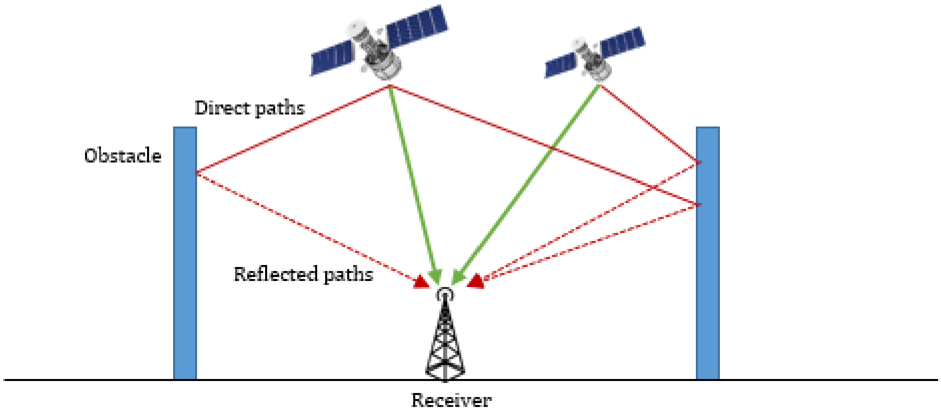


Figure 2.7. Multipath example

2.6.6 Dilution of Precision

DOP parameters are related to satellite constellation. The satellites' position geometry highly influences the positioning precision and accuracy. When the satellites used for navigation solution are aggregated into a small area, (the distance between each satellite is relatively small) the resulting position estimation gives much worse results than when the satellites are more spread over the visible area.

The DOP parameters are used to indicate this kind of error. It takes into account the position of each spacecraft on each other. Based on these DOP values, it is possible to predict the position precision. Small DOP values represent the higher precision in navigation solution.

There are different kinds of DOP parameters, which are connected to distinct measurable values.

- HDOP – measure horizontal diversions
- VDOP – measure diversion in vertical plane
- PDOP – is a position diversion based on all directions
- TDOP – is a time diversion

The scale of PDOP values is sorted into categories, which denotes the quality of satellite constellation.

- <4 – good satellite geometry
- 5-7 – more or less acceptable geometry
- >7 – poor satellite geometry

It is possible to define the limitation mask, which will discard the worse situations. (Rapant, 2002)

3 Position augmentation techniques

This section describes the most common position augmentation techniques. The main emphasis is on the system of Satellite Based Augmentation (SBAS) and principal of Differential GPS since these two methods are used for Arduino DGPS solution.

3.1 SBAS

The SBAS is a system for improving GPS accuracy designed especially for civil aviation. Commonly, the SBAS systems are developed to cover defined region by the set of satellites located on GEO orbits[3]. The satellites transmit augmentation data to the receiver on the same frequency as GPS L1, so there are no special hardware requirements.

The SBAS satellites provide corrections for GNSS satellite position (ephemeris), satellite time biases, tropospheric and ionospheric errors. It provides correction and integrity monitoring for all satellites, which are currently in the view by the ground segment. Moreover, the SBAS provide corrections to the existing system such as GLONASS, GPS, or GALILEO. The main advantage of the SBAS is the free availability for everyone. However, it is redeemed by the lack signal strength and a limited number of SBAS satellites, which result in limited coverage.

There are several types of SBAS developed to cover the main regions of the world as shown in the figure below.

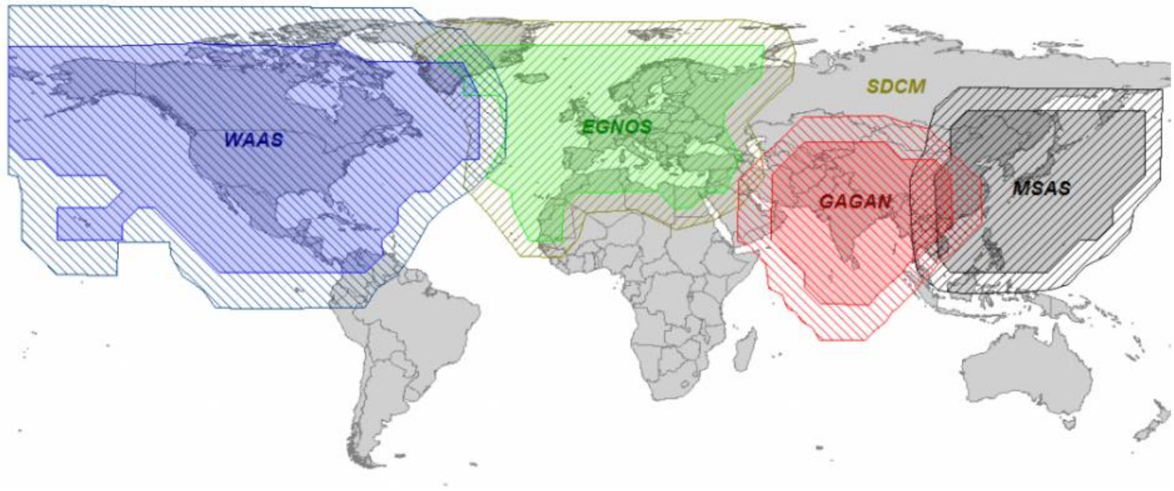


Figure 3.1. SBAS global system implementation (Gakstatter, 2013)

In the following section, the European EGNOS and the US WAAS will be described in more details.

3.1.1 EGNOS

It is a European Geostationary Navigation Overlay. The EGNOS was developed by the European Space Agency and put into operation in 2004. Its main purpose is to improve the level of accuracy of the current navigational system GPS, GLONASS. Although the EGNOS is developed to cover the European space, the compatibility with GALILEO satellites is a matter of course.

[3] GEO – Geostationary Earth Orbit – Where the position of the satellite and point on the Earth is adjacently synchronized

The EGNOS core consists of three segments:

- Ground Segment
    - It is composed of reference stations RIMS (Ranging and Integrity Monitoring Station). Its purpose is to measure actual errors from the GPS satellites and transmits them to the MCC (Master Control Centers). There is currently 39 RIMSs spread over the World. (ESA, EGNOS Open Service, 2015)
    - The ground segment also includes the NLES stations, which are used for uploading the correction data to the EGNOS satellites.
- Space Segment
    - Comprises three satellites on GEO orbits, which broadcast on GPS L1 PRN 120, 126 ,136
- User Segment
    - Include all receivers, which can parse the EGNOS signal. It does not require any special hardware, but only more advanced software.

The EGNOS correction results in 3m horizontal and 4m vertical accuracy. All these accuracy measures correspond to 95% confidence level (ESA, EGNOS Open Service, 2015).

The communication between EGNOS satellites and receivers is done by transmitting 16 different messages in 6 seconds cycle. The correction data are provided in the form of Fast and Slow corrections. Fast correction represents the rapidly changing values such as satellite clock errors. The slow correction is composed of long-term satellite clock drift and ephemeris error or ionospheric signal delay. (ESA, The EGNOS fact sheet, 2005)

Since all the EGNOS satellites are on GEO orbits, and they are hovering above the equator, the signal from the satellites is feeble, especially in higher latitudes. Due to these reasons, it is not recommended to use SBAS for surface roving devices because they can easily lose the SBAS signal. The switching between SBAS, normal mode and old SBAS data can make the receiver confused, which may result in higher errors in position computation. Also, the weaker signal is more sensitive to solar rays, which can lead to a total signal loss (ESA, EGNOS Open Service, 2015).

3.1.2 WAAS

The Wide Area Augmentation System is another satellite-based augmentation system, primarily designed to improve accuracy and integrity of the air traffic in the USA. However, the transmission of the WAAS data is done on GPS L1, so it can be used by the arbitrary receiver, which supports the SBAS.

As same as the EGNOS, the WAAS is composed of three segments. The Space segment consists of three satellites of PRN (133, 135, and 138). The 38 Wide-Area Reference Station (WRS) capturing the local GPS errors and together with three Wide Area Master Stations creates a Ground segment. Finally, the user segment mostly composed of advanced receivers as a part of an aircraft avionics. (FAA, 2008)

The corrections are broadcasted in the same way as EGNOS do. It provides the fast and slow correction messages, which may be used for pseudorange corrections.

3.2 Differential GPS (DGPS)

In this chapter, the principals of DGPS algorithms will be described. Differential GPS was developed to mitigate the systematic error of the SA (Selective Availability). It leads to improvement of the positioning error from hundreds of meters to approximately 10m accuracy. The RS (reference station) and the rover station compose the DGPS system. The reference station is located stationary at a known position. According to the difference between known and measured position the RS calculate corrections, which the rover unit uses to mitigate the positioning error and improve its accuracy. The correction estimation is typically done in two domains (position and range domain).

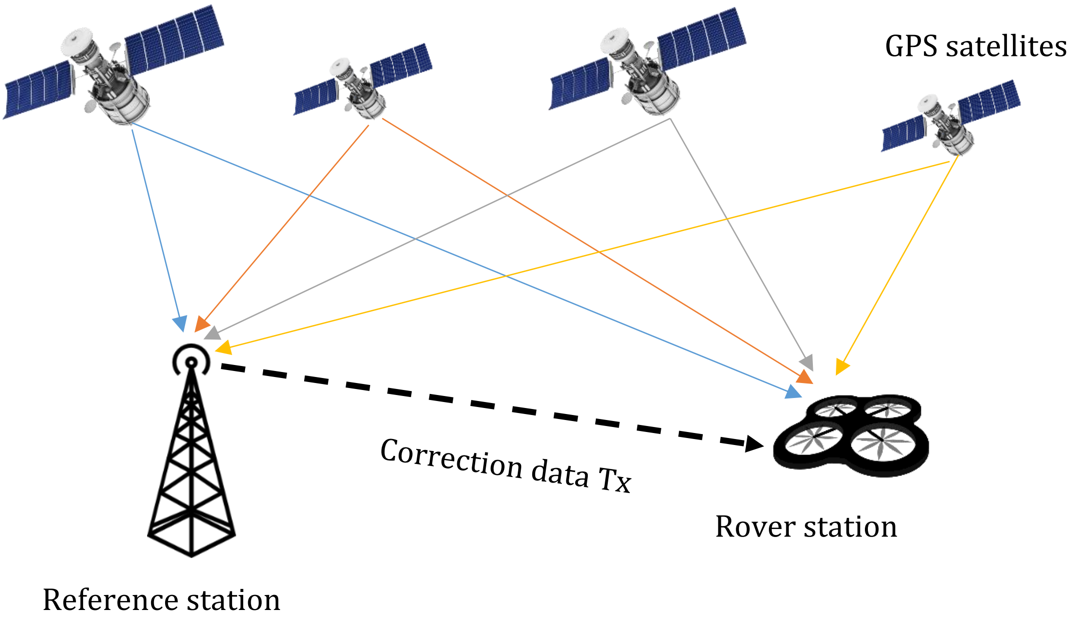


Figure 3.2 - Principal scheme of DGPS

As seen from the figure, it is required that both stations have a permanent 3D fix. It means established a connection to at least four satellites.

In early years, when the SA was active the positioning error had a high variation. The precision could jump from 10 to 200 meters in the minute. This property required a high rate of the correction data transmission because the value of the correction data regard to the position error variation. Since the SA was turned off, the position error size was also reduced. It led to the prolonged age of the correction date and possibility to rate reduction. However, the frequency of the correction data sending remains on the same level, which is increasing the robustness of the DGPS system. (Rapant, 2002)

3.2.1 Position domain DGPS

Position domain DGPS is very intuitive and straightforward to understand. It simply calculates a position difference from the known and measured position. This difference is then added to the rover's measurement, which augments its position.

The receiver in the RS iteratively measures its position using the least squares and observed pseudoranges described in Section 2.5.3.

In the position domain DPGS, the computed coordinates $\vec{X}_{RS}$ is subtracted from the known coordinates of the reference station. It gives the position difference, which is the correction of the position error. It is denoted as vector δ_X.

$$\vec{\delta}_{\mathrm{X}} = \vec{X}_{RS,known} - \vec{X}_{RS} \tag{3.1}$$

The RS transmits this correction to the rover unit by the communication link. The rover station applies this correction to its measured position, mitigate the position error, and improve its position accuracy.

$$\vec{X}_{rover,DGPS} = \vec{X}_{rover,measured} + \delta_{\vec{X}} \tag{3.2}$$

This simple solution of DGPS system is not very useful since the position correction is valid only if the satellites, used for the navigation solution, are the same at the RS and the rover station. If the RS uses another set of satellites, it will create a different observation matrix (H), and the computed error correction will do not have to improve the location. It can even increase the error up to values, which can be worse than standard GPS. This implies that both stations have to coordinate the satellite selection with each other or the RS must compute the position correction for all combination of the satellites in the view. Dale Carnegie proved this when comparing the standard deviation of differential GPS and stand-alone GPS. Without corrections, standard deviations of 2.38 m and 4.64 m were recorded in the latitude and longitude directions respectively. The resulting standard deviation results were worse while using a block-shift correction, 3.88 m and 6.90 m (Carnegie, 2005). This method does not require the raw observables, which is an advantage for the low-end receiver. However, the benefit is compensated by the necessary access to the satellite selection control. This feature does not provide even high-end GPS receivers.

3.2.2 Range domain DGPS

The second method uses the pseudorange corrections to the individual satellites instead of capturing the position correction from the coordinates shift. The receiver is located at the known location with defined coordinates. It measures the position according to the satellite constellation and respective pseudoranges. Until now, the mechanism is as simple as the stand-alone GPS. The position of each satellite is important to know to compute the pseudorange correction for particular satellites. The satellite position can be calculated from satellite ephemeris. Usage of the ephemeris is a significant issue for implementation to the low-cost GPS receivers. To get the satellite position, the raw navigation message is needed. When the satellite position is known, the true satellite range can be derived. The pseudorange correction is then obtained by subtracting the measured pseudorange and time biases from the true satellite range. There are two approaches how to calculate the pseudorange correction (PRC).

3.2.2.1 Code range based DGPS

Code based pseudorange measurement was described in Section (2.5). The code range Rdenotes the measured range observable between the transmitted and received code words. It can be summarized as follows in equation (3.3).

$$R^{j}_{RS}(t_0) = \rho^{j}_{RS}(t_0) + \Delta\rho^{j}_{RS}(t_0) + \Delta\rho^{j}(t_0) + \Delta\rho_{RS}(t_0) \tag{3.3}$$

Where $R^{j}_{RS}(t_0)$ denotes the code range between the reference station and satellite j at epoch t_0. See equation (2.11) for biases description.

The pseudorange correction (*PRC)* is added to the measured pseudorange. This adjustment creates a better approximation to the true range between the *RS* and satellites (Equation (3.4)). In addition to the pseudorange correction $PRC_{RS}^{j}(t_0)$, the time derivative of range rate correction $RRC_{RS}^{j}(t_0)$ is determined at the RS (Equation (3.5)) (Hofmann-Wellenhof, 2001).

$$PRC_{RS}^{j}(t_0) = \rho_{RS}^{j}(t_0) - R_{RS}^{j}(t_0) \tag{3.4}$$

$$PRC_{RS}^{j}(t) = PRC_{RS}^{j}(t_0) + RRC_{RS}^{j}(t_0)(t - t_0) \tag{3.5}$$

The *RRC* is used to estimate the *PRC* at the time *(t).* The value is associated with actual pseudorange variation due to some interference (especially for S/A). Since the S/A was turned off, the *RRC* is reduced to a very low value (few tens of meter/s). The natural effects in the ionosphere and troposphere now cause the major errors. Since the natural effects are hard to predict, it is better to set the value of *RRC* to zero.

The *PRC* is transmitted to the rover station in real-time. The rover station estimates its pseudo - ranges in the same way as the RS by equation (3.6).

$$R_r^{j}(t) = \rho_r^{j}(t) + \Delta\rho_r^{j}(t) + \Delta\rho^{j}(t) + \Delta\rho_r(t) \tag{3.6}$$

The sign of the *PRC* is selected in equation (3.4) to be always added at the rover side.

$$R_r^{j}(t)_{corrected} = R_r^{j}(t) + PRC_{RS}^{j}(t) \tag{3.7}$$

Compared to equation (3.4) (3.3) and (3.6), the corrected pseudorange (3.7) of the j-th satellite can be rewritten as follows.

$$\begin{gathered} R_r^{j}(t)_{corrected} = \rho_r^{j}(t) + \Delta\rho_r^{j}(t) + \cancel{\Delta\rho^{j}(t)} + \Delta\rho_r(t) + \left(-\Delta\rho_{RS}^{j}(t) \cancel{- \Delta\rho^{j}(t)} - \Delta\rho_{RS}(t)\right) \\ R_r^{j}(t)_{corrected} = \rho_r^{j}(t) + \left(\Delta\rho_r^{j}(t) - \Delta\rho_{RS}^{j}(t)\right) + \left(\Delta\rho_r(t) - \Delta\rho_{RS}(t)\right) \end{gathered} \tag{3.8}$$

As seen in equation (3.8) the satellite clock bias was subtracted, which cancel the satellite clock error. There is one critical assumption. The biases of measured pseudoranges to the set of adjacent satellites are highly correlated. This correlation allows reduction of the atmospheric refraction and orbital errors. If the multipath effect is neglected, the second part of the equation (3.8) refers to combined receiver clock bias. It gives the simplified corrected range of the rover unit.

$$R_r^{j}(t)_{corrected} = \rho_r^{j}(t) + \left(\Delta\rho_r^{j}(t) - \Delta\rho_{RS}^{j}(t)\right) = \rho_r^{j}(t) + \Delta\rho_{r,RS}(t) \tag{3.9}$$

3.2.2.2 Phase range based DGPS

Differential GPS based on carrier phase measurement also refer to real-time kinematic (RTK). It is used in high demanding measurement techniques such as geodesy or land survey. Principally, the Phase ranges DGPS works similarly as DGPS with code ranges. The pseudorange R is modeled as follows. (Hofmann-Wellenhof, 2001)

$$\lambda\phi_{RS}^{j}(t_0) = \rho_{RS}^{j}(t_0) + \Delta\rho_{RS}^{j}(t_0) + \Delta\rho^{j}(t_0) + \Delta\rho_{RS}(t_0) + \lambda N_{RS}^{j} \tag{3.10}$$

All parameters are analogous to the code range measurement. The $\lambda\phi_{RS}^{j}(t)$ represents the calculated pseudorange by multiplying wavelength cycles with wavelength itself. It gives the pseudorange in meters. Last parameter λN_{RS}^{j} is ambiguity. It is a number of complete wavelengths between RS and satellite. More about ambiguity is in Section 2.5.2.

The pseudorange correction is formulated correspondingly to the PRC in code ranges.

$$PRC_{RS}^{j}(t) = \rho_{RS}^{j}(t_0) - \lambda\phi_{RS}^{j}(t_0)$$
$$PRC_{RS}^{j}(t) = -\Delta\rho_{RS}^{j}(t_0) - \Delta\rho^{j}(t_0) - \Delta\rho_{RS}(t_0) - \lambda N_{RS}^{j} \tag{3.11}$$

Computed pseudorange correction is transmitted to the rover unit r where the PRC is applied to the pseudorange calculation.

$$\lambda\phi_{r}^{j}(t) = \rho_{r}^{j}(t) + \Delta\rho_{r,RS}(t_0) + \lambda N_{r,RS}^{j} \tag{3.12}$$

Where $\Delta\rho_{r,RS}(t_0)$ is the difference of the satellite positions based range biases and refer to combined receiver clock bias, when multipath is neglected. The $N_{r,RS}^{j}$ is the difference of the ambiguities measured at the rover and RS.

As mentioned at the beginning of the phase based DGPS, it is related to Real Time Kinematic (RTK). The relation is limited by the update frequency of the PRC to the rover station also known as latency. When the latency of the PRC update is close to zero, the precise phase DGPS is identical to RTK specifications.

3.2.3 Low cost receivers and DGPS

Implementation of the general DGPS models in the low-cost GPS receivers is very problematic. It is assumed that low-cost receivers are defined by its price, which is less than 30EUR (reflected by the component prices). These GPS receivers are mostly one channel and provide only basic navigational services in the form of NMEA output (GGA, GSV, and GGL messages) by default. The common user is satisfied with this output. However, for advanced application such as DGPS, the NMEA output is not sufficient.

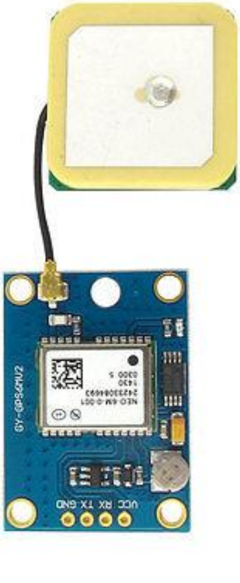


Figure 3.3. Low-cost GPS receiver U-Blox NEO6m

The raw measurement of the pseudoranges and position of each satellite must be known to estimate the pseudorange corrections by the general DGPS model. Acquisition of these data and following application of the PRC becomes impossible with the default setting. Due to this reasons, a special algorithm was developed to enable DGPS functionality with this type of receivers. (Kim, 2013) This algorithm is described in Section (4.3).

3.2.4 DGPS errors

The DGPS can mitigate or reduce mutually correlated errors. It means that errors are projected to the number of receivers, which are close to each other, in the same manner. It involves clock ephemerides and atmospheric effects. As seen from the multipath description in Section 2.6.5, the multipath effect cannot be mitigated even because of the random geometry. For example, the static RS located at the known position and moving rover station. The RS can be located in the open area with a minimal multipath error, but the rover is moving, and it can enter the area with frequent multipath error (near the building). The RS has no chance to predict or reduce this kind of mistakes.

Satellite clock biases and clock errors generated by SA are eliminated. The effect of these errors is projected into the pseudorange measurement, which is same for different receivers at different locations and they can be mutually subtracted.

Ephemeris and atmospheric errors can be reduced but not mitigated. It is because, at a different location, the different error will be measured. The variance of the error depends on the distance between receivers. With increased distance, the variance will also be higher. The amount of error suppression also depends on the age of PRC. The minimum age[4] was required especially for SA since the errors caused by SA was varying very frequently.

When talking about the distance between receivers, it means the distance between reference and rover station. The positioning error should not exceed 10m in the distant area 100km from the RS. (Madron, 2009)

The table below shows the mean amount of error reduction using DGPS presented by (Forssell, 2008).

Table 6. Error difference between GPS and DGPS (Forssell, 2008)

	GPS C/A code range (m RMS)	DGPS C/A code range (m RMS)
Satellite clock errors	1-3	0
Ephemeris errors	2.5-7	0-0.1
Ionospheric errors	2-15	0.1-1.5
Tropospheric errors	0.4-2	0.1-1.5
Multipath propagation	2-4	2-5
Resulting range error	4-18	2-6
Resulting position error H	6-27	3-9
Resulting position error V	10-45	5-15

[4] Age – mean the time between the issued PRC and time, when the PRC was applied at the rover station

4 DGPS system proposal

This section will describe the proposal of the new low-cost DGPS system. In the former part, the principal scheme, and functional proposal, which describing the core algorithms used by the both modes. This is followed by necessary GPS receiver setup respectively for RS and rover station. The latter part of the section is focused on correction data transmission and proposal of self-accuracy monitoring.

In general, the system of Arduino DGPS consists of two devices. At the RS's side, the main module considered for all the computations was Arduino MEGA. However, there were lacks of computational precision because MEGA module is equipped with 16MHz ATmega2560, which has limited memory and performance. Due to that, the MEGA has restricted the size of the double variable to 4-bytes (same as float). This lack of precision makes the calculation of PRC and satellite position very problematic. To mitigate this major issue, the Arduino DUE board is used. The Arduino Due support full sized double variables (8-bytes), and thanks to the enhanced microcontroller, the performance is improved.

The rover uses one Arduino device (version Nano or Micro), which receive the data coming from the GPS receiver and establish the Wi-Fi connection with the RS. The rover device only parses the GPS data, send them to the RS, and apply corrected position received from RS since these "small" versions of Arduino are equipped only with ATmega32. This microcontroller has only 32KB of flash memory and limited performance 16MHz. The reason for the use of this device is weight since the application of the Arduino DGPS may be utilized by drones where every gram counts. The key scheme is shown in Figure 4.1.

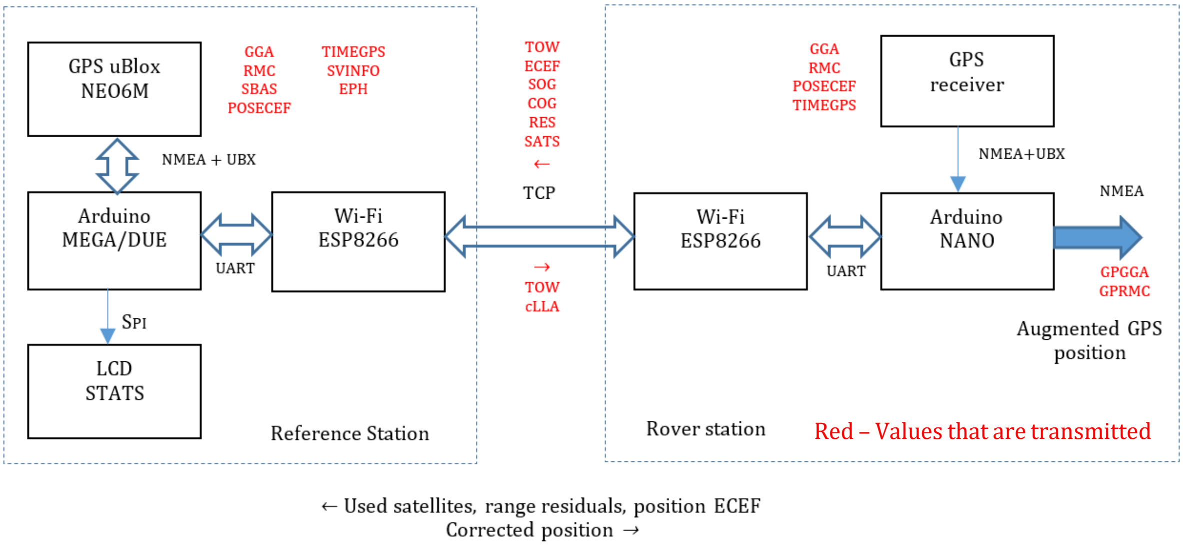


Figure 4.1. Arduino DGPS System proposal - Block Scheme

The communication between rover and RS is dedicated. Therefore, the RS has to compute corrected position for each rover station individually since the position correction processing is done in RS. In the following Arduino DGPS development, only one rover station is assumed.

4.1 Functional proposal

The system of Arduino DGPS is designed to operates in two modes. The first mode supports the Code Range based DGPS idea described in Section 3.2.2.1.

4.1.1 Range Residuals (MODE1)

The main goal in the MODE1 is to generate the PRCs and apply them to the position computed by the rover's GPS receiver. The PRC is calculated as the difference between observed pseudorange and range computed to the true position estimated from the map or by long-term measurement.

4.1.1.1 PRC estimation

The method of deriving PRC is based on the geometrical ranges from the actual measured position ρ_{RS}^{j}, true position $\rho_{RS,true}^{j}$ to the particular satellites, which are used at the rover's side, and range residuals ΔR_{RS}^{j} between observed pseudorange and computed pseudorange from the actual position. The computed pseudorange is modeled as a geometrical range ρ_{RS}^{j}, satellite clock bias $\Delta\delta_s$, and range bias $\Delta\rho_r^{j}$, which includes the ephemeris, relativistic and atmospheric errors (4.1). Since the satellite clock bias and range bias are correlated with the rover, they will be subtracted in PRC computation. This is shown in (4.2).

$$\begin{aligned} R_{RS,comp}^{j} &= \rho_{RS}^{j} - c\,\Delta\delta_s + \Delta\rho_r^{j} \\ R_{RS,observed}^{j} &= R_{RS,comp}^{j} + \Delta R_{RS}^{j} \end{aligned} \tag{4.1}$$

Residuals are contained in the NMEA GRS message and UBX-SVINFO.

Obtained observed pseudorange $R_{RS,observed}^{j}$ differs from the RAW observable since it does not contain the receiver clock bias. It is the value, which is directly used for position estimation.

The required PRC is computed as same as in the generic range domain DGPS as shown in following equation.

$$\begin{aligned} PRC^{j} &= \rho_{RS,true}^{j} - R_{RS,observed}^{j} \\ R_{r,corrected}^{j} &= R_{r,observed}^{j} + PRC^{j} \\ R_{r,corrected}^{j} &= \rho_r^{j} + \rho_{RS,true}^{j} - \rho_{RS}^{j} + \Delta R_{RS}^{j} - \Delta R_r^{j} \end{aligned} \tag{4.2}$$

The limitation is that the residuals are computed and carried by the messages only for the set of satellites, which are currently used for the navigation solution. To have the correction valid, the set of satellites used by the rover must be a part of the set of satellites used by RS (the higher elevation mask was applied at the rover side to assure that).

4.1.2 SBAS corrections (MODE2)

Another option, how to obtain the PRC is SBAS. Satellite-based augmentation was described in Section 3.1. The PRCs are provided directly by the receiver via unified UBX protocol, more about UBX in Section 2.3.4. MODE2 represents a simplified type of repeater, which repeats the SBAS corrections to the rover.

The big advantage of this method is that RS does not have to know the true position because the PRCs are already calculated. The RS just use them to compute a position correction according to rover set of used satellites. The need for the use of RS is also because the SBAS signal is weak (especially EGNOS in Europe) and it does not have to be always available for a rover itself. The RS has to be located stationary at the open area to establish the SBAS connection.

4.1.3 Application of the PRC

Calculated PRCs is applied differentially in each mode. The MODE2 involves the Correction Projection Algorithm described in Section 4.2. The solution of the position projection is then added to the measured rover position, which generates the corrected position. This method is useful since the PRC is already calculated without regard to the actual measured position by the RS. Thus, it will improve even filtered position generated by the receiver.

The MODE1 is bit trickier. It computes the difference between actual ranges (defined by the measured position of the rover) and corrected rover ranges defined by equation (4.2). This residual is then projected to the position domain and resulting position shift is added to the actual measured position. To be able to compute the corrected rover position, the range residuals have to be transmitted together with the calculated position from the rover station to RS.

The corrected position is not dependent on the receiver output filtering so the precision will be worse than actual measured position, but with improved accuracy. To get the corrected position with better precision, Kalman filter can be optionally applied at the output.

4.2 PRC Projection to Position Domain

The original idea how to use GPS receivers, which does not provide any raw pseudorange output was presented by (Kim, 2013) and (Yoon, 2014). The main formulation consists of the projection of the gathered pseudorange corrections into position domain. The algorithm allows application of the position correction directly to the rover's position coordinates. Thus, the rover's GPS device does not have to be modified and for the correction, the NMEA output is sufficient.

It is necessary to mention the method of computing the corrected coordinates using least-squares procedure in measurement domain to understand better the position projection.

$$\begin{bmatrix} \vec{X}_{DGPS} \\ B \end{bmatrix} = (\mathrm{H}^T H)^{-1} \mathrm{H}^T \begin{bmatrix} \vdots \\ -e^j \rho^j + (R^j + PRC^j) \\ \vdots \end{bmatrix} \tag{4.3}$$

Whereas e^j is a unit vector of line–of–sight directions to the j-th satellite. The ρ^j is a position vector from receiver to the j-th satellite. The R^j and PRC^j respectively denote the observed pseudorange (3.3) and computed pseudorange correction (3.4). B represents a receiver clock bias.

The equation (4.3) can be decomposed as follows.

$$\begin{bmatrix} \vec{X}_{DGPS} \\ B \end{bmatrix} = (\mathrm{H}^T H)^{-1} \mathrm{H}^T \begin{bmatrix} \vdots \\ -e^j \rho^j + R^j \\ \vdots \end{bmatrix} + (\mathrm{H}^T H)^{-1} \mathrm{H}^T \begin{bmatrix} \vdots \\ PRC^j \\ \vdots \end{bmatrix} \tag{4.4}$$

Where the former part denotes a standalone GPS position and the latter part is a correction in position domain according to equation (3.1). The last part is a crucial one and gives the desirable solution

$$\vec{\delta}_{\mathrm{X}} = (\mathrm{H}^T H)^{-1}\mathrm{H}^T\,\overrightarrow{PRC^j} \tag{4.5}$$

The important thing to this solution is to control the satellite selection as same as in the position domain DGPS described in (3.2.1). However, in the position domain DGPS, the correction is done by differencing already computed position coordinates. Due to that, it is impossible to coordinate the set of used satellites for navigation solution in both stations.

The DGPS correction projection algorithm (DGPS-CP) (stated in (4.5)) can set-up the position domain correction with an arbitrary set of satellites. To have the position correction valid, the set of j satellites must be the same as in the rover station. Otherwise, the different constellation creates definite H matrix and PRC vector. Thus, the position correction generated by (4.5) can cause a much bigger error as proved in (Kim, 2013).

4.2.1 Projection matrix H

The remaining things are to estimate parts of the equation (4.5). Projection matrix composition is the same to the observation matrix described in Section 2.5.3. H matrix consist of line-of-sight vector cosines also called direction vectors e_j (Eq. (2.15) - yellow part). It represents the direction from the receiver position to the satellite in ECEF. Figure 4.2 illustrates the situation in ECEF coordinates system.

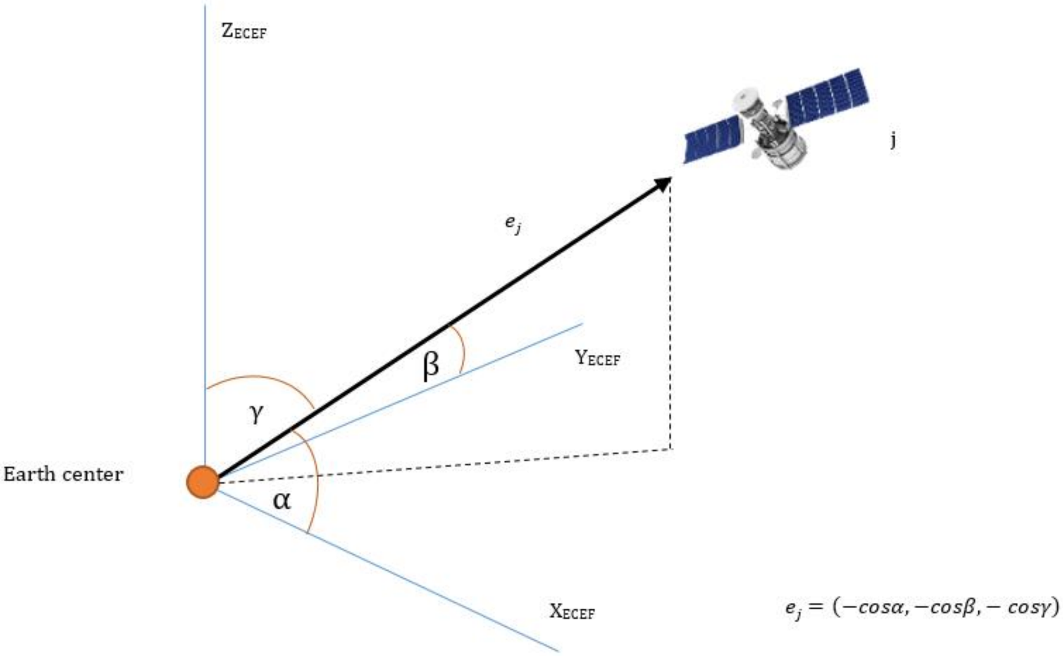


Figure 4.2. Line-of-sight vector representation in ECEF

The remaining question is how to get these parameters. There are two possible ways.

4.2.1.1 Satellite Position Calculation

The line of sight vector can be computed using the satellite position calculated from ephemeris parameters (Section 2.3.1.1). Satellite position is calculated based on actual GPS time. At first, the difference between reference time t_{ref} and actual epoch t, have to be derived.

$$\Delta t = t - t_{ref} \tag{4.6}$$

Where, both timestamps are in seconds and t_{0e} represent an ephemeris reference GPS time of the week.

$$t_{ref} = (GPSweek - 1)\times 168\times 3600 + t_{0e}\ [s] \tag{4.7}$$

Mean motion represents the average angular speed required to complete one orbit, which the body completes in variable speed.

$$n = \Delta n - n_0\ [rad/s] \tag{4.8}$$

Where n_0 is a reference mean motion based on gravitational constant and earth mass.

$$n_0 = \sqrt{\frac{GM}{a^3}} \tag{4.9}$$

The variable GM is based on the geodetic model definition (WGS84). It represents the product of the earth mass (including the atmosphere) and gravitational constant.

In the next step, the position of the spacecraft on the ellipse will be computed based on the Mean and True anomaly. The mean anomaly is an angle between body and pericenter, which the body has while moving on the circular orbit. True anomaly denotes a position on the elliptical orbit represented by the angle between directions of periapsis[5] and body seen from the focus point.

$$M = M_0 + n\Delta t\ [rad] \tag{4.10}$$

To calculate True anomaly, the eccentric anomaly[6]has to be computed by iteration of Kepler's equation of eccentric anomaly.

$$E_k = M + e\sin E_{k-1}\ \ [rad] \tag{4.11}$$

True anomaly,

$$\begin{aligned} \cos v &= \frac{\cos E - e}{1 - e\cos E} \\ \sin v &= \frac{\sqrt{1-e^2}\sin E}{1 - e\cos E} \end{aligned} \quad [rad] \tag{4.12}$$

Argument of latitude describes the position of the body moving along the Kepler's orbit in term of angle between the body and ascending node,

[5] Periapsis – is the nearest extreme point on the elliptical orbit

[6] Eccentric anomaly – is the angle between the periapsis and body, but observed from the ellipsoid center.

$$\delta u = C_{uc}\cos 2(v+\omega) + \; C_{us}\sin 2(v+\omega) \;\; [rad]$$

$$u = v + \omega + \delta u \; [rad] \tag{4.13}$$

Actual radius and radius correction from the ellipsoid center and the body is defined as.

$$\delta r = C_{rc}\cos 2(v+\omega) + \; C_{rs}\sin 2(v+\omega) \;\; [m]$$

$$r = a(1 - e\cos E) + \delta r \;\; [m] \tag{4.14}$$

Moreover, the inclination correction and actual inclination between the reference plane and the ellipsoid are computed.

$$\delta i = C_{ic}\cos 2(v+\omega) + \; C_{is}\sin 2(v+\omega) \;\; [rad]$$

$$i = i_0 + i\Delta t + \delta i \; [rad] \tag{4.15}$$

Longitude of ascending node defines an angle between the ascending node and the reference direction in the reference plane.

$$\Omega = \Omega_0 + \left(\dot{\Omega} - \omega_e\right)\Delta t - \omega_e t_{0e} \;\; [rad] \tag{4.16}$$

Where ω_e is defined by WGS84 reference geodetic system and denotes the angular speed of the earth.

Finally, the orbital plane coordinates and coordinates of the spacecraft can be derived as follows.

$$X' = r\cos u \quad [m]$$

$$Y' = r\sin u \quad [m] \tag{4.17}$$

$$X = X'\cos\Omega - Y'\sin\Omega\cos \mathrm{i} \quad [m]$$

$$Y = X'\sin\Omega - Y'\cos\Omega\cos \mathrm{i} \quad [m] \tag{4.18}$$

$$Z = Y'\sin i \quad [m]$$

By using the true position, the computed satellite position and regarding range between these, it is possible to get the full representation of LOS vector.

As shown, the calculation of the satellite position using ephemeris is quite difficult. The calculation has to be done for each SV and epoch to get the position for all SVs required for the line of sight vectors. It consumes much time and memory because each value of the ephemeris must be stored since the position changes with time. Nevertheless, the advantage of this method is a precise line of sight vector, which is useful for more accurate position correction.

Another limitation using this approach is that low-cost receivers do not provide the raw ephemeris data by default. However, there is way how to get limited (but sufficient) ephemeris information even with the low-cost receiver. (Section 4.3.2)

There is also a supplementary solution to get the ephemeris by using CORS data. CORS (Continuously Operated Reference Station) is online service, which provides ephemeris measurement on different time scales mostly in RINEX format. Delayed data are free of charge. The minimum data update rate is typically 15 minutes, which seems to be enough for the processing since the ephemeris data are valid up to an hour. On the other hand, the RS has to establish the internet connection and periodically download the ephemeris data, which is not always possible.

4.2.1.2 Satellite Azimuth and Elevation

Another way, how to get the LOS vector is to use the NMEA GSV or UBX-SVINFO message. GSV message contains angular coordinates of each satellite with the base at the receiver location (Local coordinates frame).

The structure of the NMEA GSV message is described in Section 2.3.2. The satellite azimuth and elevation angles are used to estimate the direction vector.

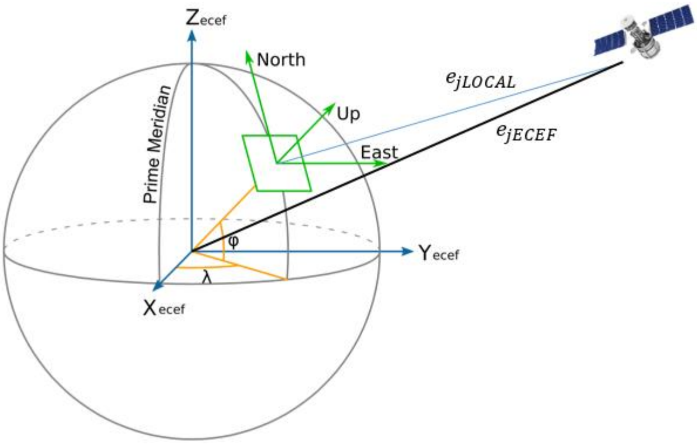


Figure 4.3. Local and ECEF coordinates relation

The local coordinates of LOS are computed directly from the azimuth Az_j and elevation El_j angles.

$$\overrightarrow{e_{J,LOCAL}} = \begin{bmatrix} \sin\ Az_j \cos\ El_j \\ \cos\ Az_j \cos\ El_j \\ \sin\ El_j \end{bmatrix} \tag{4.19}$$

The rotation matrix R is used to convert the local coordinates to ECEF coordinates system.

$$\overrightarrow{e_{J,\mathrm{ECEF}}}^T = \begin{bmatrix} -\sin\lambda & -\cos\lambda\sin\varphi & \cos\lambda\cos\varphi \\ \cos\lambda & -\sin\lambda\sin\varphi & \sin\lambda\cos\varphi \\ 0 & \cos\varphi & \sin\varphi \end{bmatrix} \times \overrightarrow{e_{J,LOCAL}} \tag{4.20}$$

Where λ and φ refer to the longitude and latitude of the receiver position. This method is very easy to implement, but it also brings some difficulty in precision. The NMEA GSV sentence provides only azimuth and elevation in full degrees (integers) that cause uncertainty in the satellite direction vector. However, as proved in (Park, 2014), the error posed by this uncertainty in position was not higher than 0.2m.

4.3 GPS Receivers Setting

The uBlox GPS receivers are used for application and testing in both stations. There are several models provided by uBlox with different prices. Concrete model for the application is uBlox NEO6m (Figure 3.3). It is a basic GPS receiver, which does not provide any RAW data output.

4.3.1 Rover Station

The main goal is to make just the minimal modification from default functionality of the GPS receiver to make the system easy to implement.

The one thing that should be configured is the minimal satellite elevation threshold. It allows filter out SVs, which have a low elevation (default 5°). By setting a higher elevation mask at the rover side, it secures that the number of the used satellite will always correspond to the RS.

The minimum GPS requirements are to have enabled NMEA GPGLL, GGA, UBX-SVINFO, UBX-POSECEF, and UBX-TIMEGPS (to get the actual TOW for synchronization).

4.3.2 Reference Station (RS)

The GPS receiver at the RS side requires a larger intervention into its setting. To enable the ephemeris, the specific setup code is sent to the device during start-up and after defined time interval. Some setting can be done directly in U-Center (selection of particular NMEA sentence, output frequency, and so on). There are two types of messages. **Polled** is refreshed when request code is sent and **Periodic**, which is updated every cycle defined by output frequency. Since the receiver sending the subframe data via UART, it is better to set the higher speed of the serial channel. In this case, the baud rate was set to 57600Bd.

To enable full functionality, following messages have to be enabled.

Table 7. Messages required by RS

Protocol	Type	Name	Description
NMEA	Periodic	GPGLL, GPRMC	Provides minimum positioning information such as (Latitude, Longitude, UTC)
UBX	Periodic	NAV-SVINFO	Provides range residual for active satellites
UBX	Periodic	NAV-SVINFO	Provides information about satellites in view (Azimuth, Elevation)
UBX	Polled	AID-EPH	Gives subframe data 1-3 from navigation message (Ephemeris)
UBX	Periodic	NAV-POSECEF	Directly provides ECEF coordinates
UBX	Periodic	NAV-TIMEGPS	Contains the GPS week/time
UBX	Periodic	NAV-SBAS	Provides SBAS correction data

🟩 Require request message

Other messages, which are provided by the receiver should be disabled to reduce the load of the Arduino device, while parsing the data. However, it will not affect the functionality if a strange frame will be received.

4.4 Algorithm Implementation

This Section discusses the application of the key parts of the algorithm to the Arduino environment. It is done separately for RS and rover station.

The Arduino environment is based on C#/C++ language with a lot of simplification, which makes them very easy to use. Arduino and the most of the developed libraries are open source, so it allows the user to realize many different projects.

Both stations using the same GPS receiver and the data, which are transmitted, are parsed by the Arduino units. For this purpose, the TinyGPS++ library was used. In general, TinyGPS provides the reading of two basic NMEA message GLL and RMC. However, the library also offers a function to call any parameter from arbitrary NMEA sentence (more in (Hart, 2016)). To be able to read the UBX data frames, the library was modified by adding special conditions. These conditions are used to identify the beginning of the UBX frame (0xB5, 0x62), which allows parsing the UBX data separately from NMEA. The parsing is done by saving the UBX stream to the array, which is then parsed to representative variables. Variables are then called from the main program as needed. Each UBX frame is recorded to the array individually.

4.4.1 Rover Station

The rover station acts as the initiator of the whole correction calculation process. The triggering event is when the new GPS measurement is received. The end of the GPS measurement is indicated by the Boolean flag (`time.end`), which is set to HIGH when the last word of the last frame is parsed. In the case when only important messages are enabled (Table 7), the last sentence in the queue is NMEA GPGGA and last word parsed is the geoid separation, so the flag is located there. It is done in the modified TinyGPS library. It is not a rule that GPGGA will always be the last, so it is important to check the message flow by using uCenter and place the flag to appropriate position in the library.

In the main sketch, there is a function `programSend()`, which is continuously called. The function is waiting for the triggering flag by calling the library function `gps.time.isFinished()`. When the respond is positive, the function `programSend()` will proceed once. The main purpose of this feature is to collect all the necessary data for the correction computation, pack them into an HEX string and send it to the RS. The first flow chart represents the algorithm principal in Figure 4.4.

The second function in the main loop `programRec()` is used for sending the received data from RS to the user. The function proceeds in specified interval `OUT_GAP`, which is equal to GPS receiver output frequency (1Hz). The function offers two options of which data will be sent to the user. The first option occurs when there is new corrected data from RS. The reception of corrected data must be done in the interval defined by `OUT_GAP`. Otherwise, the new corrected data will be discarded and function switch to the second option. The second alternative will just send the original, uncorrected NMEA data to the output.

This principal ensures that user will always be provided by the actual GPS data whether corrected or uncorrected. The rover unit provides only the basic NMEA sentences GPGGA and GPRMC since these messages are mostly used by the GPS-equipped devices because it provides a valuable positioning data.

When the corrected position is applied, it is indicated in the GPGGA fix quality word (the 6th word in the sentence). In addition, the RS ID and time from the last received correction is captured by the GPGGA (the 13th and 14th word in the sentence).

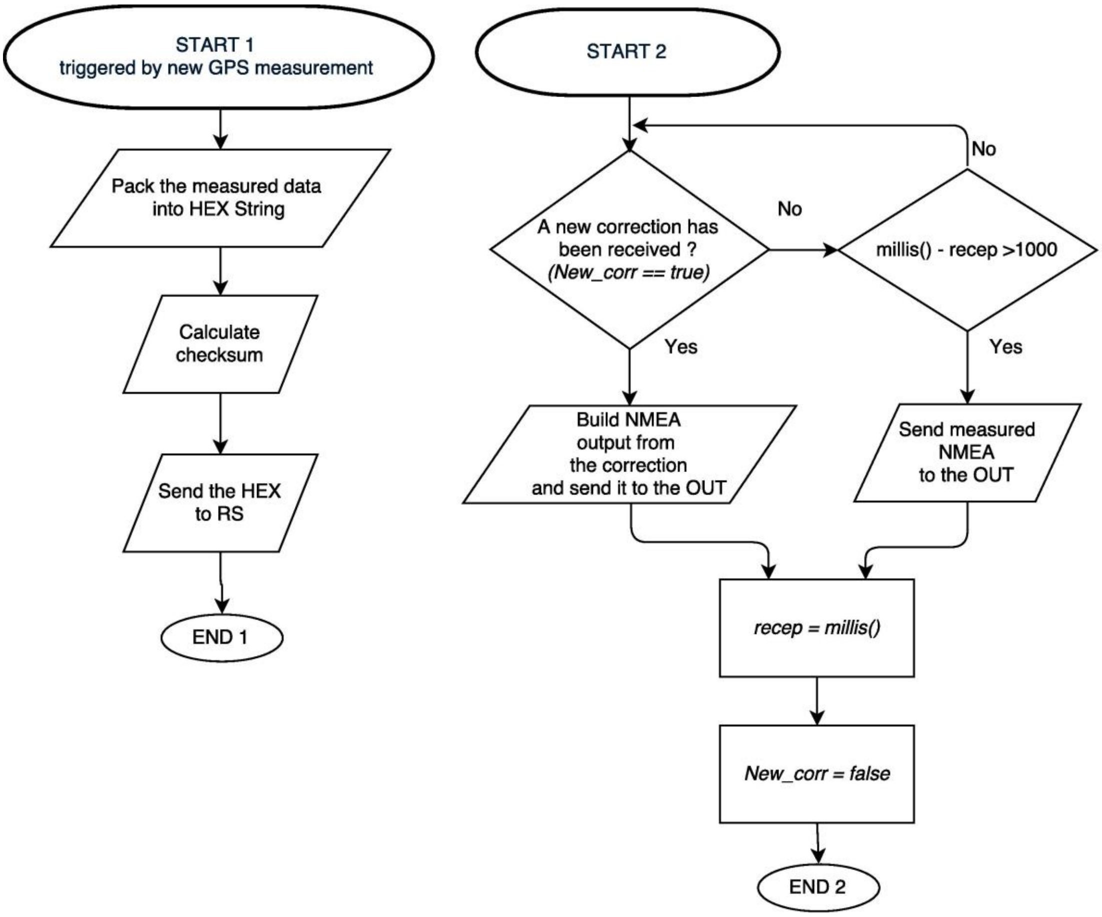

START 1
triggered by new GPS measurement
Pack the measured data into HEX String
Calculate checksum
Send the HEX to RS
END 1
START 2
A new correction has been received ? (New_corr == true)
No
millis() - recep >1000
No
Yes
Yes
Build NMEA output from the correction and send it to the OUT
Send measured NMEA to the OUT
recep = millis()
New_corr = false
END 2

Figure 4.4. Rover algorithms programSend(left) and programRec(right)

4.4.2 Reference Station (RS)

The RS is responsible for all calculations and provides system status info via attached TFT display. As same as the rover, the RS continuously capturing the GPS data from the connected GPS receiver and parsing them using the GPS library. The libraries used by RS and rover are not the same since both devices require different data. The library is adjusted individually for a particular station to save the memory space, especially on the rover side.

In the first part of the main sketch, there are optionally adjustable variables. These values should not be changed since it is set according to the shield hardware design. Section 5

The program also offers the debugging options, which can be enabled by uncommenting the line `#define DEBUG`. It will transmit a basic GPS data and RS status via the main Serial interface.

As described in Section 4.1 the RS will offer two modes of position correction computation. The main algorithm of the both modes is shown in Figure 4.5 and Figure 4.6. respectively.

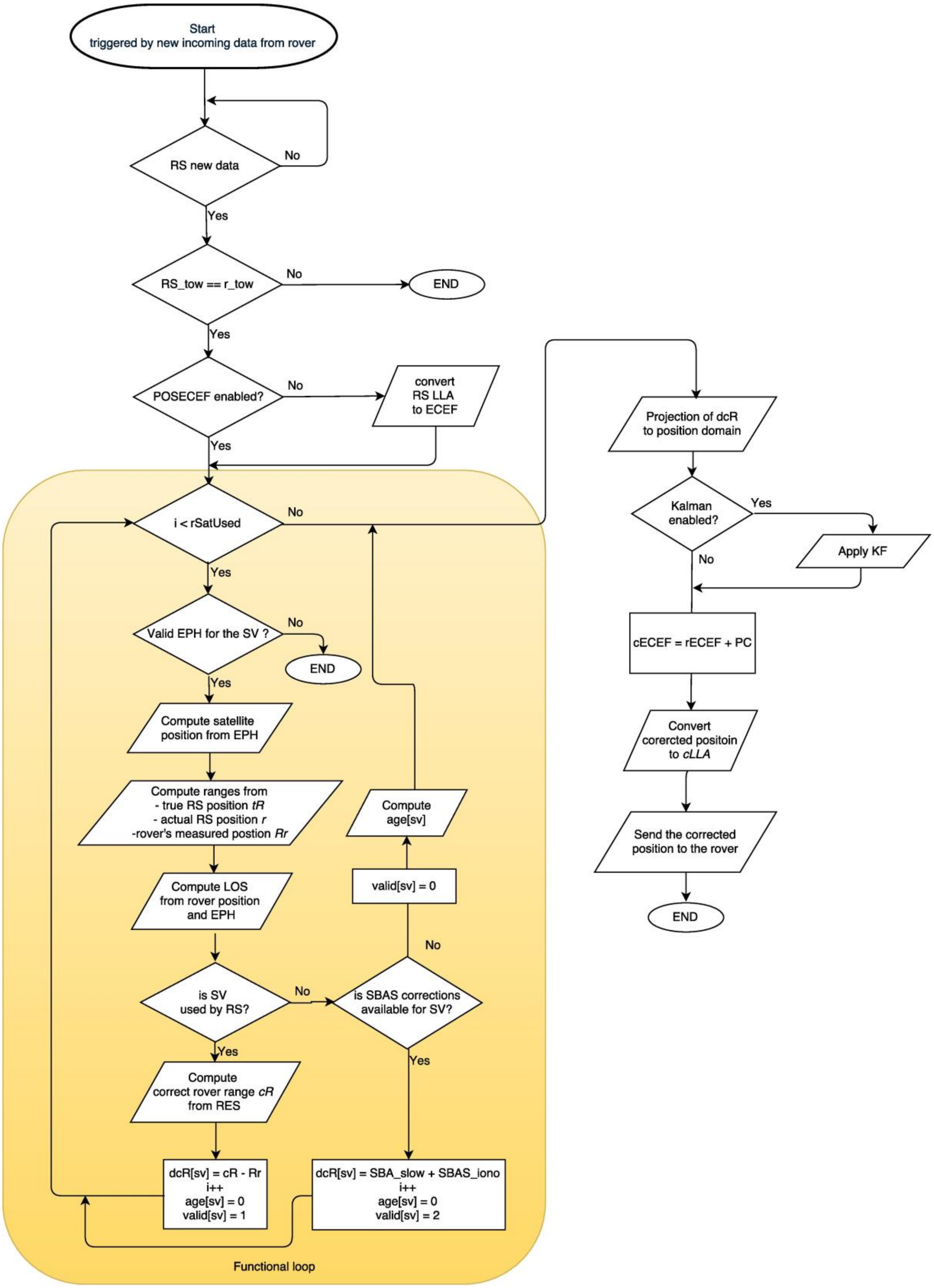


Figure 4.5. RS MODE 1

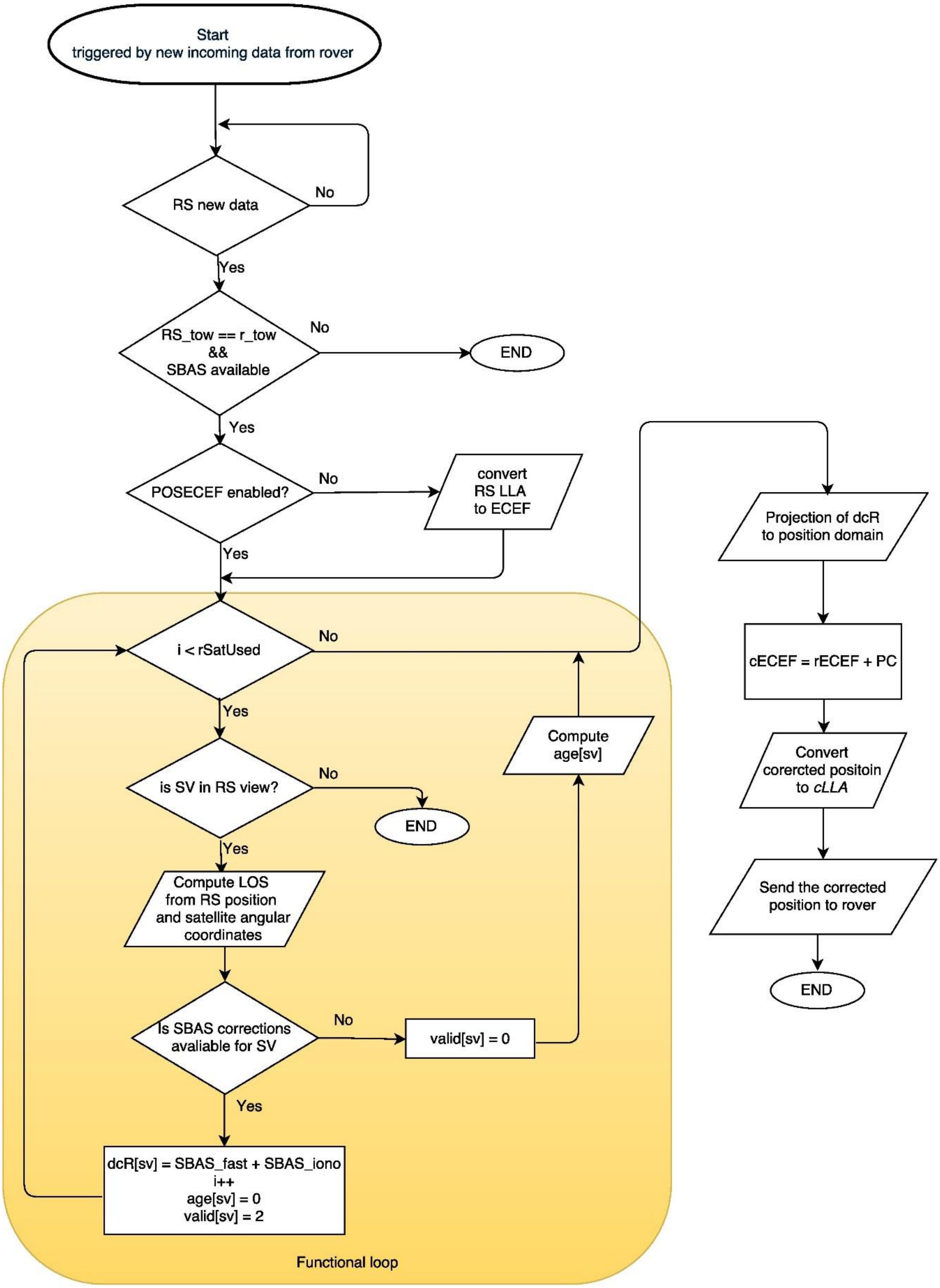


Figure 4.6. RS MODE2

4.4.2.1 MODE 1 (Range residuals)

The MODE1 algorithm is represented by the function `coreRES()` in the main Arduino sketch. Once RS receives new data from the rover, it calls the function. The next step is to check if the data are from the same sequence. This is done by comparing the timestamps `tow` from both stations. If the signal from the receiver is delayed and the data sets will be from the different sequence, the correction computation termites here and rover will not be provided by any corrected position. The GPS modules in both stations are set to synchronize the output by the GPS time. Thus, the time of the week generated by both receivers should be the same. This simple rule ensures the synchronization of both stations and avoids generating corrections for data, which are from different sequences.

In the next step, the algorithm checks if there are actual ECEF coordinates of the RS available. It is done by calling the variable `posecef`, which is set to HIGH when new UBX frame containing the ECEF coordinates has arrived. When the variable is false, the ECEF position is calculated from geodetic coordinates set.

When all input conditions are fulfilled, the algorithm proceeds to generate LOS and range corrections for all SVs used by the rover. The number of SVs used by rover is contained in the variable `rSatUsed` and specific SV values is stored in array `rSatUsedList`. The function goes through all the SVs stored in the array (Functional loop) and checks if there are valid ephemerides. If no, the function terminates, and no correction will be generated. The valid ephemerides are necessary to get the satellite coordinates (as described in Section 4.2.1.1) and for the ranges computations. Otherwise, if all ephemerides are available the program will continue to compute the satellite position, LOS from rover position, and ranges:

- From the true RS position to the satellite (`tr`)
- From the actual RS position to the satellite (`r`)
- From actual rover position to the satellite (`Rr`)

The next step is to check if RS also uses the rover's satellite. This is done in the same manner as the previous loop. However, the list of SVs used by rover is now compared with satellites employed by RS. If the result is positive, the PRC is computed by equation (4.2). In case that rover’s SV is not used by RS, the function tries to check if there is a SBAS correction available and optionally use that correction as PRC for the currently listed SV. It is indicated by setting the flag `valid[sv]` to appropriate value.

- Rover SV is used by the RS (`valid[sv] = 1`)
- Rover SV is not used by the RS, but SBAS is available (`valid[sv] = 2`)
- Rover SV is not used by the RS and SBAS is not available (`valid[sv] = 0`)
    - In case that `valid[sv] = 0` the function uses the previous value of PRC and update the age.

When the Functional loop is finished, it proceeds to the projection of PRC to position domain. The position correction `PC` is added to the actual rover position, which creates the corrected rover position `cECEF`. As discussed in Section 4.1.3, this method generates corrected position from originally observed pseudoranges, so the position is improved only in terms of accuracy. The optional Kalman filter can be applied, to improve the precision of corrected position coordinates, The corrected position is then packed to HEX String and sends back to the rover, which closes the whole process of correction computation.

4.4.2.2 MODE 2 (SBAS)

The second mode principal is shown in Figure 4.6. As seen from the figure, it works similarly as the mode 1. The initial stage of achieving synchronization has one more condition that checks if the SBAS correction data are available. If not the function terminates because the SBAS data are crucial for this mode since SBAS provides range corrections. The MODE2 functional loop is simpler since PRCs are already computed. It goes through all the satellites used by the rover and checks whether the SV used by rover is also in RS view. It is a critical condition because the RS must have the SV in view to get angular coordinates of the satellite from UBX-NAV-SVINFO. These angular coordinates are required for LOS computation, as described in Section 4.2.1.2.

When SBAS correction is available, and SV is in RS view, the functional loop proceeds to compute the LOS from Azimuth and Elevation angles and range corrections from SBAS. When the functional loop is finished, the last remaining thing is to project the range corrections to position domain and generates the corrected position, which is sent back to the rover.

Note that the UBX-NAV-SBAS provides fast and slow corrections mixed as one variable.

4.5 Data Transmission

Data transmission between rover and RS is realized by two cheap Wi-Fi modules ESP8266.

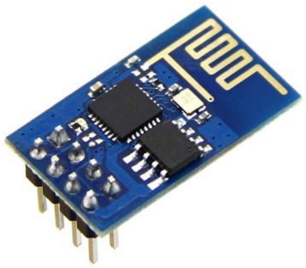


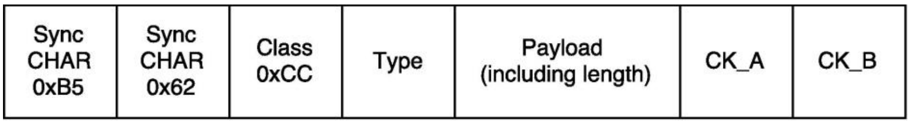
Figure 4.7. ESP8266 Wi-Fi module (www.seeedstudio.com)

This module is controlled via the serial connection by a set of AT commands. The AT commands are special ASCII strings, which are predefined for different devices (i.e. GSM). The set of AT commands defined for the ESP8266 are described in (Espressif, ESP8266 AT Instruction Set, 2015). The weakness of AT commands is the defined set of AT commands has to be sent to the unit at each startup to establish the functionality. This drawback can be removed since the ESP8266 offers direct programming in LUA language. Programming the device directly via LUA is less intuitive and more difficult. However, the program is saved in the ESP EEPROM and it is automatically loaded at each startup (LUA code for both stations is included in Appendix).

The ESP module at the RS side runs as an Access point, and rover is set to a client mode. Immediately after the startup, the units are set to their roles and client try to connect to the AP with defined SSID and password. When the connection is established, specific bytes are sent through the channel mutually to the both stations (RS, rover). It informs the stations that connection is established and the device is ready. The connection status is indicated on the display (RS) and by LED illumination (rover).

4.5.1 Data Formats

Data, which are sent over the TCP connection are carrying the position correction are using the simplified UBX communication protocol.



Sync CHAR 0xB5	Sync CHAR 0x62	Class 0xCC	Type	Payload (including length)	CK_A	CK_B

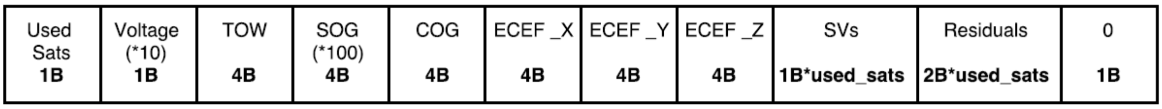
Figure 4.8. UBX packet format used for correction transmission

Reasons for using the UBX protocol is that the same algorithm can parse it as the other UBX data frames. That makes it simpler and consumes less MC memory.

The position correction packet is denoted by class 0xCC, which is not reserved by any other UBX packet. The next byte represents the data type. There several types of the transmission correction packet:

- 0xFF[7] – This type informs the station that there is no connection between rover and RS. This packet is periodically sent in a one-second interval.
- 0xFE[7] – denotes that connection was established. This byte is followed by the station ID in HEX. This type of sentence sends only once.
- 0x01 – denotes a new data from rover
- 0x02 – denotes a position correction packet

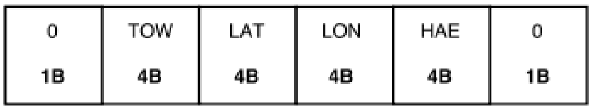
The payload length of message type 0x01 is variable, and it is derived from a number of used satellites. The payload structure is shown in *Figure 4.9*.



Used Sats	Voltage (*10)	TOW	SOG (*100)	COG	ECEF _X	ECEF _Y	ECEF _Z	SVs	Residuals	0
1B	**1B**	**4B**	**4B**	**4B**	**4B**	**4B**	**4B**	**1B*used_sats**	**2B*used_sats**	**1B**

Figure 4.9. Payload of message type 0x01

The payload of message type 0x02 has a constant length and contains correct position in LLA[8].



0	TOW	LAT	LON	HAE	0
1B	**4B**	**4B**	**4B**	**4B**	**1B**

Figure 4.10. Payload of message type 0x02

The data are sent as the ASCII string, which represents the HEX data divided by comma. This method shows better performance than sending the binary data itself and it is easier to parse the data on the other side. The data are thus packed into a HEX string and the last character "*" trigger the ESP to send the string. The receiver side then parses the data and sends it to the station in binary format via UART.

[7] This type of packet does not include the payload and checksum since they are not carry any information. They are used just for the connection status indication

[8] Corrected position is send in format dddmm.mmmm*10E5

4.6 Self-Accuracy Monitoring

This section discusses a proposal of accuracy and integrity monitoring of the Arduino DPGS.

4.6.1 Integrity Monitoring

The station monitoring is necessary to be able to detect any deficiencies, which may make the system unstable or unable to run in a proper way. The main integrity checks are done by the RS, which is more vulnerable for any malfunctions. The RS integrity monitoring runs in two steps. The first step is done in the boot sequence, which occur right after the startup. In the boot sequence, the RS checks whether there are all key features available, which are needed for proper functionality.

- RS GPS data availability
    - Check if the GPS receiver is connected
    - Check if there are enabled all required UBX sentences
- Position fix of RS (must be 3D)
- True position[9]

The RS GPS receiver activity check is done by calling the Arduino `Serial.availability()` function. This function will return zero if there are no incoming data to the defined serial port. It is useful when there is a wiring problem or basically if the GPS receiver does not work properly.

The UBX sentences are checked by calling a special boolean flag, which was set for each message type in the library. Once the UBX packet arrives, the flag is set to true, which indicates that particular messages are enabled. The 3D position fix is recognized from the NMEA GPGGA (6), and the true position of the RS must be entered by the user to enable the MODE 1.

All these requirements must be fulfilled before the RS will accept any data from the rover. It ensures that all parts of the system are accessible, and the correction computation will be done properly. All these initial steps are displayed on LCD. Thus, a user knows, which exacts part is not accessible.

The second part of integrity monitoring is done when the device is in operation. It continuously checks whether there are all required parameters available for selected mode.

- MODE 1 (Range residuals)
    - Ephemerides
    - Rover satellite is used by RS
    - SBAS correction is available
    - Ranger residuals

- MODE 2 (SBAS)
    - SBAS signal availability
    - SBAS correction for particular rover satellites
    - Rover satellites are in RS view

For the MODE 1, the ephemerides are key features since they are used for satellite position computation. The satellite position is necessary for ranges computation, which is important for range corrections estimation. Due to that importance, each

[9] Only for MODE 1

satellite ephemeris is equipped with a flag in the GPS library. It indicates whether the ephemeris was downloaded.

The comparison of used satellites is done in the functional loop. When there is a match, it sets the variable `valid[sv]`, which also indicates how the corrections are obtained. The variable also says whether there are range residuals available for defined SV. When the SV is in the RS navigation solution, this means, that the range residual is automatically computed by the receiver.

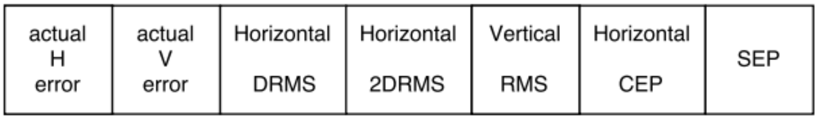
For the MODE2 the integrity monitoring is composed by checking the SBAS signal availability and the correction for particular satellites. It is done by checking the availability flag, which is carried by the UBX-NAV-SBAS message and correction validity by reading the boolean flag, which was set for each SV. The comparison of rover satellites set is made in the functional loop similarly as in MODE1.

All these values are displayed on LCD (See section 5.3)

4.6.2 Accuracy Monitoring

The accuracy monitoring is realized by computation of statistical values, which describes the actual correction behavior and give an idea about actual position error.

The RS computes its accuracy easily from the actual measured position and true position. This is done by transforming the coordinates to local coordinates system with origin in the true position. Actual measured position then shows an actual position error. The RS store the accuracy values in an array, which has following structure.

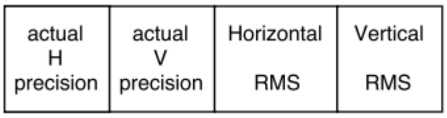


actual H error	actual V error	Horizontal DRMS	Horizontal 2DRMS	Vertical RMS	Horizontal CEP	SEP

Figure 4.11. RS Accuracy monitoring array structure

In the case of MODE2, there is no information about the true position. The accuracy values are then computed with respect to SBAS range correction.

The rover does not measure its accuracy because there is no information about the true position. The actual error measurements are computed concerning the average position. It results in precision measurements, which is stored in the array (figure 4.12). In other words, the RS is capturing the rover position from UBX frame via Wi-Fi and then computes the average. The actual position is then compared with this average, which results in actual precision measurements. This precision measurement is only valid for static measurements. When the rover produces any movement, this measurement will produce a nonsense values.



actual H precision	actual V precision	Horizontal RMS	Vertical RMS

Figure 4.12. Rover precision monitoring array structure

With each new measurement, the position is added to the previous one to be able to compute the standard deviations as described in Section 2.4.2. Standard deviations are computed for each direction. The values are stored in 8bytes variables thus; the computation of the average position is limited to approximately 1E10 measurements[10]. This number regards to hundreds of years of continuous operation. The range, over

[10] The number was computed from approximate value of ECEF coordinates

which the average is computed, can be set in the main sketch. The average variables are set to zero when the range is reached, and the measurement starts from the beginning.

5 Hardware Design

This chapter discusses the proposal of hardware structure and design of both stations. The first section briefly describes the individual parts of which the whole system is composed. It is followed by the wiring diagrams of both stations and design of the PCB shield. The last section shows how the system should be controlled and describes an information displayed on LCD.

5.1 Parts Description

This section briefly describes particular units used by the rover and RS. Both stations are using the same GPS receiver. It is the uBlox receiver NEO 6M.

5.1.1 GPS Receivers

The uBlox NEO6M is shown in Figure 3.3. NEO6M modules are designed for operational voltage 3.3V. Nevertheless, the board shown in Figure 3.3 is equipped with 3.3V stabilizer so that 5V can supply it. The device communicates with the Arduino board via Serial communication (UART). Since the module is supplied by 3.3V, the Serial connection must be shifted to 3.3V to avoid the module destruction.

From the functional point of view, the receiver offers basic positioning service in NMEA and UBX form. It can either accept the RTCM DGPS corrections. The NEO6 is equipped with SBAS engine, which can reserve up to three channels for SBAS (uBlox A. , 2011). The GPS module is one band only. It can receive the GPS L1 C/A code only. More about NEO6M can be found in the product summary (uBlox A. , 2011).

5.1.2 Wi-Fi ESP8266

See Section 4.5.

5.1.3 QVGA ILI9341

The information about station status and other information are provided via an LCD. LCD also works as GUI, which allows an initial setup of the device.

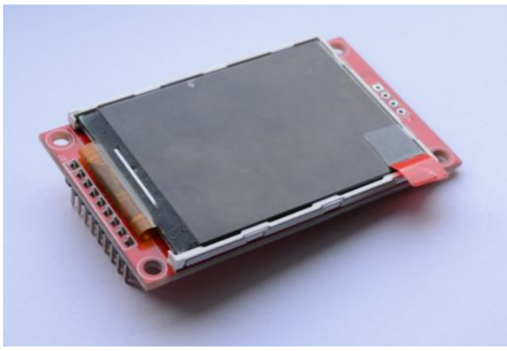


Figure 5.1. ILI9341 QVGA

The LCD has resolution 320x240px with 262K color. The supply voltage is as same as the GPS module 3.3v. The communication is provided via 8bits serial interface (SPI). More about technical data can be found in technical documentation (ILITEK).

The access to the LCD from the main program in RS is arranged via optimized ILI9341 library. This library is specially optimized for ARM microcontrollers, and it provides excellent performance than other libraries. The LCD control is very easy using this library. There are many functions, which are used for printing text, shapes, or numbers. Follow the official web pages http://pjrc.com/store/display_ili9341.html for more information about the library.

5.1.4 Arduino

The Arduino unit is a core of the whole system. It provides all the computations and control over the entire system.

The RS is using the Arduino Due. It is an enhanced unit, which is equipped with ARM 32-bit microcontroller, 54 digital IO, 4 UARTs, and 12 analog IO (Arduino, Arduino Due, 2016). The important feature of the ARM processor is that its supply power is 3.3V. That makes it easier for GPS and Wi-Fi modules since they also operate on 3.3V.

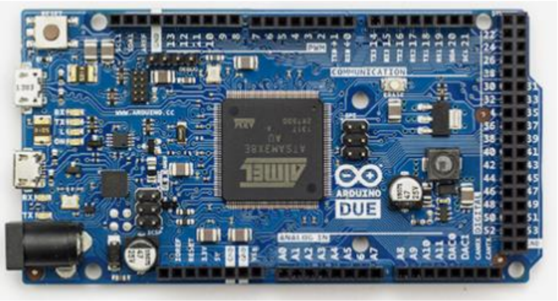


Figure 5.2. Arduino Due (Arduino, Arduino Due, 2016)

The rover station is using Arduino Mini. This device is equipped with Atmel AVR MEGA328. It is a 16-bit processor equipped with 14 digital, eight analog IO and one UART. The AVR processors operate on 5V levels. It means that the communication with GPS and Wi-Fi must be adjusted with the passive level shifter (Arduino, Arduino Pro Mini, 2016). The most important thing is to avoid overvoltage of the 3.3V modules. However, the output from these devices does not have to be adjusted since the 5V Arduino is capable of reading 3.3V logic because the minimum voltage of logic “1” is 2.7V.

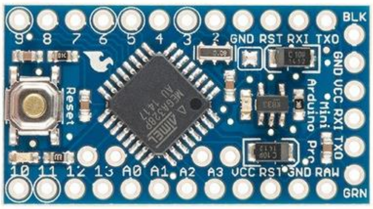


Figure 5.3. Arduino Pro Mini (Arduino, Arduino Pro Mini, 2016)

5.1.5 Wiring Diagram

The RS is designed as a shield, which fits the Arduino Due device. The RS shield consists of only of pull-up resistors and a voltage regulator. The additional Voltage regulator was used due to high power requirements of the ESP8266 unit, which internal power regulator, cannot provide.

The wiring diagrams were done in EAGLE environment. They are presented for both stations in following figures.

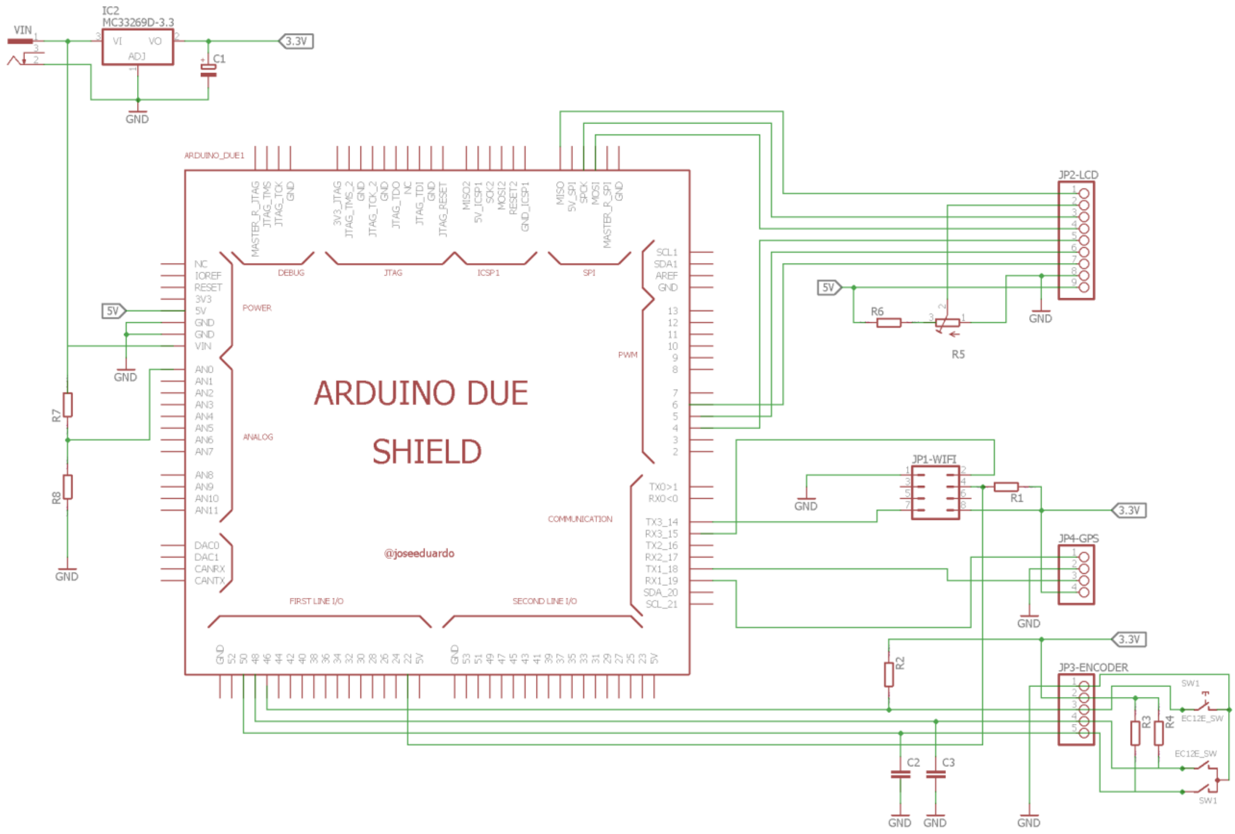


Figure 5.4. RS shield Wiring Diagram

A rover device shield (Figure 5.5) consists of a 3.3V voltage regulator for powering the Wi-Fi and GPS module. Due to that Arduino Pro MINI is a 5V logic, the signal wires must use a voltage divider to avoid overloading of the GPS and Wi-Fi modules. The input of the rover device is represented by the JP4-GPS and JP5-Wi-Fi, which collects a GPS data and RS corrections. The output is defined by JP1, which provides the NMEA GGA, and RMC sentences with the corrected position. The input voltage of both devices can be up to 16V.

The component list of both devices is included in Appendix B.

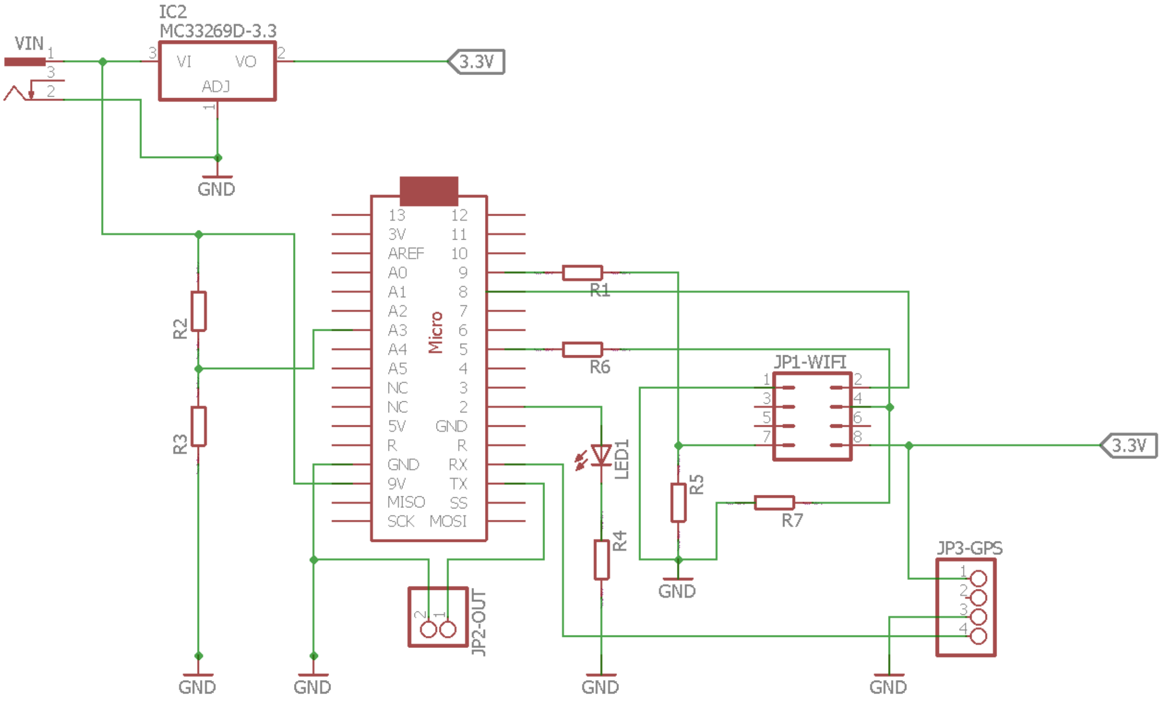


Figure 5.5. Rover shield Wiring Diagram

5.2 PCB Design

The physical shield proposal is intended on 1-sided PCB board. The PCB design was done in Sprint Layout (Figure 5.6).

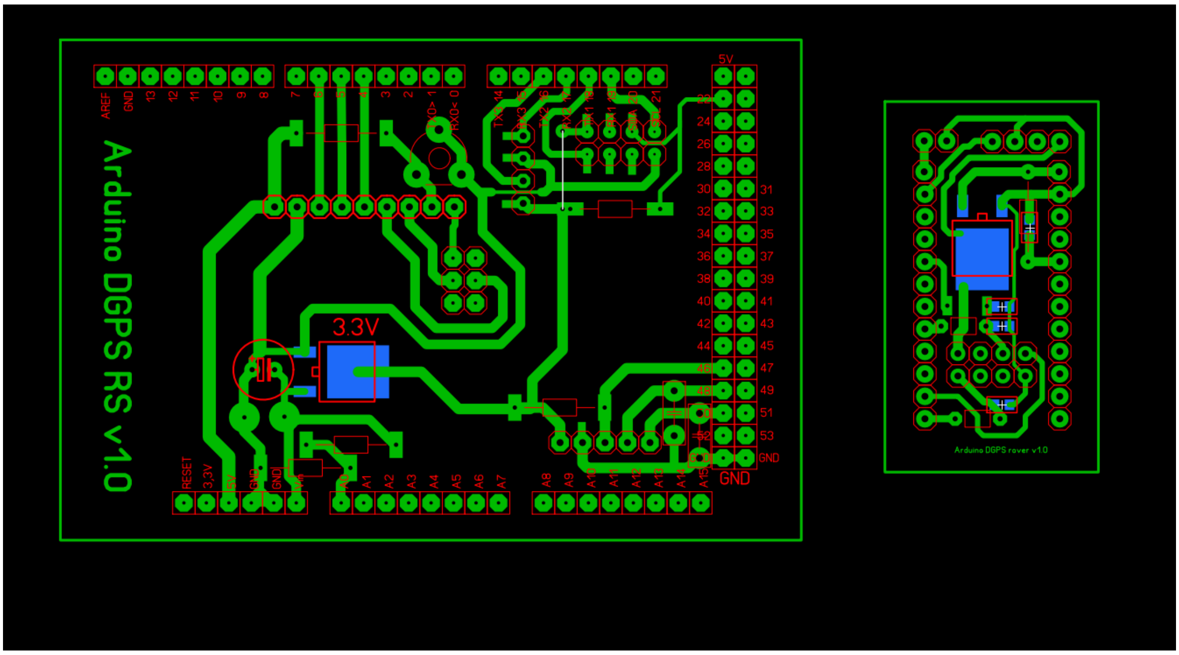


Figure 5.6. PCB layout (RS right, Rover left)

5.3 In Operation

This section contains operational instructions, which shows how to set the Arduino DGPS into operation. It will be described together with a legend, which helps to understand values on the LCD.

It is better to power up the RS first. It ensures that the RS Wi-Fi module will be set to AP mode before the rover starts surveying the network. When the rover is powered later, it may cause that connection pending will take a longer time.

After the RS is powered up, the welcome screen will be shown for a few seconds followed by the initial screen.

The initial screen shows the results of the integrity monitoring (step one) as described in Section 4.6.1. After the primary device check, a user is asked to select the mode, under which the RS will operate.

5.3.1 RS MODE1 (RES)

When Mode1 is selected, the RS checks whether there is GPS 3D position fix available. If yes, the form for the setting of the true position will appear. The user then set the true position of the RS derived from a map or gathered by a long-term measurement. Controlling of the RS is done by the rotary encoder with a confirmation button. To edit the value of the true position, the user has to select the desired value and then by pressing the button enter the editing mode. To exit the editing mode a user just presses the button again.

After the true position is set, a user selects whether the Kalman filter should be enabled or not. The selection is done by confirming the K_f Off or K_f On button on the screen. The program proceeds to the main screen after one of these selections is confirmed.

The main screen is divided into four sections. The upper (section one) is used for general information about current time and date gathered from the GPS. In the right upper corner, there is an annunciator dot, which indicates the communication with the rover. Every time the new rover data arrive, it is signalized by one blink of the dot. Similarly, when new correction is sent to the rover, it is indicated by the second blink of the dot.

The second section is used for actual condition monitoring of the RS (Section 4.6.1). This is done by displaying the most important values needed for smooth operation. See Figure 6.7.

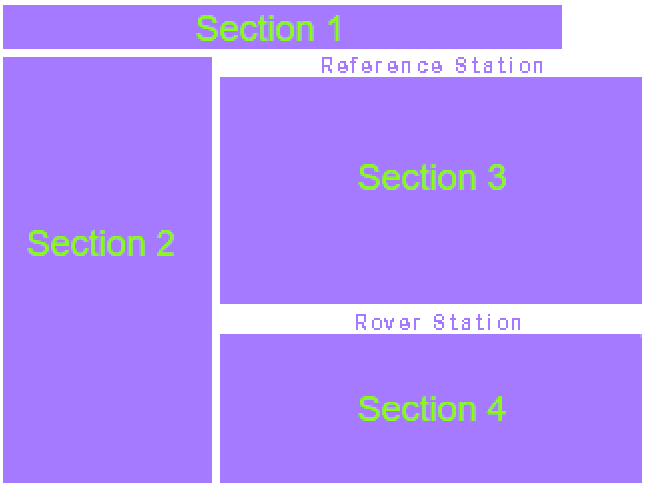


Figure 5.7. Display sections

The third and fourth section is used for the RS and rover status indication. It is structured as shown in Table 8 and Table 9.

Table 8. RS display section

Row No.	Displayed info
1	Actual measured position in LLA format
2	RS accuracy measurement
3	RS state (Week, TOW, number of used satellites, SBAS, Voltage)
4	Running time

Table 9. Rover displays section

Row No.	Displayed info
1	Corrected rover position in LLA format
2	Rover precision values
3	Rover status (Age of last correction, SOG, COG, Satellites, Voltage)

5.3.2 RS MODE2 (SBAS)

The main screen for MODE2 is practically the same as for MODE1. There is only one difference in the second section where there is only one flag signalizing the SBAS corrections availability for defined SV.

5.3.3 Rover Station

The rover is equipped with signalization LED. The LED can signalize four states.

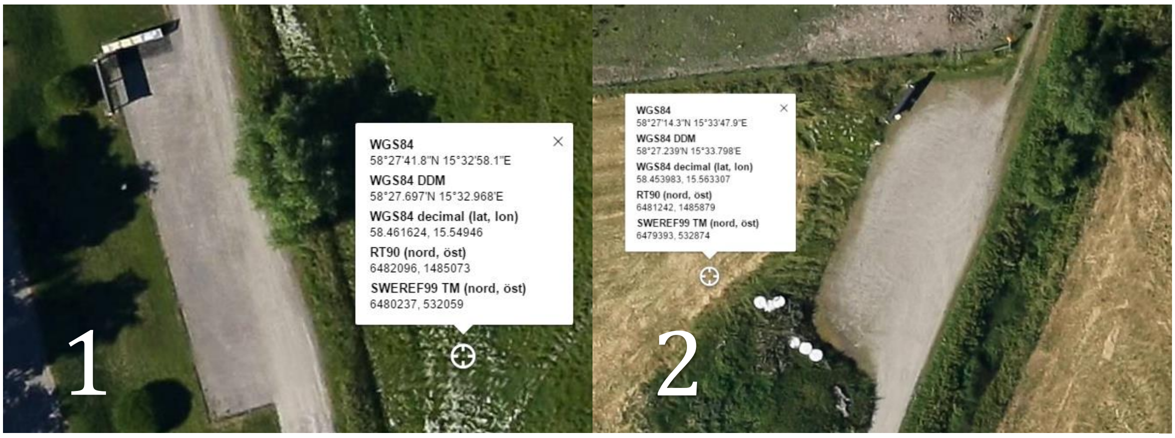
- OFF – The device has not established connection with RS
- ON – Connection with RS is established, but no data being sent to the RS
- One Blink per sec – data has been forwarded to the rover, but there is no respond from the RS
- Two Blinks per sec – the first blink signalizes that information has been sent as from the previous case and the second blink represents the received correction.

When the rover is powered up, it will automatically start surveying the network and connect to the RS SSID. After the connection was established and GPS fix is available, it automatically starts to send the data to the RS for correction. Then, the rover waits for corrected data. This algorithm proceeds as described in Section 4.4.1.

6 Test results

This chapter evaluates the Arduino DGPS solution regarding static and dynamic measurements, precision, and accuracy estimates, device technical requirements and comparison with similar projects.

Measurements were done on small parking lots near to the Roxen lake in Östergötland. There were two different places as shown in the figure below.



WGS84
58°27'41.8"N 15°32'58.1"E
WGS84 DDM
58°27.697'N 15°32.968'E
WGS84 decimal (lat, lon)
58.461624, 15.54946
RT90 (nord, öst)
6482096, 1485073
SWEREF99 TM (nord, öst)
6480237, 532059

WGS84
58°27'14.3"N 15°33'47.9"E
WGS84 DDM
58°27.239'N 15°33.798'E
WGS84 decimal (lat, lon)
58.453983, 15.563307
RT90 (nord, öst)
6481242, 1485879
SWEREF99 TM (nord, öst)
6479393, 532874

Figure 6.1. Location of the test fields (http://kartor.eniro.se)

The reason of selection of these two locations was the amount of reference points, which are used as a ground true. The Eniro maps were chosen as a source since they provide much more resolution over Sweden than i.e. Google.

6.1 Static measurements

The static measurements were done for both modes, each at different time and place. The measurements take approximately 45 minutes and during that, the rover and RS preserved its position.

6.1.1 MODE1 (RES)

The static measurement for the mode1 was done during the early afternoon on spot two, on Tuesday 31st of May 2016. The weather was partly cloudy, approximately 23°C, and light wind.

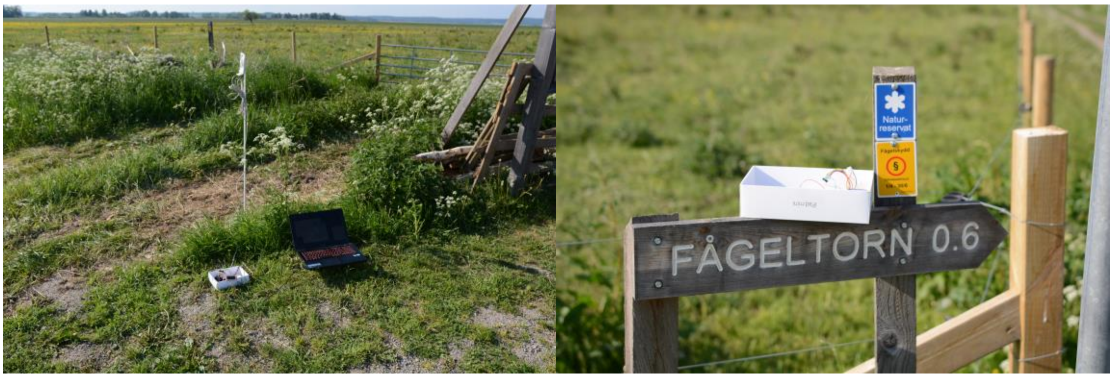


Figure 6.2. RS (left) Rover (right)

The result of the static measurement is presented in Figure 6.3. As seen from the illustration there is obvious accuracy improvement. However, the precision of the corrected position is a bit worse. The cyan measurement represents the filtered position by linear Kalman filter. The filter gives a small precision improvement, but it is not that precise as the original measurement.

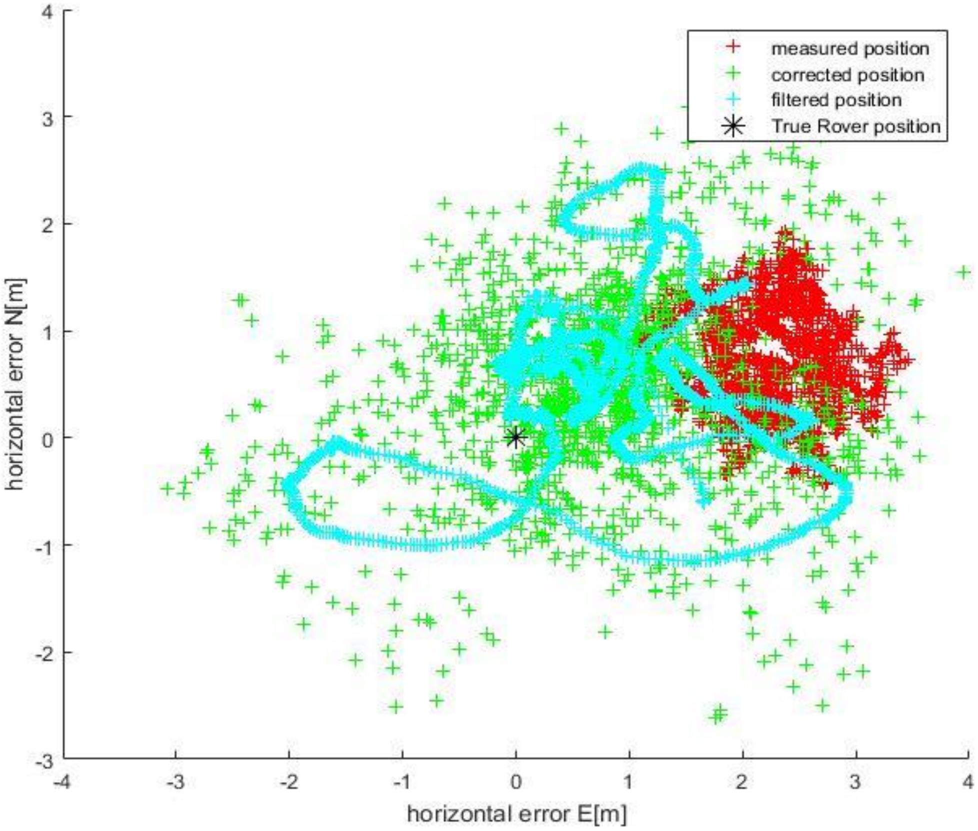


Figure 6.3. MODE1 - Static measurement

In the vertical plane, the accuracy improvement is not so apparent. The vertical data are very noisy. However, in the later part of the measurement the vertical error is quite low. From the Figure 6.4 is evident, that the largest error is resolved by the change of used satellites for which, the correction was calculated.

The horizontal accuracy improvement is also apparent from standard deviations and its RMS values as shown in table

Table 10. Accuracy comparison MODE1

Horizontal	Original	Corrected	Filtered
DRMS	2.47 m	1.72 m	1.57 m
CEP (50%)	1.93 m	1.43 m	1.31 m
Vertical			
σ_{up}	1.22 m	2.83 m	2.78 m

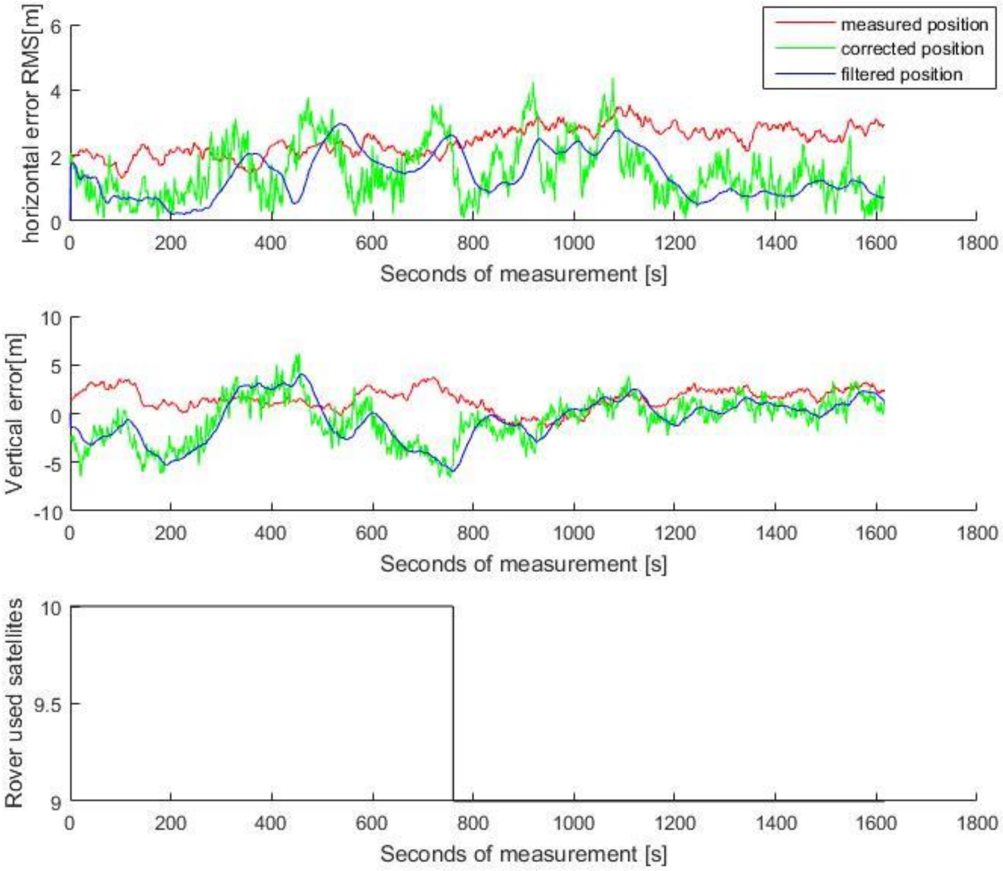


Figure 6.4. MODE1 Error comparison

6.1.2 MODE 2 (SBAS)

The SBAS static measurement was sorted out in the same conditions as the MODE1. However, it was measured at the first spot. The measurement also took approximately 45 minutes, and the results are shown in the figure bellow.

There is no such a big improvement since the original measurement was quite accurate itself. In compare with original measurement, the corrected position estimates are even slightly worse than original measurement. In the former part of the measurement, there was an error caused by unavailability of SBAS correction for the SV8, which occurs for approximately 5 minutes (figure 6.6). The time when this error was observed is also evident from the comparison figure 6.6. During this time, the correction was not generated, and it is projected as a gap in the graph. Another significant error happened when the satellite change occurred.

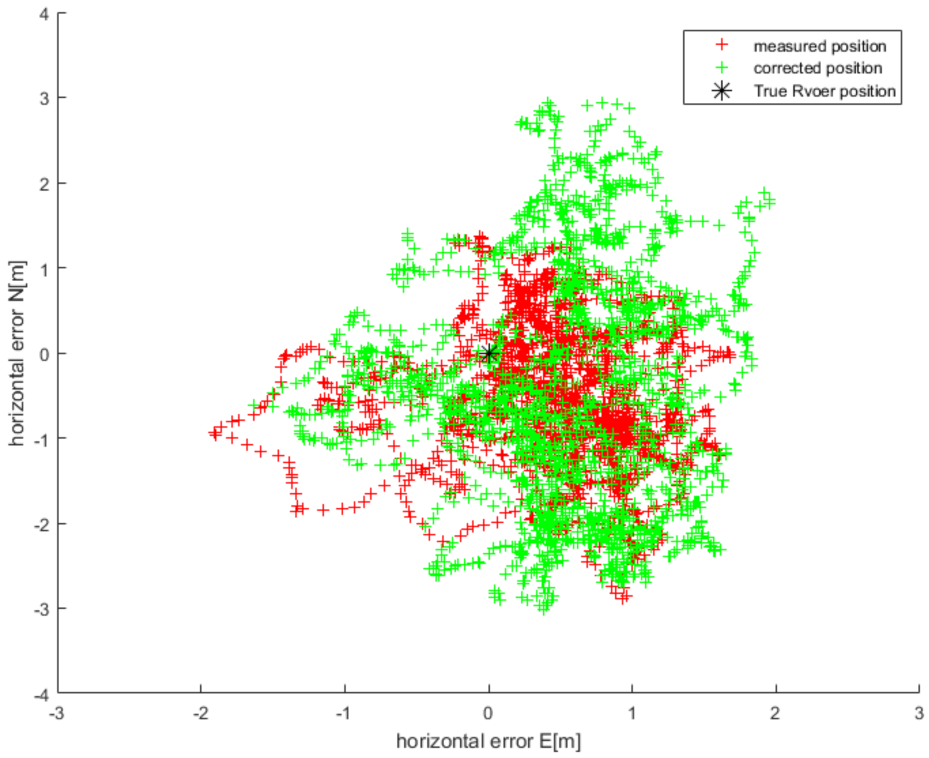


Figure 6.5. MODE2 – Static measurement

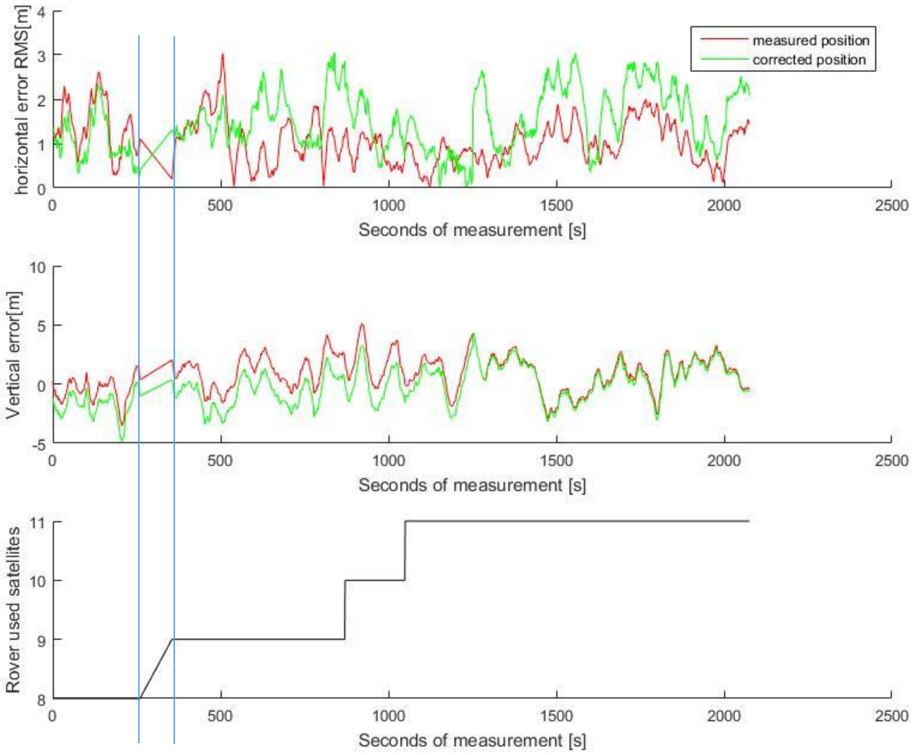
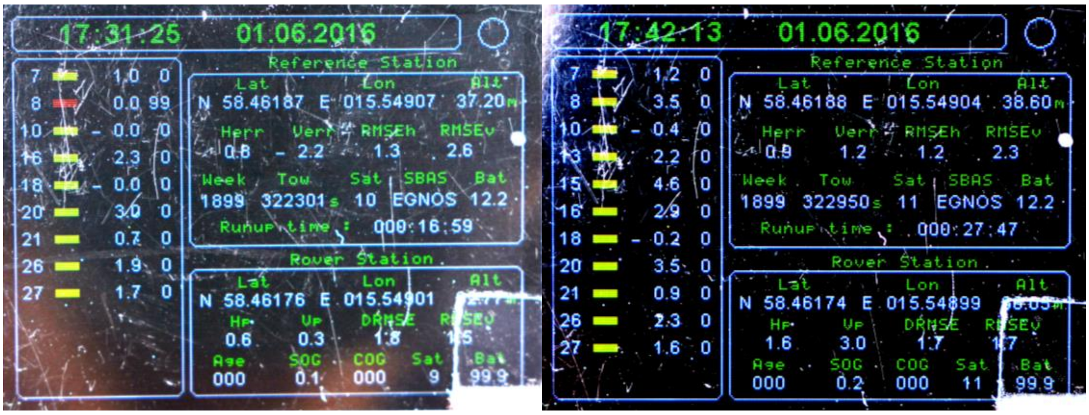


Figure 6.6. MODE2 – Error comparison

A status of the measurement was nicely recognizable from the LCD (figure 6.7)



```
17:31:25   01.06.2016
                 Reference Station
 7   1.0  0      Lat          Lon          Alt
 8   0.0 99   N 58.46187  E 015.54907  37.20m
10 - 0.0  0     Herr   Verr   RMSEh   RMSEv
16   2.3  0     0.8   - 2.2    1.3     2.6
18 - 0.0  0   Week   Tow      Sat  SBAS   Bat
20   3.0  0   1899  322301s   10   EGNOS  12.2
21   0.7  0     Runup time :  000:16:59
26   1.9  0            Rover Station
27   1.7  0      Lat          Lon          Alt
              N 58.46176  E 015.54901
                Hp     Vp    DRMSE   RMSEv
                0.6    0.3    1.8     1.5
              Age    SOG    COG    Sat    Bat
              000    0.1    000     9    99.9
```

```
17:42:13   01.06.2016
                 Reference Station
 7   1.2  0      Lat          Lon          Alt
 8   3.5  0   N 58.46188  E 015.54904  38.60m
10 - 0.4  0     Herr   Verr   RMSEh   RMSEv
13   2.2  0     0.9    1.2    1.2     2.3
15   4.6  0   Week   Tow      Sat  SBAS   Bat
16   2.9  0   1899  322950s   11   EGNOS  12.2
18 - 0.2  0     Runup time :  000:27:47
20   3.5  0            Rover Station
21   0.9  0      Lat          Lon          Alt
26   2.3  0   N 58.46174  E 015.54899
27   1.6  0     Hp     Vp    DRMSE   RMSEv
                1.6    3.0    1.7     1.7
              Age    SOG    COG    Sat    Bat
              000    0.2    000    11    99.9
```

Figure 6.7. MODE2 Error indication screenshots (occurred [Left] - resolved [right])

As can be observed from Figure 6.7, the SBAS range corrections were not significant. That was also projected to the vertical domain where the measured vertical error was slightly improved, but almost the same as an original measurement. The concrete values of accuracy are summarized in Table 11.

Table 11. Accuracy comparison MODE2

Horizontal	Original	Corrected
DRMS	1.18 m	1.65 m
CEP (50%)	0.97 m	1.34 m
Vertical		
σ_{up}	1.76 m	1.65 m

6.2 Dynamic measurements

The dynamic measurements were accomplished by walking in the defined pattern, which was delineated by reference points from Eniro maps. The pattern has the shape of the rectangle defined by the reference points in its corners.

6.2.1 MODE1 (RES)

The dynamic measurement was done during the midnight on the spot two on May 31, 2016. There was approximately 10°C, partly cloudy, and wind was calm. The measurement was done in two steps.

The first step was at a slow speed approximately 2km/h and the second about 5km/h. It can be seen in Figure 6.9**Error! Reference source not found.**. that the corrected position is closer to the ground truth, but the data are more spread around. The Kalman filter, which was used to improve this drawback, is plotted by cyan color. In the first case when speed was lower, the line is characterized by a large inertia. It is due to Kalman model, which does not take into account the actual speed of the body. In the other case, the line is conversely more smoothed, and it does not precisely follow the corrected position data when a sudden direction change occurs.

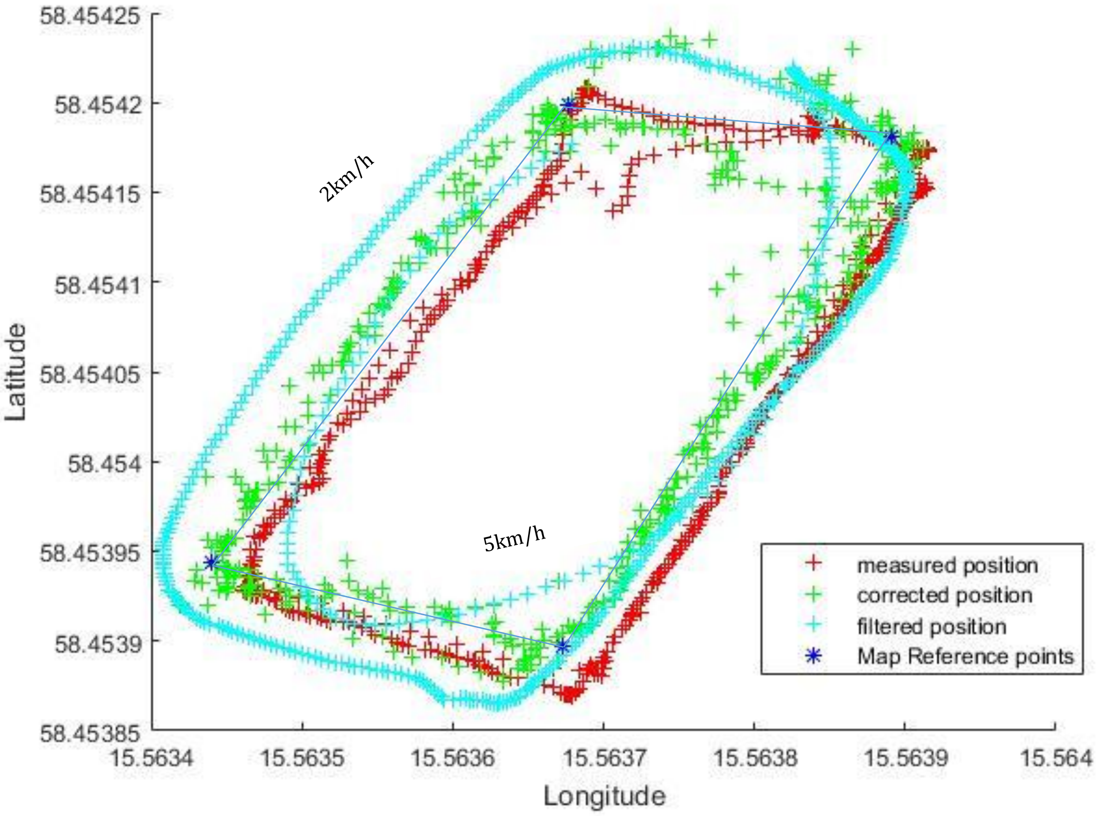
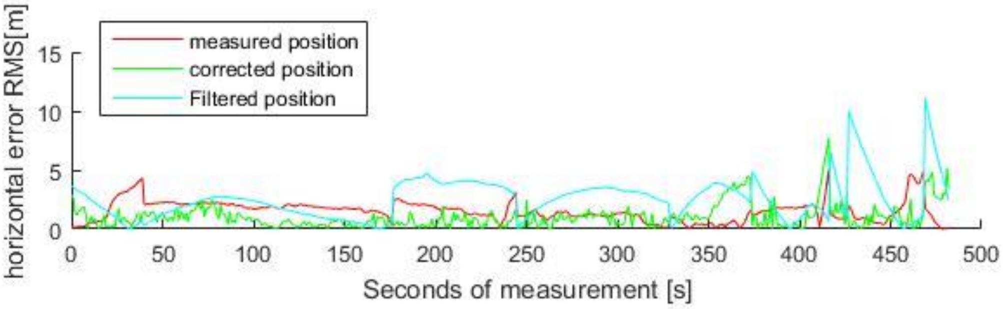


Figure 6.8. MODE1 - Dynamic measurement

The error comparison of each measurement is shown in Figure 6.9. The former part of the dynamic measurement (time 0-350s) represents the slower measurement where the filtered position is characterized by large inertia and small delay. The rest (time 350-480s) is characterized by large delay of the filtered position, which results in high inaccuracy represented by peaks, where a direction change occurs.



Horizontal	Original	Corrected	Filtered
Arithmetic Average	1.51 m	1.049 m	2.54 m
Vertical			
σ_{up}	1.49 m	2.37 m	2.14 m

Figure 6.9. MODE1 Error comparison

6.2.2 MODE2 (SBAS)

The dynamic measurement for the SBAS correction was done on spot one during late afternoon right after the static measurement. As seen from the Figure 6.10, the corrected position behaves in the proper way. It clearly follows the measurement, and because there were only small range corrections from SBAS, there is not a huge difference between measured and corrected position. However, it is apparent, that there is a small shift in the correct direction, so the accuracy in dynamical measurement

was slightly improved. The measurement was carried out in the same manner as the dynamic measurement for MODE1. However, there is not a big dilution from the true position even for fast and slow measurement. It is because of application of SBAS correction data to original filtered receiver position. It results in a position shift, which behaves similarly as the original measurement since the SBAS corrections are not so dynamic. During the dynamic measurement, there were not any satellite data unavailability.

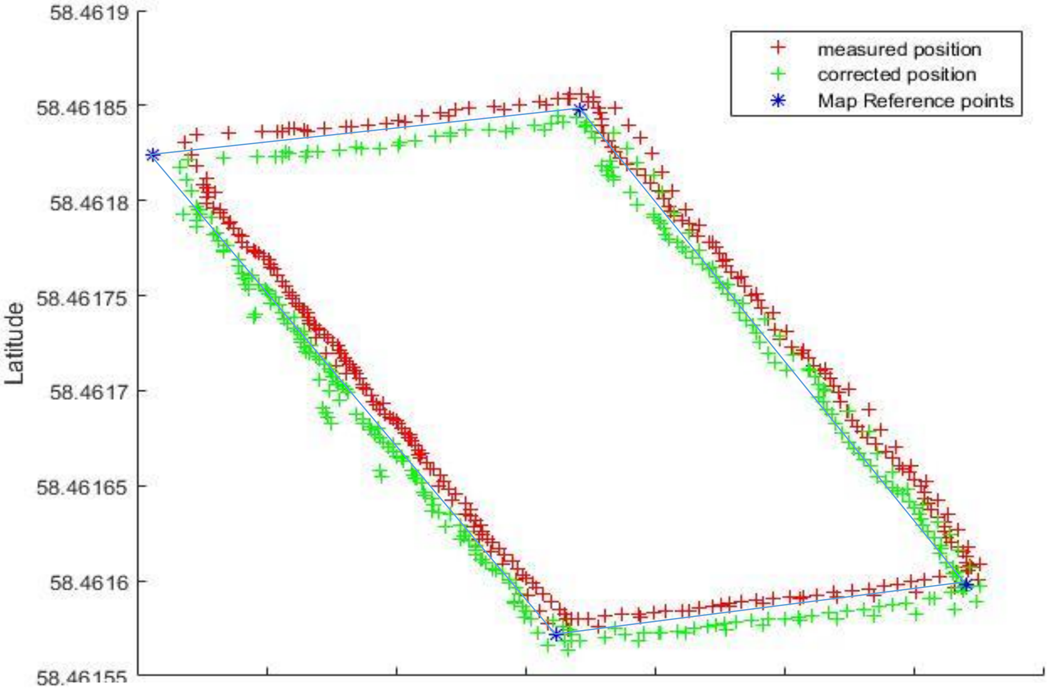
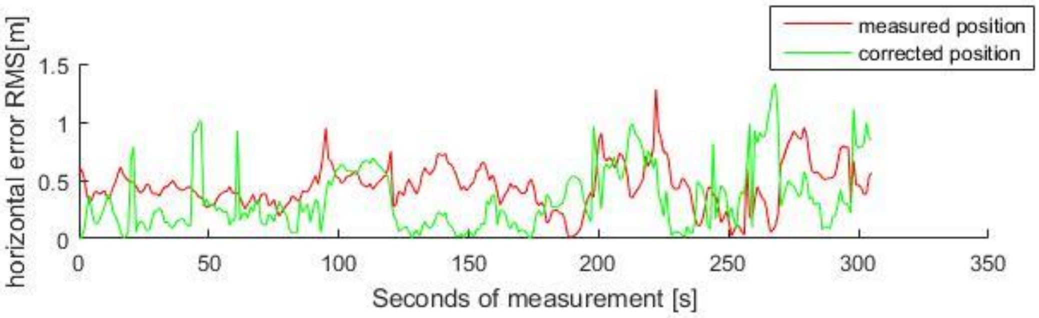


Figure 6.10. MODE2 - Dynamic measurement

The error comparison for MODE2 is shown on Figure 6.11. The generated error was not so significant since there was not a big dilution during the measurement.



Horizontal	Original	Corrected
Arithmetic mean	0.45 m	0.34 m
Vertical		
σ_{up}	1.01 m	0.96 m

Figure 6.11. MODE2 Error comparison

6.3 Device Power Requirements

The RS and rover power requirements are described in terms of measured power consumption during the IDLE state and the OPERATIONAL state. A seen from *Table 12*. the current consumed by both units is similar to both states. It is caused by powering the GPS and Wi-Fi module since they are designed as independent modules, so they are equipped with several peripherals.

Table 12. Stations Power Consumption comparison

	IDLE	OPERATIONAL
RS	270mA	Up to 300mA
Rover	170mA	Up to 220mA

The highest power requirement has a Wi-Fi module, which has a peak current up to 220mA (Espressif, ESPRESSIF SMART CONNECTIVITY PLATFORM, 2013). This fact raises demands for a power source, which has to be optimized according to maximal device requirements.

6.4 Result Benchmarking

In this section, there will be several comparisons with similar project done before. There will be an emphasis on two projects, which were also used as references.

The first project was done by Younsil Kim at Seoul National University. It refers to the implementation of DGPS principal to the NMEA data output. It was based on a projection of the range correction to position domain as same as the examples described above. However, the ephemerides and range corrections required for position correction estimation were generated by GUMC CORS server and downloaded by the RS. These data are sent to the rover stations where the position correction is calculated. The project was not focused on the standalone device design, but it only proceeds the application on a standalone computer. More information about this project can be found in (Kim, 2013).

The Kim's DGPS solution provides quite impressive results. With this method, the DGPS provides Accuracy about 0.946 m RMS horizontally and 1.09 m vertically. (Kim, 2013)

The second project done by Dong Hwan Yoon was dealing with same problematic as described by (Kim, 2013). He tried to make the DGPS more self - consistent, which was achieved by replacing the CORS data with SBAS correction and angular coordinates of the satellites in the same way as MODE2. The resulting accuracy levels were 1.048mRMS horizontally and 0.939 vertically.

7 Discussion

This Section provides an evaluation of the results of proposed Arduino DGPS solution. The discussion follows the main steps done in the thesis and examines possible applications and drawbacks, which may influence the Arduino DGPS operation.

7.1 Functional limitations

The Arduino DGPS solution is designed to augment the GPS positioning by using the popular Arduino devices and cheap and well available GPS receivers Ublox. These devices have several functional restrictions. The main limitation is in the usage of uBlox GPS receivers since the PRC computations are based on the UBX protocol, which others brands does not have to provide. The proposed solution is based on Arduino programming language, so the implementation of another platform could be complicated. The reason of using the Arduino in this project is the availability, price, and easy and understandable programming language. However, the Arduino devices provide limited performance, which makes them difficult to use in wider application (multiple rover stations).

There are multiple alternatives, which might be used. For example, Teensy 3.2 has the same size as Arduino NANO, but it is equipped with ARM processor as a CPU, which means that the performance will be much better than NANO. However, Teensy price is quite higher than Arduino. The platform selection highly depends on the application, on which the Arduino DGPS will be used. For example, if the system consists of only one reference station, it can use the Arduino. However, for multiple reference stations, it is better to use enhanced devices, which allows the PRC computation at the rover's side.

Proposed Arduino DGPS is developed to work with only one rover station. The rover station is attached to a body, which can be represented by flying objects (drones) or ground objects (robotics lawnmowers and autonomous vehicles). Especially for the flying bodies, the dimension and weight of the rover unit play an important role. Both applications (ground, air) are able to use both modes (RES, SBAS) without any restrictions. Everything depends on the RS, which has to have a good view to be able to get all necessary data for PRC estimation.

The correction data transmission is designed to be dedicated since only one rover station is assumed. This is a major limitation. The correction computation at the RS takes up to 50ms (depends on the number of used satellites), so theoretically, it would be able to handle more than one rover. In that case, the RS will have to compute the PRCs for each station individually since each station can work with different set of satellites. This fact will cause increase of the memory usage and CPU load. Application with more than one rover station was not tested.

7.2 Measurement results evaluation

The measurement was done separately for each mode. The RS was located at the open area during calm weather. The distance between RS and rover station does not exceed more than 200m. However, the distance can be much longer as proved in another project (up to 400m with integrated antennae and up to several kilometers when using external antennae (Benchoff, 2014)), but it was not tested in this project. The proposed distance is sufficient for small ranges lawnmowers and autonomous vehicles. For drones and autonomous flying objects, it is better to equip the rover device by external antennae to avoid unpleasant signal lost.

The MODE1 results presented in Sections 6.1 and 6.2 shows that there is an accuracy improvement in horizontal plane when using the range residuals for PRC computation. On the other hand, there is quite a huge noise in this results caused by range residuals inaccuracy. During the static measurement, there was apparent inaccuracy in original measurement, which was improved by applying corrections computed from range residuals. As showed on Figure 6.3, the accuracy was improved by about 30% (2.47m – 1.57m DRMS) in the horizontal plane. In the vertical plane, the accuracy improvement was not as good as in the horizontal plane (1.22 – 2.78m). However, the inaccuracy is resolved by a change of rover's used satellites. That indicates that the error caused before this change was due to lack of data from one satellite (it was indicated by the red flag when the rover satellite was not included in the RS solution).

It seems that dynamic measurement, done for the first mode, produces similar results. It also shows visible improvement in the position accuracy, but the noise makes an appreciable position distortion caused by the range residuals model. The vertical distortion was also even worse than the static measurement. The accuracy values were computed as the average perpendicular distance from the ground truth position, which is represented by a line between two reference points.

The dynamical changes in the position are reduced by applying a Kalman filter. However, the Kalman filter requires a time demanding tuning of covariance matrices. For the test measurement, the covariance matrices were set according to Matlab simulations. As seen from Figure 6.8, the filter intensively influences the corrected position and causing the softening of the data. Another drawback of the wrongly set filter is the position delay, which is apparent from Figure 6.9. The error caused by the filter delay can be very high when the rover moves faster. The well-designed filter should provide very good performance for dynamic applications; otherwise, the error caused by the filter can cause much higher error than original measurement, which may lead to fatal consequences.

During the MODE2 measurement, there was a small error, so the correction is not as evident as in the MODE1. In some phases, the accuracy is an even little bit worse than original measurement (1.18 – 1.65m DRMS in vertical domain and 1.76-1.65 in vertical domain). That might be caused by incomplete SBAS correction data carried by the NAV-SBAS sentence. However, the error variance is very small in compare to original measurement. It might be caused coincidentally during the position-domain correction computation.

In the dynamic measurement, there was the same error as in the static measurement. However, in dynamic measurement, the correction algorithm produces slightly better accuracy in the vertical and horizontal domain.

Errors in the dynamic measurements are highly influenced by the ground truth estimation. The ground truth for the test measurements was set according to ENIRO map bases. It may cause a high uncertainty in true position and reference points estimations. This issue can be resolved by estimating the reference point true positions by long-term measurements and comparing the measured data with CORS server data to acquire the errors, which might influence the long-term measurement.

7.3 Future development

The next step in the Arduino DGPS development will be an implementation of the DGPS units to the real rover device (RC drone using GPS for home returning) and verify the functionality in the real application. It will involve proving the ESP8266 performance by estimation of the possible range between rover and RS with integrated PCB and external, antennae.

During and after this testing, the necessary Kalman filter tuning will be done in order to find proper values of Kalman filter error covariance matrices and implementation of Extended Kalman Filter (EKF), which could provide more accurate filtering results.

8 Conclusion

The Arduino DGPS solution is designed with respect to component availability and prices. The realization costs for two modes Arduino DGPS unit does not exceed 100EUR, and all components are easily available on today's market. All of the Arduino DGPS modules can be easily assembled, which make it easy to use for amateurs at home environments.

To avoid unpleasant inaccuracy, the device provides the function of integrity monitoring, which is used to detect any missing input data. This integrity monitoring shows its usefulness during the MODE2 measurements, where the missing data could cause a big accuracy dilution.

The measurements showed that the accuracy improvement in vertical domain highly depends on the system integrity. It was proved that one missing satellite correction might cause a big variance in vertical domain. The MODE1 measurements showed better accuracy improvements in static terrestrial measurements (1,57m horizontally). Well-optimized Kalman filter should make the MODE1 also usable in a dynamic manner.

The MODE2 measurement showed a good accuracy improvement in dynamic measurements with average accuracy 0.34m horizontally and 0.96m vertically.

The proposed Arduino DGPS solution has a big potential for private owners of autonomous devices such as drones or lawnmower where the expensive solutions are not possible to implement due to high costs.

9 References

Arduino. (2016). *Arduino Due*. Retrieved May 29, 2016, from https://www.arduino.cc/en/Main/ArduinoBoardDue

Arduino. (2016). *Arduino Pro Mini*. Retrieved May 29, 2016, from https://www.arduino.cc/en/Main/ArduinoBoardProMini

Bailey, J. (1962, February). *THE LORAN-C SYSTEM OF NAVIGATION.* Retrieved January 21, 2016, from http://www.loran-history.info/: http://www.loran-history.info/Loran-C/Jansky%20_%20Bailey%201962.pdf

Bauer, A. (2004, December 26). *Some historical and technical aspects of radio navigation, in Germany, over the period 1907 to 1945.* Retrieved March 12, 2016, from http://www.cdvandt.org/Navigati.pdf

Benchoff, B. (2014, September 26). *ESP8266 Distance testing.* Retrieved July 1, 2016, from http://hackaday.com/2014/09/26/esp8266-distance-testing/

Carnegie, D. (2005). The design of Pair of Identical Mobile Robots to Investigate Co-Operative Behaviours. *i-techoline*, 390-391.

CDDIS-NASA. (n.d.). *Geopotential Model EGM96.* Retrieved April 17, 2016, from The Development of the Joint NASA GSFC and the National Imagery and Mapping Agency (NIMA): http://cddis.nasa.gov/926/egm96/TOC.HTML

Dana, P. H. (1994). *Global Positioning System Overview*. Retrieved February 13, 2016, from http://www.colorado.edu/geography/gcraft/notes/gps/gps_f.html

Dunn, M. J. (2013, September 24). *GPS space segment/Navigation User Interfaces.* Retrieved March 24, 2016, from http://www.gps.gov/technical/icwg/IS-GPS-200H.pdf

ESA. (2005). The EGNOS fact sheet. Retrieved May 25, 2016, from http://www.mi.gov.pl/1/files/0/3141/fact_sheet_12.pdf

ESA. (2015). EGNOS Open Service. Retrieved May 24, 2016, from https://egnos-user-support.essp-sas.eu/new_egnos_ops/sites/default/files/library/official_docs/egnos_os_sdd_v2_2.pdf

Espressif. (2013). *ESPRESSIF SMART CONNECTIVITY PLATFORM.* Retrieved June 3, 2016, from https://cdn-shop.adafruit.com/datasheets/ESP8266_Specifications_English.pdf

Espressif. (2015). ESP8266 AT Instruction Set. Retrieved May 28, 2016, from https://cdn.sparkfun.com/assets/learn_tutorials/4/0/3/4A-ESP8266__AT_Instruction_Set__EN_v0.30.pdf

FAA. (2008). WIDE AREA AUGMENTATION SYSTEM (WAAS) PERFORMANCE STANDARD. Retrieved May 25, 2016, from http://www.gps.gov/technical/ps/2008-WAAS-performance-standard.pdf

Forssell, B. (2008). *Radionavigation systems.* Norwood MA: Artech House.

Gakstatter, E. (2013). *Sources of Public, Real-Time, High-Precision Corrections.* GPS - world. Retrieved May 24, 2016, from http://gpsworld.com/sources-of-public-real-time-high-precision-corrections/

Guard, U. C. (2015, September 04). *LORAN-C REFERENCE INFORMATION*. Retrieved December 30, 2015, from navcen.uscg.gov: http://www.navcen.uscg.gov/?pageName=loranReference

Gurtner, W. (2007). *The Receiver Independent Exchange Format.* UNVACO. Retrieved May 24, 2016, from ftp://ftp.unibe.ch/aiub/rinex/rinex301.pdf

Hart, M. (2016). TinyGPS++. Retrieved May 26, 2016

Hegarty, C. J. (2008). *Evolution of the Global Navigation SatelliteSystem (GNSS).* IEEE. doi:10.1109/JPROC.2008.2006090

Hofmann-Wellenhof, B. (2001). *GPS - Theory and Practice.* Wien: Springer Wien New York.

Hornish, A. (n.d.). *LORAN Day and Night Coverage, 1945*. Retrieved January 21, 2016, from timeandnavigation.si.edu: https://timeandnavigation.si.edu/multimedia-asset/loran-day-and-night-coverage-1945

ILITEK. (n.d.). a-Si TFT LCD Single Chip Driver 240RGBx320 Resolution and 262K color Specification. Retrieved May 29, 2016, from https://cdn-shop.adafruit.com/datasheets/ILI9341.pdf

Instruments, N. (2016). *GPS Receiver Testing.* National Instruments. Retrieved May 23, 2016, from http://www.ni.com/white-paper/7189/en/

Kim, Y. (2013). *DGPS enhancement to GPS NMEA output data: DGPS by correction projection to position-domain.* Research gate. Retrieved January 21, 2016, from https://www.researchgate.net/publication/259431030_DGPS_enhancement_to_GPS_NMEA_output_data_DGPS_by_correction_projection_to_position-domain

Madron, T. (2009). *DIFFERENTIAL GPS.* Retrieved March 4, 2016, from https://www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=14866

NAP. (2010). Precise Geodetic Infrastructure: National Requirements for a Shared Resource. (pp. 89-100). National Academic Press. doi:10.17226/12954

NAVSTAR. (1995, June 2). *GLOBAL POSITIONING SYSTEM STANDARD POSITIONING SERVICE SIGNAL SPECIFICATION.* Retrieved May 23, 2016, from http://www.gps.gov/technical/ps/1995-SPS-signal-specification.pdf

NGA, N. G.-I. (n.d.). *EGM96 geoid*. Retrieved April 17, 2016, from NGA GEOnet Names Server: http://earth-info.nga.mil/GandG/images/ww15mgh2.gif

NovaTel. (2000). *Discussions on RF Signal.* NovTel. Retrieved April 9, 2016, from http://www.novatel.com/assets/Documents/Bulletins/apn008.pdf

NPTEL. (n.d.). *Carrier phase based measurements*. Retrieved February 10, 2016, from http://nptel.ac.in/: http://nptel.ac.in/courses/Webcourse-contents/IIT-KANPUR/ModernSurveyingTech/lecture4/4_3_Carrier_phase.htm

Park, B.-W. (2014). *Algorithm, A Study on the DGPS Service Utilization for the Low-cost GPS Receiver Module Based on the Correction Projection.* Seoul: School of Aerospace Engineering - Sejong University. Retrieved February 5, 2016, from https://www.researchgate.net/publication/263629615_A_Study_on_the_DGPS_Service_Utilization_for_the_Low-cost_GPS_Receiver_Module_Based_on_the_Correction_Projection_Algorithm

Rapant, P. (2002). *Družicové polohové systémy.* Retrieved January 21, 2016, from http://gis.vsb.cz/vojtek/content/gnps/files/_source/RAP02.pdf

Slavíček, J. (2007). *Navigační systémy.* České Budějovice: Soukromá vyšší odborná škola a Obchodní akademie s.r.o. Retrieved March 16, 2016, from http://www.petrpexa.cz/diplomky/slavicek.pdf

Stansell, T. (1978). *The Transit.* Magnavox. Retrieved January 21, 2016, from https://www.ion.org/museum/files/TransitBooklet.pdf

Subirana, J. S. (2011). *GPS Navigation Message.* Retrieved May 23, 2016, from Navipedia: http://www.navipedia.net/index.php/GPS_Navigation_Message

Subirana, J. S. (n.d.). *Klobuchar Ionospheric Model.* Retrieved February 29, 2016, from www.navipedia.net: http://www.navipedia.net/index.php/Klobuchar_Ionospheric_Model

The Department of Energy, M. a. (1995). *GPS POSITIONING GUIDE.* Otawa: Natural Resources Canada. Retrieved February 6, 2016, from http://www.utdallas.edu/~aiken/GPSCLASS/GPS_Guide_e.pdf

U.S. Air Force. (n.d.). *GPS Space Segment*. Retrieved January 6, 2016, from http://www.gps.gov/systems/gps/space/

U.S.A.F, U. A. (n.d.). *GPS - Control Segment*. Retrieved March 22, 2016, from GPS.gov: http://www.gps.gov/systems/gps/control/

uBlox. (2013, April 18). *uBlox 6 Receiver Description and Protocol Specification.*

uBlox, A. (2011). *uBlox NEO6 Series Product Description.* Retrieved March 21, 2016, from https://www.u-blox.com/sites/default/files/products/documents/u-blox6_ReceiverDescrProtSpec_%28GPS.G6-SW-10018%29_Public.pdf

Yoon, D. H. (2014, March). A Feasibility Teston the DGPS by Correction Projectio Using MSAS Correction. *JPNT - Journal of Positioning NAvigation and Timing*, 2-4. doi:10.11003/JPNT.2014.3.1.025

Zornoza, J. J. (n.d.). *Tropospheric Delay.* Retrieved February 29, 2016, from www.navipedia.net: http://www.navipedia.net/index.php/Tropospheric_Delay

Appendix A (CD content)

Due to length of the program sketches, the Arduino and Lua codes are included on the attached CD.

- Reference Station
  - Arduino sketch (DGPS_core.ino)
  - ESP8266 LUA code (RS.lua)
  - Reference Station schematic Eagle (RS.sch)
  - Reference Station PCB layout (RS.lay)

- Rover Station
  - Arduino sketch (rover.ino)
  - ESP8266 LUA code (rover.lua)
  - Rover Station schematic Eagle (rover.sch)
  - Rover Station PCB layout (RS.lay)

- Datasheets
  - ESP8266.pdf
  - NEO-6_DataSheet_(GPS.G6-HW-09005).pdf

Appendix B (list of components)

Table 13. RS Shield

Component name	Component ID	Value
Resistor	R1, R2, R3, R4	10kΩ
Resistor	R6	100Ω
Resistor	R7	12k7Ω
Resistor	R8	3k3Ω
Trimer	R5	2k5 Ω
Capacitor Electrolyte	C1	10uF/16V
Capacitor Ceramic	C2, C3	100nF
Voltage regulator 3.3V	IC2	MC33269D
Pin header	JP1, JP2, JP3, JP4	2x4, 9, 5, 4

Table 14. Rover Station Shield

Component name	Component ID	Value
Resistor	R1, R6	4k7Ω
Resistor (SMD 0603)	R5, R7	10kΩ
Resistor	R2	12k7Ω
Resistor (SMD 0603)	R3	3k3Ω
Resistor	R4	330 Ω
Voltage regulator 3.3V	IC2	MC33269D
Pin header	JP1, JP2, JP3, JP4	2x4, 9, 5, 4