Design and Evaluation of various miniaturized GPS modules

eingereichte
PROJEKTPRAKTIKUM
von

Akram Ben Halima

Hanen Soufi

wohnhaft in:
München
Tel.: 017632026167
Tel.: 017663045209

Lehrstuhl für
STEUERUNGS- UND REGELUNGSTECHNIK
Technische Universität München
Univ.-Prof. Dr.-Ing../Univ. Tokio Martin Buss
Univ.-Prof. Dr.-Ing. Sandra Hirche

Betreuer:	Prof. Dr. sc. nat. Jörg Conradt
Beginn:	19.04.2010
Zwischenbericht:	07.06.2010
Abgabe:	16.07.2010

Abstract:

In the last decade, the Global Positioning System has proved his relevance being one of the most accurate and the most reliable navigation systems available in the market. The remarkable characteristic about GPS is that the satellite navigational information is available to anyone, anywhere, free of charge. Of course, you need to buy a GPS receiver to use the data, but it's a pretty small investment compared to the amount of information you can get.

Through this project, we discovered the endless possibilities offered by the GPS. Its usability does exceed its indispensability for many travellers. GPS has indeed many functions that make it still very useful even in the research area.

Nowadays, around me, you and us, the scientific world is only talking about miniaturisation. The miniaturisation announces new fascinating world, unlimited research fields and promising perspective. Through it, manufacturers aim to reach better performance, more resistance, less pollution and less charge. But with miniaturisation other aspects are affected, other problems must be taken into consideration; magnetic and electric fields, heating, and noise...

Having the GPS miniaturised is a great achievement. Then they are easily embedded, applicable in the robotic field. They are light and very convenient for the localisation of our flying robots and for the recognition of their path.

Zusammenfassung:

In der letzten Dekade hat das Global Positioning System rasch zugenommen an Bedeutung. Es ist einer der präzisesten und genauesten Navigationssystemen verfügbar im Markt.

Das merkwürdige an GPS ist dass die Satelliten Information verfügbar für alle, überall und kostenlos ist. Natürlich muss man dafür ein GPS Empfänger kaufen, aber es ist richtig eine kleine Investition im Vergleich mit der Menge der Informationen erhältlich.

Durch dieses Projekts haben wir die Geheimnisse vom GPS entdeckt. Sein Verbrauch beschränkt sich nicht nur auf Reisenden, es hat zwar auch verschiedene Funktionen, die im Forschungsbereich sehr hilfreich sind.

GPSs sind in stendingen Entwicklung. Um uns herum, die Wissenschaftler reden nur über miniaturizierung. Miniaturisierung öffnet fasziniernden Horizont, erfolgversprechenden Perspectiven und unendliche Forchungsbereiche. Dadurch wollen die Hersteller bessere Leistung, bessere Robustheit, weniger Kosten und weniger Schadstoffemission erreichen.

Aber wenn wir die Bauteile verkleinern, treten dann andere unerwünschte Effekt, die auch berücksigtigt werden müssen.

Die miniaturieseirte GPS sind eine sehr große Herausforderung. Sie sind sehr einfach eingebettet, einsetzbar im Bereich der Robotik. Sie sind leicht und geeignet für die Lokalisierung von unseren fliegenden Robotern und für die Erkennung von deren Pfad.

TECHNISCHE UNIVERSITÄT MÜNCHEN
LEHRSTUHL FÜR STEUERUNGS- UND REGELUNGSTECHNIK
ORDINARIUS: UNIV.-PROF. DR.-ING./UNIV. TOKIO MARTIN BUSS
EXTRAORDINARIA: UNIV.-PROF. DR.-ING. SANDRA HIRCHE

29.3.2010

P R A C T I C A L   C O U R S E
for
Hanen Soufi, Mat.-Nr. 3123093
Akram Ben Halima, Mat.-Nr. 03281634

Design and Evaluation of various miniaturized GPS modules

Problem description:

Knowing where you are is essential for many mobile robotic scenarios. GPS (global positioning system) is a technology that allows outdoor localization around the world, currently with a precision in the range of 10s of centimeters. Various commercially available modules exist, such as navigation systems for cars or mobile telephones. The core of all such systems are tiny GPS-receiver modules, that exist in at least three different chip sets (SiRF star III, MTK, and u-blox) from different manufacturers. For flying robot experiments, such modules provide valuable location information; but most commercially available sensor solutions are by far too heavy for lightweight flight experiments. Here we are investigating in designing and building miniaturized versions of GPS modules.

Tasks:

In this practical, the students will evaluate existing GPS systems (such as USB-connected GPS-mice) and record performance and accuracy (as baseline for later comparisons). In the next step, the students shall design circuit boards with miniature antennae and several commercially available GPS chips. These size of such chips is about 10x10mm, so the final circuit board should not be much larger. Several different existing antennae and chip modules will be designed, produced (PCB-mill and miniature SMD soldering!) and compared in terms of performance against each other and against the commercial baseline.

- study performance of existing GPS solution
- evaluate modules and antennae / receiver circuits
- design and implement miniature versions of GPS receiver (includes miniature SMD soldering!)
- program basic software (or use existing tools) for performance evaluation

This practical requires some hands-on experience with electronic components, and soldering.

Supervisor: Jörg Conradt

Conradt
(J. Conradt)
Univ.-Professor

Contents

Word of Thanks:

We would like to express our sincere thank to Professor Conradt for his support and comprehension.
Over the hole project, you have been tremendously helpful with regards to finding mistakes and inconsistencies within our work.
We are very grateful for your advices, suggestions and recommendations.
Many thanks for your ongoing interest and support.

1 Introduction

1.1 History of GPS

Throughout the century, mankind has been trying to figure out a dependable way to know once location and to find the suitable path for once destination. People first navigated only by means of landmarks - mountains, trees, or leaving trails of stones. For travelling across the ocean a process called dead reckoning, which used a magnetic compass and required the calculation of how fast the ship was going, was applied. It was a very complicated process and inaccurate. Celestial navigation was our primary means of navigation for hundreds of years. But its degree of precision was limited. The sextant was developed during this time but since it only measured latitude, a timepiece was also invented so that the longitude could also be calculated.

The ground-based radio navigation systems were in the early 1940s developed. Some are still in use today. An example of such a system is LORAN. Long Range Navigation is using low frequency radiotransmitters that uses multiple transmitters (multilateration) to determine the location and speed of the receiver. It is based on the principle of the time difference between the receipt of signals from a pair of radio transmitters. LORAN could neither be used over the ocean nor applied to aviation. The accuracy of such systems could be affected by geography as well. In the 1970s, the ground-based Omega Navigation System, based on phase comparison of signal transmission from pairs of stations, became the first worldwide radio navigation system. However, limitations of these systems drove the need for a more universal navigation solution with greater accuracy.

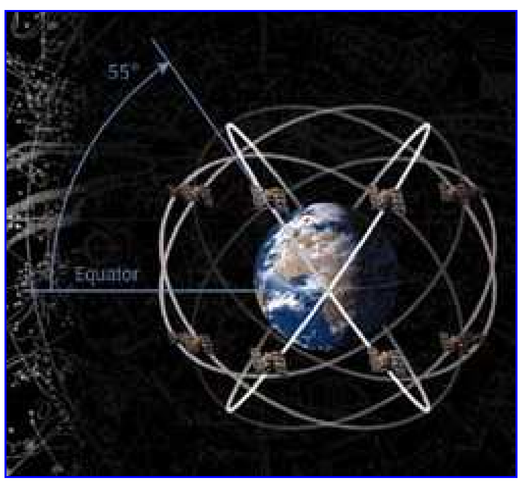

Figure 1.1: GPS constellation

The first satellite navigation system, Transit, used by the United States Navy, was first successfully tested in 1960. Transit's operation was based on the Doppler effect: the satellites travelled on well-known paths and broadcast their signals on a well known frequency. It used a constellation of five satellites and could provide a navigational fix approximately once per hour.

Recognizing the importance of GPS to civilian use as well as military uses, U.S. President Bill Clinton issued in 1996 a policy directive declaring GPS to be a dual-use system and establishing an Interagency GPS Executive Board to manage it as a national asset.

1.2 Why GPS?

In the 1860s, the first prototype of gyroscompasses was created. The first functional marine gyrocompass was patented in 1908 by German inventor Hermann Anscütz Kämpfe. It is a device for measuring or maintaining orientation, based on the principles of conservation of angular momentum. Due to higher precision, gyroscopes are also used to maintain direction in tunnel mining. However, for long duration accuracy, the gyroscopes are not stable.

Figure 1.2: Gyroscope

There is also the Inertial Navigation System (INS). It is a navigation aid that uses an accelerometer to continuously calculate via dead reckoning the position, orientation, and velocity (direction and speed of movement) of a moving object without the need for external references. Other terms used to refer to inertial navigation systems or closely related devices include inertial guidance system, inertial reference platform, and many other variations.

An accelerometer alone is unsuitable to determine changes in altitude over distances where the vertical decrease of gravity is significant, such as for aircraft and rockets. In the presence of a gravitational gradient, the calibration and data reduction process is numerically unstable.
Though, satellite navigation systems can provide high frequency signals allowing high accuracy, as well as a global access because the satellites are so high up that remaining within the line of sight of the satellites is easy. Its measurement are accurate global and stable for long periods of times.

1.3 Principles of GPS

GPS receivers passively receive satellite signals; they do not transmit.
There are at least 24 operational GPS satellites orbiting with a period of 12 hours around the earth. Their orbits are precisely tracked by ground stations to precisely localize them.

Each GPS satellite transmits data that indicates its location and the current time. They synchronize operations so that these repeating signals are transmitted at the same instant. The signals, moving at the speed of light, arrive at a GPS receiver at slightly different times because some satellites are farther away than others. The distance to the GPS satellites can be determined by estimating the amount of time it takes for their signals to reach the receiver.

By estimating how far away a satellite is, the receiver also estimates its location somewhere on the surface of an imaginary sphere centred at the satellite. It then determines the sizes of several spheres, one for each satellite. When the receiver estimates the distance to at least four GPS satellites, it can calculate its position in three dimensions. The receiver is located where these spheres intersect.

The GPS system provides very accurate timing and frequency signals. GPS operations depend on a very accurate time reference, which is provided by atomic clocks at the U.S. Naval Observatory. Each GPS satellite has three or four onboard caesium or rubidium atomic clocks. The four relative times are mathematically transformed into three absolute distance coordinates and one absolute time coordinate. The time is accurate to within about 50 nanoseconds. To ensure accurate measurements, GPS receivers require an unobstructed view of the sky. So they are used only outdoors and they often do not perform well within forested areas or near tall buildings.

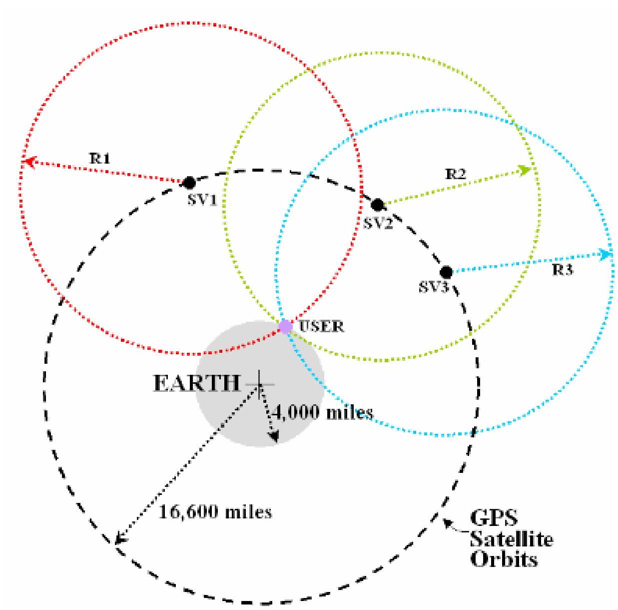

Figure 1.3: Functioning principle of GPS

1.4 Perspective

Performances of GPS are still assigned with error. The augmentation of a global navigation satellite system (GNSS) is a method of improving the navigation system's attributes, such as accuracy, reliability, and availability, through the integration of external information into the calculation process. There are many such systems in place and they are generally named or described based on how the GNSS sensor receives the external information. Some systems transmit additional information about sources of error (such as clock drift, ephemeris, or ionospheric delay), others provide direct measurements of how much the signal was off in the past, while a third group provide additional vehicle information to be integrated in the calculation process.

The Wide Area Augmentation System (WAAS) is an air navigation aid developed by the Federal Aviation Administration (FAA) and the Department of Transportation (DOT) for use in precision flight approaches eventually civilian aviation. Currently, GPS alone does not meet the FAA's navigation requirements for accuracy, integrity, and availability. WAAS corrects for GPS signal errors caused by ionospheric disturbances, timing, and satellite orbit errors, and it provides vital integrity information regarding the health of each GPS satellite. WAAS is essentially intended to enable aircraft to rely on GPS for all phases of flight, including precision approaches to any airport within its coverage area.

The European Union and European Space Agency introduced on March 2002 their own alternative to GPS, called the Galileo positioning system. At a cost of about GBP 2.4 billion, the system is scheduled to be working from 2012. The first experimental satellite was launched on 28 December 2005. Galileo is expected to be compatible with the modernized GPS system. The receivers will be able to combine the signals from both Galileo and GPS satellites to greatly increase the accuracy.

1.5 Flying Saucer

The Micro R/C 4-Channel Flying Saucer incorporates technology in a small package not much larger than an outstretched palm. With 4-propellers, the Micro Flying Saucer would be hard to keep balanced in the air, however this problem is solved by the built in solid-state accelerometer. Similar to the technology in the iPhone and Nintendo Wii controller, the accelerometer senses tilt on two axes and adjusts the power of each propeller to keep the Flying Saucer rock solid in the sky.

Figure 1.4: Flying Saucer

The small R/C flying vehicle is very sophisticated. It is neither built with trim dials nor buttons to adjust. The remote needs simply to be power-cycled before the flight and the Flying Saucer auto trims itself using the accelerometer as a reference. Four-channel control allows a forward, reverse, left or right movement, plus rotation in place, and up and down flight.

A network of Flying Saucer is preferably localisable; the problem is that its load of charge is very limited, so we need to find a very light navigation system for this intended purpose

2 Norms

2.1 Frames - NMEA 0183

2.1.1 NMEA

NMEA 0183 is a combined electrical and data specification for communication between marine electronic devices such as echo sounder, sonars, anemometer (wind speed and direction), gyrocompass, autopilot, GPS receivers and many other types of instruments. It has been defined by, and is controlled by, the U.S.-based National Marine Electronics Association.

The NMEA 0183 standard uses a simple ASCII, serial communications protocol that defines how data is transmitted in a "sentence" from one "talker" to multiple "listeners" at a time. Through the use of intermediate expanders, a talker can have a unidirectional conversation with a nearly unlimited number of listeners, and using multiplexers, multiple sensors can talk to a single computer port.

At the application layer, the standard also defines the contents of each sentence (message) type so that all listeners can parse messages accurately.

NMEA 2000 (IEC 61162-3) can be considered a successor to the NMEA 0183 (IEC 61162-1) serial data bus standard. It has a significantly higher data rate (250k bits/second vs. 4800 bits/second for NMEA 0183). It uses a compact binary message format as opposed to the ASCII serial communications protocol used by NMEA 0183. Another improvement is that NMEA 2000 supports a disciplined multiple-talker, multiple-listener data network whereas NMEA 0183 requires a single-talker, multiple-listener (simplex) serial communications protocol.

Most GPS manufacturers include special messages in addition to the standard NMEA set in their products for maintenance and diagnostics purposes. These extended messages are not standardized at all and are normally different from vendor to vendor.

However, there are some standard rules in frame analysis:

- Each message's starting character is a dollar sign.
- The next five characters identify the talker (two characters) and the type of message (three characters).
- All data fields that follow are comma-delimited.
- Where data is unavailable, the corresponding field contains *NUL* bytes (e.g., in "123,,456", the second field's data is unavailable).

- The first character that immediately follows the last data field character is an asterisk.
- The asterisk is immediately followed by a two-digit checksum representing a hex number. The checksum is the exclusive OR of all characters between the *$* and *. According to the official specification, the checksum is optional for most data sentences, but is compulsory for RMA, RMB, and RMC (among others).
- <CR><LF> ends the message.

Figure 2.1: NMEA logo

2.1.2 Input frames

Input frames are specific command data used to configure the GPS chip.

They are listed below:

Message	Name	Description
100	SetSerialPort	Set PORT A parameters and protocol
101	NavigationInitialization	Parameters required for start using X/Y/Z1. Input coordinates must be WGS84.1
102	SetDGPSPort	Set PORT B parameters for DGPS input
103	Query/Rate Control	Query standard NMEA message and/or set output rate
104	LLANavigationInitialization	Parameters required for start using Lat/Lon/Alt2
105	Development Data On/Off	Development Data messages On/Off
106	Select Datum	Selection of datum used for coordinate transformations
107	Proprietary	Extended Ephemeris Proprietary message
108	Proprietary	Extended Ephemeris Proprietary message
110	Extended Ephemeris Debug	Extended Ephemeris Debug
200	Marketing Software Configuration	Selection of Marketing Software Configurations
MSK	MSK Receiver Interface	Command message to a MSK radio-beacon receiver

The relevant frame for our tests is Query/Rate Control to be adaptable to every chip.

This command is used to control the output of standard NMEA messages GGA, GLL, GSA, GSV, RMC, and VTG. It also controls the ZDA message in software that supports it. Using this command message, standard NMEA messages may be polled once, or setup for periodic output. Checksums may also be enabled or disabled depending on the needs of the receiving program. NMEA message settings are saved in battery-backed memory for each entry when the message is accepted.

An example to query the GGA message with checksum enabled would be:

$PSRF103,00,01,00,01*25

The input values are:

Name	Example	Description
Message ID	$PSRF103	PSRF103 protocol header
Msg	00	Message to control. See Table 2-9
Mode	01	0 = Set Rate, 1 = Query one time
Rate	00	Output Rate, 0 = Off, 1 – 255 = seconds between messages
CksumEnable	01	0 = Disable Checksum, 1 = Enable checksum
Chechsum	*25	
<CR><LF>		End of message termination

2.1.3 Output frames

Output frames are the data lines that the GPS chip sends via the serial port.

They are listed below:

Message	Description
GGA	Time, position and fix type data
GLL	Latitude, longitude, UTC time of position fix and status
GSA	GPS receiver operating mode, satellites used in the position solution, and DOP values
GSV	Number of GPS satellites in view satellite ID numbers, elevation, azimuth, & SNR values
MSS	Signal-to-noise ratio, signal strength, frequency, and bit rate from a radio-beacon receiver
RMC	Time, date, position, course and speed data
VTG	Course and speed information relative to the ground
ZDA	PPS timing message (synchronized to PPS

Depending on the content, we made the choice of conserving 4 particular frames which are GGA, GSA, GSV and RMC in order to collect all relevant information.

- **GGA -** *Global Positioning System Fixed Data*
- **GSA -** *GNSS DOP and Active Satellites*
- **GSV -** *GNSS Satellites in View*
- **RMC -** *Recommended Minimum Specific GNSS Data*

2.2 Projection - WGS 84

2.2.1 Reference ellipsoid

In geodesy, a **reference ellipsoid** is a mathematically-defined surface that approximates the geoid, the truer figure of the Earth, or other planetary body. Because of their relative simplicity, reference ellipsoids are used as a preferred surface on which geodetic network computations are performed and point coordinates such as latitude, longitude, and elevation are defined.

A primary use of reference ellipsoids is to serve as a basis for a coordinate system of latitude (north/south), longitude (east/west), and elevation (height). For this purpose it is necessary to identify a *zero meridian*, which for Earth is usually the Prime Meridian. For other bodies a fixed surface feature is usually referenced, which for Mars is the meridian passing through the crater Airy-0. It is possible for many different coordinate systems to be defined upon the same reference ellipsoid.

The longitude measures the rotational angle between the zero meridian and the measured point. By convention for the Earth, Moon, and Sun it is expressed as degrees ranging from −180° to +180° For other bodies a range of 0° to 360° is used.

The latitude measures how close to the poles or equator a point is along a meridian, and is represented as angle from −90° to +90°, where 0° is the equator. The common or *geodetic latitude* is the angle between the equatorial plane and a line that is normal to the reference ellipsoid. Depending on the flattening, it may be slightly different from the *geocentric (geographic) latitude*, which is the angle between the equatorial plane and a line from the center of the ellipsoid. For non-Earth bodies the terms *planetographic* and *planetocentric* are used instead.

Currently the most common reference ellipsoid used, and that used in the context of the Global Positioning System, is WGS 84.
Traditional reference ellipsoids or *geodetic datums* are defined regionally and therefore non-geocentric, e.g., ED50. Modern geodetic datums are established with the aid of GPS and will therefore be geocentric, e.g., WGS 84.

The following table lists some of the most common ellipsoids:

Name	Equatorial axis (m)	Polar axis (m)	Inverse flattening, $1/f$
Clarke 1866	6 378 206.4	6 356 583.8	294.978 698 2
Bessel 1841	6 377 397.155	6 356 078.965	299.152 843 4
International 1924	6 378 388	6 356 911.9	296.999 362 1
Krasovsky 1940	6 378 245	6 356 863	298.299 738 1
GRS 1980	6 378 137	6 356 752.3141	298.257 222 101
WGS 1984	6 378 137	6 356 752.3142	298.257 223 563
Sphere (6371 km)	6 371 000	6 371 000	∞

2.2.2 Models

The World Geodetic System is a standard for use in cartography, geodesy, and navigation. It comprises a standard coordinate frame for the Earth, a standard spheroidal reference surface (the *datum* or *reference ellipsoid*) for raw altitude data, and a gravitational equipotential surface (the *geoid*) that defines the *nominal sea level*.

The latest revision is **WGS 84** (dating from 1984 and last revised in 2004), which will be valid up to about 2010. Earlier schemes included **WGS 72**, **WGS 66**, and **WGS 60**. WGS 84 is the reference coordinate system used by the Global Positioning System.

The coordinate origin of WGS 84 is meant to be located at the Earth's center of mass; the error is believed to be less than 2 cm.

In WGS 84, the meridian of zero longitude is the IERS Reference Meridian. It lies 5.31 arc seconds east of the Greenwich Prime Meridian, which corresponds to 102.5 metres (336.3 feet) at the latitude of the Royal Observatory.

As of the latest revision, the WGS 84 datum surface is a pole-flattened (*oblate*) spheroid, with major (*transverse*) radius a = 6,378,137 m at the equator, and minor (*conjugate*) radius b = 6,356,752.314 245 m at the poles (a flattening of 21.384 685 755 km, or 1/298.257 223 563 ≈ 0.335% in relative terms). The b parameter is often rounded to 6,356,752.3 m in practical applications.

Presently WGS 84 uses the 1996 Earth Gravitational Model (EGM96) geoid, revised in 2004. This geoid defines the nominal sea level surface by means of a spherical harmonics series of degree 360 (which provides about 100 km horizontal resolution). The deviations of the EGM96 geoid from the WGS 84 reference ellipsoid range from about -105 m to about +85 m. EGM96 differs from the original WGS 84 geoid, referred to as EGM84.

2.2.3 UTM - Mercator

The **Universal Transverse Mercator** (**UTM**) coordinate system is a grid-based method of specifying locations on the surface of the Earth that is a practical application of a 2-dimensional Cartesian coordinate system. It is a horizontal position representation, i.e. it is used to identify locations on the earth independently of vertical position, but differs from the traditional method of latitude and longitude in several respects.

The UTM system is not a single map projection. The system instead employs a series of sixty zones, each of which is based on a specifically defined secant transverse Mercator projection.

The **Mercator projection** is a cylindrical map projection presented by the Flemish geographer and cartographer Gerardus Mercator, in 1569. It became the standard map projection for nautical purposes because of its ability to represent lines of constant course, known as rhumb lines or loxodromes, as straight segments. While the linear scale is constant in all directions around any point, thus preserving the angles and the shapes of small objects (which makes the projection conformal), the Mercator projection distorts the size and shape of large objects, as the scale increases from the Equator to the poles, where it becomes infinite.

Mathematically, a reference ellipsoid is usually an oblate (flattened) spheroid with two different axes: An equatorial radius (the semi-major axis a), and a polar radius (the semi-minor axis b). More rarely, a scalene ellipsoid with three axes (triaxial—— a_x, a_y, b) is used, usually for modeling the smaller, irregularly shaped moons and asteroids. The polar axis here is the same as the axis with the great moment of inertia and is approximately aligned with the rotational axis (not the magnetic nor orbital pole). The geometric center of the ellipsoid is placed at the center of mass of the body being modeled, and not the barycenter in a multi-body system.

In working with elliptic geometry, several parameters are commonly utilized, all of which are trigonometric functions of an ellipse's **angular eccentricity**, $o\!\varepsilon$:

$$o\!\varepsilon = \arccos\left(\frac{b}{a}\right) = 2\arctan\left(\sqrt{\frac{a-b}{a+b}}\right);$$

Due to rotational forces, the equatorial radius is usually larger than the polar radius. This ellipticity or *flattening*, f, determines how close to a true sphere an oblate spheroid is, and is defined as

$$f = \mathrm{ver}(o\!\varepsilon) = 2\sin^2\left(\frac{o\!\varepsilon}{2}\right) = 1 - \cos(o\!\varepsilon) = \frac{a-b}{a}.$$

For Earth, f is around 1/300, and is very gradually decreasing over geologic time scales. For comparison, Earth's Moon is even less elliptical, with a flattening of less than 1/825, while Jupiter is visibly oblate at about 1/15 and one of Saturn's triaxial moons, Telesto, is nearly 1/3 to 1/2.

Such flattening is related to the eccentricity, e, of the cross-sectional ellipse by

$$e^2 = f(2-f) = \sin^2(o\!\varepsilon) = \frac{a^2-b^2}{a^2}.$$

It is traditional when defining a reference ellipsoid to specify the semi-major equatorial radius a(usually in meters) and the inverse of the flattening ratio $1/f$. The semi-minor polar radius is then easily derived.

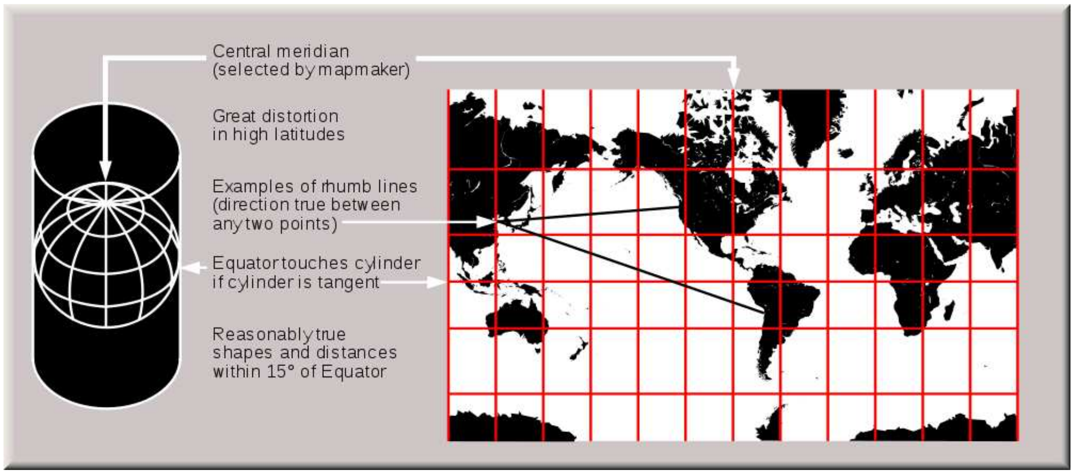

Figure 2.2: The Mercator projection

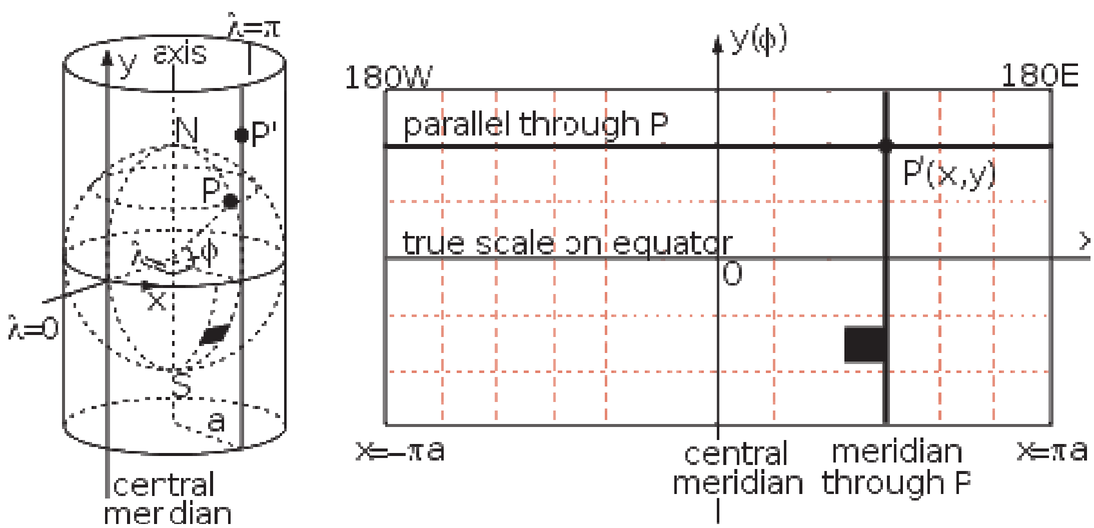

Figure 2.3: Cylindrical projection

3 Software

3.1 Choice steps

3.1.1 Serial communication

Serial communication is a popular means of transmitting data between a computer and a peripheral device such as a programmable instrument or even another computer. Serial communication uses a transmitter to send data, one bit at a time, over a single communication line to a receiver. You can use this method when data transfer rates are low or you must transfer data over long distances. Serial communication is popular because most computers have one or more serial ports, so no extra hardware is needed other than a cable to connect the instrument to the computer or two computers together.

Serial communication requires that you specify the following four parameters:

- The baud rate of the transmission
- The number of data bits encoding a character
- The sense of the optional parity bit
- The number of stop bits

Each transmitted character is packaged in a character frame that consists of a single start bit followed by the data bits, the optional parity bit, and the stop bit or bits.

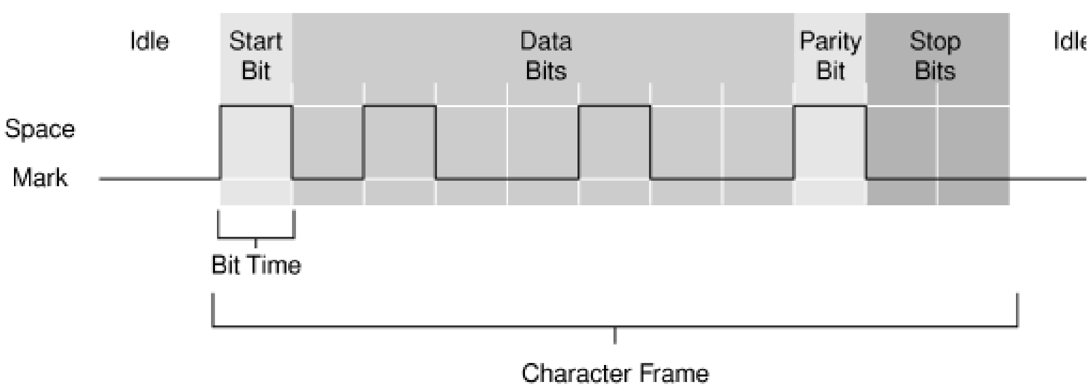

Figure 3.1: Typical character frame encoding the letter **m**

Baud rate is a measure of how fast data are moving between instruments that use serial communication. RS-232 uses only two voltage states, called MARK and SPACE. In such a two-state coding scheme, the baud rate is identical to the maximum number of bits of information, including control bits, that are transmitted per second.

The output signal level usually swings between +12 V and -12 V. The dead area between +3 V and -3 V is designed to absorb line noise.

A start bit signals the beginning of each character frame. It is a transition from negative (MARK) to positive (SPACE) voltage. Its duration in seconds is the reciprocal of the baud rate. If the instrument is transmitting at 9,600 baud, the duration of the start bit and each subsequent bit is about 0.104 ms. The entire character frame of eleven bits would be transmitted in about 1.146 ms.

Data bits are transmitted upside down and backwards. That is, inverted logic is used, and the order of transmission is from least significant bit (LSB) to most significant bit (MSB). To interpret the data bits in a character frame, you must read from right to left and read 1 for negative voltage and 0 for positive voltage. This yields 1101101 (binary) or 6D (hex). An ASCII conversion table shows that this is the letter m.

An optional parity bit follows the data bits in the character frame. The parity bit, if present, also follows inverted logic, 1 for negative voltage and 0 for positive voltage. This bit is included as a simple means of error handling. You specify ahead of time whether the parity of the transmission is to be even or odd. If the parity is chosen to be odd, the transmitter then sets the parity bit in such a way as to make an odd number of ones among the data bits and the parity bit. This transmission uses odd parity. There are five ones among the data bits, already an odd number, so the parity bit is set to 0.

The last part of a character frame consists of 1, 1.5, or 2 stop bits. These bits are always represented by a negative voltage. If no further characters are transmitted, the line stays in the negative (MARK) condition. The transmission of the next character frame, if any, is heralded by a start bit of positive (SPACE) voltage.

3.1.2 HyperTerminal

HyperACCESS was the first software product from Hilgraeve Comapny, and it was initially designed to let 8-bit Heath computers communicate over a modem. In 1985 this same product was ported to IBM PCs and compatible systems, as well as Heath/Zenith's Z-100 non PC-compatible MS-DOS computer. Over the years the same version of this technology would be ported to other operating systems including OS/2, Windows 95 and Windows NT. It has earned a total of five Editor's Choice awards from PC Magazine.

In 1995 Hilgraeve licensed a low-end version of HyperACCESS, known as **HyperTerminal** (essentially a "Lite" version) to Microsoft for use in their set of communications utilities. It was initially bundled with Windows 95, and subsequently all versions of Windows up to and including Windows XP. Windows Vista and Windows 7 do not include HyperTerminal, though the commercial product HyperTerminal Private Edition (latest version being 7.0) does support Vista. HyperACCESS will release version 9.0 in early 2010 - this will support Vista and 7 and have SSH functionality.

Hyperterminal is a simple way to establish a connection with a virtual serial port such as detected when the modules are connected with USB. It allows read and write, serial port parameters including mainly the baud rate configuration. The received data can simply scroll over the monitor if the port is well configured and all needed drivers are installed. So, it is a reliable method to test the readiness of our modules. We can also manage to save these results in real time in a text file in order to process it.

3.1.3 Excel macros

Microsoft Office Excel is a spreadsheet application written and distributed by Microsoft for Microsoft Windows and Mac OS X. It features calculation, graphing tools, pivot tables and a macro programming language called VBA (Visual Basic for Applications). It has been a very widely applied spreadsheet for these platforms, especially since version 5 in 1993. Excel forms part of Microsoft Office. The current versions are Microsoft Office Excel 2010 for Windows and 2008 for Mac.

The Windows version of Excel supports programming through Microsoft's Visual Basic for Applications (VBA), which is a subset of Visual Basic. Programming with VBA allows spreadsheet manipulation that is impossible with standard spreadsheet techniques. Programmers may write code directly using the Visual Basic Editor (VBE) accessed using the Visual Basic icon on the Developer Tab. Clicking that icon opens a window to the Visual Basic editing environment, which includes a window for writing code and a sophisticated debugging and code module organization environment. The user can implement virtually any numerical method in VBA and guide the calculation using any desired intermediate results reported back to the spreadsheet.

VBA code interacts with the spreadsheet through the *Object Model*, a vocabulary identifying spreadsheet objects, and a set of supplied functions or *methods* that enable reading and writing to the spreadsheet and interaction with its users (for example, through custom toolbars or *command bars* and *message boxes*). User-created VBA subroutines execute these actions and operate like macros generated using the macro recorder, but are more flexible and efficient.

In order to establish a serial connection with Excel, an additional tool must be installed in Windows latest than XP which is MSCOMM. This feature enables the use a new tool in the control panel to open a serial port, but only one at the time. Then, the received data are saved in a buffer and transmitted in order to the sheet columns depending on the tabulator.

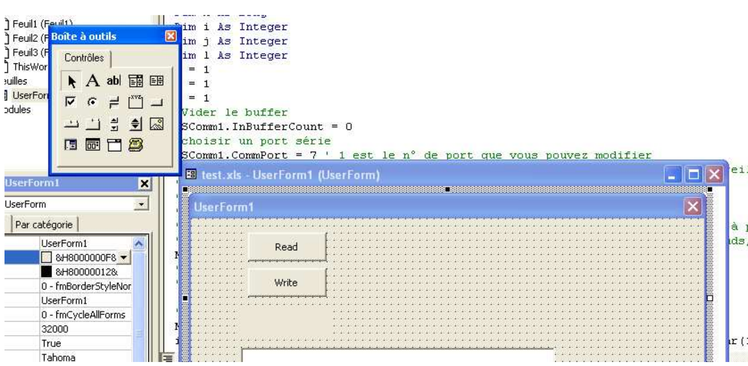

Figure 3.2: VBA interface

3.1.4 LabVIEW

LabVIEW (short for Laboratory Virtual Instrumentation Engineering Workbench) is a platform and development environment for a visual programming language from National Instruments. The graphical language is named "G". Originally released for theApple Macintosh in 1986, LabVIEW is commonly used for data acquisition, instrument control, and industrial automation on a variety of platforms including Microsoft Windows, various flavors of UNIX, Linux, and Mac OS X. The latest version of LabVIEW is version LabVIEW 2009, released in August 2009.

LabVIEW Tool for serial communication is VISA. The VISA Write and VISA Read functions work with any type of instrument communication and are the same whether doing GPIB or serial communication. However, because serial communication requires to configure extra parameters start the serial port communication with the VISA Configure Serial Port VI.
The VISA Configure Serial Port VI initializes the port identified by VISA resource name to the specified settings. timeout sets the timeout value for the serial communication. baud rate, data bits, parity, and flow control specify those specific serial port parameters. The error in and error out clusters maintain the error conditions for this VI.

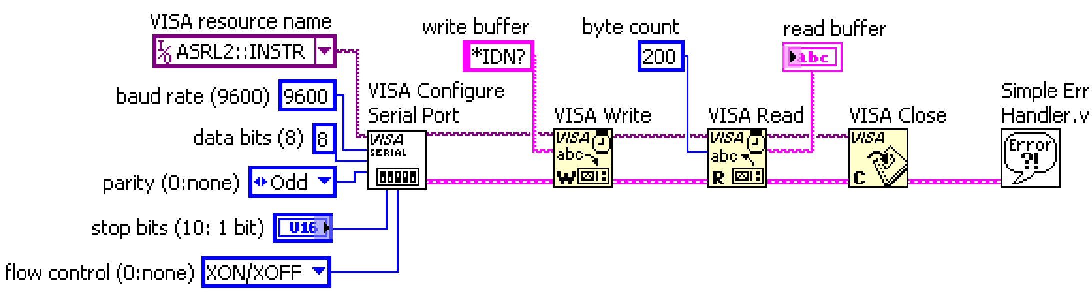

Figure 3.3: LabVIEW Graphic Programming

3.2 Final solution

3.2.1 VB.NET

Visual Basic .NET (**VB.NET**) is an object-oriented computer programming language that can be viewed as an evolution of Microsoft's Visual Basic (VB) which is generally implemented on the Microsoft .NET Framework. Microsoft currently supplies Visual Basic Express Edition free of charge.

In 2007, Microsoft planned to use the Dynamic Language Runtime (DLR) for the upcoming Visual Basic 10 (2010), formerly known as VBx. However, Microsoft shifted to a co-evolution strategy between Visual Basic and sister language C# to bring both languages into closer parity with one another. Visual Basic's innate ability to interact dynamically with CLR and COM objects has been enhanced to work with Dynamic languages built on the DLR such as IronPython and IronRuby. The Visual Basic compiler was improved to infer line continuation in a set of common contexts in this version, lifting in many cases the requirement for the " _" line continuation character. Also in version 10.0 existing support of inline Functions was complemented with support for inline Subs as well as multi-line versions of both Sub and Function lambdas.

VB.NET appears nowadays to be a universal programming language adaptable in compiling to any other structure and giving many possibilities of development. As far as serial communication is concerned, it supports thanks to multi object programming the use of simultaneous serial ports.

3.2.2 Functions & screenshots

The designed interface and functions are inspired from HyperTerminal working process. However, sending frames to the GPS chips in order to configure them is much more easier and reliable. Then, data export is also allowed in text file format. Choosing standard baud rates and selecting the communication port from a list of active ones is accordingly available as well as all required configuration. See Appendix for the source code.

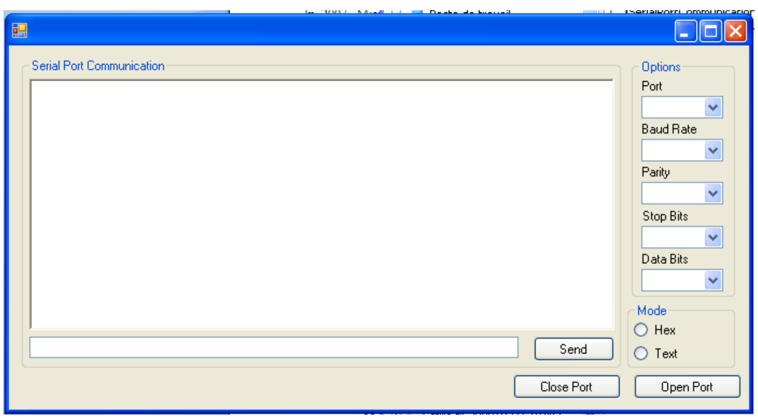

Figure 3.4: VB.NET Program Interface

3.3 Data processing

3.3.1 Document transformation and organization

The role of the software is limited to exporting a raw text file for every GPS chip containing different sorts of output frames (see 2.1.3).

It is important to notice that all modules were previously configured to only send the required frame types. So, we reduce the amount of useless data as the global quantity is quite big: 6 lines, of about 40 characters, each every 1 second.

The first step is to open the document with excel and choosing the comma “,” as tabulator (or separator) to have the frames organized in cells. It is necessary not to consider a sequence of commas as only one so that only one field is counted. In fact, some parts of a frame may be hidden as proof of error detection or signal problem or invalidity, these phenomena must be taken into consideration in our analysis.

When opened with excel, the text document is arranged in one sheet. We should first fix the start time (when all modules where connected and data sent being saved by the program). Then the effective departure time (when first valid is frame detected).

After that, we can apply a filter to sort the frames and place every single type in a sheet with naming the fields (columns) corresponding to the frame parts. Here are the partitions of the chosen frames explained with examples.

- ***GGA*** $GPGGA,002153.000,3342.6618,N,11751.3858,W,1,10,1.2,27.0,M,-34.2,M,,0000*5E

Name	Example	Unit	Description
Message ID	$GPGGA		GGA protocol header
UTC Time	002153.000		hhmmss.sss
Latitude	3342.6618		ddmm.mmmm
N/S Indicator	N		N=north or S=south
Longitude	11751.3858		dddmm.mmmm
E/W Indicator	W		E=east or W=west
Position Fix Indicator	1		
Satellites Used	10		Range 0 to 12
HDOP	1.2		Horizontal Dilution of Precision
MSL Altitude	27.0	Meters	
Units	M	Meters	
Geoid Separation	-34.2	meters	Geoid-to-ellipsoid separation. Ellipsoid altitude = MSL Altitude + Geoid Separation.
Units	M	Meters	
Age of Diff. Corr.		Sec	Null fields when DGPS is not used
Diff. Ref. Station ID	0000		
Checksum	*5E		
<CR> <LF>			End of message termination

- ***GSA*** $GPGSA,A,3,07,02,26,27,09,04,15, , , , , ,1.8,1.0,1.5*33

Name	Example	Unit	Description
Message ID	$GPGSA		GSA protocol header
Mode 1	A		
Mode 2	3		
Satellite Used1	07		SV on Channel 1
Satellite Used1	02		SV on Channel 2
…			…
Satellite Used 12			SV on Channel 12
PDOP	1.8		Position Dilution of Precision
HDOP	1.0		Horizontal Dilution of Precision
VDOP	1.5		Vertical Dilution of Precision

Checksum	*33		
<CR> <LF>			End of message termination

- ***GSV*** $GPGSV,2,2,07,09,23,313,42,04,19,159,41,15,12,041,42*41

Name	Example	Unit	Description
Message ID	$GPGSV		GSV protocol header
Number of Messages	2		Total number of GSV messages to be sent in this group
Message Number	1		Message number in this group of GSV messages
Satellites in View	07		
Satellite ID	07		Channel 1 (Range 1 to 32)
Elevation	79	degrees	Channel 1 (Maximum 90)
Azimuth	048	degrees	Channel 1 (True, Range 0 to 359)
SNR (C/N0)	42	dBHz	Range 0 to 99, null when not tracking
…			…
Satellite ID	27		Channel 4 (Range 1 to 32)
Elevation	27	degrees	Channel 4 (Maximum 90)
Azimuth	138	degrees	Channel 4 (True, Range 0 to 359)
SNR (C/N0)	42	dBHz	Range 0 to 99, null when not tracking
Checksum	*71		
<CR> <LF>			End of message termination

- ***RMC*** $GPRMC,161229.487,A,3723.2475,N,12158.3416,W,0.13,309.62,120598, ,*10

Name	Example	Unit	Description
Message ID	$GPRMC		RMC protocol header
UTC Time	161229.487		hhmmss.sss
Status1	A		A=data valid or V=data not valid
Latitude	3723.2475		ddmm.mmmm
N/S Indicator	N		N=north or S=south
Longitude	12158.3416		dddmm.mmmm
E/W Indicator	W		E=east or W=west
Speed Over Ground	0.13	knots	
Course Over Ground	309.62	degrees	True
Date	120598		Ddmmyy
Magnetic Variation	degrees		E=east or W=west

3.3.2 Coordinates change

The normal cylindrical projections are described in relation to a cylinder tangential at the equator with axis along the polar axis of the sphere. The cylindrical projections are constructed so that all points on a meridian are projected to points with $x = a\lambda$ and y a prescribed function of φ. For a tangent Normal Mercator projection the (unique) formulae which guarantee conformality are:

$$x = a\lambda\,, \qquad y = a\ln\left[\tan\left(\frac{\pi}{4}+\frac{\phi}{2}\right)\right] = \frac{a}{2}\ln\left[\frac{1+\sin\phi}{1-\sin\phi}\right].$$

Conformality implies that the point scale, *k*, is independent of direction: it is a function of latitude only:

$$k(\phi) = \sec\phi.$$

For the secant version of the projection there is a factor of k_0 on the right hand side of all these equations: this ensures that the scale is equal to k_0 on the equator.

The position of an arbitrary point (φ,λ) on the standard graticule can also be identified in terms of angles on the rotated graticule: φ' (angle M'CP) is an effective latitude and -λ' (angle M'CO) becomes an effective longitude. (The minus sign is necessary so that (φ',λ') are related to the rotated graticule in the same way that (φ,λ) are related to the standard graticule). The Cartesian (*x*',*y*') axes are related to the rotated graticule in the same way that the axes (*x*,*y*) axes are related to the standard graticule.

The tangent Transverse Mercator projection defines the coordinates (*x*',*y*') in terms of -λ' and φ' by the transformation formulae of the tangent Normal Mercator projection:

$$x' = -a\lambda' \qquad y' = \frac{a}{2}\ln\left[\frac{1+\sin\phi'}{1-\sin\phi'}\right].$$

This transformation projects the central meridian to a straight line of finite length and at the same time projects the great circles through E and W (which include the equator) to infinite straight lines perpendicular to the central meridian. The true parallels and meridians (other than equator and central meridian) have no simple relation to the rotated graticule and they project to complicated curves.

The direct formulae giving the Cartesian coordinates (*x*,*y*) follow immediately from the above. Setting $x = y'$ and $y = -x'$ (and restoring factors of k_0 to accommodate secant versions)

$$x(\lambda,\phi) = \frac{1}{2}k_0 a\ln\left[\frac{1+\sin\lambda\cos\phi}{1-\sin\lambda\cos\phi}\right],$$
$$y(\lambda,\phi) = k_0 a \arctan\left[\sec\lambda\tan\phi\right],$$

The above expressions are given in Lambert and also (without derivations).

After having this transformation, we should fix an origin to work properly in a Cartesian reference. The chosen one is the statute in front of the Audimax in TUM.

All these operations are made automatically thanks to an excel document (see screenshot in appendix) with the required formula and an ergonomic interface of capture. All steps are detailed there.

3.3.3 DGPS

Differential Global Positioning System (DGPS) is an enhancement to Global Positioning System that uses a network of fixed, ground-based reference stations to broadcast the difference between the positions indicated by the satellite systems and the known fixed positions. These stations broadcast the difference between the measured satellite pseudoranges and actual (internally computed) pseudoranges, and receiver stations may correct their pseudoranges by the same amount. The correction signal is typically broadcast over UHF radio modem.

Differential GPS is a method to enhance the results precision. In fact, the idea is to eliminate the static error resulting from signal variation. We can notice easily, that the data coming out a fixed GPS are not constant; however his position did not change.

So, when doing our tests, we should always keep our reference fixed and move with the other modules. Then, in the data processing we subtract the coordinates of the reference from the coordinates of the moving modules as if the synchronization was made in real time.

Thus, we obtain new data and a better analysis.

3.3.4 Satellite detection

It is important to know when the GPS chip changes from a satellite to another (when evaluating the signal coming from somewhere else is better).

NMEA protocol could not unfortunately force the GPS chip to connect with a given satellite. It can only suggest using a satellite rather than another one at the start of the chip.

3.3.5 Curves

Many curves are interesting for the analysis and relevant to plot.

First we can estimate the reference error by evaluating the variation of Xr(t) and Yr(t) while moving and in fix position.

The static error is the maximum scale of variation.

Then we will mainly focus on drawing the trajectories of the other tested modules by plotting Y=f(X) especially in the DGPS mode.

4 Hardware

4.1 Reference

Figure 4.1: BLUMAX 4013

Our aim is the comparison of different miniaturized GPS chips. Starting with a car GPS receiver is not really a good launch out for our project. So we had to find a relatively small GPS as reference for the next following steps.
Comparing the GPS mouses available on the market, we tried to find a compromise between price and performance. The range of choice was wide; there are many brands, different chips different prices... After comparing the most important of them, we chose the BLUMAX Bluetooth GPS-4013 receiver. Its Chipset is MTK MT3318 version v3.01 compatible with the NMEA command 0183 and able to give, as output; GGS, GSA, GSV and RMC messages. However for inputs, the chip is only compatible with the MTK NMEA packet format composed of:

*Preamble TalkerID PktType DataField * CHK1 CHK2 CR LF*
The high-performance 66 channel GPS is able to be used either as AGPS or as DGPS. It only needs 1s for the Hot Start, 33s for the Warm Start and 36s for the Cold Start.

You may probably ask, what do hot, warm and cold start mean?
During the *Hot Start* the GPS receiver remembers its last calculated position and which satellites were in view, the almanac used, and the UTC Time. It then performs a reset and attempts to acquire satellites and calculate a new position based upon the previous information. This is the quickest re-acquisition of a GPS lock. Though during the Warm Start, the GPS receiver remembers its last calculated position, almanac used, and knows the UTC Time, but not which satellites were in view. It then performs a reset and attempts to obtain the satellite signals and calculate a new position. The receiver has a general idea of which satellites to look for because it knows its last position and the almanac helps identify which

satellites are visible in the sky. That is why it takes longer. During the Cold Start, the GPS receiver dumps all information and resets. It then attempts to locate satellites and then calculate a GPS lock. This takes the longest because there is no known information. The GPS receiver has to attempt to lock a satellite signal from all of the satellites, basically like polling, which takes a lot longer than knowing which satellites to look for.

When a GPS receiver is put into operation for the first time, it takes too long time to detect a signal from the GPS satellite. This is not due to the bad performance of the GPS bought but the devise needs, during its first switch on, a lot of information in order to give back a correct calculated position. The GPS then saves this data and uses it for the next operation. The data stills available for a long period. By the next activation, the GPS needs only to update the data and save the new one.

One of the biggest problems we faced in our project is; how to configure the GPS mouse, how to set the appropriate frames, accordingly, the desired output messages.
Following the MTK NMEA protocol it only gives the command
$PMTK314,-,-,-,-,-,*2C<CR><LF>
Example: $PMTK314,1,1,1,1,1,5,1,1,1,1,1,1,0,1,1,1,1*2C<CR><LF>
This command set GLL output frequency to be outputting once every 1 position fix, and RMC to be outputting once every 1 position fix, and so on.
Applying this command on our GPS mouse was effectless, because it was not sufficient to configure the MTK chip.

However, when we tried to configure it using its official program ‘GPSView’, consisting in a GUI interface, it works, the configuration is possible!
So the question is: what command is compatible with Blumax 4013?
In order to figure this out, we built two USB ports, we connected two pins of each one to the other; the input pin is connected with the output pin and reciprocally. Then we wired them with the computer.

Using the program ‘GPSView’, we changed the configuration of the GPS; we set the desired output messages. Simultaneous we read the correspondent command in HyperTerminal.
The result is for example this command
$PMTK314,0,1,0,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0*28
$PMTK414*33
*1E
Now we could set up the GPS mouse and choose the appropriate output messages needed in the project.

4.2 D2523T

The technology D2523T is a complete GPS smart antenna engine board. It includes a built-in Sarantel’s GeoHelix high-gain (Omni-directional) low-noise amplifier, active antenna and GPS receiver circuits. The engine board is powered by the high performance 50-channel u-blox 5 technology. These modules provide good performance at an economical price. The receiver is optimized for applications requiring high performance, low power, and low cost, thus its position error is less than 2.0m and its time error is limited to 30ns. To receive a signal from the GPS satellites, the technology D2523T needs only 1s for Hot Start and 29s for Cold Start.

Figure 4.2: D2523T

A massive correlator (over 1 million) signal parameter search engine enables rapid search of all available satellites and acquisition of very weak signals. Once acquired, satellites are passed on to a power-optimized dedicated tracking engine. This arrangement allows the GPS and GALILEO engine to simultaneously track up to 16 satellites while searching for new one. The D2523T GPS receiver's -160dBm tracking sensitivity allows weak signal tracking and positioning in severe environments such as urban canyons and under deep foliage.

The D2523T is compatible with the NMEA protocol, UBX binary and gives with the baud rate 9600 bps the output messages GGA, GLL, GSA, GSV, RMC, VTG, TXT.
Its miniature size, makes it suitable to be integrated into portable devices, such as portable media players and portable navigation devices.

4.3 SUP500F

SUP500F Flash-based is a low-power high-performance 65 Channel GPS Smart Antenna Module. It is a compact solution intended for a broad range of Original Equipment Manufacturer (OEM) products, where fast and easy system integration and minimal development risk is required.

The SUP500F GPS receiver's -161dBm tracking sensitivity allows continuous position coverage in nearly all application environments. Its high performance search engine is capable of testing 8,000,000 time-frequency hypotheses per second. The receiver is optimized for applications requiring high performance, low power, and low cost, thus its position error is less than 2.5m, velocity accuracy about 0.1m/s and time error is limited to 300ns. To receive a signal from the GPS satellites, it needs only 1s for Hot Start and 29s for Cold Start, both under open sky condition. SUP500F is compatible with the NMEA-0183 protocol. Having the baud rate 9600, it gives the output messages GGA, GLL, GSA, GSV, RMC, VTG. Choosing the correct baud rate is very important to have meaningful frames otherwise it is unreadable.

Figure 4.3: SUP500F

The receiver, version v3.01 is suitable for a wide range of OEM configurations including mobile phone, PND, asset tracking, and vehicle navigation products.

4.4 Self-built GPS receivers

4.4.1 EB-230

EB-230 is an ultra miniature 12 x 12 mm² GPS engine board. It provides high navigation performance under dynamic conditions in areas with limited sky view like urban canyons. The EB-230 GPS receiver's -158dBm tracking sensitivity is up to detect weak signal operation without compromising accuracy.

EB-230 is a good choice for embedded applications. It is a low-power-consumption module, able to track up to 51-Channel of satellites. But this chip still requires an antenna and an amplifier, to work properly.

3V power supply, 50 Ohm output impedance and a frequency of 1575.42 ± 1.032MHz must be the characteristics of the antenna. To ensure an optimal system performance, the antenna must preferably have less or equal to 2.0dB low noise figure and LNA gain between 15dB and 20dB.

Figure 4.4: EB230

4.4.2 ET-318

ET-318 is a compact, high performance, and low power consumption GPS engine board. This GPS receiver' -159dBm tracking sensitivity allows continuous position coverage in nearly all application environments. Its version v2.2 is supported by the NMEA protocol. The module uses SiRF Star III chipset which can track up to 20 satellites at a time and perform fast TTFF* in weak signal environments.

For a good reception of the GPS satellite signal, it is recommended to use an antenna with 50 Ohm output impedance, a frequency of 1575.42 + 2 MHz, an amplifier gain between 18 and 22 dB and a noise figure less than 2.0dB.

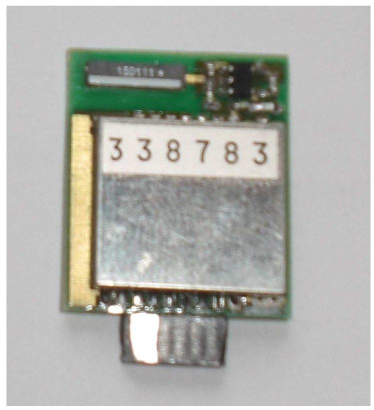

Figure 4.5: ET318

4.4.3 MT-662

MT-662 is a compact high performance GPS engine board. It is a low power consumption GPS, able to operate between the range of -30°C and 85°C. It uses MTK MT3318 chipset which can track up to 32 satellites at the same time and perform fast TTFF* in low signal environments. The chip version v3.01 is supported by the NMEA 0183 protocol and gives as default the output GGA, GSA, GSV, RMC, VTG, GLL, ZDA. However, only the MTK NMEA command is able to configure it.

MT661 is suitable for portable electronic devices such as automotive navigation devices, handheld navigation devices mobile phones and other GPS applications. The GPS receiver's -158dBm tracking sensitivity is up to detect weak signal operation without compromising accuracy. Its position accuracy is about 3m and 2.4m with DGPS.

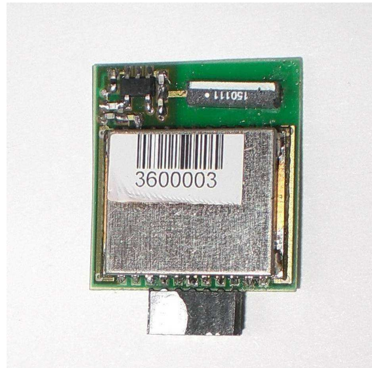

Figure 4.6: MT662

4.4.4 LEA 5

The LEA-5 module series is a family of self-contained GPS and GALILEO receivers featuring the powerful 50-channel u-blox 5 positioning engine. These modules provide exceptional GPS performance in a compact form factor and at a reasonable price. U-blox 5 sets a new standard in GPS receiver technology. A 32-channel acquisition engine with over 1 million effective correlators is capable of massive parallel searches across the time/frequency space. This enables a TTFF* of less than 1 second, while long correlation, once acquired, satellites are passed on to a dedicated tracking engine. This arrangement allows the GPS engine to simultaneously track up to 16 satellites while searching for new ones.

U-blox 5's advanced jamming suppression mechanism and innovative RF architecture provides a high level of immunity to jamming, ensuring maximum GPS performance. U-blox 5 has been designed to be able to support the GALILEO system currently being developed by European authorities. The capability of receiving GALILEO L1 signals will provide increased coverage and even better positioning accuracy when this system comes into operation. Their small size makes LEA-5 modules one of the ideal GPS solutions for applications with stringent space requirements. The packaging makes expensive RF cabling obsolete, with the RF input being available directly on a pin. The LEA-5 series are SMT solderable and can be handled by standard pick and place equipment.

LEA-5 modules come equipped with a serial port, which can handle NMEA and UBX proprietary data formats, as well as a high speed USB port. The optional FLASH EPROM provides the capacity to store user-specific configuration settings as well as future software updates.

Figure 4.7: LEA 5

*Time to First Fix (**TTFF**) is the time, a GPS receiver needs, to detect his position for the first time after 1its activation.

4.5 Amplifiers

4.5.1 RF2373

The RF2373 is a low noise amplifier with a very high dynamic range designed for WLAN WiMAX, and digital cellular applications. The device operates as an outstanding low noise amplifier or driver amplifier in the transmit chain of digital subscriber units where low transmit noise power is a concern.
When used as an LNA, the bias current can be set externally. When used as a PA driver, the IC can operate directly from a single cell Li-ion battery and includes a power down feature that can be used to completely turn off the device.

Figure 4.8: LEA 5

At a frequency of 1575 MHz, the RF2373, operating as a low noise GPS amplifier, has a maximal noise figure of 1.3dB and a gain of 17dB up to 21dB. The IC is featured in a standard SOT 5-lead plastic package and has a low power supply of 1.8V to 6.0V.

4.5.2 SMA-661

The **SMA661** is a low-noise amplifier with integrated matching networks and embedded power-down function. The chip, which requires only one external input capacitor, drastically reduces the application bill of materials and the PCB area, resulting in an ideal solution for compact and cost-effective GPS LNA. The SMA661AS, using the ST's leading-edge 70GHz SiGe BiCMOS technology, achieves excellent RF performance at the GPS frequency of 1.575GHz, in terms of power gain(18 dB), noise figure(1.15 dB) and linearity with a current consumption of 8.5mA. The device is unconditionally stable and ESD protected. All these features are steady over the operating temperature range of -40°C to +85°C.
It's housed in ultra-miniature SOT666 plastic package and operates with a power supply of 3.3V.

4.5.3 BGA-428

BGA428 is a low noise amplifier ideal for GSM, DCS1800, PCS1900 but not really specified for GPS. The BGA428, using the SIEGET®-45 technology, achieves excellent RF performance at low power supply.
It has in term of noise figure 1.4dB and a gain of 20dB at a frequency of 1.8GHz. It's housed in an ultra-miniature SOT363 plastic package and operates in a temperature range of -40°C to +85°C.

→ In order to have an accurate comparison between the three GPS chips, then our purpose is to choose the best GPS for the flying robots, we decided to work with only one amplifier RF2373.
And though all GPS were compatible with this amplifier we chose it.

4.6 Antenna

To work properly these chips must be associated to an efficient antenna. Then the antenna is the interface between the signal of the GPS satellite and the GPS. Without good antenna the performance of the GPS chip would be corrupted.

And though, we had to choose between two antennas:
The 922D03E15X11113 is a high performance 1.575GHz band small antenna. It is implementable in navigation system, aero plane servicing automotive application, and other portable communication devices …

Figure 4.9: 922D03E15X11113

The GPS antenna comprises a radiating structure of multiple meandered conduction strips, which are developed on tiny pieces of printed circuit board (PCB) and packed with Liquid Crystal Polymer (LCP) dielectric composite material to achieve size performance characteristics and cost effectiveness superior to other designs. The compact surface mountable package measures 8.0mm (L) * 2.0mm (W) * 1.5mm (H) in dimensions and is fully compatible with handmade and reflow attachment processes. It is Ultra thin and weights only 0.06g.

The antenna's favourable electrical specifications, stability and cost effectiveness make it the first logical choice for a wide variety of applications in the 1.575GHz GPS and accordingly for our project.

4.7 Soldering

While soldering, any person handling the ICs should be grounded either with a wrist strap or ESD-protective footwear used in conjunction with a conductive or static-dissipative floor or floor mat. The work surface where devices are placed for handling, processing, testing, etc must be made of static-dissipative material and be grounded to ESD ground. All insulator materials must either be removed from the work area or must be neutralized with an ionizer. Static-generating clothing must be covered with an ESD-protective smock. When ICs are being stored, transferred between operations or workstations, or shipped, they must be kept in a Faraday shield container with inside surfaces (surfaces touching the ICs) that are static-dissipative.

When positioning the antenna on the PCB board, we must pay attention to its polarisation and the place to mount it in. When a satellite signal reflects off building and other objects creating multiple paths to the receiver, results its polarisation inverted from right hand circular to left hand circular. Our GPS antenna has *Elliptical Polarisation* and the *Axis Ratio* is about 1:3, therefore, when placing our GPS antenna on the top of the PCB we should choose the right hand circular polarisation (RHCP) type, on the other hand, when we place it on the bottom, then we should choose the left hand circular polarisation (LHCP) type. For an optimal result the chip antenna 922D03E15X11113 should be mounted on one corner of 0.8mm thick PCB with 10*17 mm² empty area and 50 Ohm micro strip line input.

Having the modules soldered, most of them didn't work from the first switch on, we had to verify the proper flux of current, check the soldering point wether they are conductive or not and control the power supply.
To read the data captured by the GPS modules, we had to connect them to a computer.
To establish this communication, we built USB modules.
Universal Serial Bus (USB) is a set of connectivity specifications developed by Intel in collaboration with industry leaders. USB allows high-speed, easy connection of peripherals to a PC. When plugged in, everything configures automatically. USB is the most successful interconnect in the history of personal computing and has migrated into consumer electronics (CE) and mobile products. The Universal Serial Bus system has an asymmetric design, consisting of a host, a multitude of downstream USB ports, and multiple peripheral devices connected in a tiered-star topology.
The FT232R is the latest device added to FTDI's range of USB UART interface Integrated Circuit Devices. The FT232R is a USB to serial UART interface with optional clock generator output. It is available in Pb-free (RoHS compliant) compact 28-Lead SSOP and QFN-32 packages.
In a PCB, we soldered the FT232R device, we add some conductors, resistors, and two input output connectors.

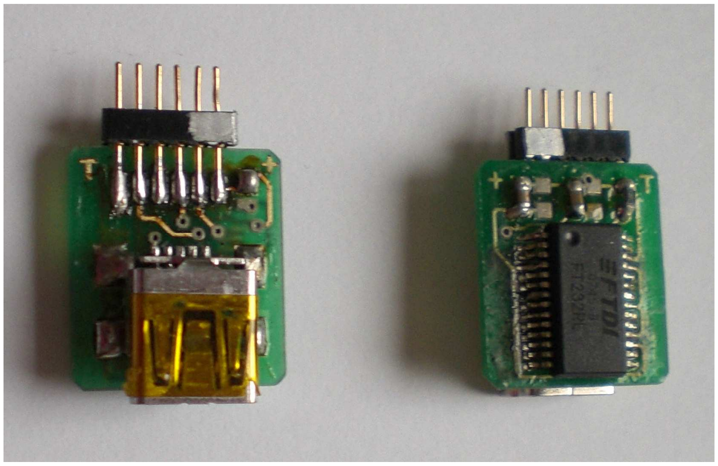

Figure 4.10: USB-connector

4.8 Table

	Blumax 4013	**D2523T**	**SUP500F**	**MT-662**	**ET-318**	**EB-230**
Chiset	MT3329	UBX-G5010	-	MT3318	SiRF star III	MTK
Frequency	L1, 1575.42 MHz C/A code	GPS L1, C/A code GALILEO L1	L1 C/A code	L1, 1575.42 MHz C/A code	L1, 1575.42 MHz C/A code	L1 frequcney, C/A code (SPS)
Baud rate (bps)	115200	9600	9600	4800	4800	4800
Protocol	NMEA0183 v3.01	NMEA, UBX binary	NMEA-0183 V3.01	NMEA 0183 v3.1 MTK NNEA command	NMEA 0183 v2.2 SiRF binary and NMEA Command	NMEA 0183 v3.1
Output messages (Default)	GGA, GSA, GSV, RMC	GGA, GLL, GSA, GSV, RMC, VTG, TXT	GGA, GLL, GSA, GSV, RMC, VTG	GGA, GLL, GSA, GSV, RMC, VTG, ZDA	GGA,GSA,GSV,RMC Support:VTG, GLL,ZDA)	GGA, GLL, GSA,GSV, RMC and VTG
Channels	66	50	65	32	51	20
Sensitivity: *Acquisition *Tracking	-165 dBm	-160dBm -160dBm	-161dBm	-146 dBm -158 Bm	-159dBm	-146dBm -158dBm
Update	1 Hz ~ 5 Hz	4 Hz (Max.)	10Hz	5 Hz	1 Hz	Up to 5Hz
Hot Start	1.5s	<1s	1s	1s	1s	1s
Warm Start	34s	29s		33s	32s	33s
Cold Start	35s	29s	29s	36s	42s	36s
Position accuracy	< 3 m CEP	<2.5m CEP	2.5m CEP	3m	10 meters, 2D RMS	<3m CEP
Velocity accuracy	0.1 m/s	-	0.1m/s	0.1m/s	0.1m/s	-
Time accuracy	50ns	30ns	300ns	1 ms RMC	1000ns	-
Input Voltage	DC 5.0 V±10 %	DC 3.3V ±10%	DC 3.0V ~ 5.5V	DC 3.3V ±5%	DC 3.3V ±5%	DC 2.8~3.3V

5 Tests

5.1 Principle

5.1.1 Normal mode

- Plugging all modules on the same PC
- Waiting until they receive satellite signal
- Saving received data continuously
- Moving in a known trajectory and timing

5.1.2 DGPS mode

The difference is on keeping the „Reference“ fixed to the ground and connected to the PC with the 10-meter cable while moving with the others modules and taking measures.
The trajectory is also supposed to be more complicated to highly evaluate the accuracy.

5.2 Results

5.2.1 Electrical analysis

The GPS modules take too long to catch signal from the satellites. In order to check whether the antenna is working properly or not, we used the oscilloscope. We tried to analyze which signal is coming out of the antenna.
This test we was done with open sky conditions, then in the room the antenna can’t detect any signal!

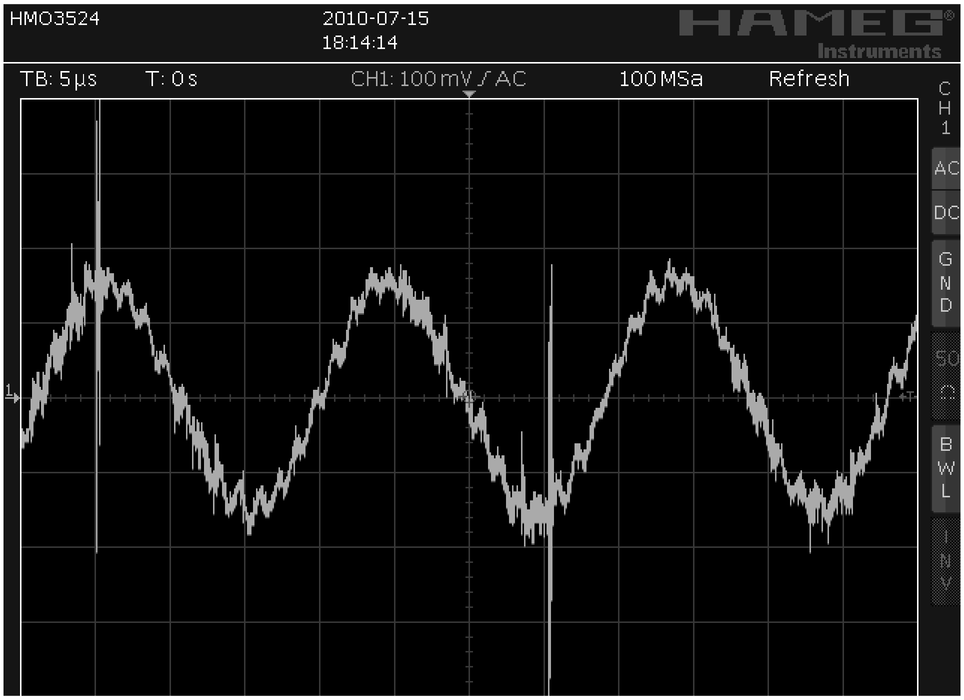

Figure 5.1: Oscilloscope measurement using a probe head with ten times amplification

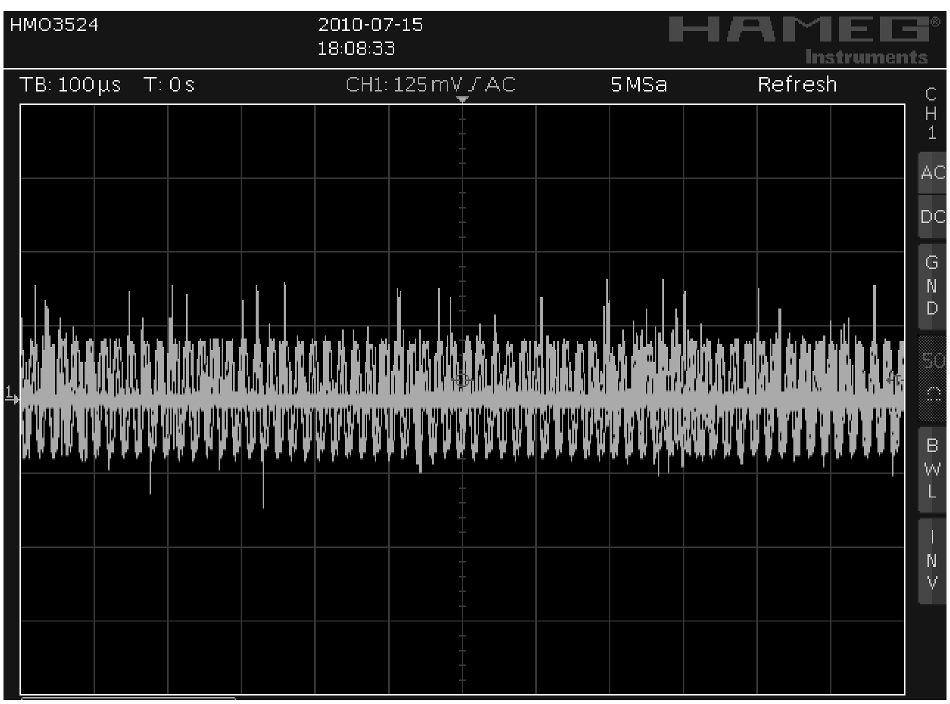

Figure 5.2: Oscilloscope measurement using a probe head with one time amplification

5.2.2 Data analysis

After resolving the start time problem, we have made 3 experiences with the GPS modules.

- ***Normal mode***

  It is about evaluating the GPS precision while moving.

  Example: Reference (Xm, Ym)

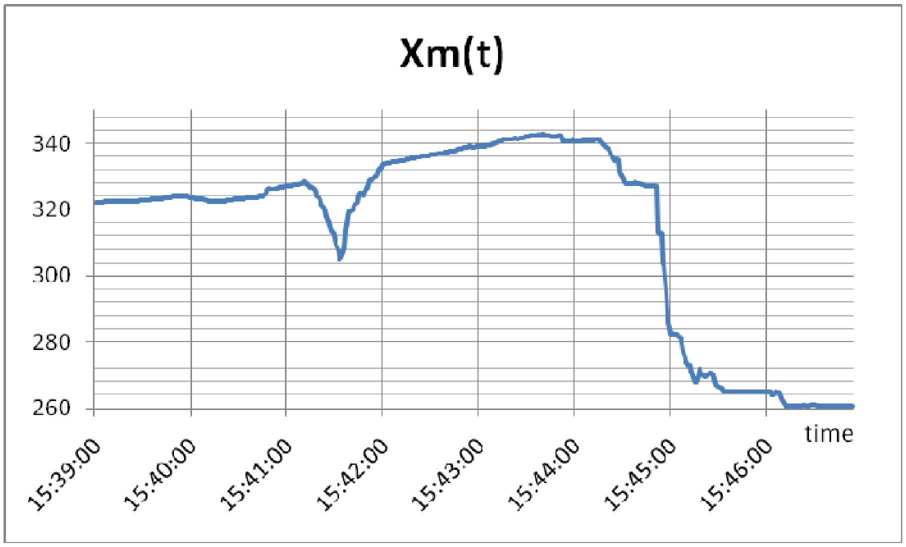

Figure 5.3: Variation in time for X coordinate of reference in normal mode

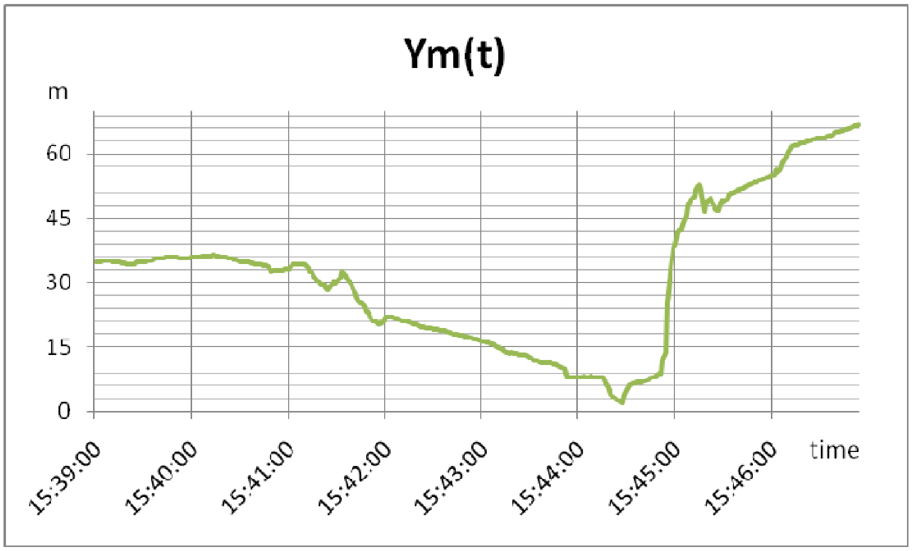

Figure 5.4: Variation in time for Y coordinate of reference in normal mode

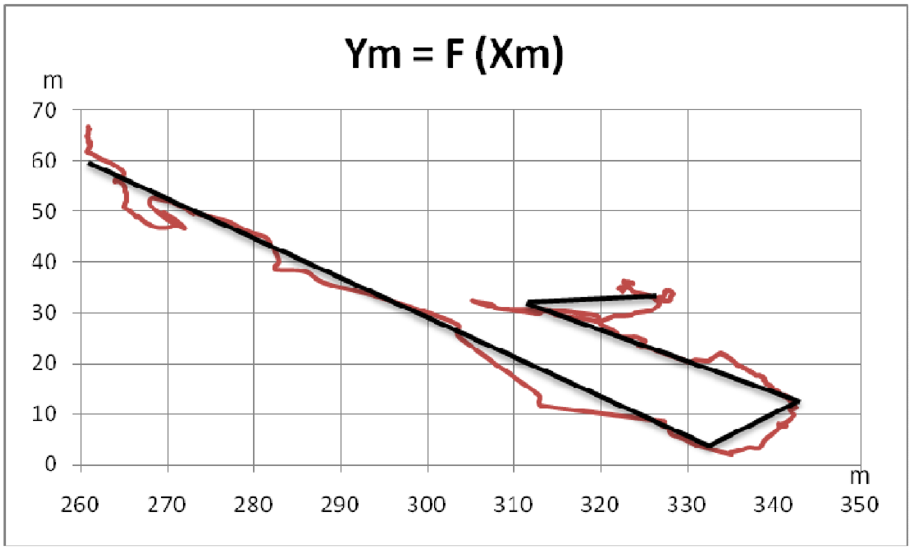

Figure 5.5: Trajectory of the reference in normal mode (real in black)

➔ The approximation is quite good (2m), the reference is working properly.

- ***Static error***

When the reference is fixed to the ground there is a little variation inducing an error in the received data when moving.

To evaluate it, we drew Ye = F(Xe) while executing a first DGPS experiment

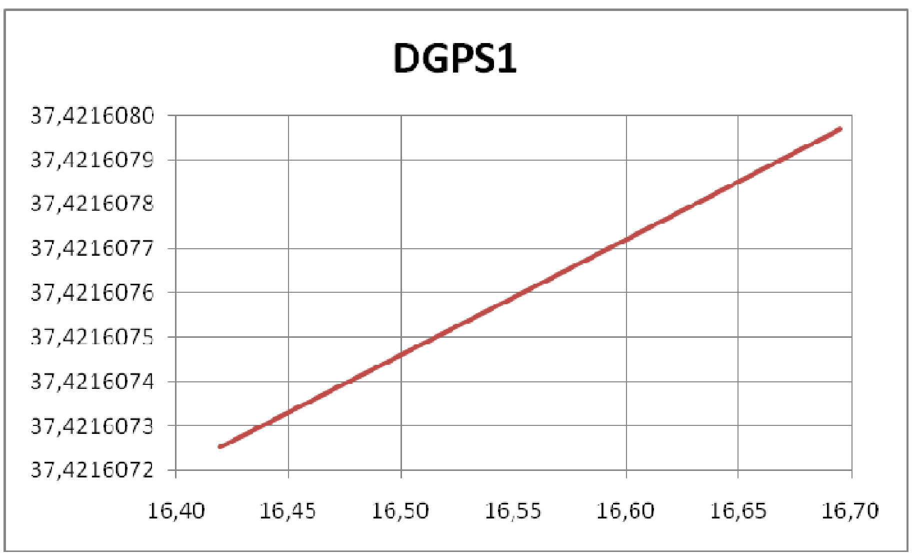

Figure 5.6: Coordinates variation for reference while fix

➔ The reference is very accurate, especially in Y!

- ***DGPS mode***

Here is the main part of the test that will really compare the acquired modules. We always plot trajectories and compare them to the real one.

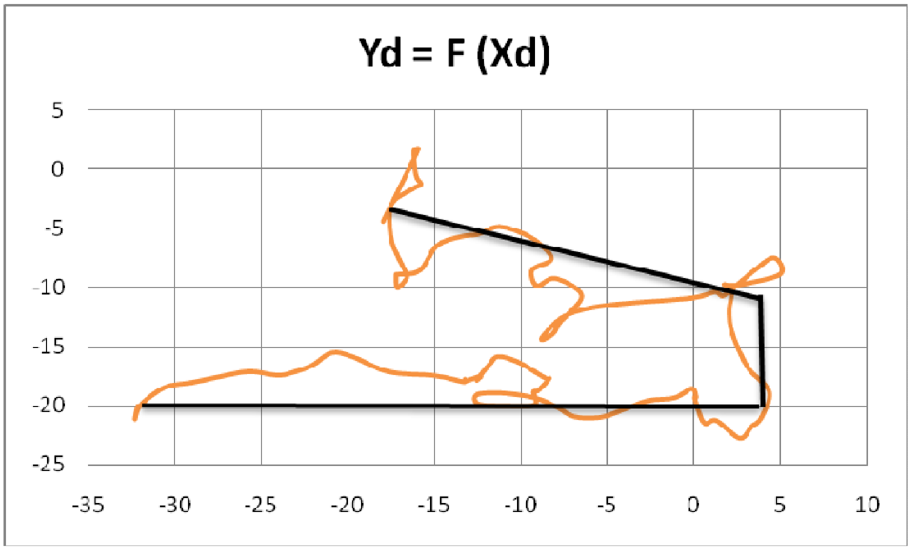

Figure 5.7: Trajectory of D2523T in DGPS mode (real in black)

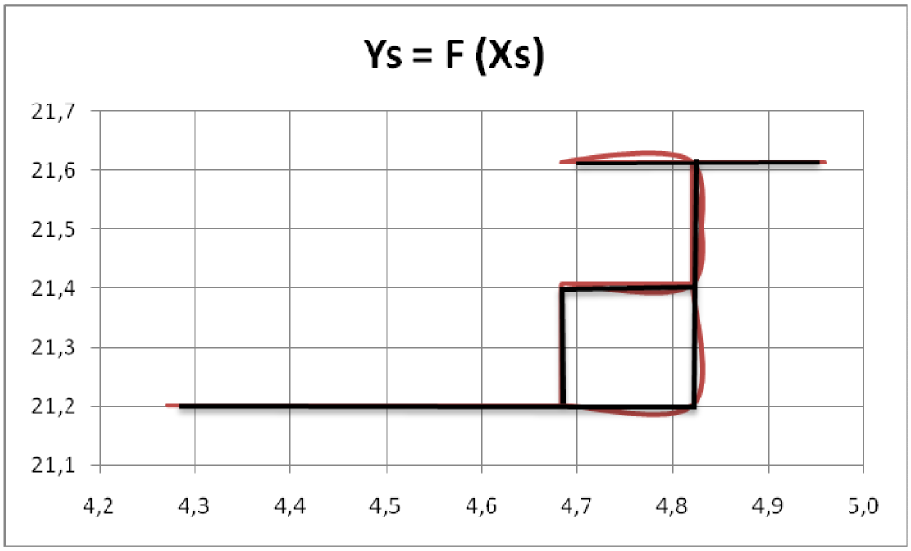

Figure 5.8: Trajectory of SUP500F in DGPS mode (real in black)

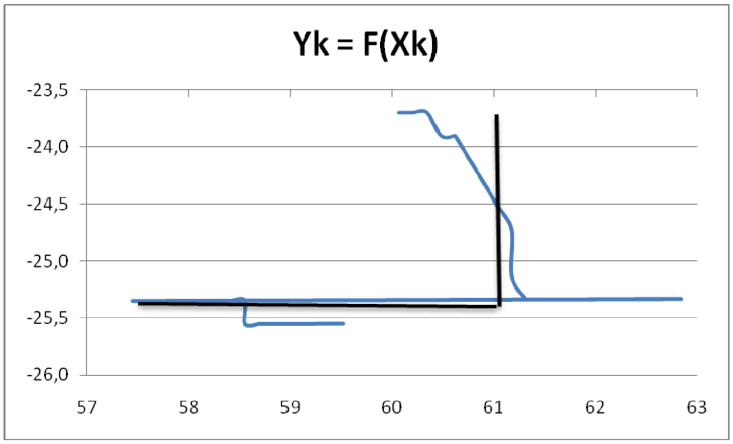

Figure 5.9: Trajectory of MT662 in DGPS mode (real in black)

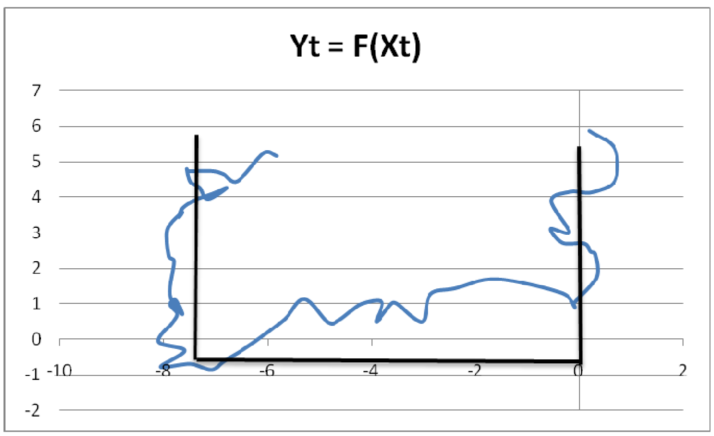

Figure 5.10: Trajectory of ET318 in DGPS mode (real in black)

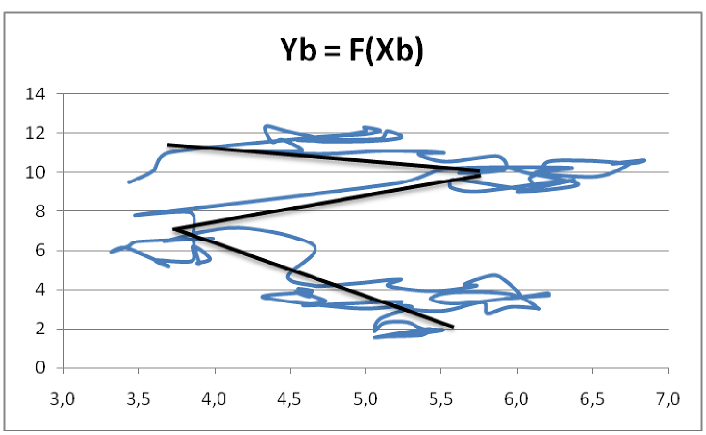

Figure 5.11: Trajectory of EB230 in DGPS mode (real in black)

5.3 Conclusions

* The Reference module is very accurate. The static error is quite tiny.

* There are some gaps in data capture for the tiny others modules: some frames are missing or bad recognized. That is mainly due to signal reception problems.

* The signal quality remains constrained by weather, buildings and foliage.

* The wait time for the tiny modules until they receive data is ~15 minutes. Saving the initial parameters of starting (time, location, satellites…) is crucial.

* When we disconnect and then reconnect the modules, the recognized position strangely changes. The initialization problem again!

* The functioning of the tiny modules is really random and the test is not 100% reproducible. That is why it was not possible to ensure the same DGPS test for all the modules; we cannot manage to make them all work at the same time.

* Data processing cannot be done automatically due to multiple errors of reception.

* DGPS enhances the results and gives a relatively similar trajectory to the real one. But, the approximation is still not enough: 1m for X and Y, 20cm for Z.

* Changing satellites connection occurs rarely in a small scale (10m radius for our tests).

* MTK chips seem to be the best (at least from the sample we had)

* In the tiny modules, the reliability of MT662 makes it the better (less than 10 minute wait time and works 50% of the time well). However, when it is plugged with a Windows PC, there is a risk to get the mouse moving aimlessly on the screen: Windows recognize this GPS chip as a ball point and tries every time to install the required driver for it, so the virtual serial port is blocked and the mouse is not controllable anymore. This driver should be always uninstalled to get a coherent functioning.

* The uBlox module Lea 5 is technically very good. And it has an interesting value for money. Unfortunately, its conditioning with the theoretically required components does not work.

* In the middle size chips, SUP500F is better than D2523T (his trajectory is more straight, few oscillations). It has a little more errors though. Furthermore, its antenna size is very reasonable and it could be compared with the tiny modules (except maybe for the weight).

6 Summary

Through this project, we got in depth in the world of navigation systems. During three months, we discovered the functioning way of GPS. We accordingly enriched our scope of knowledge either in the hardware field or in software programming, thanks to the support of our supervisor.

The GPSs are nowadays an indispensable tool of navigation; they are accurate global and precise, practical in many fields.
Through miniaturization, GPS acquired more importance in research field. They have a better size ratio, they need less space requirement and they are more compact. The GPS are easily embedded, in any electronic device, which is for the robotic field very promising.

But the size reduction has influenced other aspect in the functioning way of GPS.
The characteristics that made the GPS so popular have scaled down, such as precision, accuracy and robustness. The small modules require optimal condition to work properly, in fact they do capture valid satellite signal only on open sky conditions, good weather, and they can't work in urban canyons or under deep foliage.
We made several tests, we took various measurements in different weather conditions, we noticed that the small modules work mostly randomly.
Therefore, we concluded to the fact that the modules' unexpected behaviour is mainly due to the improper work of the antenna, on which the detection of satellite signals is depending. The miniaturisation of the antenna has corruptly influenced its working behaviour and its performance.

The measurements, we achieved with the DGPS mode, were really more accurate and more precise than the normal ones. But to reach a static error in the interval of some centimetres is not really feasible and to guarantee their good working in hostile environment is not really possible. For these reasons, we ended up to the fact that the miniaturised GPS are not really an optimal solution for our flying robots, as this operation requires high accuracy and precision.

The miniaturization of GPS has promising perspective but till today their performance are still under the standard requirement of the market. The developers still need to enhance robustness and precision.

Appendix

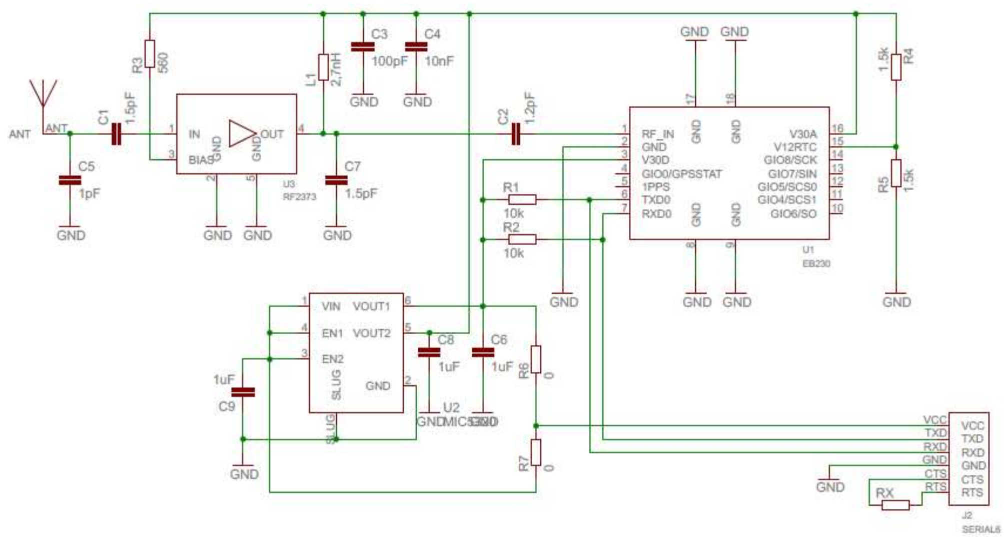

Figure 6.1: Layout EB230

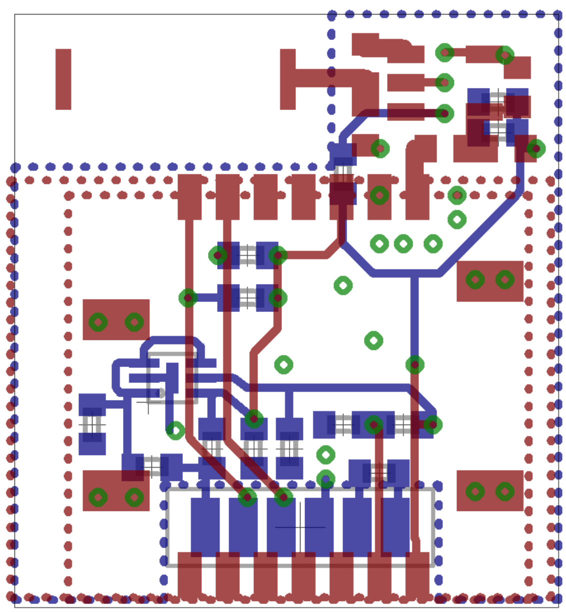

Figure 6.2: PCB EB230

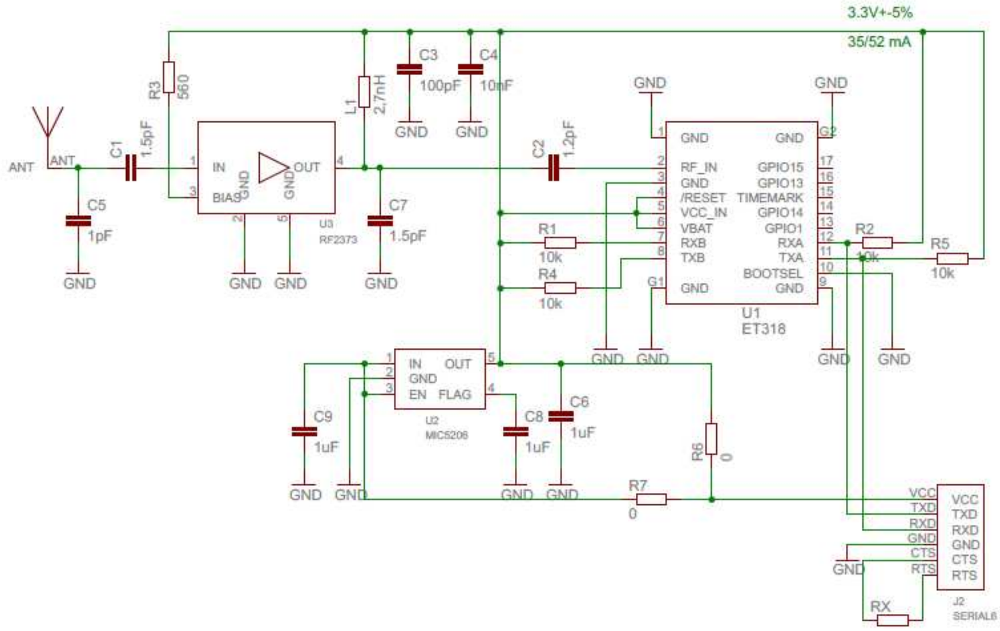

Figure 6.3: Layout ET318

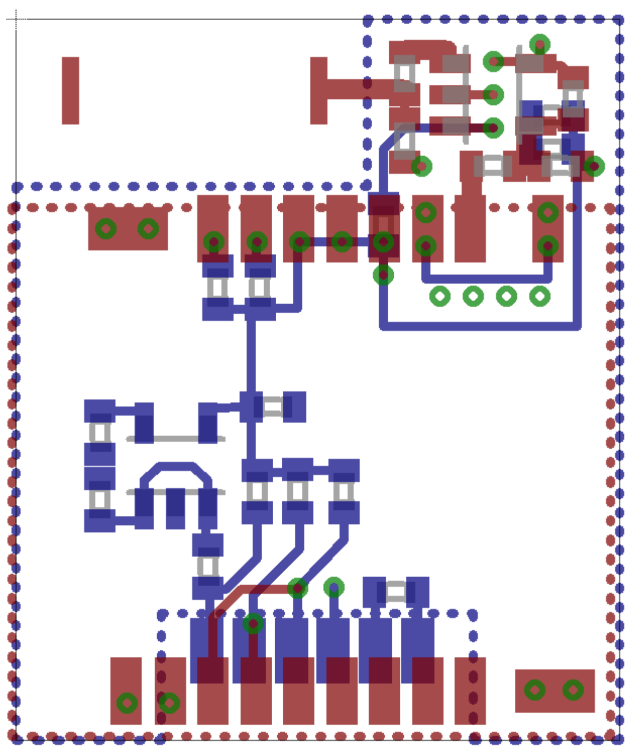

Figure 6.4: PCB ET318

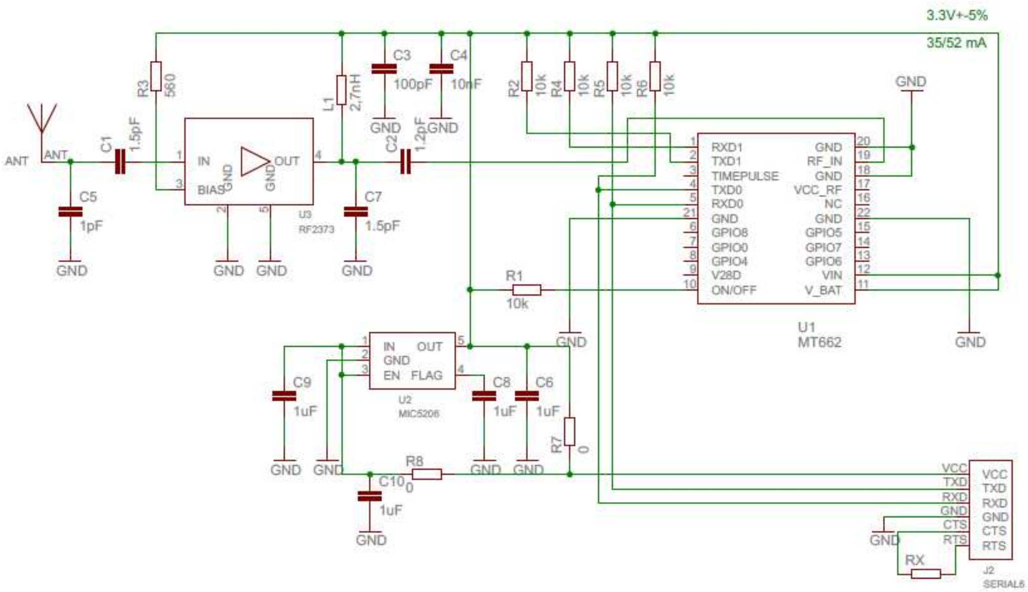

Figure 6.5: Layout MT662

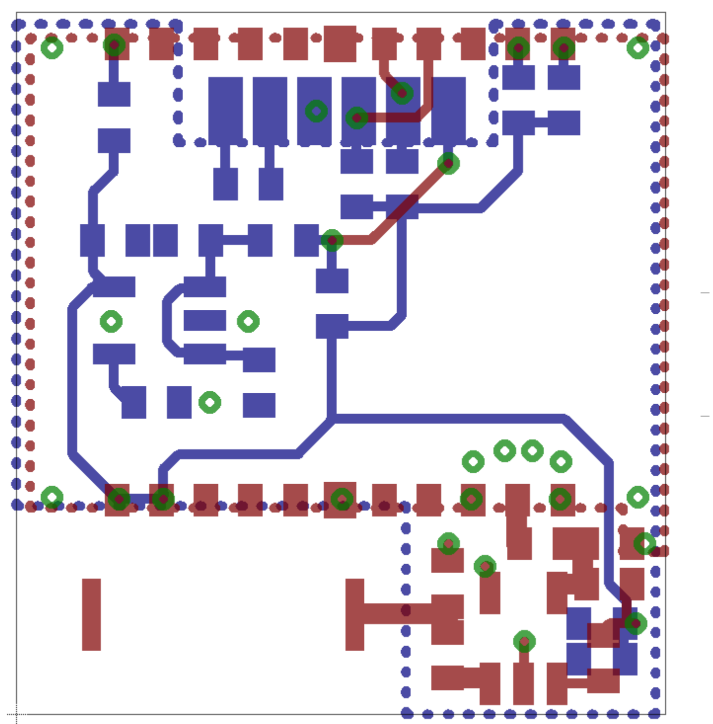

Figure 6.6: PCB MT662

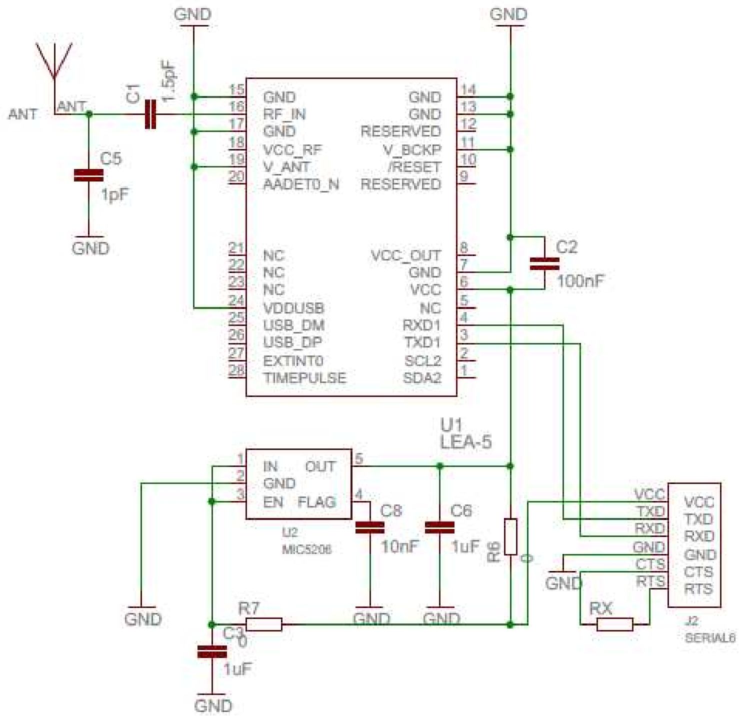

Figure 6.7: Layout LEA U-Blox 5

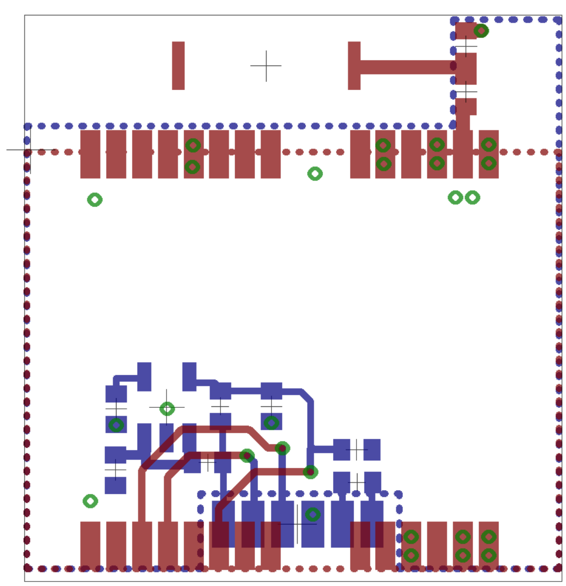

Figure 6.8: PCB LEAU-Blox 5

```
Imports PCComm
Public Class frmMain
    Private comm As New CommManager()
    Private transType As String = String.Empty

    Private Sub cboPort_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles cboPort.SelectedIndexChanged
        comm.PortName = cboPort.Text()
    End Sub

    ''' <summary>
    ''' Method to initialize serial port
    ''' values to standard defaults
    ''' </summary>
    Private Sub SetDefaults()
        cboPort.SelectedIndex = 0
        cboBaud.SelectedText = "9600"
        cboParity.SelectedIndex = 0
        cboStop.SelectedIndex = 1
        cboData.SelectedIndex = 1
    End Sub

    ''' <summary>
    ''' methos to load our serial
    ''' port option values
    ''' </summary>
    Private Sub LoadValues()
        comm.SetPortNameValues(cboPort)
        comm.SetParityValues(cboParity)
        comm.SetStopBitValues(cboStop)
    End Sub

    ''' <summary>
    ''' method to set the state of controls
    ''' when the form first loads
    ''' </summary>
    Private Sub SetControlState()
        rdoText.Checked = True
        cmdSend.Enabled = False
        cmdClose.Enabled = False
    End Sub

    Private Sub frmMain_Load(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles MyBase.Load
        LoadValues()
        SetDefaults()
        SetControlState()
    End Sub

    Private Sub cmdClose_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles cmdClose.Click
        comm.ClosePort()
        SetControlState()
        SetDefaults()
    End Sub

    Private Sub cmdOpen_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles cmdOpen.Click
        comm.Parity = cboParity.Text
        comm.StopBits = cboStop.Text
```

```
        comm.DataBits = cboData.Text
        comm.BaudRate = cboBaud.Text
        comm.DisplayWindow = rtbDisplay
        comm.OpenPort()

        cmdOpen.Enabled = False
        cmdClose.Enabled = True
        cmdSend.Enabled = True
    End Sub

    Private Sub cmdSend_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles cmdSend.Click
        comm.Message = txtSend.Text
        comm.Type = CommManager.MessageType.Normal
        comm.WriteData(txtSend.Text)
        txtSend.Text = String.Empty
        txtSend.Focus()
    End Sub

    Private Sub rdoHex_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rdoHex.CheckedChanged
        If rdoHex.Checked() Then
            comm.CurrentTransmissionType = PCComm.CommManager.TransmissionType.Hex
        Else
            comm.CurrentTransmissionType = PCComm.CommManager.TransmissionType.Text
        End If
    End Sub

    Private Sub cboBaud_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles cboBaud.SelectedIndexChanged
        comm.BaudRate = cboBaud.Text()
    End Sub

    Private Sub cboParity_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles cboParity.SelectedIndexChanged
        comm.Parity = cboParity.Text()
    End Sub

    Private Sub cboStop_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles cboStop.SelectedIndexChanged
        comm.StopBits = cboStop.Text()
    End Sub

    Private Sub cboData_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles cboData.SelectedIndexChanged
        comm.StopBits = cboStop.Text()
    End Sub
End Class
```

Figure 6.9: frmMain.vb code source

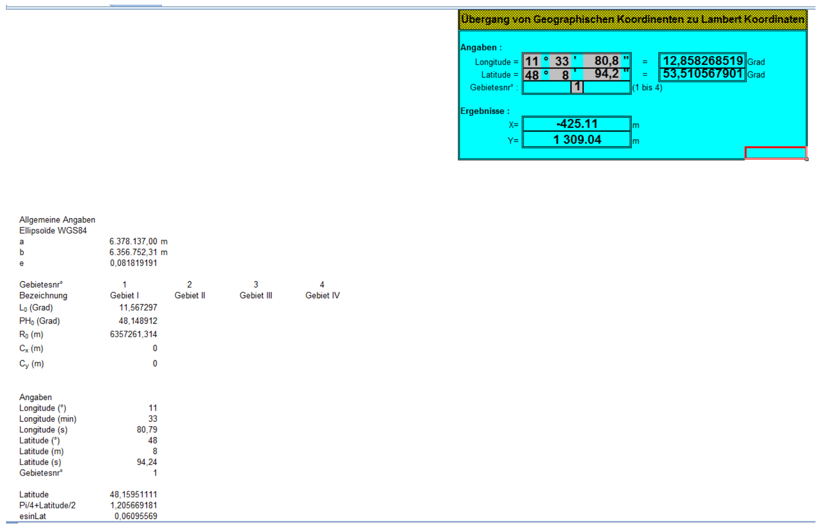

Figure 6.10: Excel document for coordinates transformation

List of figures

Bibliography

[1] Alfred Leick, GPS satellite survey.USA, 3rd edition, 2004
[2] Ahmed El-Rabbany, Introduction to GPS: the Global Positioning System. Britain, 2002
[3] Jesse Liberty, Learning visual basic, 1st edition, 2002
[4] www.ftdichip.com/Products/FT232R.htm
[5] http://www.u-blox.ch
[6] www.gadgetfind.com/4channelrcflyingsaucer.html
[7] www.nasm.si.edu/gps/work.html
[8] www.math.tamu.edu/~dallen/physics/gps/gps.htm
[9] www.sco.wisc.edu/gps/
[10] www.wikipedia.de
[11] www.dafu.de/redir/dgps.html
[12] www.nmea.org/
[13] www.intel.com/technology/usb/