Software defined GPS/Galileo receiver

Filipe Coelho, Gonçalo Tavares(Advisor).

Instituto Superior Tecnico - Instituto de Engenharia de Sistemas e Computadores Investigação e Desenvolvimento em Lisboa (INES-C-ID)
Lisbon 1000-029, Portugal

Abstract: The interest in the Global Navigation Satellite Systems (GNSS) has been increasing in the last decade. In recent years, the GNSS receivers became a present in most of the cars, boats, airplanes and even in smartphones and laptops.

Today, United States’ Global Position System (GPS) is the most utilized GNSS. The Russian GLONASS was a fully functional navigation constellation but it felt in disrepair, leading to gaps in coverage. Restoration and modernization is in underway. The European Union and European Space Agency are developing a public GNSS called Galileo. This system has two test satellites in orbit. China intends to expand their regional navigation system into a GNSS.

The development of this systems and the modernization of the GPS put challenges in signal processing of the signals received.

The present work addresses the problem of the signal processing. The platform to make the signal processing has to be flexible. The software defined radio (SDR) is a system where the hardware part of the receiver has been reduced and most of the signal processing is done in a general purpose processor like the ones found in PCs. A system that used this concept of SDR has developed to address this problem. An analog front-end make the filtering and down-conversion of the frequency and make the conversion for digital form. After the data has to be transferred to a PC for processing, a USB interface has utilized for this transmission.

With the data in the PC, the signal processing begin and the navigation solution are obtained.

Keywords: Software defined radio, SDR, GNSS, GPS, Galileo, GLONASS, USB, software defined GNSS receiver

1. INTRODUCTION

The outline of this paper is structures as follows. Section 2 includes a description of most important GNSS. In Section 3, the system architecture is described. The Section 4 describes the signal processing necessary for obtain a navigation solution. In Section 5, the work developed is showed. The Section 6 shows the results. Conclusions and topics for further research are included in Section 7.

2. GNSS DESCRIPTION

GNSS are based in a simple method to determine the position of the receiver. This method consists in measure the distance between the receiver and some known positions. Knowing a position and the distance to that position, the receiver only can be on a sphere with the center in the known position. With two position known and the distance to those positions, a receiver only can be in the intersection of the two spheres resulting of the two positions. With three positions and three distances it is possible to determine the position of the receiver, because the intersection of the three sphere is only two points and one of them can be eliminated because down make sense French (2006).

In a GNSS system, the satellites broadcast its position and the time of transmission. The time of transmission is compared with the time of reception in the receiver, this gives the time of transmission. The transmission time multiplied by the velocity of the signal (close to the velocity of light) gives a measurement of the distance called pseudorange. The data broadcast by the satellites contains information necessary to calculate the position of the satellite and contain the time of transmission and also include parameter to calculate some correction for the pseudorange calculated. Figure 1 shows the method utilized by the GNSS to determine the position of the receiver. All satellites contain a clock used to determine the time of transmission, this clock can have an offset and drift from the receiver’s clock. This is the reason why normally a minimum four satellites are utilized to obtain a solution.
Today, the most utilized GNSS is the United State GPS. This system provides reliable location and time information in all weather and at all times anywhere on or near the Earth. The complete system utilizes a minimum of 24 satellites to provide global coverage.

The GPS satellite broadcast in two frequency band: one called L1 (1575.42MHz) and other called L2 (1227.60MHz).

Figure 2 shows the modulation of the signal form one GPS signal. The L1 carrier is modulated with a code called Coarse/Acquisition (C/A). The Precision code P(Y) is modulated the L2 carrier. The C/A and P(Y) codes belong to a family of codes called Pseudo Range Codes (PRN). These codes have some desirable properties:

- Satellites can transmit on the same frequency;
- Precise ranging;
- Process gain due to despreading of PRN code;
- Rejection of reflected signal;
- Anti-jamming properties.

The navigation data is modulated in the code C/A for the L1 band and in the code P(Y) for the L2 band at 50 bps. Figure 3 shows the spectrum for the L1 signal. In this figure, the main lobe that contain 90% of the energy of the signal are represented with a bandwidth of 2MHz Borre (2007).

The Russian GLONASS utilized other frequencies and other type of solution for the transmission of all satellites. The full system is composed of 24 satellites. These satellites all have the same C/A code. Figure 4 shows the channels utilized by the GLONASS and the GPS. The GLONASS utilize a system called frequency domain multiple access (FDMA). In this system, the different satellites utilize a slightly different frequency. The system is composed of 12 channels and the two satellites on opposite sides of the earth utilize the same channels.

The Galileo system is the European GNSS. It is in the development phase and at the moment, two test satellites are in orbit. These satellites have the objective of validate the signal utilized in this GNSS. The Galileo is different from the other GNSS present because this system is controlled by different countries and is not controlled by military. The Galileo system has been developed to work with GPS in the same receiver utilizing same antenna and filter for the L1 band. To make this possible, a different type of modulation is utilized to reduce the interference between these two signals. The modulation utilized is the Binary-offset code (BOC) modulation. The effect of this modulation is to make two lobes centered at L1 - 1.023MHz and L1 + 1.023MHz. This doubles the minimum bandwidth of the receiver but make less interference with the GPS.

Figure 5 shows the relation between the GPS and Galileo signal spectrum.

3. SYSTEM ARCHITECTURE

Software defined receiver

Figure 6 shows the differences between the software defined receiver, hardware receiver and “ideal” software receiver. The ideal software receiver only has an antenna and ADC for convert the signal to digital. After the entire signal processing is done in software. The hardware receiver
implements most of the function in hardware and few in software. And the software defined receiver is the close possible to the ideal software receiver that can be made. In a software receiver, most of the hardware of the receiver is removed, and the only essential parts remain. The term Software defined GNSS receiver refers to the application of the SDR to a GNSS system. In this a SDR system, an antenna is used to receive the signal, after the signal is downconverted to intermediate frequency (IF). After the signal is converted to digital. The data converted has to be transferred to a PC for the software to perform the signal processing.

In the hardware receiver, most of the function are implemented in the hardware and the a microprocessor is only utilized to control some hardware parts (Leveson 2006).

The software defined receivers have several advantages in comparison to the hardware receivers. The SDR are much more flexibility, allow the system to be upgraded to new signal and systems only utilizing a software update. The signal also can be recorded to perform post processing. This allows the development of different algorithm and permit the algorithm to be optimized utilizing always the same data. The hardware receivers also have some advantages. The hardware receivers are cheaper than the SDR because it requires a more powerful processor, also have lower power consumption especial important for smartphones.

**Hardware**

Figure 7 shows the block diagram of the MAX2769. This IC performs all the analog processing of the signal. The MAX2769 has two internally matched Low Noise Amplifiers (LNA) used to amplifies the signal that come from a passive antenna or a active antenna. When using the chip to bias an active antenna, the LNA port is automatically selected. The outputs of the LNAs are connected to an output pin. There is an input pin for the mixer stage of the front-end. Between this two pins a Surface Acoustic Wave (SAW) filter can be placed filter the unwanted frequencies. After the signal is filtered, it enters the mixer stage. The MAX2769 is a direct frequency conversion. A Temperature-compensated crystal oscillator (TCXO) is utilized to achieve a better clock stability. The mixer stage down convert the signal form the L1 frequency band to an Intermediate Frequency (IF) of few Megahertz or even 0 Hz. The IF signal is then filtered using the filter included. After the signal is filtered, the signal is amplified using a Programmable Gain Amplifier and converted to digital using the ADC from the chip. The gain of the amplifier can be automatically controlled to obtain a determined bit density. This front-end was chosen because it has a flexible architecture. There several parameters that can be configured, has the IF frequency, the sampling rate, the type of filter, the number of bits and channel to utilize.

The configuration of the MAX2769 is saved in internal registers. This registers can be programmed utilizing the SPI interface. This interface is made by three wire, clock, data and chip select. This interface only allows data to be written in the device. No data can be read from the MAX2769 utilizing the SPI interface.

The FT2232H form FTDI is the chip utilized to implement the USB interface. This chip can convert interfaces like SPI, JTAG, parallel to USB. It has two channels capable of implementing two different interfaces. The FT2242H makes the design easier and faster because it implements the entire protocol in the chip and is not necessary to develop any firmware. For communication with the chip and transfer of data, FTDI provides drivers for every major Operational Systems and there are several code libraries that make the communication very straightforward. These libraries can access directly the chip and is no low level programming is needed. The two channels are utilized, one channel is utilized to perform the SPI communication and programming the MAX2769 and the other is used to transfer the data from the MAX2769 to the PC using a FIFO interface commanded by the clock of the ADC.

The CY7C68013 EZ-USB FX2LP from the Cypress is a High Speed USB peripheral controller. It integrates in a single chip the USB2.0 transceiver, Serial Interface Engine(SIE) and an enhanced 8051 microprocessor. This chip has the disadvantages in relation with the FT2232H.
chip because this chip requires development of firmware to implement the USB protocol and drivers for the OS. But this gives the developer much more flexibility. Also the sustain data transfers of this chip is considerable bigger than in the FT2232H. The type of interface used in this case as a FIFO interface with 8 bits and controlled by the clock of the ADC of MAX2769.

4. SIGNAL PROCESSING

**Algorithms**

The signal processing needed to determine the position of the receiver can be divided in three big parts. The acquisition, the tracking and the navigation solution calculation.

The first step is acquisition. The objective of the acquisition is to determine which satellite signal are in the data collected. Some receivers save some parameters of the navigation message that makes part of the “almanac”. Using the almanac and the time for the clock of the receiver is possible to determine the rough position of the satellites, with a rough position of the receiver is possible to determine the position of the satellites and to determine if there is a line of sight for this satellite. This information can be also used to determine an approximate Doppler frequency for each satellite. The Doppler frequency is a deviation from the nominal frequency of emission. The relative motion of the satellite in relation with the receiver, make the frequency of the signal slightly different from the nominal frequency. This difference is estimate by the acquisition. After which satellites could be in the data using the almanac, the search begins to find the signals. This search is made for different Doppler frequencies and different code phases. There are three major algorithms for this search. One is called serial search acquisition. In this acquisition algorithm, the search for the Doppler frequency is divided in several frequency bins. These bins are in the range of the expected maximum Doppler frequency. For each frequency, a code generator is used to remove the code of the signal. After the results is multiplied by the local generated carrier and by a 90 shifted version to remove the carrier from the signal generating a I and Q values. These results are integrated during a minimum of one code period and the results are squared and summed. If the result is above a determined threshold, the satellite is acquired and the code phase and the frequency of this result are passed to the tracking function. This algorithm as to be performed for all the code phases and for the entire frequency bin.

Other algorithm is called parallel frequency search. In this algorithm, the data is multiplied by a locally generated code and the result is transformed to frequency domain utilizing a Fourier transform. The result is squared and if the result is bigger that a determined threshold the satellite is acquired and the Doppler frequency and the code phase is found. This algorithm as to be performed to all the code phases possible in the code.

The last algorithm is called parallel code phase. In this algorithm, the carrier is removed from the data using a locally generated carrier generating an I and Q signals. These signals are transformed to frequency domain using a Fourier transform a local PRN code is transformed to frequency domain and conjugated. This is multiplied by the frequency domain of the data without carrier. The result is transformed to time domain using a inverse Fourier transform. The result is squared. This implements a circular correlation between the code and the data without the carrier. If the satellite signal is in the data, a peak appears in the results. This algorithm has to be performed to all frequency bins Ziedan (1996).

All these algorithms have advantages and disadvantages. The serial search algorithm is the easier the implement but is the most time consuming. The parallel frequency search has the bigger resolution in the frequency but as to be performed with all the different code phases- The parallel code search is the most complex to implement but gives more precision on the code because the resolution is at the sample. To refine the acquisition results, in the case of the parallel code phase search, a parallel frequency can be implemented with the determined code phase to obtain a more precise frequency results.

With the results of the acquisition, the tracking of the signal can begin. The objective of the tracking is to refine the coarse results of the acquisition and to follow the changes in the acquired signal. During the tracking parameters like the Doppler carrier frequency and code phase, have to be tracked because the relative motion of the receiver and the satellite is not always the same, so this values can change. In the case of the GPS (it is similar for other GNSS), the tracking of the signal is obtain using two loops, one to lock the carrier frequency and other to lock the code phase. A Costas loop is utilized to track the frequency of the carrier. GPS utilize the Costas loop because this loop is insensitive to 180 phase shifts. In the Costas loop, the incoming signal is multiplied by a locally generated code replica to remove the code of the carrier, after the result is multiplied by a locally generated carrier replica and a 90 version of this carrier to remove the carrier. The output is filtered and the result is utilized in a discriminator to determine the error and the correction to make in the local carrier replica. The objective of this filter is to maintain the carrier and the local carrier replica in phase, in the I arm. There are difference types of discriminator but the function is the same for all. If the local carrier replica has a positive phase shift, the frequency is reduced to reduce the phase shift. If the local carrier replica has a negative phase shift, the frequency increased to reduce the phase shift.

The code tracking loop is a Delay Lock Loop (DLL). In this loop, the incoming signal is multiplied by the local carrier replica. After the signal with the carrier removed is multiplied by three different version of the code. This version of the code is called early, late and prompt. The early is a version of the prompt code with a negative phase shift, the late is a version of the prompt signal with a positive phase shift. These three versions of the code are multiplied by the incoming signal without carrier and the results are integrated and dumped. These results are used to determine what correction to make is necessary utilizing a discriminator. The objective is to make the I prompt result bigger than the early and late results. The discriminator tries to make this. Some discriminators look at the values of the early and late results and if the late result is bigger than the early results, the local code phase is behind the received code phase. If the early code result is
bigger than the late result, the local code phase is advance in relation to the received code phase. Most of the time, six correlators are used to a better isolation of the dependence of a good carrier tracking. The other three correlators are used to track the code in the Q signal resulting of the carrier removing.

The results of the I prompt correlators are used to determine the navigation data. The first step of the recovering of the navigation data is find the preamble of the each subframe. To confirm that the correct begin of subframe is founded, the distance of 6 seconds between subframe can be checked. After the subframe is found, the bit transitions are also found and a mean is done in the results of the I prompt correlator to determine the bit. After the navigation is decoded, the necessary information for the precise location of the signal became available and the pseudoranges can be calculated.

After the pseudoranges are calculated, the navigation solution can be found if four or more measures are made.

Software

There are several open source softwares for processing the sampled data. One of these software comes in the book ?.

5. DEVELOPED WORK

Board 1

The first prototype board was build utilizing the Evaluation kit (EV kit) for MAX2769. This evaluation kit contains the MAX2769 and the entire necessary component for the analog front-end. The EV kit includes a MAX2769, a TCXO for the frequency reference, power regulators and transceiver to make accessible some signals of the MAX2769 using the SMA connectors. A parallel port and drivers are included to make the programming the MAX2769. The programming of the MAX2769 can be made utilizing a program made by MAXIM, that allows to choose the configuration of the MAX2769 using a graphical interface. The analog from are used in conjunction with the FT2232H for transmitting the digital data to the PC. For a easier implementation of the USB interface, a module for FT2232H is used. This module have a FT2232H chip, a crystal for the oscillator, a USB mini connector and a Linear Low Dropout (LDO) regulator for conversion of the 5V for 3.3V. Most modern laptops do not have a parallel port. To program the MAX2769 using the SPI, one channel of the FT2232H chip is utilized.

To transfer the data form the EV kit to the PC, a FIFO interface is used in one channel of the FT2232H. This interface takes 8 bits at the falling edge of the write signal. This is a problem for the design because the front end can convert the signal to digital with the maximum of 2 bits for the I channel and other 2 bits for Q channel. So at the minimum, half of the data send through the USB interface do not contains data form the ADC. To resolve this problem, a data packer is developed using a circuit that divide clock by two and have two groups of flip-flops. One group of flip-flops saves the even samples, and the other saves the odd samples. So the data is only transfered when two new samples are in the flip-flops.

This board do not utilize a SAW filter.

Some software has to be developed to this board. To program the MAX2769, a MATLAB graphical interface was developed. This interface utilizes some libraries developed to communicate with the FT2232H device. In the interface, the value of the MAX2769 can be chosen by clicking in the bit that the user wants to change. It also possible to determine the parameters of the transfer of data form the MAX2769 to the PC, like the size of data to acquire, the name of the file to create and the format of the digital data. This program calls another program that make transfered of data to the PC. The sampler program was developed using the C++ language because the libraries that can for the FTDI are made in this language. The idea of this program is to utilize a circular buffer for transferring the data to the PC. One thread is created to puts as many request of USB data as possible in the queue so the data transfer never stops and put the data transferred to the circular buffer. Other thread was created to transfer the data from the circular buffer to one file. The buffer is utilized to never stop the transmission of the data because of the write to disk operation.

Board 2

The second prototype board was developed with the first prototype board as a reference. This board replaces the need for a EV kit, which is expensive and big, and tries to obtain a solution using the MAX2769 directly in the main board. The signals from the GNSS system are very weak so obtain a solution using the MAX2769 directly in the main board. The signals from the GNSS system are very weak so a good layout and noise isolation is needed. All the power pins of the MAX2769 have two decoupling capacitors, there two LDOs to give power to the analog and digital parts of the chip.

The LNAs are internally matched to 50 Ω and the antenna have a characteristic impedance of 50 Ω so a 50 Ω transmission line should be used in this line. Also the line between the LNA output and the input of the mixer stage must have 50 Ω. The calculation done show that the width of the track must be 3mm for the technology used. This is not possible to make, so the bigger size is used. The module of the FT2232H is placed in two headers. The logic that perform the packing the two data samples in ones is placed below the module and in the other side of the board.

This board utilizes the same software of the first prototype board because the hardware that makes the communication is the same.
This a board with free and open hardware and software from the web site http://www.gnss-sdr.ru. This system utilizes the same analog front end utilized by me in the two first prototype boards. But it utilizes the chip CY7C68013A for the USB interface. This chip has some advantages as the higher throughput of the data connection. Also for the data packing, this board utilizes a Complex Programmable Logic device (CPLD). These devices perform the control of several signals and makes the packing of the data. This IC has the option of In-System Programming. This allows the IC to be programmable while placed in the board. This gives the used a bigger flexibility in the design. This board also has a clock section more complex than in my case. This board is prepared to receive a external clock and to put the clock generated in a SMA connector.

**GNSS SDR board 2**

This a board made with the reference of the first GNSS SDR board but removing all the unnecessary parts and changing the layout to make the board smaller. The clock section is removed and only the TXCO is left. Some component like the leds and switch are removed to make a smaller board. The result is a smaller board. This has the advantage that is cheaper to make.

The software for this board is essential the same as the before. Only in code for the CPLD, the not necessary features are removed.

6. RESULTS

**Board 1**

Figure 10 shows the results of the acquisition for one satellite utilizing the parallel code phase search algorithm for the board 1. In this figure the satellite signal is not present in the data. In the x axis, the code phase of the search. The Y axis shows the frequency bins. The Z axis shows the result of the acquisition. In this figure it is not visible a peak that indicates a high value of correlation.

Figure 10 shows the results of the acquisition for one satellite. In this figure is visible a distinct peak, this indicate that a high correlation is found. The frequency bin and code phase for which the result is maximum is the parameters of the acquisition for this satellite. The coarse value of the carrier frequency will be refined.

Figure 12 shows the result of the tracking function. In the first row in the left, the results of the I correlator and the Q correlator are plotted. Two groups of results are plotted close to the I axis. This demonstrates that the Costas loop is tracking the carrier and almost all the energy of the I correlator. In the right of the first row, the I correlator result plotted in relation to the time. Here is visible the transition of the bit of the navigation data. In the plot called “Raw PLL discriminator”, the error of the carrier tracking loop is plotted. The carrier stays always
in track because the error does not diverge from zero. The plot called “Correlation results” shows the results of the correlation for the early, prompt and late codes. The objective of the code tracking loop is to maintain the prompt code correlation always bigger than the early and late correlation results. The results of the early and late correlators should be equal. This plot shows that the prompt correlation is almost always bigger than the other correlations. The late and early correlations are also very similar. The “Filtered PLL discriminator” shows the value that enter the Numeric Controlled Oscillator (NCO) of the carrier. This value only account for the difference from the nominal frequency value. The “Raw DLL discriminator” shows the error in the code tracking loop. These values not diverge from zero, so the code is always locked. The “Filtered DLL discriminator” plot shows the value of the NCO for the code frequency.

Figure 13 shows the navigation solution obtain utilizing the Board 1. In the plot “Coordinated variations in the UTM system”, the variations of the coordinates are showed in relation to a mean position. The GPS signal is sampled at a fixed position, so the variations are in solutions. The line in red represent the variation in the Up coordinate in a Earth Center Earth Fixed system, the blue line represent the variation in the East coordinate and the green line represent the variation in the North coordinate. The variation of the Up coordinate is bigger than the variation of the North and East coordinates. This can be explained by the position of the satellites. All the satellites are above the receiver so the Up coordinate are the less precise. The plot “Position in UTM system” shows the position of the mean solution. This solution is marked in red, the other solution are displayed in blue. In the plot “Sky plot” show the location in the sky, the position of the acquired satellites.

Fig. 13. Board 1 navigation solution results.

Figure 14 show the tracking results for the Board 2. In the “Discrete-Time Scatter Plot”, the I and Q correlators result show two groups of results much this is much closer than in the Board 1 tracking results. In the plot in the right, the results of the for the I correlator is plotted in time domain. The bits transitions are clearly visible, but the results of the correlation are much smaller than in the Board 1 tracking results. In the plot the “Correlations results” shows the early, late and prompt correlator results show two groups of results much this is much closer than in the Board 1 tracking results. In the plot “Position in UTM system” shows the position in the sky, the position of the acquired satellites.

Fig. 14. Board 2 tracking result.

Fig. 15. Board 2 navigation solution.

Board 2
loose tracking. The results for the error in the case of the Board 2 than the results of the Board 1.

Figure 15 shows the navigation solution for the board 2. The first line plot shows the variation of the coordinates in the ECEF system. The variation in the Up coordinate is much bigger than in the Board 1. In the East and North coordinates the variation is also bigger. The plot “Position in UTM system” show the mean position in red and the other coordinates in blue. In the last plot, the position of the satellites in the sky is plotted.

**GNSS SDR 1 Board**

The figure 16 shows the results of the Probe data function for the GNSS SDR Board 1. The spectrum of the signal is visible on the top and it is centered in the IF frequency and have a passband with 3B of bandwidth.

The figure 17 shows the tracking function plot for one of the signals. In this figure, in the first row is possible to see that the I and Q plot are concentrated around two unique spot close to the I axis. This is what is expected in this type of demodulation. The result of the correlation must be higher for the prompt I correlator.

**GNSS SDR 2 Board**

The figure 18 show the navigation solution for the board 3. In the first row, the results for the navigation solution are plot in North, East and Up coordinates are plotted. The solution in the North and East coordinates are close to the mean position. The variation in the solution in the Up coordinate is bigger than the other coordinates variation. In the last row in the left, the solution is plotted in the North and East coordinates. The solution in blue is close to mean position in red. In the left of this row, the sky plot of the satellites is plotted.

**GNSS SDR 2 Board**

The figure 19 shows the plot of the probe data function. The shape of the IF filter comes is plot the first row. The filter look like the expected, with 2 MHz bandwidth and centered on the IF.
Fig. 20. Board 4 navigation Solutions

The figure 20 shows the results for the navigation solution. The solution is showed in the first row plot, the coordinates are in the Earth Center Earth Fixed. The North and East coordinates, the solutions are close to the mean position. In the Up coordinate, the variation is larger than in the others coordinates. In the last row the navigation solutions are plot in the East and North coordinates. The mean position is showed in red and the other solutions in blue. The solutions are close to the mean position. In the right of the last row, the plot of the position of the satellites in the sky is showed.

7. CONCLUSIONS

The goal of this work was to develop a complete solution for a software defined GNSS receiver. The goal was achieved. The results show that the system is able to perform all the function of a hardware receiver, the acquisition, tracking and navigation solutions. This work also shows that there are different algorithms that can be implemented using a software defined receiver. The flexibility provided by this type of receiver can be used to test the need GNSS like the Galileo and makes possible to utilize and test different algorithms. This work also shows that the performance of the receiver is closed connected to the layout and noise isolation of the PCB. In the second board, the usage of a dual layer board, the mismatch of the 50 Ω transmission lines and the proximity of the switching signals make the results more noisy and the sensibility is smaller because a small number of satellites are acquired.

With a proper layout and better noise isolation, the GNSS SDR boards achieve better performance. This is result of the four layers PCB and the usage of smaller package and close position of the decoupling capacitors make the system performance better.

This work show that the open source community has several project for the Software defined GPS receiver and even Software defined GLONASS receiver.

REFERENCES


