GRAS Meteorology SAF

Assisted Raw Sampling Report

Version 1.2
26 May 2006

Prepared by

Danish Meteorological Institute
The Met Office of the United Kingdom
Institut d'Estudis Espacials de Catalunya

SAF/GRAS/IEEC/ALG/OL/003

APPROVAL LIST		
NAME	SIGNATURE	DATE
Prepared by: Oleguer Nogues, Santi Oliveras, Laust Olsen, Antonio Rius		26/05/2006
Approved by: GRAS SAF Project Manager		26/05/2006

Document Distribution

Name	Position	Company
GRAS SAF Team		DMI, Met Office, IEEC

Document Status Sheet

Author	Issue	Date	Pages	Change description
Oleguer Nogues	v 0.1	10/12/2005		First Draft
Oleguer Nogues	v 1.0	22/12/2005		Several modifications after first revision done by A. Rius
Santi Oliveras	v 1.1	11/01/2006	sec 3.4	Expanded the ARS website description
Laust Olsen	v 1.2	26/05/2006	chap 4-5	Added chapters 4-5

Table of Contents

List of Figures

1 Introduction

1.1 Contents and Scope

This report summarizes the activities performed during the period September 2005 to March 2006 within the ARS project as was established in the ARS Proposal [RD1]. In a first section we restate the objectives. It continues with a detailed description of the work done to implement the data acquisition system. And ends with an Outlook section.

The subsections 3.1, 3.2 and 3.3 give implementations details, which could be used to understand the level of the implementation effort. Our system is based in a Novatel GPSCard, but the approach taken could be easily applied to other equivalent receivers.

The subsection 3.4 (ARS website) is oriented towards the actual use of this data.

The section 4 describes the algorithms required to remove the effect of the navigation message bits in the signal phase, and in section 5 these concepts are applied to the data acquired during an experiment aimed to obtain GPS data in a coastal occultation experiment.

The instrument used to validate the concept experimentally, the GOLD-RTR, produce in-phase and quadrature samples (I and Q) of the downconverted L1 band GPS signals, sampled at the same rate (1 kHz) as the highest available in the GRAS receiver.

2 Objectives

During the last GOLW meeting, it was made evident that to obtain the full benefit of the GRAS instrument it was needed to find a mechanism to extract the navigation bit from the phases produced by such instrument, when it is in a high sampling rate mode. At the end it was considered, within the GRAS SAF, the interest of an action to study this question. The points, which this study should cover, are:

- To extract the Navigation Bits (NB) from the GPS signals using simple and reliable equipment.
- To propose means to transfer the NBs to a GRAS SAF server.
- To define algorithms to correct, in post processing, the GRAS observed phase when sampling rates are 50 Hz or higher.
- To produce, and to test, a prototype implementing the required functionalities to deliver to the GRAS SAF servers navigation bit streams.

While this activity is within the GRAS SAF responsibility, we expect that the results could be implemented in the future within the EUMETSAT processing chain.

3 Navigation Bit Extraction Activities

In this chapter, the activities involved with the first part of the project are described. This first phase consist in the setting up of the system to extract the NBs from a GPS receiver and save all the collected data in a server; and then, make available this data in a visualization way, here implemented in a website. These activities are basically described in four points:

- The hardware identification to support the functionalities.
- The software implementation, with the programs to retrieve the data from the receiver and send this information to the server, which controls that the receivers are working well and saves the available data to a database.
- The tests or validations used to check that the NBs are correct.
- The website implementation, that shows the NBs stored in a database in a suitable graph.

This four points are described below in the corresponding sections.

3.1 Hardware Identification

To get the navigation bits, we need a GPS antenna connected to a GPS receiver, and also a PC connected to the receiver via serial port to collect the subframes and send them to a server via Internet. There are other possibilities as i.e. a GPS Card integrated into the PC or other types of GPS antenna/receiver.

Figure 3.1 shows the hardware used to get the NBs at the receiver station. The antenna is a GPS positioning avionics antenna; the receiver is a Millennium GPSCard,model OEM-3 and software version 4.45/2.03 (it works at 12V); and we use a conventional PC. During the a operational phase, it would be suitable to use a UPS to supply both PC and GPS receiver.

Figure 3.2 shows the connection between the server and the receiver stations. This connection is via Internet and we use SSH protocol to transfer the data from the receiver to the server. In the current demo phase, only one receiver located at IEEC in Bellaterra is available, but the server is prepared to support more than one receiver. The server is also located at IEEC in Bellaterra.

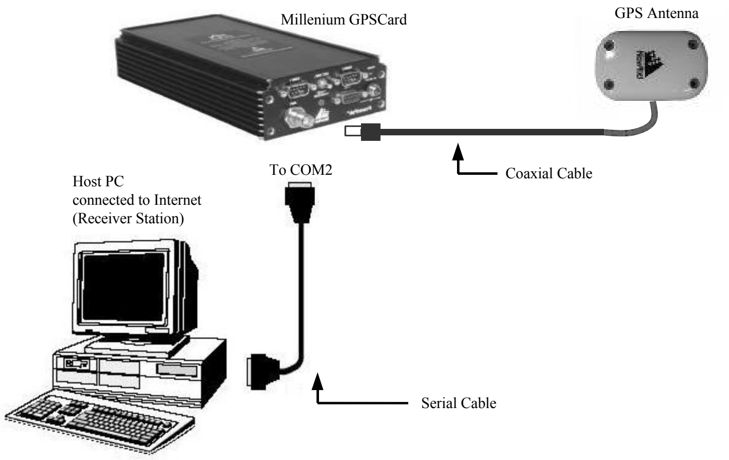

Figure 3.1: Receiver Station Hardware PDF1.

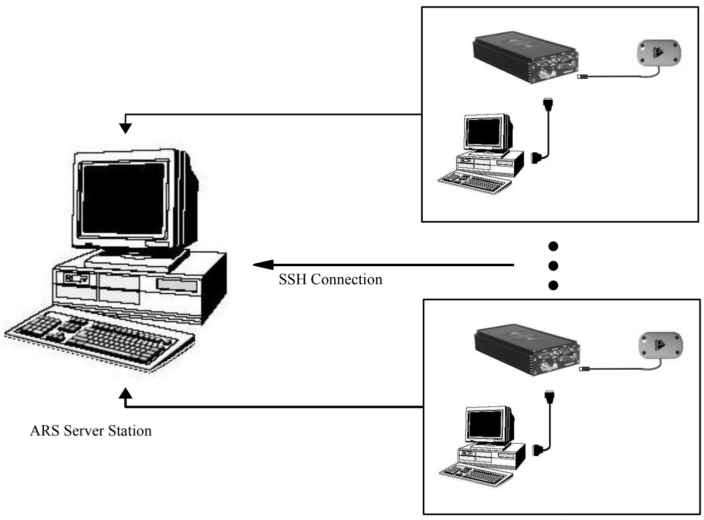

Figure 3.2: Connection between ARS Server and several Receiver Stations.

3.2 Software Development

This section describes the software implementation to get the data from the receiver and send it to the server, which stores the data in a data base. All the PCs are Linux platforms machines. We have to bear in mind two main blocks, one at receiver station PC, and the other at server station. Figure 3.3 shows the essential software block diagram of the ARS project.

In the receiver station side there's a process called *getfrma*, a C program, that opens a connection via serial port (/dev/ttyS0) to communicate with the GPS receiver. It resets the receiver at the beginning, to set it in a known state, and then starts to retrieve the NBs. As shown in the referred figure, *getfrma* process writes two output files; the first one is a file that contains the NBs of the current GPS week (one file per week) and stored in the local hard disk as a backup, and second one is a file labeled *ars.txt* that contains the last NBs data and also error or message lines. This file is the one read the server via SSH channel to get the NBs and to save them in a database. *Getfrma* process can run independently of the server; this fact makes the system more robust.

In the server side, as shown in figure 3.3, there's a main process called *ARS Controller* that controls all the subprocesses or threads in the server side to retrieve the data from the different remote receiver stations and save it to a data base. For each receiver station there's a thread called *ARS Receiver Thread* that retrieves the NBs from the correspondent receiver; it opens a SSH Channel and executes *getfrma* process (if it doesn't exist), and another remote process that reads continuously *ars.txt* file which contains the last subframes and also error or message lines. It also calls at the beginning a remote program called *requestNAVB* that tries to retrieve the NBs in the backup files from the last SOW subframe stored in the MySQL database; this process is called because could be the possibility that the receivers work well, but the server falls. If some error lines appear or the communication is broken (a long period of time without data or SSH connection broken), then the server restarts *getfrma* remote process. All this actions increase the data availability and robustness of the whole system, obtaining a low data loss probability.

All the code implemented in the server is programmed in Java, so the processes are very stable, and with platform independence. Also, with Java it's easier to implement a system with different subprocesses and i.e. to avoid collisions when data is stored in the database by different receivers simultaneously.

As mentioned before there's a communication protocol between the receiver and *getfrma* process; this communication protocol follows the commands described in the Millennium GPSCard Command Descriptions Manual [RD2]; the commands are in ASCII format and the data logs could be both ASCII or binary formats, but in the implementation only ASCII format is used because it is standard.

It's important to know that all the Raw Navigation Data is checked and validated; so it's practically impossible to save wrong data. The next section explains these checks and data validations.

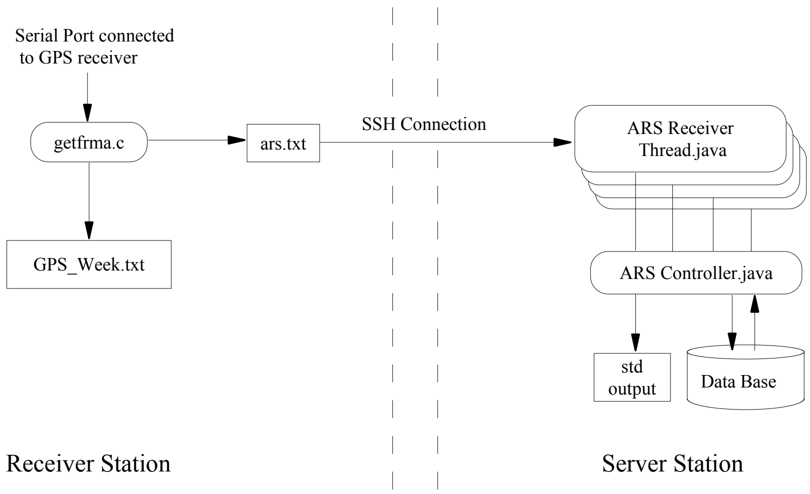

Figure 3.3: Software block diagram.

3.3 Testing and Validation Data

This section describes the control protocol to ensure the success of the system, and the tests carried out. Also, the steps to validate the data read by the GPS receiver.

This control is done in *getfrma* process and also in the server with the reception of error or warnings messages. The process have a main loop with a counter. This counter counts the seconds between received subframes, so when the receiver don't send subframes during a time over 6 seconds, here in the code a time out of 10 seconds, then a RVSA[1] log is sent to the receiver to request the status of the receiver; depending on this status some actions are done. There's also a handler that activates a routine when a new line is sent by the GPS receiver throw the serial port connection; so when a new line is received, if this line belongs to a FRMA log or a RVSA log then this counter is set to 0. If the counter in the main loop is up to 40 seconds, then we understand that there's a problem with the serial port connection and the process send an error message to the server and stops the execution; then the server tries to restart the remote process and notifies this fact in the standard output of the server process.

So, with this implementation, some tests where done to check the robustness of the system. This tests are listed below:

- Test 1: Disconnection of the antenna from the receiver. The system outputs error messages indicating this fact; after antenna reconnection, the system comes back to work well.

- Test 2: Disconnection of GPS receiver (getting serial cable out of the PC, or switching off the receiver). An error message with no response by the receiver is sent to the server and this tries to restart the process, also a message is sent to standard output; after receiver reconnection, the system comes back to work well.

- Test 3: Disconnection of the remote station. The server gets a ssh connection error and notifies it to the standard output; after one minute tries to reconnect with the remote station. When the remote station is reconnected, then the system comes back to work well.

- Test 4: Disconnection of the network (Internet disconnection by one or both stations). A similar behavior as last test, but in that case the remote stations are still getting the data from the receiver and saving the data into the backup files. So, when Internet connection is reestablished,the system can recover the lost data and comes back to work well.

- Test 5: Disconnection of the server. The remote stations are still getting the data from the receiver and saving the data into the backup files. When the server is set up, and the *ARS Controller* process reestablished, this process executes both *requestNAVB* and *getfrma* processes at each

[1] This log message sends information about GPS receiver status. For more information see in the documentation [RD2], page 212 to see RVSA log format, and page 198 to see table D-5 with the status field description.

remote station to recover the lost data and restarts the connection to keep the data retrieval. So, the system doesn't lose any subframe, and comes back to work well.

Note that only in the first three cases we lose data (only the subframes during the disconnection, the stored data is still standing).

Now, we have checked that the system is robust, so the unique way to lose data is a hardware failure in the receiver stations. Then, the critical point of the system is in the receiver station side. So, to strengthen the system we can install a UPS in the remote station, and also use a GPS receiver with a buffer.

But, how we are sure that the saved data is valid data? This question is answered below, using some images.

In figure 3.4 there's a diagram that shows the validations done by the *getfrma* process when new data is available from the GPS receiver. When new data is available, a signal is thrown by the serial port and catched by the *signal_hander_IO* routine. As we see in the figure, then this data (an ASCII line) is read and processed. As mentioned before, the receiver only sends, after the setting up, two types of log messages[2]; FRMA logs with the NBs, and RVSA logs with receiver status (requested when no data is received during more than 6 seconds that corresponds to the subframe regular recurrence). So, the first validation is to check the LOG structure and checksum, this checksum is an exclusive OR (XOR) of all bytes in the log. If this validation succeeds then depending on the log type, different actions are done.

In the RVSA log cases, the receiver status is checked; so, if critical flags are activated, then error messages or warning messages are sent to the server via *ars.txt* file which notifies them in the standard output, and in case of fatal errors, the process is stopped and restarted by the server.

In the FRMA log cases, the common ones, different validations are done. First, you have to take into account that we are receiving subframes (300 NBs); so, in the figure 3.5 you can see a subframe structure. A subframe is divided into 10 words of 30 bits, and each word has 6 parity bits. To validate the subframe, the process calls the *Parity_check()* routine, which checks the 10 words of a subframe following the standard interface specification IS-GPS-200 [RD3]. This function, first checks the preamble in the telemetry word; and after that, checks the parity bits according to the Parity Encoding Equations (see table 20-XIV in IS-GPS-200 document [RD3]). Figure 3.7 shows the user implementation of parity algorithm. After this parity check, an additional

[2]See chapter 3, of Millennium manual [RD2], with the data log description.

check is done; if you see in the figure 3.6 you can see the telemetry and HOW words. From the HOW word you can extract the GPS second of week (SOW); and on the other hand, the receiver gives us this same information into FRMA log. So, the process checks that both informations are in accordance with themselves. If there's an error higher than one second, then the subframe is rejected; if the error is one second, the SOW sent to the database is the SOW from the extracted HOW word; in both cases a warning message are sent to the server.

After these checks, if there hasn't been any error, the subframe is written into *ars.txt* and backup files (every GPS week, the backup file is closed and GZIP compressed, and another one with the new GPS week is opened to content the new subframes).

So, bearing in mind that the data is read by the server via SSH protocol, that controls errors itself, we could be sure that all saved data is valid data after all these validations.

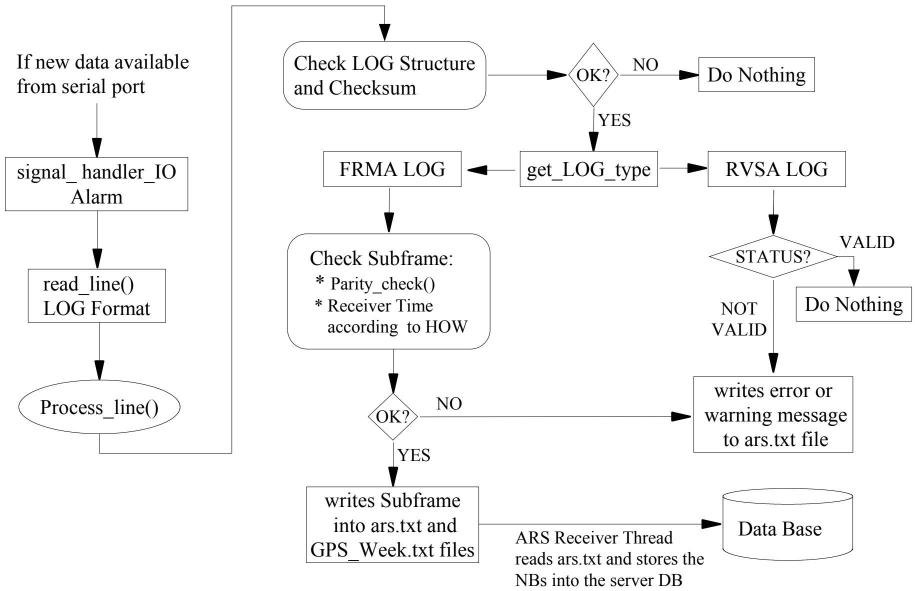

Figure 3.4: Flow data and Validation Diagram.

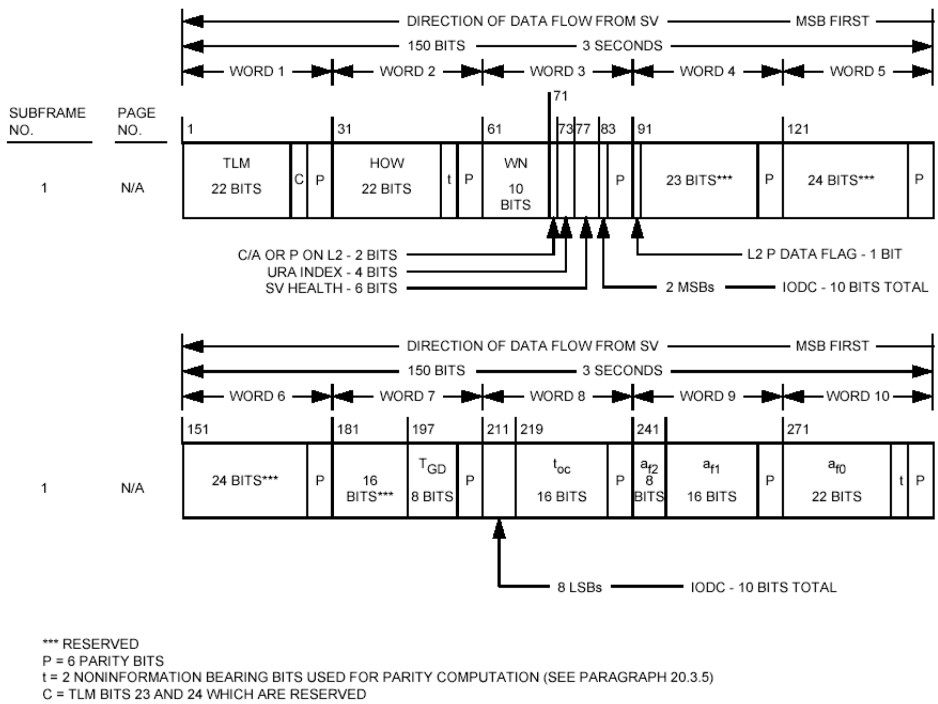

Figure 3.5: Subframe Structure [src: IS-GPS-200 [RD3]].

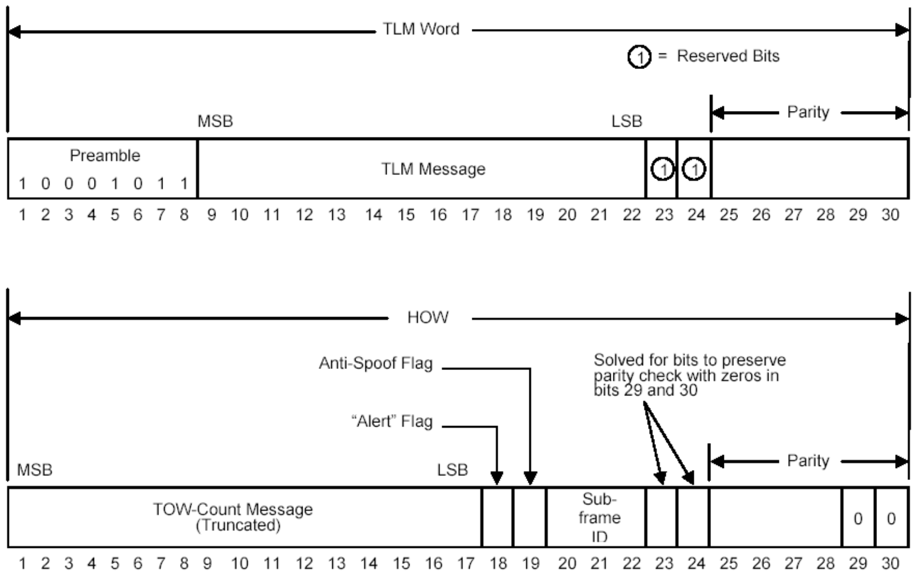

Figure 3.6: Telemetry and HOW Words Structure [src: IS-GPS-200 [RD3]].

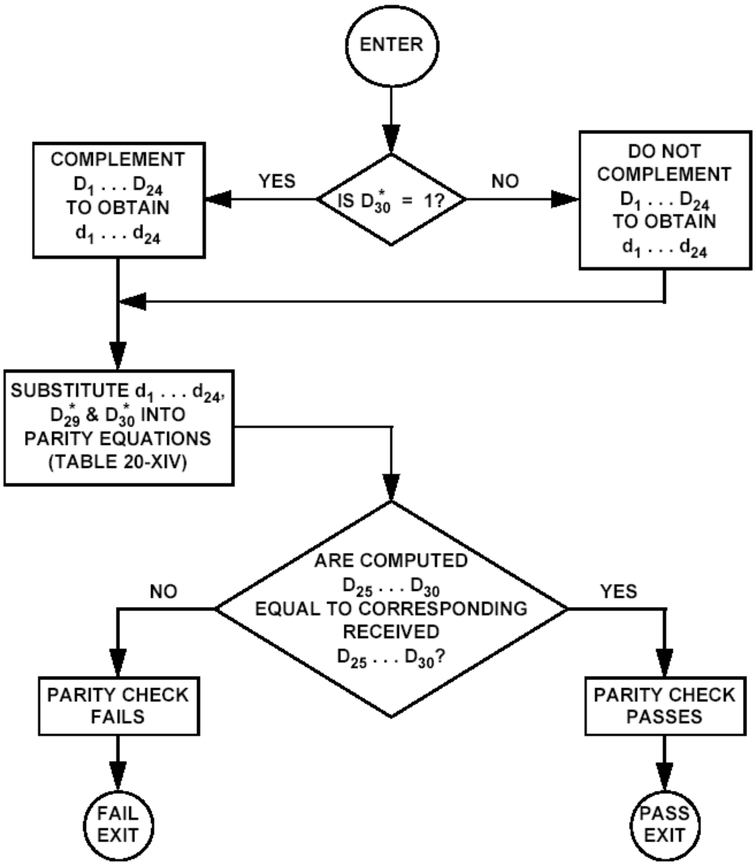

Figure 3.7: Example Flow Chart for User Implementation of Parity Algorithm [src: IS-GPS-200 [RD3]].

3.4 ARS Website

This section explains how the available data is shown in a visual way. In our implementation, the data could be seen in a website. This website is a simple page, easy to use, that shows us the NBs availability, see figure 3.8, and the NBs into an interval time of 5 minutes per each PRN, see figure 3.9.

The graphics are computed in real time with the data available from the MySQL database. As each frame from each PRN has a unique ID, it's impossible to replace or delete any subframe. We are not sure if the database could support a huge amount of information during a long period, because we haven't had enough time to check this issue. At the moment the system works well. If appear some problems in the future, related with the database capacity, a new implementation, i.e. exporting monthly the data into files, could be done.

The URL website is: **https://bond.ieec.uab.es/ars/**

In the website, you see an introduction to the project, a time selector to choose other GPS time window, and an availability graphic with the data from the last hour[3]. This plot (see figure 3.8) shows the Navigation Bit Message hourly subframe availability recorded during last or selected GPS hour; in each row there's the information that corresponds to the PRN; so, there are 32 PRNs possibles and, for that reason, 32 rows. If you click into one 5 minutes block from a PRN, then a second plot appears, see figure 3.9. This second plot, shows the NBs; each row corresponds to one subframe, and as we are showing 5 minutes and there's one subframe each 6 seconds, 50 rows are shown. Notice that you could see regular recurrence each 5 subframes (5 subframes builds a frame), and also from bit 31 till 47 a counter that corresponds to the HOW word (see figure 3.6).

At the moment there's no possibility to retrieve from the website the NBs because no formats are defined, and because till now is not an objective in a demonstration phase.

[3]The availability in our set up is limited to one receiver.

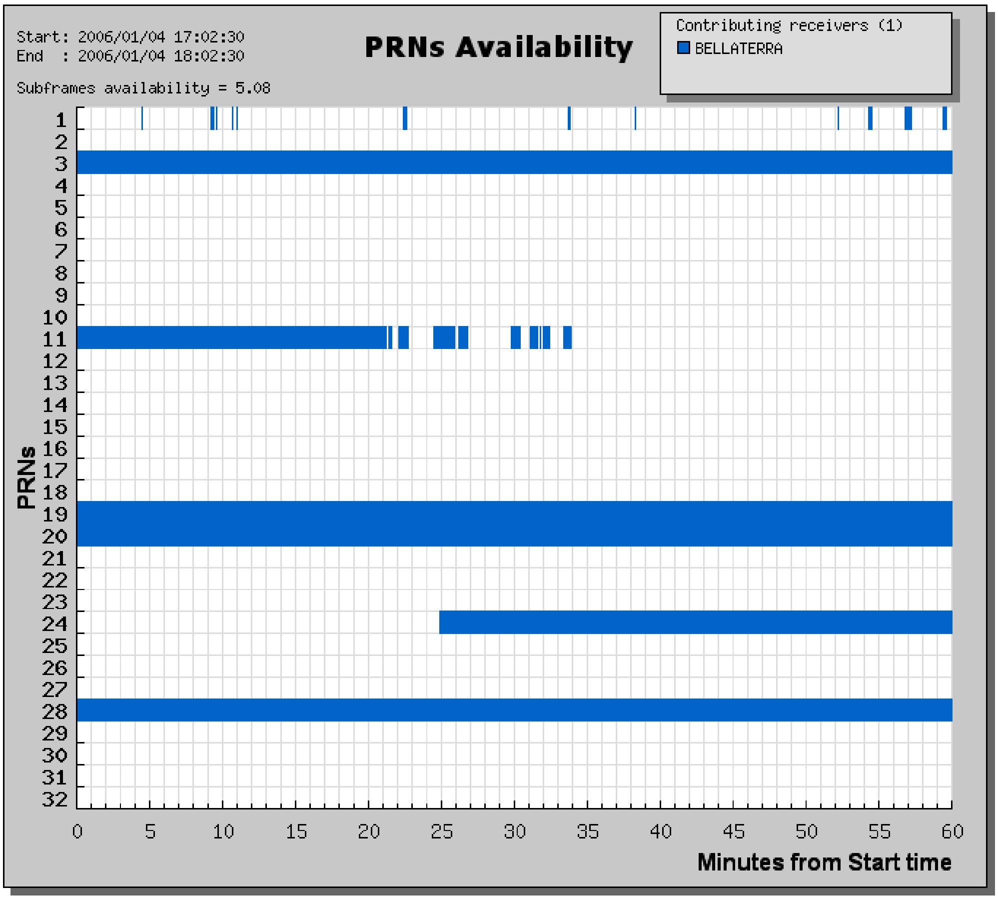

Figure 3.8: Availability Graphic from website.

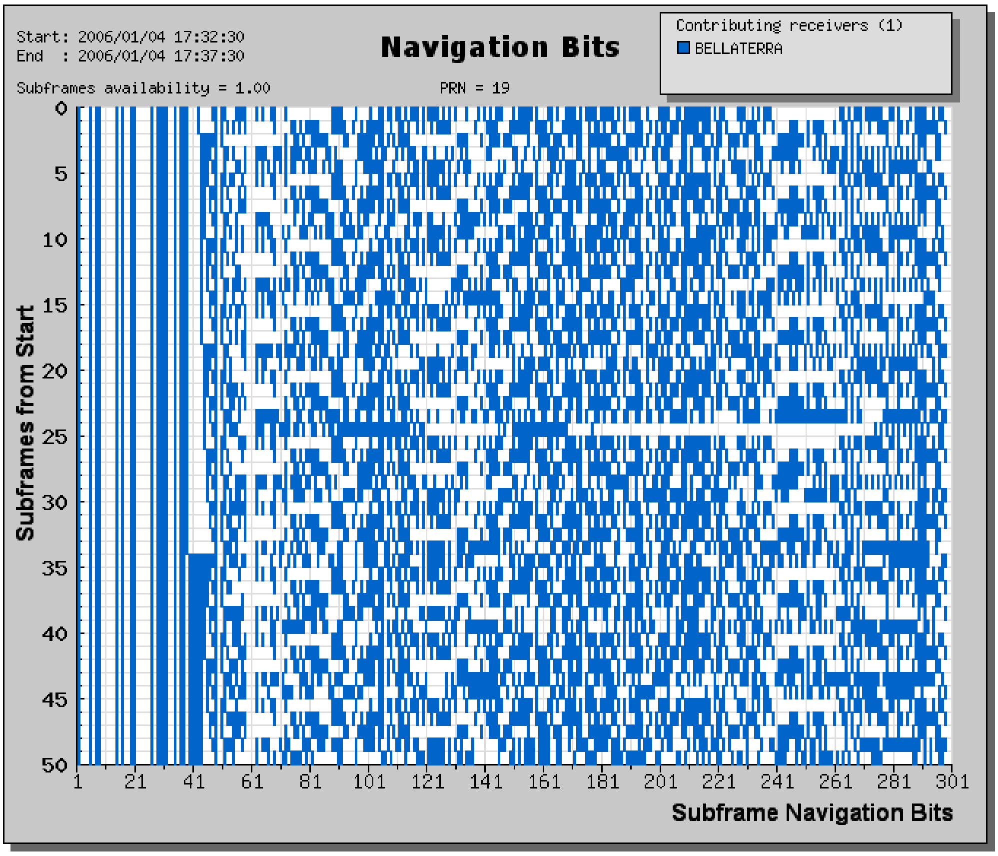

Figure 3.9: Navigation Bits Graphic from website.

4 Algorithms

In this chapter it is described how to implement assisted navigation bit removal for raw sampling radio occultation baseband (I and Q) data.

The following basic scheme is used:

- Retrieve raw sampling in-phase and quadrature data I, Q
- Retrieve navigation bit data B from an external source
- Retrieve auxiliary information to determine signal travel time
- Synchronize raw sampling data epochs to navigation bit data epochs
- Add 180° phase shift to the raw sampling data whenever the navigation bit changes state.

4.1 Data Synchronization

The core of the navigation bit removal algorithm is the synchronization of the data. The raw sampling data and the navigation bit data are referred to epochs measured in different time frames. The GRAS raw sampling data timestamps, $t_k^{IMT}; k \in K$, where K is the discrete set of indices of the raw sampling data samples, are based on Instrument Measurement Time (IMT). The ARS data time stamps, $t_l^{GPST}; l \in L$, where L is the discrete set of indices for the navigation bit data, are referred to GPS Time (GPST). Furthermore the GRAS data time stamps refer to time of signal reception while the ARS data time stamps refer to the time of signal transmission. These facts might of course differ for other sources of radio occultation (RO) data and/or other implementations of external navigation bit providers.

In retreiving the timestamps of the navigation bits in the ARS system from the time specified in the Hand-Over Word of the navigation message, it is important to note that it refers to the beginning of the following subframe, see section 3.3, each subframe lasting 6 s. The navigation bit data are stored in these blocks of 6 s of data, corresponding to 300 navigation bits. The format is 75 hexadecimal numbers with each bit representing a navigation bit. The order is straight forward, i.e. the most significant bit in the first hexadecimal number is the first navigation bit in the 6 s block, the second most significant bit in the first hexadecimal number is the next, and so on, until the least significant bit in the last hexadecimal number.

According to [RD4] (section 4.6) the various time frames used in the GRAS instrument can be related at each OnePPS instant. Hence the differences of the time bases can be estimated once every second. Specifically, IMT can be synchronized to GPST, as estimated in the navigation solution by GRAS, and the timestamp t_k^{IMT} can be converted to GPST using interpolation between the OnePPS instances. Let t_k^{GPST} denominate the raw sampling data timestamps referred to GPST.

Next, the signal travel time τ, which is now the remaining difference of the time systems of the raw sampling data and the navigation bit data, shall be estimated. Assume that the values of t_k^{GPST} and t_l^{GPST} are exact. Then, since the correlators in GRAS integrate over 1 ms the signal travel time must be known with a precision better than 0.5 ms in order to determine unambiguously the bit value for a given correlator output.

The signal travel time can be estimated within the required resolution in one of two ways:

- Use the code phase estimation.
- Calculate the straight line distance between the transmitting satellite and the receiver.

The code phase is the actual pseudorange, i.e. the signal travel time when given in units of time. It is estimated from the *navigation data*, the *GPS data* and the *code phase data* which are all output of the GRAS instrument, see [RD5]. The straight line distance is not accounting for the atmospheric, ionospheric and instrumental excess path length. It can be estimated using the *navigation data* and *GPS data* which are outputs of the GRAS instrument, see [RD5]. Both methods are accurate enough, since the atmospheric delay (which is the difference between the two) is at the most $\sim$ 200 m, corresponding to 0.67 μs, during an occultation. This is much less than the 0.5 ms requirement stated above. Hence, the choice of which to use, is to be based on other implementation issues such as e.g. program execution speed.

The signal travel time must be calculated at every raw sampling data epoch, and the times of transmission of the raw sampling data samples are then

$$t_k^{GPST,trans} = t_k^{GPST} - \tau_k \tag{4.1}$$

In the generation of the GPS signals the C/A code periods are synchronized with the databits, i.e the data bit transitions occur at instances where a new repetition of the C/A code begins. This feature is true for the received signal, too. Since the integration epochs of the raw sampling GRAS data are aligned with code periods of the received signal, then the every 20'th measurement epoch referred to transmission times, $t_k^{GPST,trans}$ must coincide, to the degree of accuracy in time conversions and signal travel time estimation, with the potential occurrence of a data bit transition.

Note that it is important to keep track of whether the time stamps refer to the beginning, end or some other instance of the integration periods. For example, the time stamps of the GRAS raw instrument carrier phase data refer to the middle of the integration period, [RD4] (section 5.3.1), while the timestamps in the ARS navigation bit data refer to the beginning of the bits. We could translate the GRAS time stamps to the beginning of the integration periods by subtracting 0.5 ms in equation 4.1.

If the above mentioned timing accuracy requirement of 0.5 ms (half a C/A code period) is meet, then the selection of which ARS navigation bit $B(t_l)$ to use for correcting the GRAS raw sampling data sample at t_k is done by

$$\mathrm{round}(t_l \cdot 1000) \leq \mathrm{round}(t_k^{GPST,trans} \cdot 1000) < \mathrm{round}(t_l \cdot 1000) \tag{4.2}$$

anticipating that $t_k^{GPST,trans}$ and t_l is in seconds. Occultation events overlapping week transitions must be treated slightly differently, e.g. one could add the number of seconds per week, 604800, to the times stamps after the week transition before applying equation 4.2. Equation 4.2 also holds if the GRAS time stamps has been translated by 0.5 ms to define the time of the beginning of integration periods.

4.2 Data Bit Removal

Once the raw sampling in-phase and quadrature data, $I(t_k)$ and $Q(t_k)$, and the navigation bit data, $B(t_l)$ has been synchronized, the navigation bit is removed from the complex signal $z(t_k) = I(t_k) + Q(t_k)$ according to

$$\tilde{z}(t_k^{GPST,trans}) = z(t_k^{GPST,trans}) \cdot \exp\left(i \cdot \pi \cdot B(t_l)\right) \tag{4.3}$$

where $t_k^{GPST,trans}$ and t_l is selected as in equation 4.2 and B is represented as zeros and ones.

5 Experiment

During 27-28 February 2006 an experiment for collecting open loop ground based occultation data concurrently with the operation of the ARS system were conducted. In sections 5.1 and 5.2 the experiment setup and the implementation of assisted navigation bit removal is described, respectively, while in section 5.4 the results are presented.

5.1 Experiment Setup

Figure 5.1 shows a topographic map of the surroundings of the observation site in the village Llafranc north of Barcelona, Spain. Relatively steep cliffs are rising from the sea, and at the top of the cliffs a church tower is situated, see picture in figure 5.2. Placing the equipment on top of the tower, provide an unobstructed antenna field of view over the Mediterranean sea in an azimuth range from approximately 50° to 178°. The height of the antenna above the sea surface is approximately 120 m. The position of the site is $41°53'53.6''$ N and $3°12'14.17''$ E.

Figure 5.3 and 5.4 shows the antenna setup on the tower roof. The ARS receiver was connected to an Ashtech RHCP choke-ring antenna model ASH701933 which was tilted somewhat from zenith towards the direction in which the GOLD-RTR receiver was tracking occulting GPS signals. The raw sampling measurements where made with either the same choke-ring antenna or one of the RHCP and LHCP patch antennas mounted on the same metal plate and directed with their main gain lobes towards the horizon.

The investigations here are limited due to the fact that the GOLD-RTR and the ARS receivers are located the same place, and hence experience more or less the same SNR conditions. Hence when the ARS receiver looses track of the signal we cannot continue the investigation to lower elevation angles. Optimally, and the way the ARS concept are intended to be used, the ARS receiver shall be situated in a place with a better view to the GPS satellite in terms of SNR, than the receiver measuring the occulting signal. The experiment nevertheless allows to demonstrate the ARS concept: Navigation bit data retrieval, distribution and application to assist RO raw sampling data analysis.

Figure 5.1: Topographic map of the Llafranc village and the observation site area. There is an unobstructed view over the ocean from 50° to 178° azimuth. Courtesy of ICC.

Figure 5.2: View of the tower from which the measurements where conducted.

Figure 5.3: The antenna setup. In the front is the near zenith directed choke-ring antenna which feeds the ARS receiver. Behind is seen the metal plate on which the RHCP and LHCP antennas used for the raw sampling occultation measurements.

Figure 5.4: The antenna setup. In the back is the near zenith directed choke-ring antenna which feeds the ARS receiver. In front is seen the RHCP and LHCP antennas used for the raw sampling occultation measurements.

Figure 5.5: ARS and GOLD-RTR receivers and auxiliary equipment. Explained in the text.

For studies of the improvement in RO data retrievals with and without assisted navigation bit removal, the ARS and the RO receivers must be located far apart, preferably several thousand kilometers. Alternatively, artificial noise could be added to the raw sampling data, before the navigation bit retrieval, in order to simulate worse SNR conditions.

The ARS and GOLD-RTR receivers and computers for receiver operation and data collection were placed on the top floor in tower, see figure 5.5. In the background is seen the ARS receiver and the car-battery that powers it. To the left is the labtop which retrieves and stores the ARS data. The gray rack to the right contains the GOLD-RTR receiver. The PC on top of the GOLD-RTR rack configures and retrieves data from the GOLD-RTR and stores the data on the external disc in the middle on the floor. Adetailed description of the instrument can be found in Nogués et al[1].

[1] *A GPS-reflections receiver that computes Doppler-Delay maps in real time*, O. Nogués et al., submitted to IEEE Transactions on Geoscience and Remote Sensing, 2006.

5.2 Implementation

Automatic assisted navigation bit removal is implemented as described in this section. In addition to the GOLD-RTR raw sampling data files, the navigation bits derived from an external source and an estimate of the signal travel time is needed. In the Llafranc Experiment setup the navigation data bits is extracted in the ARS receiver and stored with the corresponding transmission time (GPS time frame). The estimated signal travel times (or pseudoranges) is derived by the Novatel GPS receiver card in the GOLD-RTR and stored using the RINEX format, i.e. the pseudorange is given as function of the signal reception time (GPS time frame). The GOLD-RTR data is the complex correlator values as function of signal reception time (in the GPS time frame). The time stamp of these refers to the beginning of the integration period (not to the middle as the GRAS data discussed in section 4.1).

The pseudorange information in the RINEX files is stored once every second, hence the pseudorange values is interpolated to the 1 ms spaced sample times. Linear interpolation is used. The GOLD-RTR data is matched with the correct navigation bit value in the ARS data as described in equation 4.2. Since the correlators in the GOLD-RTR integrate over 1 ms the pseudorange must be known with a precision better than 0.5 ms in order to determine unambiguously the bit value for a given correlator output, except for the correlations spanning over a bit shift.

The selection criteria in equation 4.2 is used. In this case, when the integrations may overlap bit transitions, it matters whether the raw sampling data time stamps refer to the beginning or the middle of the integration periods. Referring to the beginning ensures that the navigation bit value which corresponds to the largest fraction of the integration time is used. Only in case the bit transition occurs very close to the middle of the integration period this might be altered due clock noise and inaccurate signal travel time estimation. In the next section, data samples overlapping bit transitions is discussed further, while in section 5.4 the results of the experiment is given.

5.3 Correlations Containing Bit Transitions

As opposed to the GRAS instrument the GOLD-RTR receiver used in this experiment, do not align the 1 ms integration periods with the 1 ms C/A code periods in the received signal. It might happen, but it is not generally true. Therefore, the possible navigation bit transitions generally occur at some instance within every 20'th integration period. This could also be the case for other GPS receivers for RO measurements and is therefore given attention in this section.

How is the complex correlator value altered by a navigation bit shift during the integration period? Denominate the duration of the first part of the integration period, before the bit transition, τ_1, and the duration of the second part τ_2 and the corresponding partial complex correlator integration values z_1 and z_2 respectively. The amplitude of the total correlator integration is diminished since the correlation is of opposite sign in the two parts of the integration. Assuming a stationary signal with random zero mean noise (over the integration time, i.e. 1 ms) the correlator amplitude goes towards zero as the ratio $\frac{\tau_1}{\tau_2}$ goes towards one.

The phase of the correlator output depends strongly on the ratio $\frac{\tau_1}{\tau_2}$. Again, assuming a stationary signal with random zero mean noise, for $\tau_1 \gg \tau_2$ the phase will be close to its value before the navigation bit shift. Similarly, for $\tau_1 \ll \tau_2$ it will be close to its value after the bit shift. If $\frac{\tau_1}{\tau_2} \approx 1$ then $z_1 \approx -z_2$ and the amplitude of the total integration is very small. In this case the noise will have a major impact on the phase which will be unpredictable then.

The raw sampling data based on correlations spanning over navigation bit transitions can be treated in different ways. For example:

- Either they could be omitted from the data analysis,
- or, it could be estimated how large a fraction of the integration time the integration runs over which data bit and the bit corresponding to the largest fraction selected,
- or, either the first or the second data bit could be used mechanically.

Which of the three options shall be used, depend on the intended use for the data and the following data processing.

5.4 Results

In figure 5.6 raw sampling data recorded when satellite PRN 5 was setting behind the horizon in local morning time, is shown. The satellite elevation angle these 5 s was approximately 1°. The top plot shows the raw sampling data phase before navigation bit removal while the middle plot show the raw sampling data phase after the automatic *assisted navigation bit removal* and the the bottom plot shows the navigation bit state, represented as zeros and ones. There is a data gap at approximately 14.3 s to 14.5 s due to a problem

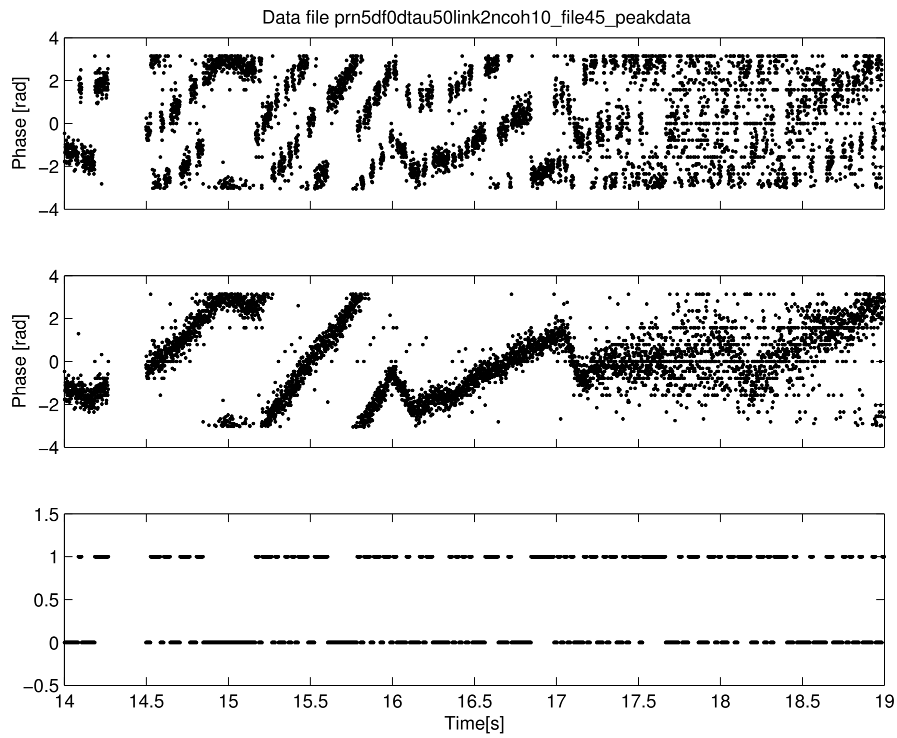

Figure 5.6: Raw sampling phase data before navigation bit removal (top plot), raw sampling phase data after navigation bit removal (middle plot) and navigation bit state (bottom plot).

storing the large amounts of data output from the GOLD-RTR continuously on the external disc. The zero of the time axis is arbitrary.

Figures 5.7 and 5.8 show zoom in's on the time intervals [14.5; 15.5] and [17.5; 18.5], respectively. These intervals correspond to two different regimes with higher and lower SNR. Before navigation bit removal it is hard to distinguish the navigation bit transitions from the random phase fluctuations in the low SNR data. It is especially in this regime that the *assisted raw sampling* concept will become valuable for raw sampling RO data retrievals.

In figure 5.9 a comparison of the ARS navigation bits and navigation bits derived in postprocessing of the GOLD-RTR data is shown in the bottom plot. The value is 1 if they agree and is 0 if they are different. Due to the extensive quality checks that the ARS data undergoes, see 3.3, the disagreements between the two set of navigation bits is believed to be due to the algorithm used for detecting phase transitions in the GOLD-RTR data. In the top plot is shown the raw sampling data amplitude while in the middle plot the raw sampling phase data before navigation bit removal is shown. We see, as expected that the disagreement of navigation bits is associated with enhanced phase noise which again is corresponds to low signal strength.

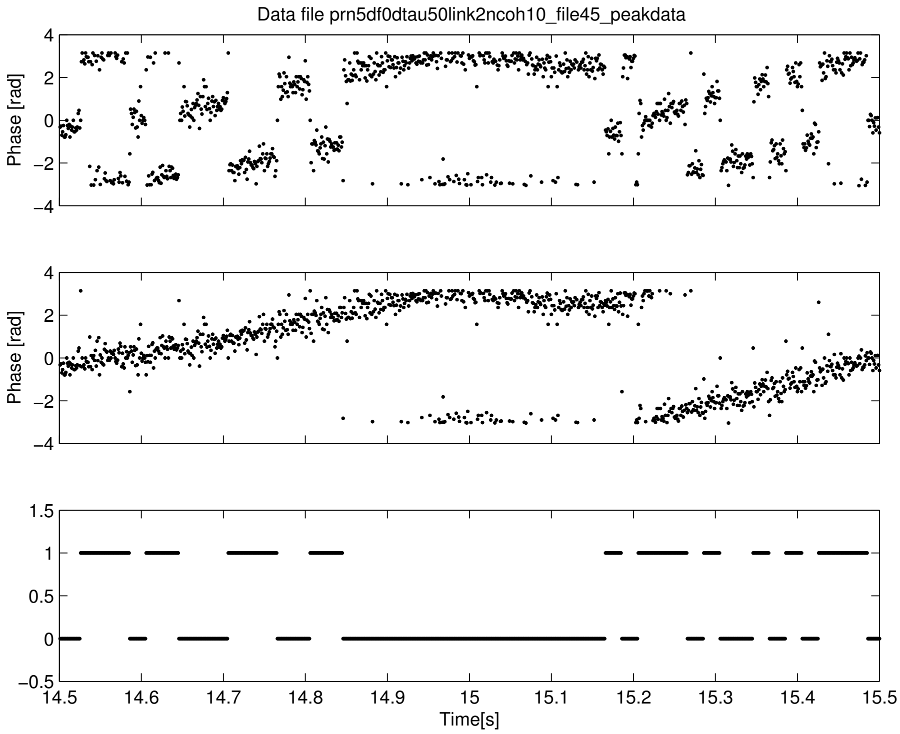

Figure 5.7: Raw sampling phase data before navigation bit removal (top plot), raw sampling phase data after navigation bit removal (middle plot) and navigation bit state (bottom plot).

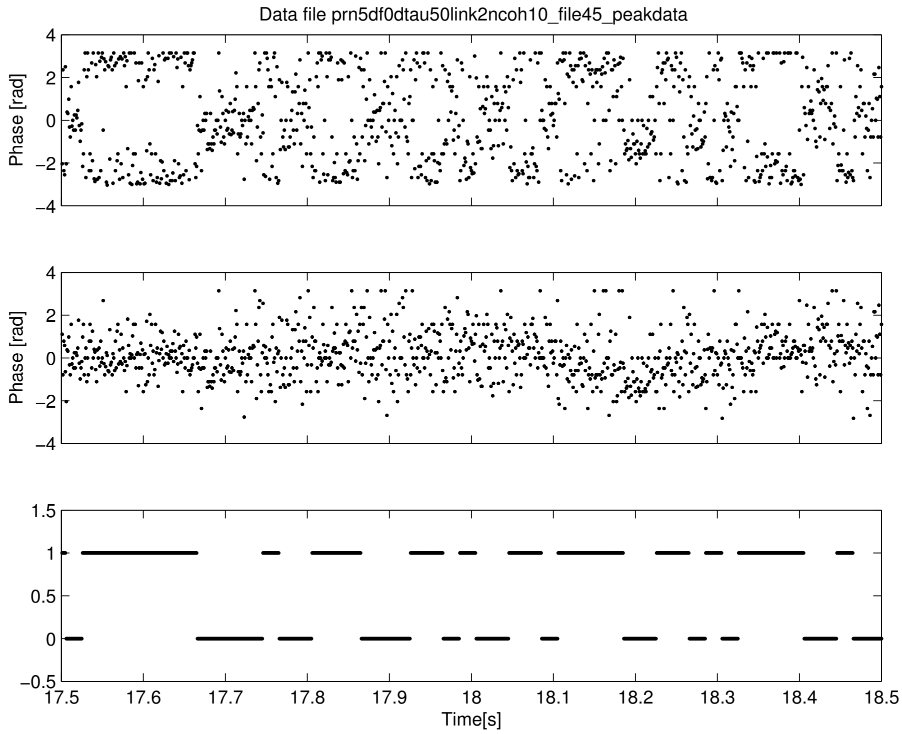

Figure 5.8: Raw sampling phase data before navigation bit removal (top plot), raw sampling phase data after navigation bit removal (middle plot) and navigation bit state (bottom plot).

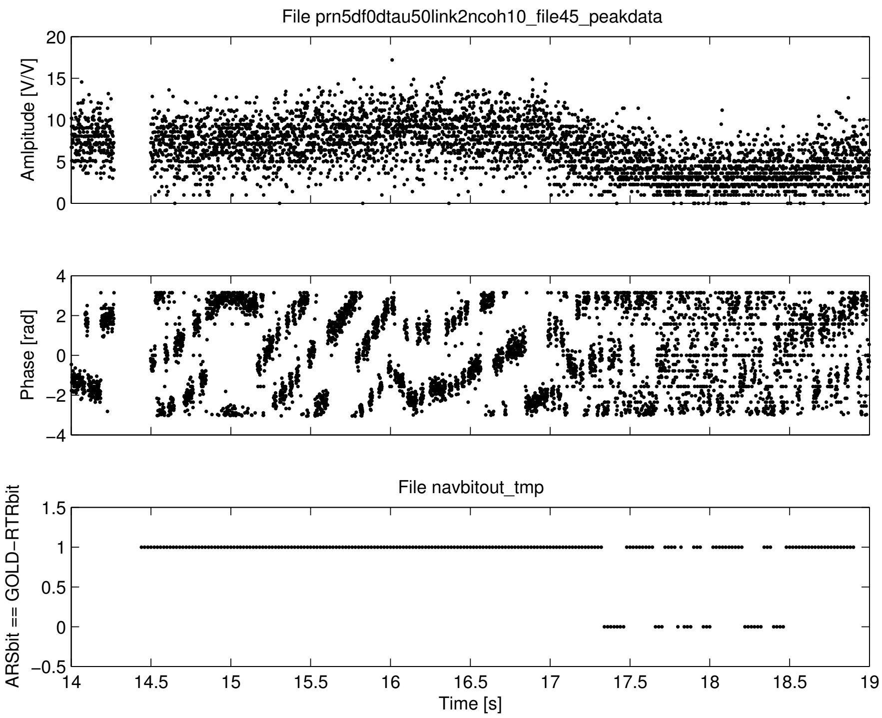

Figure 5.9: Raw sampling data amplitude (top plot), raw sampling phase data before navigation bit removal (middle plot) and comparison of ARS navigation bit state and navigation bit derived in postprocessing from the GOLD-RTR raw data, 1 if equal, 0 if different. (bottom plot).

6 Conclusion

We have developed a system able to detect the navigation bits inserted in the GPS signals. The system stores the data in a computer for further processing.

Methodologies to remove navigation bits in the baseband image of the incoming GPS signals is proposed. The methodology should be easily applicable to the GRAS instrument raw sampling data gathered at 1 kHz.

Finally, we have applied successfully the instrumental setup and algorithms to the data gathered in an occultation experiment performed from the top of a mountain.

7 Acronyms

ARS	Assisted Raw Sampling
ASCII	American Standard Code for Information Interchange
DMI	Danish Meteorological Institute
EUMETSAT	European Organisation for the Exploitation of Meteorological Satellites
FRMA	Framed Raw Navigation Data
GOLD	GPS Open Loop Differential
GOLW	GRAS SAF Open Loop Workshop
GPS	Global Positioning System
GPST	Global Positioning System Time
GRAS	Global Navigation Satellite System Receiver for Atmospheric Sounding
GZIP	GNU Zip
HOW	Hand-Over Word
ICC	Institut Cartogrfic de Catalunya
IEEC	Institut d'Estudis Espacials de Catalunya
IMT	Instrument Measurement Time
LHCP	Left Hand Circular Polarized
NB	Navigation Bits
OEM	Original Equipment Manufacturer
OL	Open Loop
OM	Original Manufacturer
PC	Personal Computer
PRN	Pseudo-Random Noise
RHCP	Right Hand Circular Polarized
RINEX	reciever INdependent EXchage format
RO	Radio Occulation
RVSA	Receiver Status
SAF	Satellite Application Facility
SNR	Signal to Noise Ratio
SOW	Second Of Week
SSH	Secure Socket Shell
UPS	Uninterruptible Power Supply
URL	Uniform Resource Locator (world wide web address)

XOR Exclusive OR

8 Reference documents

Ref.	Title	Number	Issue	Date
RD1	Assisted Raw Sampling Mode	SAF/GRAS/IEEC/ALG/OL/002	v. 0.2	2005/07/25
RD2	Millennium GPSCard Software Versions 4.503 and 4.52, Command Descriptions Manual	OM-20000053	Rev 2	2001/01/16
RD3	Navstar GPS Space Segment/Navigation User Interfaces	IS-GPS-200D	Rev D	2004/12/07
RD4	METOP-GRAS Measurement Data Interpretation and Description	P-GRM-SPC-0036-SE	6	2004/12/01
RD5	METOP-GRAS Measurement Data Interface Control Document	P-GRM-ICD-0008-SE	8	2003/14/10