Ephemeris Push for GSC3-based Products

A Description of Ephemeris Push Software Feature on Vincotech's GPS modules based on SiRFstarIII – GSC3 A1080, A1084, A1088, A1035-D, A1035-H

Application Note

Version 1.0
Firmware Revision 3.5.0

Revision History

Rev.	Date	Description
0.1	12-09-08	Initial version
1.0	03-24-09	Small design changes, renamed (GSC3-generic)
	mm-dd-yy	

Disclaimer

THIS DOCUMENT CONTAINS PROPRIETARY INFORMATION OF VINCOTECH GMBH. IT MAY NOT BE COPIED OR TRANSMITTED BY ANY MEANS, PASSED TO OTHERS, OR STORED IN ANY RETRIEVAL SYSTEM OR MEDIA, WITHOUT PRIOR CONSENT OF VINCOTECH OR ITS AUTHORIZED AGENTS.

THE INFORMATION IN THIS DOCUMENT IS, TO THE BEST OF OUR KNOWLEDGE, ENTIRELY CORRECT. HOWEVER, VINCOTECH CAN NEITHER ACCEPT LIABILITY FOR ANY INACCURACIES, OR THE CONSEQUENCES THEREOF, NOR FOR ANY LIABILITY ARISING FROM THE USE OR APPLICATION OF ANY CIRCUIT, PRODUCT, OR EXAMPLE SHOWN IN THE DOCUMENT.

THE PRODUCT (HARD- AND SOFTWARE) DESCRIBED IN THIS DOCUMENTATION IS NOT AUTHORIZED FOR USE IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF VINCOTECH.

THIS DOCUMENT MAY PROVIDE LINKS TO OTHER WORLD WIDE WEB SITES OR RESOURCES. BECAUSE VINCOTECH HAS NO CONTROL OVER SUCH SITES AND RESOURCES, VINCOTECH SHALL NOT BE RESPONSIBLE FOR THE AVAILABILITY OF SUCH EXTERNAL SITES OR RESOURCES, AND DOES NOT ENDORSE AND IS NOT RESPONSIBLE OR LIABLE FOR ANY CONTENT, ADVERTISING, PRODUCTS, OR OTHER MATERIALS ON OR AVAILABLE FROM SUCH SITES OR RESOURCES. VINCOTECH SHALL NOT BE RESPONSIBLE OR LIABLE, DIRECTLY OR INDIRECTLY, FOR ANY DAMAGE OR LOSS CAUSED OR ALLEGED TO BE CAUSED BY OR IN CONNECTION WITH USE OF OR RELIANCE ON ANY SUCH CONTENT, GOODS OR SERVICES AVAILABLE ON OR THROUGH ANY SUCH SITE OR RESOURCE.

VINCOTECH RESERVES THE RIGHT TO CHANGE, MODIFY, OR IMPROVE THIS DOCUMENT OR THE PRODUCT DESCRIBED HEREIN, AS SEEN FIT BY VINCOTECH WITHOUT FURTHER NOTICE.

Table of Contents

1 Introduction 5
2 Technical Background and Motivation 5
2.1 Push-to-Fix (PTF) Mode 6
2.2 Ephemeris Push 6
2.2.1 Local Systems 7
2.2.2 Widespread Systems 7
3 Tips and Tricks 8
3.1 Hexadecimal Notation 8
3.2 Calculating the SiRF Binary Checksum 8
3.3 HTerm 8
3.4 Using SiRFDemo to Send SiRF Binary 9
4 Ephemeris Push Using SiRFDemo 10
4.1 Initial Preparation 10
4.2 Ephemeris Poll from Reference Receiver 11
4.3 Erase Ephemeris Data Stored in GPS Receiver 11
4.3.1 Receiver Switched off Completely (Cold Start) 12
4.3.2 Backup Supply Present (Warm Start) 12
4.4 Ephemeris Push to Moving GPS Receiver 13
5 Ephemeris Push Using SiRF Binary 14
5.1 Switch to SiRF Binary Mode (Set Serial Port) 14
5.2 Switch off All Periodic Output Messages 14
5.3 Ephemeris Poll from Reference GPS Receiver 15
5.3.1 Poll Ephemeris Data of a Single Satellite 15
5.3.2 Poll All Available Ephemeris Data at Once 16
5.4 Prepare the Input Message 16
5.5 Ephemeris Push to Moving GPS Receiver 17
5.6 Switch Back to NMEA Mode 18
5.7 Summary 18
6 Related Information 20
6.1 Contact 20
6.2 Related Documents 20
6.3 Related Tools 20
7 List of Figures 21
8 List of Tables 21

1 Introduction

The intention of this application note is to give an overview over the new ephemeris push feature implemented in the **firmware revision 3.5.0** that is available on all of Vincotech's SiRFstarIII based GPS receiver and antenna modules (currently A1080, A1084, A1088; A1035-D, A1035-H).

The next paragraph presents a short technical introduction and a comparison to another possible technique of providing valid ephemeris data.

The subsequent paragraphs cover two ways of using the ephemeris push feature. The main focus of the first method is mainly on evaluation and demonstration purpose, the second one covers the implementation in an embedded system.

2 Technical Background and Motivation

One crucial figure – beside the accuracy of the position and the sensitivity – is the time that elapses between the moment the GPS receiver is switched on and the first valid position output, also referred to as *time to first fix* (TTFF).

A short TTFF is not only a question of convenience - like in consumer navigation devices - but also an integral requirement of specific applications:

- For security applications it is very important to know quite immediately where the reported incident is happening.
- Systems designed to work from battery can benefit from a short TTFF, because the GPS section needs less power to deliver a valid fix and therefore extends battery life.

One key point to a short TTFF is the fast presence of valid Ephemeris data in the GPS receiver. If this information is not available the data need to be downloaded from each individual satellite, which takes approximately 30 seconds under good conditions. Fortunately the download is done in parallel for all detected and decoded satellites. Things become worse under bad conditions, as download time is increasing then.

Once the Ephemeris data is known for a specific satellite, this satellite can be used to calculate a position fix, even when its signal strength is too low to download and decode ephemeris data. Or in other words: a receiver with valid Ephemeris information might provide a fix in situations where a receiver that starts from scratch is not able to calculate a fix at all.

Unfortunately the Ephemeris data is valid for a period of a few hours (max. four) only, and updated information is provided approximately every two hours. Thus any

receiver switched off for a time longer than this period will need to start with a new Ephemeris download.

In order to overcome this handicap, all SiRFstarIII based receivers offer two methods to keep ephemeris data up-to-date without full operation of the receiver throughout the whole time. Both are briefly introduced in the following sections, together with their respective benefits and drawbacks.

2.1 Push-to-Fix (PTF) Mode

The basic idea of PTF is to keep the ephemeris data up-to-date without operating the receiver continuously. Most of the time the receiver is in hibernate mode (current consumption approximately 20 µA) and will wake up from time to time to update ephemeris data and to calibrate the clock. To request a position fix during the sleep period the ON/OFF pin needs to be toggled. This mode is covered more detailed in the *SiRF Binary Reference Manual* and in the Vincotech application note *GPS Firmware GSC3*.

+ Self-contained, no additional infrastructure and/or communication channel is necessary.
+ RTC is running and up-to-date.
+ Last known position present.

– Permanent current draw of GPS receiver.
– Receiver is only aware of the currently tracked satellites. This number might be smaller than the number of potentially visible satellite (e.g. covered parking lot or even underground car park).
– Will not help in bad conditions with very low GPS signals (e.g. personal tracker in the inner pocket of a wet coat).

2.2 Ephemeris Push

The idea of Ephemeris push is to provide up-to-date Ephemeris data to a GPS receiver that has just been switched on without having valid Ephemeris data (e.g. powered off completely or Ephemeris information outdated). This relieves the burden of collecting Ephemeris data from the GPS receiver, by transmitting the information via a high speed channel like GPRS or WLAN and UART.

+ The complete system including GPS receiver can be switched off completely.
+ It is possible to provide Ephemeris data for more satellites than the currently tracked ones (e.g. the whole set of information is available before the car leaves the underground parking lot).
+ Enables position fixes in weak signal situations where a direct download of Ephemeris data from satellites is not possible.

- Additional communication channel needed. This will add extra cost and system complexity.
- Ephemeris data has to be collected externally

As already stated above the Ephemeris data has to be collected externally and converted into the necessary format prior to downloading it to the GPS receiver. The actual source of the data is not relevant, as long as the information is up-to-date.

The firmware release 3.5.0 enables both basic functionalities necessary to implement a working system that makes use of transmitted Ephemeris data. The receiver may be used as the moving system (Ephemeris Push) and as a reference station (Ephemeris poll).

Depending on the application several ways are possible and reasonable.

2.2.1 Local Systems

With a local system (e.g. logistics on large company premises or local taxi dispatch) usually one reference station collecting Ephemeris data is sufficient. The antenna of this reference station is best placed in an elevated position with good view to the sky, in order to ensure, that all relevant Ephemeris information valid for this area is collected.

2.2.2 Widespread Systems

With a wider spread of the system (e.g. worldwide truck fleet) one single reference station is not sufficient, as each station can only collect data of satellites that are currently visible at its location.

To overcome this it is necessary to place several reference stations throughout the intended area of the system. Now it is possible to combine the collected data to a complete set of ephemeris data, or – in case the coarse position of the receiver is known (e.g. WLAN base station, GSM cell, last known position) to transmit the Ephemeris data of the relevant reference station.

3 Tips and Tricks

3.1 Hexadecimal Notation

Throughout this application note the hexadecimal notation is used to describe the SiRF binary commands. Here each byte of data that is sent over the serial port is represented by a pair of ASCII characters.

3.2 Calculating the SiRF Binary Checksum

Vincotech offers a simple command line tool (see ”6.3 Related Tools”) to calculate the checksum of a SiRF binary command. No installation is required. The executable file can be invoked directly from the command line prompt. It takes the payload as input and provides the complete SiRF binary command as output.

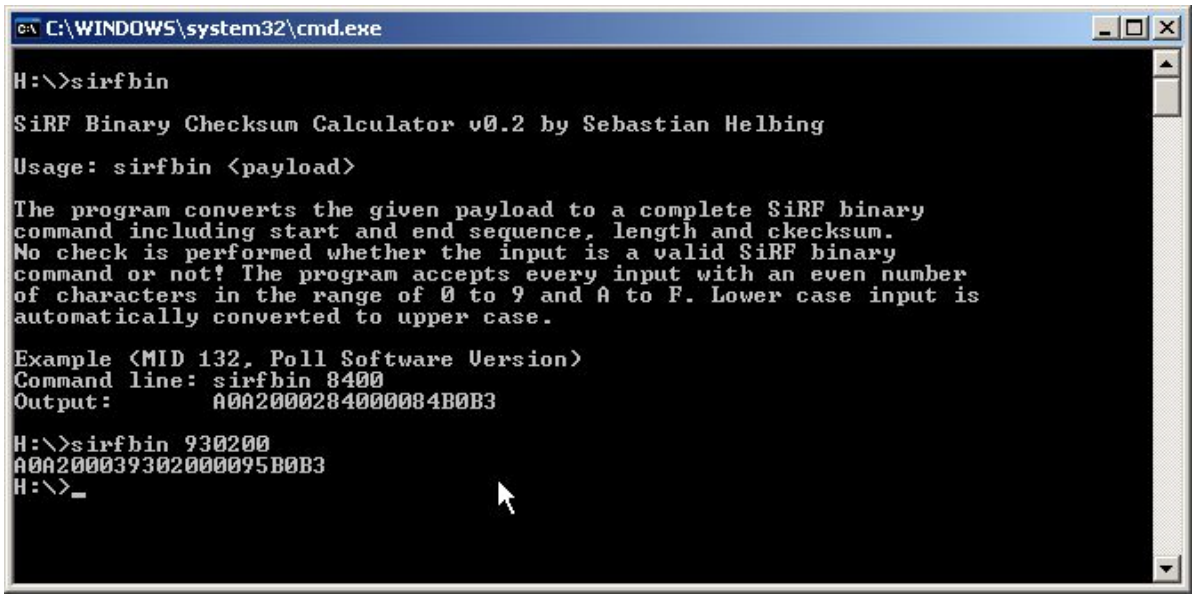

Figure 1: sirfbin command window

3.3 HTerm

HTerm is a freeware tool. Fortunately its use is not restricted to non-commercial applications. It can be obtained directly from the website of the originator (see ”6.3 Related Tools”). Unlike a lot of other terminal software programs it offers the following unique features:

- Separate input and output window
- Binary input and output in different notations (decimal, hexadecimal, binary and ASCII)
- Definition of end-of-line delimiters, e.g. <CR><LF> (NMEA) or B0B3 (SiRF binary)

3.4 Using SiRFDemo to Send SiRF Binary

SiRFDemo can be used to send binary commands to the GPS receiver. It will take the payload to calculate the checksum and embed it between start and end sequence prior to transmitting it.

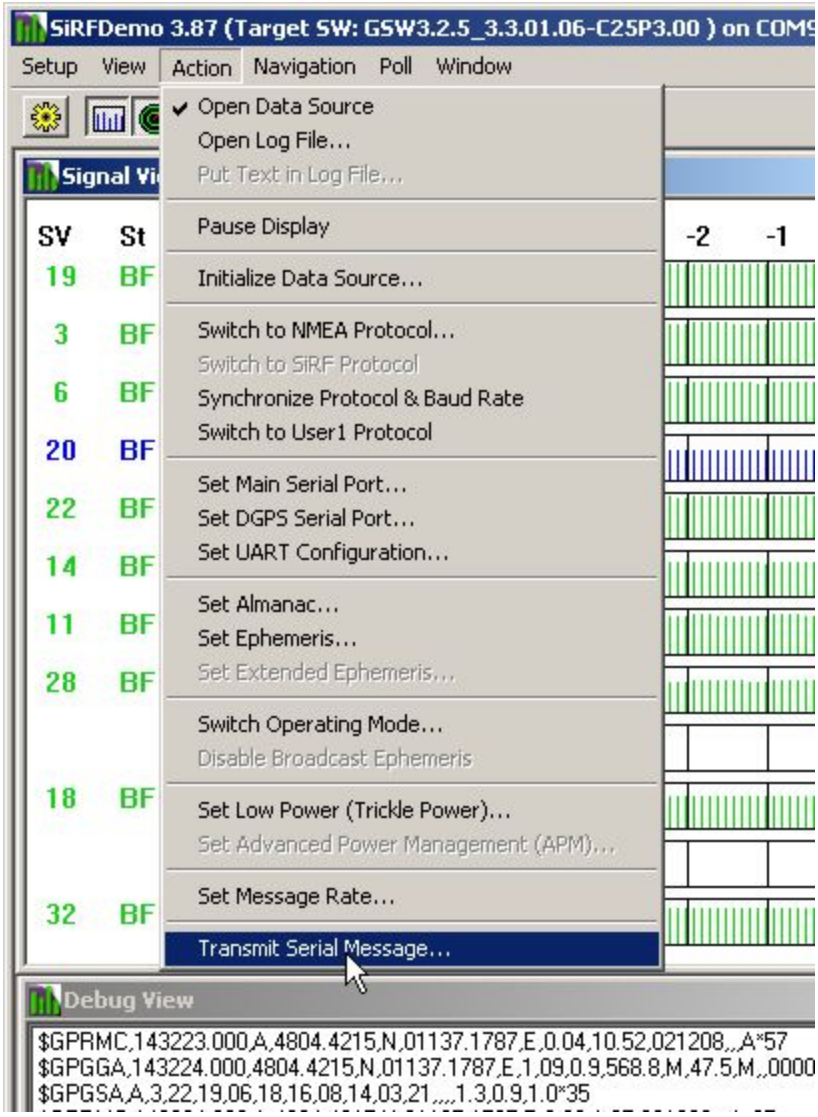

Figure 2: SiRFDemo – Transmit Serial Message

4 Ephemeris Push Using SiRFDemo

This approach is best used for testing, evaluation and demonstration purpose and not designed for the final application. However, it is an easy way to get into contact with the concept of Ephemeris push and poll.

It is assumed, that the reader of this application note is already familiar with the GPS demonstration kit and that all necessary drivers and the SiRFDemo software are installed. For a more detailed explanation of these steps it is recommended to consult the respective documentation mentioned in chapter “6.2 Related Documents”.
The following paragraphs will explain the demonstration procedure step by step. The actual look of the SiRF Demo window might differ from the presented screenshots, because the user may adapt the appearance of this tool to her/his taste and needs.

4.1 Initial Preparation

Here it is assumed that the user is working with a Vincotech evaluation kit. But this chapter also applies to a setup where a module on a custom design can be connected to a PC.

Connect the evaluation kit to the PC and start the SiRFDemo software. As soon as the connection to the GPS receiver is established switch the receiver to SiRF binary mode.

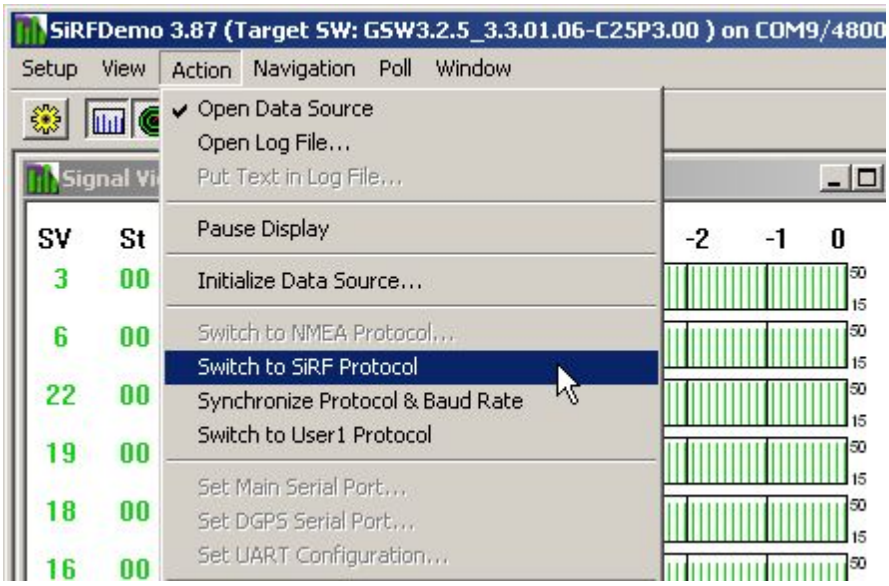

Figure 3: SiRFDemo – Switch to SiRF Protocol

4.2 Ephemeris Poll from Reference Receiver

Connect the GPS antenna and wait until a valid position fix is calculated. Now the latest Ephemeris data are available and can be downloaded and saved to disk.

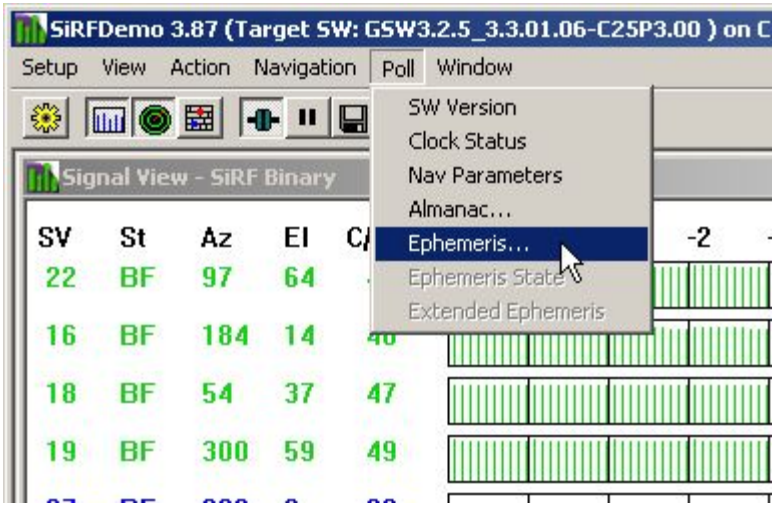

Figure 4: SiRFDemo – Poll Ephemeris data

4.3 Erase Ephemeris Data Stored in GPS Receiver

This intermediate step is necessary due the fact that the reference receiver and the moving GPS is the very same device throughout this evaluation procedure.

Disconnect the GPS antenna from the evaluation kit and select the receiver initialization option within SiRFDemo.

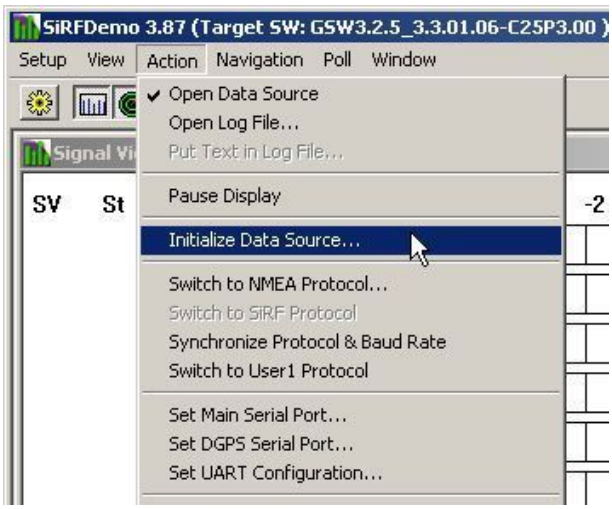

Figure 5: SiRFDemo – Erase data on receiver

The window that appears now will offer several ways to initialize the GPS receiver. Two of them are used in this demonstration as they simulate typical scenarios: *Cold Start* and *Warm Start (No init).* Select the desired option and press SEND.

4.3.1 Receiver Switched off Completely (Cold Start)

The *Cold Start* option simulates a GPS receiver that has been switched off completely with no backup supply provided. Absolutely no information gathered during previous position fixes is available now. This is the worst case scenario a GPS receiver has to cope with.

Figure 6: SiRFDemo – Cold Start Scenario

4.3.2 Backup Supply Present (Warm Start)

The *Warm Start (No Init)* option simulates a GPS receiver that awakes out of the hibernate state. Either the main power supply was disconnected, but battery backup supply attached, or the receiver was set into hibernate mode by toggling the ON/OFF pin. Only the Ephemeris information is erased, but last known position, date and time is still present. This is the typical situation seen with outdated Ephemeris data.

Figure 7: SiRFDemo – Warm Start Scenario

4.4 Ephemeris Push to Moving GPS Receiver

Now it is time to push the Ephemeris back to the receiver. Select the file that was obtained previously in chapter 4.2.

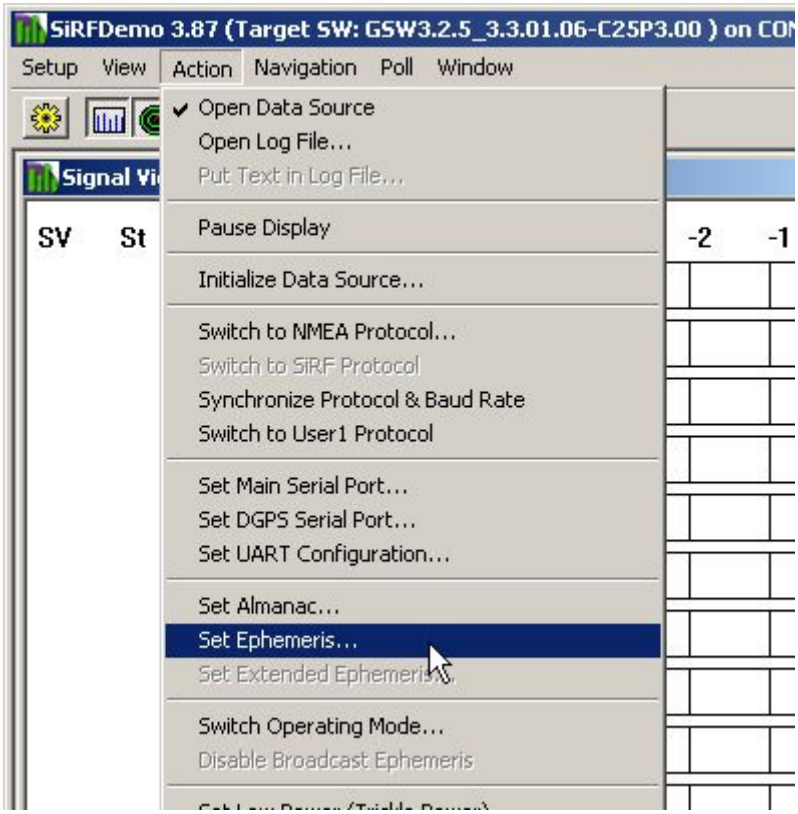

Figure 8: SiRFDemo – Push Ephemeris data

After this procedure the receiver is aware of the latest ephemeris data. Once the GPS antenna is reconnected the position can be calculated quickly because there is no need to wait for the download of the Ephemeris data from each satellite.

5 Ephemeris Push Using SiRF Binary

This approach is the preferred one to be used in the final application. The SiRF binary protocol was specifically designed to allow for a quick and efficient communication between the GPS receiver and the host processor. Unlike NMEA it is not plain ASCII text and therefore difficult to read by human beings.

It is assumed, that the reader of this application note is already familiar with the basic principle of the SiRF binary protocol, its notation and how to send/receive messages to/from the receiver. For a more detailed description it is recommended to consult the respective documentation mentioned in chapter “6.2Related Documents”.

5.1 Switch to SiRF Binary Mode (Set Serial Port)

Commands involved:
Set Serial Port, $PSRF100 (NMEA input message)

Ephemeris push is only supported in the SiRF binary protocol. Thus it is necessary to switch the receiver from NMEA to binary protocol prior to pushing or polling Ephemeris data. This is obtained with the NMEA command:

```
$PSRF100,0,57600,8,1,0*37
```

This is the default SiRF binary message configuration, however it is possible to select different baud rates up to 115200 baud. In case the baud rate has changed, make sure to set the same baud rate in the host processor.

Wait until the receiver sends data in SiRF binary mode before proceeding to the next step. This ensures that the receiver is ready to accept binary commands.

5.2 Switch off All Periodic Output Messages

Commands involved:
Set Message Rate, Message ID 166 (binary input message)
Command Acknowledgement, Message ID 11 (binary output message)

This step is not mandatory. However, it ensures that the host processor has to cope only with the relevant data throughout the next steps.

The following example shows how to disable the *Visible List* (MID 13):

`A0A20003` – Start sequence and payload length
`A6000D0000000000` – Payload, MID 13 (decimal) = 0x0D (hex)
`00B3B0B3` – Message checksum and end sequence

Upon completion of the command the receiver will send a *Command Acknowledge* (MID 11).
The following table shows the default binary output messages and the command to switch them off.

MID dec	MID hex	Function	Command to switch off
2	0x02	Measured Navigation Data	A0A20008A60002000000000000A8B0B3
4	0x04	Measured Tracking Data	A0A20008A60004000000000000AAB0B3
7	0x07	Clock Status	A0A20008A60007000000000000ADB0B3
9	0x09	Throughput	A0A20008A60009000000000000AFB0B3
13	0x0D	Visible List	A0A20008A6000D000000000000B3B0B3
27	0x1B	DGPS Status	A0A20008A6001B000000000000C1B0B3
41	0x29	Geodetic Navigation Data	A0A20008A60029000000000000CFB0B3

Table 1: Switching off periodic output message

5.3 Ephemeris Poll from Reference GPS Receiver

Commands involved:
Poll Ephemeris, Message ID 147 (binary input message)
Ephemeris Data (response to Poll), Message ID 15 (binary output message)
Command Acknowledgement, Message ID 11 (binary output message)

The output of the Ephemeris data is requested with the Poll Ephemeris (MID 147) command. This command allows the request of all known ephemeris data at once or per each satellite individually. After the transmission of all requested data via the Ephemeris Data (MID 15) the Command Acknowledgement (MID 11) is sent.

5.3.1 Poll Ephemeris Data of a Single Satellite

The Ephemeris data of the satellite with PRN02 is requested with the command:

`A0A20003` – Start sequence and payload length
`930200` – Payload
`0095B0B3` – Message checksum and end sequence

The answer (MID 15) to this request will look like:

`A0A2005C` – Start sequence and payload length
`0F02000200370004790000000000000000000000000000DB05313700FFEB`
`1638A8000200370008050030373 5B67A49D1004204854E5E0F03A10CA1`
`4131370000020037000CFFCE6F14E0D2FFCE2665BFCF1D186D299D00FF`
`A65A05F68C` – Payload
`18B6B0B3` – Message checksum and end sequence

The output is terminated with the acknowledgement (MID 11):

A0A20002 – Start sequence and payload length
0B93 – Payload
009EB0B3 – Message checksum and end sequence

5.3.2 Poll All Available Ephemeris Data at Once

The following command (MID 147) is used to request all available ephemeris information at once. Here the specific satellite PRN is substituted with the value “00”:

A0A20003 – Start sequence and payload length
930000 – Payload
0093B0B3 – Message checksum and end sequence

The answer to this request is similar to the first example in 5.3.1. For each available satellite a MID 15 is transmitted. The whole set of data is terminated with a command acknowledgement (MID 11).

5.4 Prepare the Input Message

Commands involved:
Ephemeris Data (response to Poll), Message ID 15 (binary output message)
Set Ephemeris, Message ID 149 (binary input message)

The ephemeris data output message MID 15 holds all information to build the ephemeris input message (MID 149). The following example for PRN02 will illustrate this conversion of the payload.

MID 15 payload output for satellite PRN02:
0F – Message ID 0x0F(hex) = 15(decimal)
02 – Satellite PRN
000200370004790000000000000000000000000DB05313700FFEB1638
A800020037000805003037 35B67A49D1004204854E5E0F03A10CA14131
370000020037000CFFCE6F14E0D2FFCE2665BFCF1D186D299D00FFA65A
05F68C – Ephemeris data (45 bytes)

The payload of the set ephemeris message (MID 149) consists of the hexadecimal message ID 95 and exactly the same 45 bytes data block obtained from the ephemeris output message (MID 15). The satellite PRN is also part of the data block, thus it is not specified explicitly.

MID 149 payload input for satellite PRN02:
95 – Message ID 0x95(hex) = 149(decimal)
000200370004790000000000000000000000000DB05313700FFEB1638
A800020037000805003037 35B67A49D1004204854E5E0F03A10CA14131
370000020037000CFFCE6F14E0D2FFCE2665BFCF1D186D299D00FFA65A
05F68C – Ephemeris data (45 bytes)

5.5 Ephemeris Push to Moving GPS Receiver

Commands involved:
Set Ephemeris, Message ID 149 (binary input message)
Command Acknowledgement, Message ID 11 (binary output message)
Error ID Data, Message ID 10 (binary output message)

This set Ephemeris message (MID 149) payload compiled in 5.4 is completed with the start and end sequences, the payload length and the checksum in order to form a valid SiRF binary input message. The checksum depends on the actual content of the message, thus it has to be calculated for each message individually:

A0A2005B – Start sequence and payload length
95
0002003700047900000000000000000000000000DB05313700FFEB1638
A800020037000805003037 35B67A49D1004204854E5E0F03A10CA14131
37000002003700 0CFFCE6F14E0D2FFCE2665BFCF1D186D299D00FFA65A
05F68C – Payload
193AB0B3 – Checksum and end sequence

The Ephemeris data of each satellite has to be sent individually. A successful transmission is confirmed with the command acknowledge message MID 11. In case this message fails to appear, the input is not accepted, possibly due a miscalculated checksum:

A0A20002 – Start sequence and payload length
0B95 – Payload
009A00B3 – Message checksum and end sequence

In case the additional message error ID data (MID 10) appears before the command acknowledge, the ephemeris data is very likely outdated. The example MID 10 below indicates this for the ephemeris data of satellite PRN02:

A0A2000D – Start sequence and payload length
0A000C0002000000 0200002B49 – Payload
008EB0B3 – Checksum and end sequence

The payload of this message can be fractioned into the following pieces of information (see also SiRF binary reference manual):

0A – Message ID 0x0A(hex) = 10(binary)
000C – Error ID, sub message type: Satellite ephemeris age has exceeded 2 hours
0002 – Count, 2x4 bytes
00000002 – Satellite PRN02
00002B49 – Age of Ephemeris 11081sec = 3h 4min 41sec

In this case it is recommended to use the latest available Ephemeris data from the reference receivers.

5.6 Switch Back to NMEA Mode

Commands involved:
Switch to NMEA Protocol, Message ID 129 (binary input command)

In case NMEA output is required again, the following command will restore the default settings of baud rate and message set:

`A0A20018` – Start sequence and payload length
`810201010001010105010101000100010000000100012C0` – Payload
`0165B0B3` – Message checksum and end sequence

5.7 Summary

The complete procedure can be summarized to the following flow:

1. Switch the reference receiver to SiRF binary mode ($PSRF100)
2. Poll ephemeris data (MID 147, MID 15)
3. Prepare input message (MID 15, MID 149)
   MID 15: A0A2005C0F [PRN] **[45 byte of data]** [checksum_1] B0B3
   MID 149: A0A2005B95 **[45 byte of data]** [checksum_2] B0B3
   The [45 byte of data] block is the same for MID 15 and MID 149, the checksum cannot be copied and has to be recalculated for MID 149!
4. Switch moving GPS receiver to SiRF Binary mode ($PSRF100)
5. Disable all periodic output messages (MID 166)
6. Push ephemeris data to moving GPS receiver (MID149)
7. Switch moving GPS receiver back to NMEA (MID129)

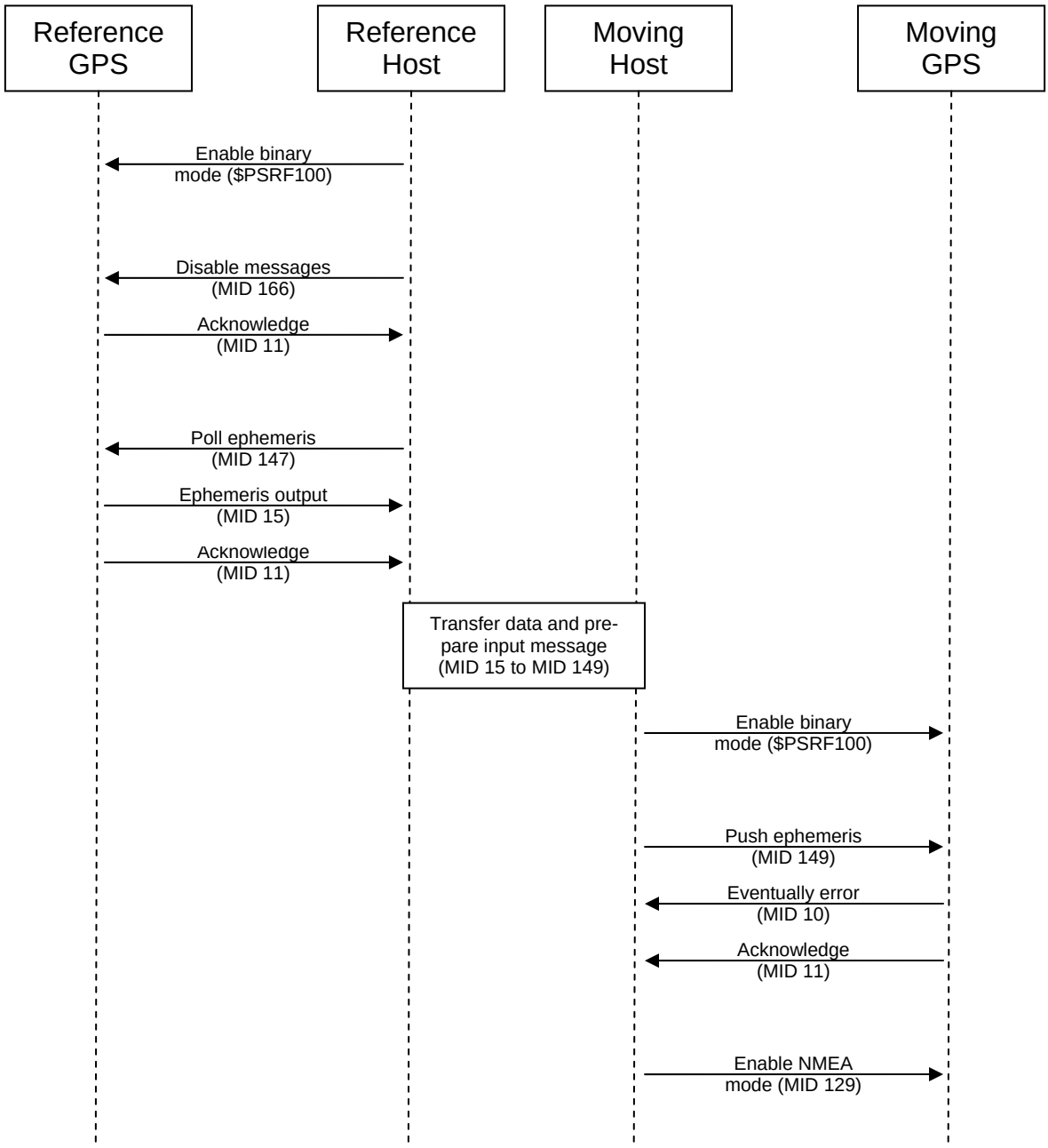

Figure 9: Ephemeris Poll and Push – typical flow

6 Related Information

6.1 Contact

This manual was created with due diligence. We hope that it will be helpful to the user to get the most out of the GPS modules.

Inputs regarding errors or mistaken verbalizations and comments or proposals to Vincotech, Germany, for further improvements are highly appreciated.

Vincotech GmbH
Bibergerstr. 93
82008 Unterhaching (Munich)
Germany

Tel.: +49 89 8780 67 0
Fax: +49 89 8780 67 351
gps@vincotech.com
www.vincotech.com/gps

6.2 Related Documents

- SiRF_NMEA_Reference_Manual_2.2 (SiRF)
- SiRF_Binary_Reference_Manual_2.4 (SiRF)

- GPS Receiver A1080 (Vincotech)
- GPS Evaluation Kit EVA1080 (Vincotech)
- GPS Receiver A1084 (Vincotech)
- GPS Evaluation Kit EVA1084 (Vincotech)
- GPS Receiver A1088 (Vincotech)
- GPS Evaluation Kit EVA1088 (Vincotech)
- GPS Receiver A1035-D (Vincotech)
- GPS Evaluation Kit EVA1035-D (Vincotech)
- GPS Receiver A1035-H (Vincotech)
- GPS Evaluation Kit EVA1035-H (Vincotech)

6.3 Related Tools

- GPS Cockpit (Vincotech)
- SiRF Demo (SiRF)
- SiRF Flash (SiRF)
- sirfbin.exe (Vincotech)
- HTerm: freeware tool, available through http://www.der-hammer.info/terminal/ (homepage in German only, SW in English)

7 List of Figures

Figure 1: sirfbin command window 8
Figure 2: SiRFDemo – Transmit Serial Message 9
Figure 3: SiRFDemo – Switch to SiRF Protocol 10
Figure 4: SiRFDemo – Poll Ephemeris data 11
Figure 5: SiRFDemo – Erase data on receiver 11
Figure 6: SiRFDemo – Cold Start Scenario 12
Figure 7: SiRFDemo – Warm Start Scenario 12
Figure 8: SiRFDemo – Push Ephemeris data 13
Figure 9: Ephemeris Poll and Push – typical flow 19

8 List of Tables

Table 1: Switching off periodic output message 15