



# **Extended Ephemeris Receiver**

**A description on how to load  
Vincotech's GPS receiver and antenna modules  
with EE data files**

**Application Note**  
Version 1.0

## Revision History

Rev.	Date	Description
1.0	01-15-10	Release
	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

<b>1 Introduction</b> .....	<b>5</b>
<b>2 Preparation</b> .....	<b>6</b>
2.1 Setting the Baud Rate .....	6
2.2 Duration of Transfer .....	7
2.3 Data Transfer .....	8
2.3.1 Initialization Stage .....	9
2.3.2 Transfer Stage .....	9
2.3.3 Additional Commands .....	9
<b>3 NMEA Data Transfer Messages</b> .....	<b>10</b>
3.1 Start Download.....	10
3.2 Send File Size .....	11
3.3 Send Data .....	12
3.4 Request EE Age.....	13
3.5 Ack / Nack Responses .....	14
3.6 Get Age Response .....	15
<b>4 SiRF Binary Data Transfer Messages</b> .....	<b>16</b>
4.1 Start Download.....	16
4.2 Send File Size .....	17
4.3 Send Data .....	18
4.4 Request EE Age.....	20
4.5 Ack / Nack Responses .....	21
4.6 Get Age Response .....	22
<b>5 Related Information</b> .....	<b>23</b>
5.1 Contact.....	23
5.2 Related Documents.....	23
5.3 Related Tools .....	23
<b>6 List of Figures</b> .....	<b>24</b>
<b>7 List of Tables</b> .....	<b>24</b>

# 1 Introduction

This document describes the way to work with a GPS receiver or antenna module incorporated within a unit with an additional processor and memory resources. The document covers the third part of a complete process of retrieving Extended Ephemeris (EE) data from Vincotech’s EE server, transferring them to the units in the field and then finally loading the receiver with the EE data. The following picture gives an overview:

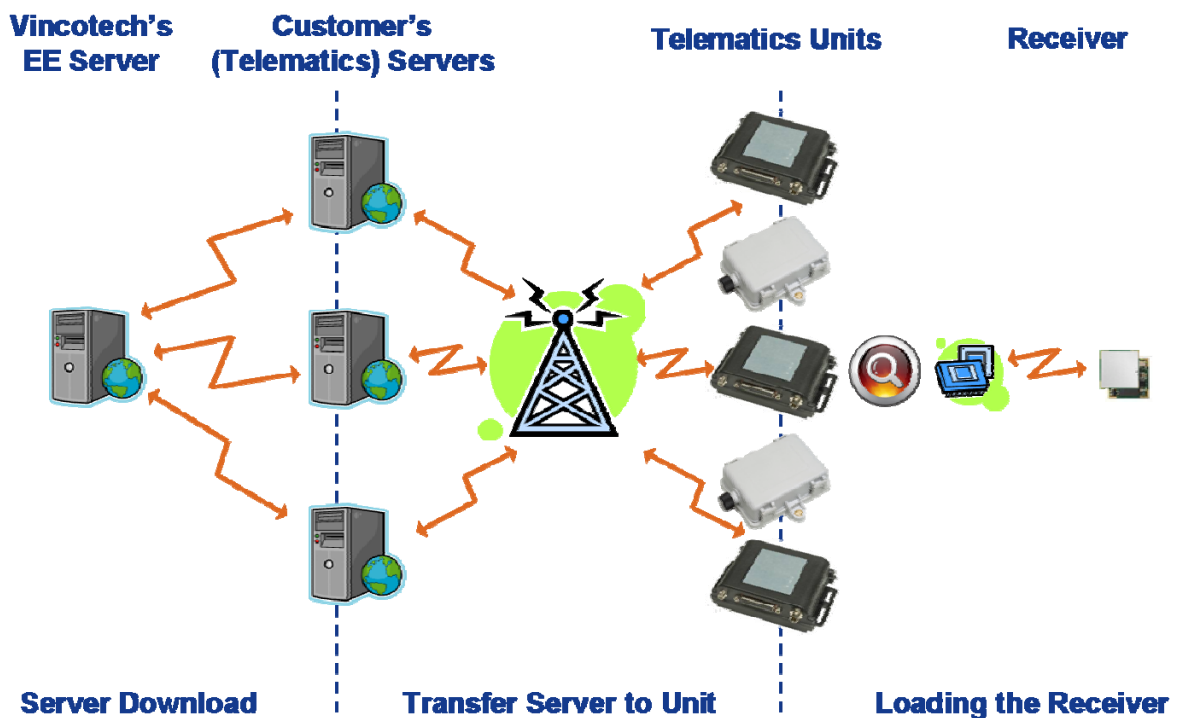


Figure 1: Overview of EE use

The first part “**Server Download**” is described within the application note “GPS AppNote EE Server Download”. The second part “**Transfer Server to Unit**” is a custom specific implementation of a file transfer and storage. The overview shows GSM / GPRS as the transfer medium. Typically ftp could be used to transmit the file from a server to a unit in the field. The file can then be stored in flash, SRAM, on an SD card, with or without file system in the unit. Alternative transmission methods could be WLAN or simply via a direct serial (e.g. USB connection). In the following chapters the third part “**Loading the Receiver**” of the complete process is described.

**Note:** The GPS receiver or GPS antenna module must hold SiRF FW revision 3.6.0 or higher. Please verify the FW on your receiver (according information is giving on the label of the module! The part of the firmware supporting Extended Ephemeris is referred to as Embedded Client Location Manager (ECLM).

## 2 Preparation

### 2.1 Setting the Baud Rate

Dependent on the validity of the EE data (currently 1, 3, 5, and 7 days are supported) the file size is between 9.5 and some 65 kBytes. The default setting of firmware 3.6.0 is NMEA and 4,800 baud. It is possible to use both SiRFbinary and NMEA mode for the transfer. Anyhow it is recommended to switch the baud rate to a higher value, e.g. 57,600 baud, in order to speed up this process. This can be done using the command **PSRF100**, described here in detail with an example for NMEA mode and 57,600 baud:

- \$PSRF100,1,57600,8,1,0\*36

Name	Example	Description
Message ID	\$PSRF100	PSRF100 protocol header
Protocol	1	0 SiRF binary <sup>(1)</sup> / 1 NMEA
Baud	57600	4800, 9600, 19200, 38400, 57600, 115200
DataBits	8	8, 7 <sup>(1)</sup>
StopBits	1	0, 1 <sup>(1)</sup>
Parity	0	0 none / 1 odd / 2 even <sup>(1)</sup>
Checksum	*36	End of message termination

(1) SiRF protocol is only valid for 8 data bits, 1 stop bit and no parity

Table 1: Serial Port Set-up

One can return to the original baud rate of 4,800 baud by sending the following command:

- \$PSRF100,1,4800,8,1,0\*0E

For switching to binary mode and back to NMEA, please refer to the Description in "GPS AppNote Firmware GSC3".

## 2.2 Duration of Transfer

Protocol and baud rate will have an affect on the amount of time required to transfer data to the receiver. Refer to the following tables when deciding whether to use SiRF binary or NMEA and which baud rate (not complete, but an excellent reference):

Baud Rate	NMEA	SiRF Binary
4800	160s	80s
9600	55s	30s
57600	10s	4s

Table 2: Transfer time for a 1 Day Extended Ephemeris File

Baud Rate	NMEA	SiRF Binary
4800	300s	180s
9600	130s	80s
57600	33s	20s

Table 3: Transfer time for a 3 Day Extended Ephemeris File

Baud Rate	NMEA	SiRF Binary
4800	660s	330s
9600	270s	170s
57600	77s	30s

Table 4: Transfer time for a 7 Day Extended Ephemeris File

## 2.3 Data Transfer

The data transfer can be split into two stages, the initialization and the real transfer following it. The flow below shall serve as an overview and explanation:

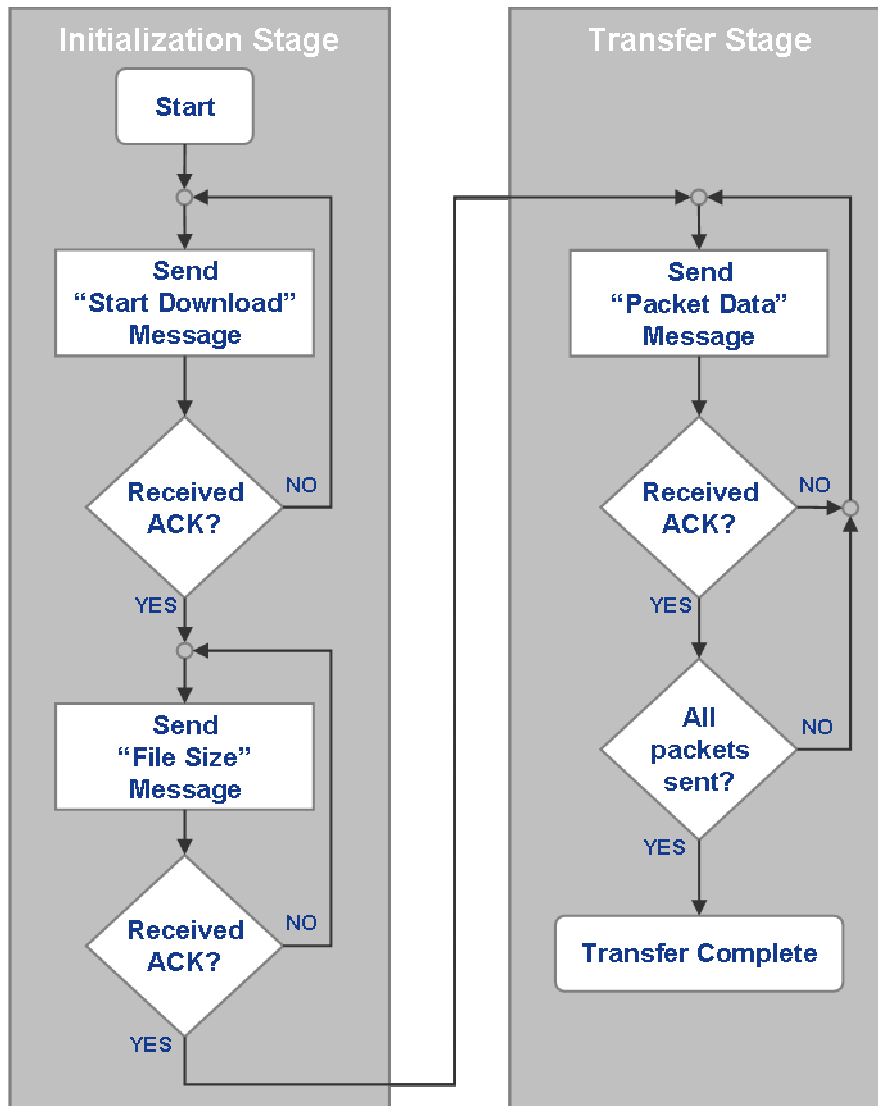


Figure 2: Data transfer



### 2.3.1 Initialization Stage

Before the Extended Ephemeris File can be transferred to the GPS receiver, the following process must be carried out:

- 1) Send “Start Download” Message
- 2) Wait for “Ack/Nack” Message from receiver
- 3) Send “File Size” Message
- 4) Wait for “Ack/Nack” Message from receiver

Please see the following sections for descriptions of each message.

#### **NMEA:**

Start Download – “3.1 Start Download”

File Size – “3.2 Send File Size”

Ack/Nack – “3.5 Ack / Nack Responses”

#### **SiRF Binary:**

Start Download – “4.1 Start Download”

File Size – “4.2 Send File Size”

Ack/Nack – “4.5 Ack / Nack Responses”

### 2.3.2 Transfer Stage

Once the initialization stage has completed, the following process may take place:

- 1) First file packet is sent to the receiver with the “Packet Data” message.
- 2) Wait for “Ack/Nack” message from receiver.
- 3) Loop between steps 1 and 2 until the entire file is transferred.

See the following sections for descriptions of each message.

#### **NMEA:**

– Packet Data – “3.3 Send Data”

– Ack/Nack – “3.5 Ack / Nack Responses”

#### **SiRF Binary:**

– Packet Data – “4.3 Send Data”

– Ack/Nack – “4.5 Ack / Nack Responses”

### 2.3.3 Additional Commands

In order to request the age of the EE data already loaded into the receiver it is possible to request their age. The response will return the age in seconds along with the total prediction interval. Therefore the remaining prediction time can be calculated.

### 3 NMEA Data Transfer Messages

#### 3.1 Start Download

ECLM Start Download - MsgID \$PSRF114, SubMsgID 0x16

This message is to indicate to the GPS receiver that the host processor wants to initiate the EE file download. The SubMsgID for this message is fixed to 0x16.

**Example:** \$PSRF114,16\*08

Name	Example	Unit	Description
Message ID	\$PSRF114	-	ECLM Download Packet Header
Sub Message ID	0x16	-	SubID 22(dec) : Sub Id for Start Download
Checksum	*08	-	-
<CR><LF>	-	-	End of message termination.

Table 5: ECLM Start Download Message Description

- Note:** This message would be acknowledged to indicate SUCCESS/FAILURE.
- SUCCESS: would be acknowledged with “Ack: \$PSRF156” using Command Acknowledgment – NMEA Message ID \$PSRF156, Sub Msg ID 0x20 – see paragraph “3.5 Ack / Nack Responses”.
  - FAILURE: would be acknowledged with “Nack: \$PSRF156” using Command Negative Acknowledgment – NMEA Message ID \$PSRF156, Sub Msg ID 0x20 – see paragraph “3.5 Ack / Nack Responses”.

### 3.2 Send File Size

ECLM Send File Size - MsgID \$PSRF114, SubMsgId 0x17

This message is sent from the host processor to the GPS Receiver to indicate the size of the EE file to be downloaded. The SubMsgID for this message is fixed to 0x17.

**Example:** \$PSRF114,17,2859\*23

In the example a file size of 10329 bytes is used which corresponds to 0x2859.

Name	Example	Unit	Description
Message ID	\$PSRF114	-	Message Id
Sub Message ID	0x17	-	SubID: EE File Size Message
File Length	0x2859	-	Length of the Extended Ephemeris File to be downloaded in bytes. Length of file does not include any overhead from adding start sequence, message checksums, or end sequences.
Checksum	*23	-	-
<CR><LF>	-	-	End of message termination.

Table 6: ECLM File Size Message Description

- Note:** This message would be acknowledged to indicate SUCCESS/FAILURE.
- SUCCESS: would be acknowledged with "Ack: \$PSRF156" using Command Acknowledgment – NMEA Message ID \$PSRF156, Sub Msg ID 0x20 – see paragraph "3.5 Ack / Nack Responses".
  - FAILURE: would be acknowledged with "Nack: \$PSRF156" using Command Negative Acknowledgment – NMEA Message ID \$PSRF156, Sub Msg ID 0x20 – see paragraph "3.5 Ack / Nack Responses"

### 3.3 Send Data

ECLM Packet Data - MsgID \$PSRF114, SubMsgID 0x18

This message is sent from the host processor to the GPS receiver to actually transmit the file's data. The SubMsgID for this message is fixed to 0x18.

**Example:** \$PSRF114,18,1,32,62,12,31,6,3,2,7,d9,7,7,0,0,39,6d,8f,12,0,0,0,0,0,1,2d,9a,e7,5,2,ff,fe,28,5\*3d

This is the first package, so packet Sequence No = 0x1 and the packet length is 20 (0x14).

Name	Example	Unit	Description
Message ID	\$PSRF114	-	Message Id
Sub Message ID	0x18	-	SubID: SGEE Packet Data
Packet Sequence No	0x1	-	Packet Sequence number of the current packet Starting from 1
Packet Length	32	bytes	Length of this packet. Number of bytes in decimal. Max. number is 400 (decimal).
Packet Data	62,12,31,6,3,2,7,d9,7,7,0,0,39,6d,8f,12,0,0,0,0,0,1,2d,9a,e7,5,2,ff,fe,28,5	-	EE Data in this packet of length PacketLength
Checksum	*3d	-	-
<CR><LF>	-	-	End of message termination.

Table 7: ECLM Send Data Message Description

- Note:** This message would be acknowledged to indicate SUCCESS/FAILURE.
- SUCCESS: would be acknowledged with "Ack: \$PSRF156" using Command Acknowledgment – NMEA Message ID \$PSRF156, Sub Msg ID 0x20 – see paragraph "3.5 Ack / Nack Responses".
  - FAILURE: would be acknowledged with "Nack: \$PSRF156" using Command Negative Acknowledgment – NMEA Message ID \$PSRF156, Sub Msg ID 0x20 – see paragraph "3.5 Ack / Nack Responses".

### 3.4 Request EE Age

ECLM Get EE Age - MsgID \$PSRF114, SubMsgId 0x1A

This message is sent from the host processor to the GPS receiver to get the age of the EE data of single satellites. The SubMsgID for this message is fixed to 0x1A.

**Example:** \$PSRF114,1a,1\*42

This message requests age of satellite #1.

Name	Example	Unit	Description
Message ID	\$PSRF114	-	Message Id
Sub Message ID	0x1a	-	SubID: EE Age
Satellite ID	0x1	-	Satellite ID (in hex) for which age is requested
Checksum	*42	-	-
<CR><LF>	-	-	End of message termination.

Table 8: ECLM Get EE Age Message Description

- Note:** This message would be acknowledged to indicate SUCCESS/FAILURE.
- SUCCESS: would be acknowledged with "Ack: \$PSRF156" using Command Acknowledgment – NMEA Message ID \$PSRF156, Sub Msg ID 0x22 with EE Age – see paragraph "3.6 Get Age Response" – this is the specific response for this request
  - FAILURE: would be acknowledged with "Nack: \$PSRF156" using Command Negative Acknowledgment – NMEA Message ID \$PSRF156, Sub Msg ID 0x20 – see paragraph "3.5 Ack / Nack Responses" – this is the standard response

### 3.5 Ack / Nack Responses

Ack/Nack Response - Message ID \$PSRF156, SubMsgID 0x20

This is the receiver's response to the message \$PSRF114 with SubMsgIDs 0x16, 0x17, 0x18, 0x1a.

The SubMsgID for this message is fixed to 0x20.

The Ack Sub ID corresponds to the SubMsgID of the original message that triggered this response.

**Example:** \$PSRF156,20,72,16,0,0\*09

This example holds Sub Message ID = 0x20, AckMid = 114, AckSid = 0x16, Ack, Reason = SUCCESS. AckMid and AckSid are the Message IDs and SubID of the message for which this response Message will be sent.

Name	Example	Unit	Description
Message ID	\$PSRF156	-	ECLM Response
Sub Message ID	0x20	-	SubID 32(dec) : Sub Id for Ack/Nack
Ack Message ID	0x72	-	MsgID 114(dec) : Msg Id for ECLM Download
Ack Sub ID	0x16	-	SubID: Sub Id for ECLM Start Download This field may take values 0x16, 0x17, 0x18, or 0x1a to ACK corresponding SID's.
Ack/Nack	0x0	-	0: Ack / 1: Nack
Reason	0x0	-	ECLM_SUCCESS = 0, ECLM_SPACE_UNAVAILABLE = 1, /* if flash space is not sufficient */ ECLM_PKT_LEN_INVALID = 2, /* if pkt length field is out of range */ ECLM_PKT_OUT_OF_SEQ = 3, /* if pkt received is out of sequence */ ECLM_DOWNLOAD_SGEE_NONEWFILE = 4, /* no new file */ ECLM_DOWNLOAD_CORRUPTFILE_ERROR = 5, /* corrupt file */ ECLM_DOWNLOAD_GENERIC_FAILURE = 6, /* generic failure */ ECLM_API_GENERIC_FAILURE = 7 /* generic failure calling CLM API */
Checksum	*08	-	-
<CR><LF>	-	-	End of message termination.

Table 9: ECLM Message Response Description

### 3.6 Get Age Response

Get Age Response - Message ID \$PSRF156, SubMsgID 0x1a

This is the receiver's response on the message \$PSRF114, SubMsgID 0x1a. The SubMsgID for this message is fixed to 0x22.

**Example:** \$PSRF156,22,7da8,15180\*3e

This example holds Sub Message ID = 0x22, the acknowledge itself, the age value of 0x7da8 and the prediction interval of 0x15180.

Name	Example	Unit	Description
Message ID	\$PSRF156	-	ECLM Response
Sub Message ID	0x22	-	SubID 34(dec) : Sub Id for EE age
EE Age	0x7da8	-	Age of the satellite data in seconds
Prediction Interval	0x15180	-	Prediction interval in seconds: Total validity period of the extended ephemeris in seconds. Prediction interval minus EE age would result in the remaining validity period of the EE file.
Checksum	*3e	-	-
<CR><LF>	-	-	End of message termination.

Table 10: ECLM Get EE Age Response

## 4 SiRF Binary Data Transfer Messages

### 4.1 Start Download

ECLM Start Download - Msg ID 0xe8, SubMsgID 0x16

This message is to indicate to the GPS receiver that the host processor wants to initiate the EE file download. The SubMsgID for this message is fixed to 0x16.

#### Example:

Message = a0 a2 00 02 e8 16 00 fe b0 b3

a0 a2 00 02— Start Sequence and Payload Length

e8 16 — Payload

00 fe b0 b3— Message Checksum and End Sequence

Name	Bytes	Value (Hex)		Unit	Description
		Scale	E.g.		
Message ID	1U	-	0xe8	-	Standard host applications to receive communication Message ID
Sub Message ID	1U	-	0x16	-	SubID to indicate Start Download Message

Payload length: 2 bytes

Table 11: ECLM Start Download Payload Description

**Note:** This message would be acknowledged to indicate SUCCESS/FAILURE.  
 – SUCCESS/FAILURE: would be acknowledged with Command Acknowledgment – SSB Message ID 0x38, Sub Msg ID 0x20 – see paragraph “4.5 Ack / Nack Responses”



## 4.2 Send File Size

ECLM Send File Size - Msg ID 0xe8, SubMsgID 0x17

This message is sent from the host processor to the GPS Receiver to indicate the size of the EE file to be downloaded. The SubMsgID for this message is fixed to 0x17.

### Example:

Message = a0 a2 00 06 e8 17 00 00 28 59 01 80 b0 b3

a0 a2 00 06— Start Sequence and Payload Length

e8 17 00 00 28 59 — Payload

01 80 b0 b3 — Message Checksum and End Sequence

In the example a file size of 10329 bytes is used which corresponds to 0x2859.

Name	Bytes	Value (Hex)		Unit	Description
		Scale	E.g.		
Message ID	1U	-	0xe8	-	Standard host applications to receive communication Message ID
Sub Message ID	1U	-	0x17	-	SubID to indicate File Size Message
File length	4U	-	0x00 0x00 0x28 0x59	-	Length of the Extended Ephemeris File to be downloaded in bytes. Length of file does not include any overhead from adding start sequence, message checksums, or end sequences. (big endian)

Payload length: 6 bytes

Table 12: ECLM File Size Payload Description

**Note:** This message would be acknowledged to indicate SUCCESS/FAILURE.  
 – SUCCESS/FAILURE: would be acknowledged with Command Acknowledgment – SSB Message ID 0x38, Sub Msg ID 0x20 – see paragraph “4.5 Ack / Nack Responses”

### 4.3 Send Data

ECLM Packet Data - Msg ID 0xe8, SubMsgID 0x18

This message is sent from the host processor to the GPS receiver to actually transmit the file's data. The SubMsgID for this message is fixed to 0x18.

**Example:**

Message = a0 a2 00 26 e8 18 00 01 00 20 62 12 31 06 03 02 07 d9 07 07 00  
00 39 6d 8f 12 00 00 00 00 00 01 2d 9a e7 05 02 ff fe 28 05 07 e6 b0 b3

a0 a2 00 26 — Start Sequence and Payload Length

e8 18 00 01 00 20 62 12 31 06 03 02 07 d9 07 07 00 00 39 6d 8f 12 00 00  
00 00 00 00 01 2d 9a e7 05 02 ff fe 28 05— Payload

07 e6 b0 b3— Message Checksum and End Sequence

As this is the first package packet Sequence No = 0x1; packet length is 32 (0x20).

Name	Bytes	Value (Hex)		Unit	Description
		Scale	E.g.		
Message ID	1U	-	0xe8	-	Standard host applications to receive communication Message ID
Sub Message ID	1U	-	0x18	-	SubID to indicate packet data
Packet Sequence Number	2U	-	0x00 0x01	-	Packet sequence number of the current packet starting from 1. (big endian)
Packet length	2U	-	0x00 0x20	-	Length of the Extended Ephemeris data in current packet in bytes (hex value). Value includes only size of EE data and not start sequence, payload length, message checksums, or end sequences. Max. number is 400 (decimal) / 0x190 (hex). (big endian)
Packet data	Packet length	-	62 12 31 06 03 02 07 d9 07 07 00 00 39 6d 8f 12 00 00 00 00 00 00 01 2d 9a e7 05 02 ff fe 28 05	-	EE data of length indicated in packet length of the message. (big endian)

Payload length: 6 + <packet length> bytes

Table 13: ECLM Packet Data Payload Description

**Note:** This message would be acknowledged to indicate SUCCESS/FAILURE.  
– SUCCESS/FAILURE: would be acknowledged with Command Acknowledgment – SSB Message ID 0x38, Sub Msg ID 0x20 – see paragraph “4.5 Ack / Nack Responses”

## 4.4 Request EE Age

ECLM Get SGEE Age - Msg ID 0xe8, SubMsgID 0x11

This message is sent from the host processor to the GPS receiver to get the age of the EE data of single satellites. The SubMsgID for this message is fixed to 0x1a.

### Example:

Message = a0 a2 00 03 e8 1a 01 01 03 b0 b3

a0 a2 00 03— Start Sequence and Payload Length

e8 1a 01 — Payload

01 03 b0 b3— Message Checksum and End Sequence

This message requests age of satellite #1.

Name	Bytes	Value (Hex)		Unit	Description
		Scale	E.g.		
Message ID	1U	-	0xe8	-	Standard host applications to receive communication Message ID
Sub Message ID	1U	-	0x1a	-	SubID to indicate File Size Message
Sat ID	1U	-	0x01	-	PRN (in hex) for which EE age is requested

Payload length: 3 bytes

Table 14: ECLM File Size Payload Description

- Note:** This message would be acknowledged to indicate SUCCESS/FAILURE.
- SUCCESS: would be acknowledged along with EE data using Command Acknowledgment – SSB Message ID 0x38, Sub Msg ID 0x22 – see paragraph “4.6 Get Age Response” – this is the specific response for this request
  - FAILURE: would be acknowledged with “Rejected: MID\_ECLMAckNack” using Command Acknowledgment – SSB Message ID 0x38, Sub Msg ID 0x22 – see paragraph “4.5 Ack / Nack Responses” – this is the standard response

## 4.5 Ack / Nack Responses

Ack/Nack Response - Msg ID 0x22, SubMsgID 0x20

This is the receiver's response to the message ID 0xE8 with SubMsgIDs 0x16, 0x17, 0x18, or 0x1a.

The SubMsgID for this message is fixed to 0x20.

The Ack Sub ID corresponds to the SubMsgID of the original message that triggered this response.

### Example:

Message = a0 a2 00 06 38 20 e8 16 00 00 01 56 b0 b3

a0 a2 00 06— Start Sequence and Payload Length

38 20 e8 16 00 00 — Payload

01 56 b0 b3 — Message Checksum and End Sequence

This example is an acknowledge to message 0xe8 with SubID 0x16 (ECLM Start Download).

Name	Bytes	Value (Hex)		Unit	Description
		Scale	E.g.		
Message ID	1U	-	0x38	-	Standard host applications to receive communication Message ID
Sub Message ID	1U	-	0x20	-	SubID ECLM Ack/Nack
Ack Msg ID	1U	-	0xe8	-	Ack Message ID 0xe8
Ack Sub ID	1U	-	0x16	-	Ack SubID, ECLM Start Download 0x16
Ack/Nack	1U	-	0x00	-	0 = Ack 1 = Nack
Ack Nack Reason	1U	-	0x00	-	ECLM_SUCCESS = 0, ECLM_SPACE_UNAVILABLE = 1, /* if flash space is not sufficient */ ECLM_PKT_LEN_INVALID = 2, /* if pkt length field is out of range */ ECLM_PKT_OUT_OF_SEQ = 3, /* if pkt received is out of sequence */ ECLM_DOWNLOAD_SGEE_NONE_WFILE = 4, /* no new file */ ECLM_DOWNLOAD_CORRUPTFILE_ERROR = 5, /* corrupt file */ ECLM_DOWNLOAD_GENERIC_FAILURE = 6, /* generic failure */ ECLM_API_GENERIC_FAILURE = 7 /* generic failure calling CLM API */

Payload length: 6 bytes

Table 15: ECLM Start Download Ack/Nack Message Description

## 4.6 Get Age Response

Get Age Response - Msg ID 0x38, SubMsgID 0x22

This is the receiver's response to the message Get EE Age with Message ID 0xe8, SubMsgID 0x1A. The SubMsgID for this message is fixed to 0x22.

### Example:

Message = a0 a2 00 0a 38 22 00 00 80 ea 00 01 51 80 02 96 b0 b3

a0 a2 00 0a— Start Sequence and Payload Length

38 22 00 00 80 ea 00 01 51 80 — Payload

02 96 b0 b3 — Message Checksum and End Sequence

This example holds Sub Message ID = 0x22, the acknowledge itself, the age value of 0x80ea and the prediction interval of 0x15180.

Name	Bytes	Value (Hex)		Unit	Description
		Scale	E.g.		
Message ID	1U	-	0x38	-	Standard host applications to receive communication Message ID
Sub Message ID	1U	-	0x22	-	SubID ECLM Get EE Age
EE Age	4U	-	0x00 0x00 0x80 0xea	-	Age of the satellite data in seconds. (big endian)
Prediction Interval	4U	-	0x00 0x01 0x51 0x80	-	Prediction interval in seconds: Total validity period of the extended ephemeris in seconds. Prediction interval minus EE age would result in the remaining validity period of the EE file. (big endian)

Payload length: 10 bytes

Table 16: ECLM Get EE Age Response Description

## 5 Related Information

### 5.1 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 module.

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

#### **Vincotech GmbH**

Biberger Str. 93  
82008 Unterhaching (Munich)  
Germany

Tel.: +49 89 8780 67 0

Fax: +49 89 8780 67 398

[gps@vincotech.com](mailto:gps@vincotech.com)

[www.vincotech.com/gps](http://www.vincotech.com/gps)

### 5.2 Related Documents

- GPS AppNote Server Download (Vincotech)
- GPS AppNote Firmware GSC3 (Vincotech)
- SiRF\_Binary\_Reference\_Manual (SiRF)
- SiRF\_NMEA\_Reference\_Manual (SiRF)

### 5.3 Related Tools

- GPS Cockpit (Vincotech)
- SiRF GetEE (SiRF)
- SiRF Flash (SiRF)

## 6 List of Figures

Figure 1: Overview of EE use .....	5
Figure 2: Data transfer .....	8

## 7 List of Tables

Table 1: Serial Port Set-up.....	6
Table 2: Transfer time for a 1 Day Extended Ephemeris File.....	7
Table 3: Transfer time for a 3 Day Extended Ephemeris File.....	7
Table 4: Transfer time for a 7 Day Extended Ephemeris File.....	7
Table 5: ECLM Start Download Message Description .....	10
Table 6: ECLM File Size Message Description .....	11
Table 7: ECLM Send Data Message Description .....	12
Table 8: ECLM Get EE Age Message Description .....	13
Table 9: ECLM Message Response Description.....	14
Table 10: ECLM Get EE Age Response .....	15
Table 11: ECLM Start Download Payload Description .....	16
Table 12: ECLM File Size Payload Description.....	17
Table 13: ECLM Packet Data Payload Description .....	18
Table 14: ECLM File Size Payload Description.....	20
Table 15: ECLM Start Download Ack/Nack Message Description .....	21
Table 16: ECLM Get EE Age Response Description .....	22