AirPrime GMM-G3, XA11xx and XM11xx GNSS Aiding Application Note

2174195-2.2

Author:	Sierra Wireless	**Date:**	June 03, 2019

APN Content Level	**BASIC**		**INTERMEDIATE**	✓	**ADVANCED**		**Confidentiality**	**Public**	✓	**Private**	

Hardware Compatibility	**Product Line**	**AirPrime**	**Series**	**GMM-G3**	
				XA11xx	**XM11xx**

Software Compatibility	**ALL**	**Document Type**	**Application Note**	✓	**Technical Note**	

1 Version

These documents may be updated over their lifetime. To ensure you design with the correct version, please check the Source for latest versions.

2 Introduction

This Application Note (APN) is provided to Sierra Wireless distributors and clients to aid more rapid development using the Sierra Wireless portfolio of GNSS solutions. To request a new application note, contact your regional Sierra Wireless Product Marketing Manager.

3 Glossary

Abbreviation	Definition
AN	Application Note
EPO	Extended Predictive Orbit
FTP	File Transfer Protocol
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
MTK	Mediatek
PTS	Product Technical Specification
RTC	Real Time Clock
SEQ	Sequence
SV	Satellite Vehicle
TTFF	Time To First Fix
UTC	Universal Time Coordinate
ASCII	American Standard Code for Information Interchange

4 Overview

To enable a GNSS engine to acquire a faster TTFF, aiding data can be applied to it while it is searching for SVs; this is generally referred to as aiding. This aiding data can consist of one or several of the following.

- Current UTC
- Current position
- Almanac and ephemeris information to be applied to a given SV

There are two aiding methods – host aiding and flash aiding. This application note describes these aiding sources and shows the data that Sierra Wireless can supply, its format, and how to download it to one of the MTK-based GNSS engines currently being supplied by Sierra Wireless.

5 Almanac and Ephemeris

The definition of the almanac and ephemeris information is given below.

- Almanac – This is a general file which enables the a GNSS engine, with a given time and date, to determine which SVs it should be able to see. This was important in the past when modules had limited numbers of tracers that could be used during acquisition. Given that GNSS receivers now have in the hundreds of trackers, it is less critical.
- Ephemeris – This is the detailed position information for a given SV. Along with the time from the SV, it is what is required to be able to calculate a position. Once the module can see the SV and gets the time from it, getting this data is what will determine the TTFF. Ephemeris data has a life of 6 hours before becoming invalid.

Information supplied by Sierra Wireless is given in an EPO file. There are two versions of EPO data – version 1, which is GPS only; and version 2, which is GPS and GLONASS. Beidou and Galileo are currently not supported.

5.1 EPO Version 1

EPO version 1 supports GPS only. The basic unit of an EPO file is SAT Data, and the data size of a SAT Data is 60 bytes. One EPO SET contains 32 SAT Data, the data size for an EPO SET is 1920 bytes. Each EPO file contains several EPO SETs. The file size must be a multiple of 1920. An EPO SET is valid for 6 hours. Therefore, there will be 4 EPO SETs for one day. A 7-day EPO file, for example, will be 7*4 = 28 EPO SETs and the file size will be 28*1920 = 53760 bytes.

The format of the EPO file as it is downloaded is shown in the figure below.

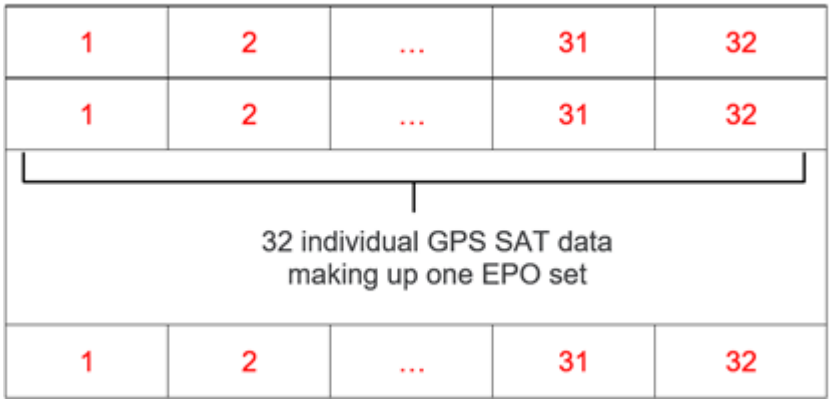

Figure 1. GPS-only EPO Data Format

5.2 EPO Version 2

The EPO version 2 supports GPS-only and GPS + GLONASS. The basic unit of an EPO file is SAT Data, the data size of a SAT Data is 72 bytes.

5.2.1 GPS-only

One EPO SET for GPS-only contains 32 SAT Data, the data size for an EPO SET is 2304 bytes. Each EPO file contains several EPO SETs. The file size must be a multiple of 2304. An EPO SET is valid for 6 hours. Therefore, there will be 4 EPO SETs for one day. A 3-day EPO file, for example, will be 3*4 = 12 EPO SETs and the file size will be 12*2304 = 27648 bytes.

5.2.2 GPS+GLONASS

One EPO SET for GPS+GLONASS contains 56 SAT Data, the data size for an EPO SET is 4032 bytes. Each EPO file contains several EPO SETs. The file size must be a multiple of 4032. An EPO SET is valid for 6 hours. Therefore, there will be 4 EPO SETs for one day. A 3-day EPO file, for example, will be 3*4 = 12 EPO SETs and the file size will be 12*4032 = 48384 bytes.

The format of the EPO file as it is downloaded is shown in the figure below.

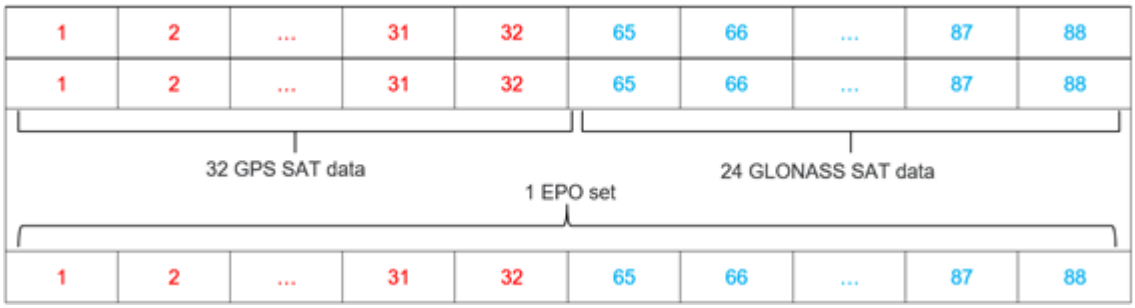

Figure 2. GPS+GLONASS EPO Data Format

5.3 FTP Server

Customers can download the current EPO data files from an FTP server. There are several files available:

- GPS-only
  - Version 1
    - 7-day
    - 14-day
  - Version 2
    - 6-hour
    - 3-day
    - 30-day
- GPS+GLONASS
  - Version 2
    - 6-hour
    - 3-day

The files on the server will be automatically updated everyday. Additionally, 6-day EPO files will be updated every hour.

For connection details of the FTP server, please contact your Sierra Wireless FAE or distribution contact.

6 Host Aiding

To facilitate faster TTFF, the host can add the reference UTC time, reference location and EPO data into the GNSS module. Adding assistance data requirement is shown in the following table. Some of the assistance data can be skipped if it is not available or if the data might be wrong. Please note that no assistance data is always better than an incorrect one. Also, host aiding doesn't support EPO version 1. Assistance data is temporarily stored in the GNSS module RAM. When the GNSS module powers off, the assistance data will be gone.

Table 1: Adding Data Requirement

Assistance Data	Requirement
Reference UTC time	Deviation of the reference UTC should be less than 3 seconds.
Reference location	Deviation of the reference location should be less than 30 kilometers.
EPO data	EPO set of the current UTC time

6.1 Recommend Implementation

The sequence of assistance data transfer is shown as the figure below. After powering on, the GNSS module will output a start-up message **`$PMTK010,001*2E`** to notify the host that it has finished initialization and is capable of receiving commands. After the host receives the system startup message, it can decide to send the assistance data as per the figure below. The sequence of assistance data is time, location, and then the EPO data. If any of the assistance data is not available at the host, the host could skip sending certain assistance data. For example, if the system time of the host is invalid, please do not send the time assistance to the GNSS module. Similarly, if the EPO data at the host is corrupt, the host should not send EPO assistance data to the GNSS module. No assistance data is always better than bad assistance data.

In the original design, the purpose of acknowledgment packets is to prevent the UART buffer of the GNSS module from overflowing. However, firmware design has improved so that the host software does not need to be concerned about acknowledgments. Assistance packets can be sent without waiting for any acknowledgment.

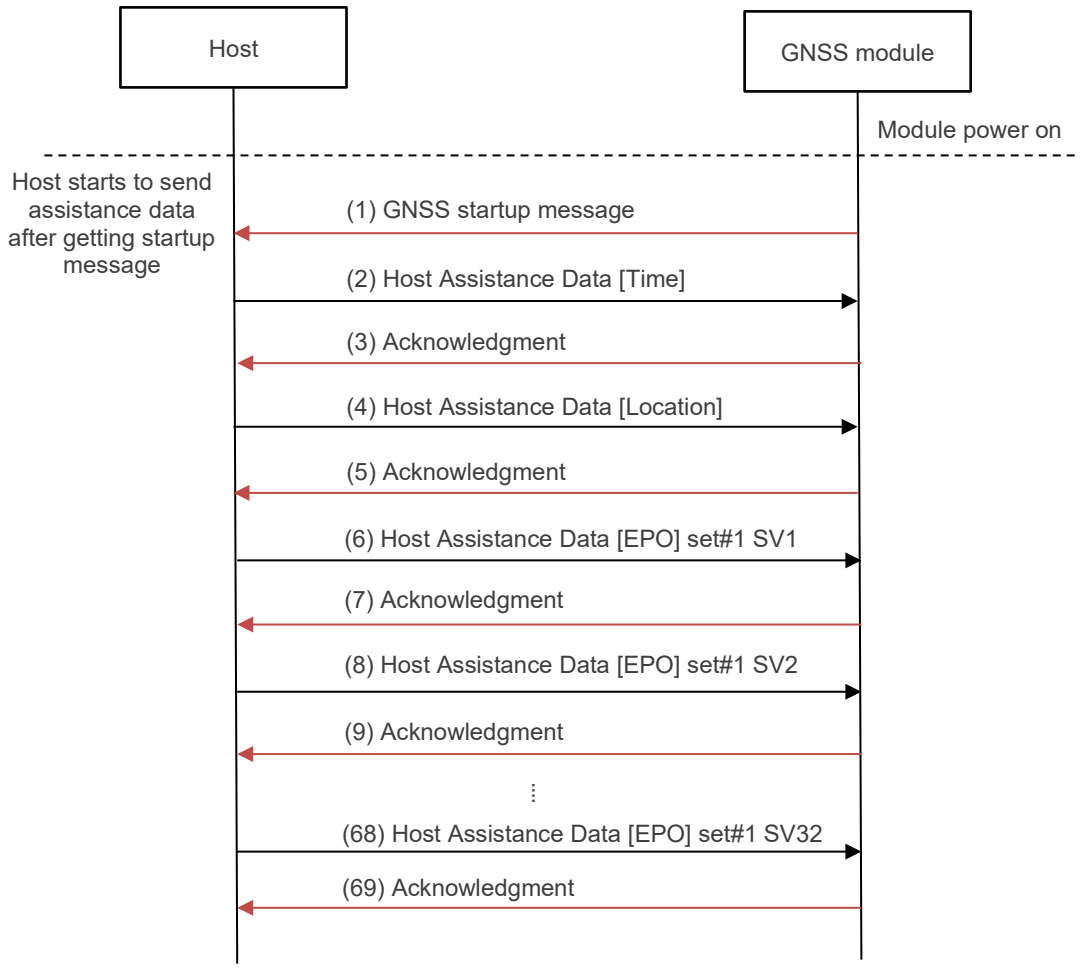

Figure 3. Sequence of Assistance Data Transfer

6.2 PMTK Commands

6.2.1 PMTK721 Transmit EPO Data

Description:

Transmits EPO data of specified SV to GNSS module.

Data Field:

```
PMTK721,SatID,W[0],...,W[17]
```

- **SatID:** SV PRN number in hexadecimal
  - 1 – 32: GPS
  - 65 – 88: GLONASS
- **W[0] – W[17]:** EPO data of a satellite divided into 18 words. The size of each word is 4 bytes. Byte order of each word is little-endian.

Example:

Command	Response

Transmits GPS EPO data:
`$PMTK721,01,010542D5,B4780416,299BDE72,…*Checksum <CR><LF>`
`$PMTK721,02,020542D5,AF780326,299BEE69,…*Checksum <CR><LF>`
:
`$PMTK721,20,200542D5,9778398A,29993151,…*Checksum <CR><LF>`
	`$PMTK001,721,3, 1,00000000*25<CR><LF>`
	`$PMTK001,721,3, 2,00021798*23<CR><LF>`
	:
	`$PMTK001,721,3,32,00021798*30<CR><LF>`

Transmits GLONASS EPO data:
`$PMTK721,41,410542D5,05783FE1,299F21C3,…*Checksum <CR><LF>`
`$PMTK721,42,420542D5,10783FFF,299F11D6,…*Checksum <CR><LF>`
:
`$PMTK721,58,580542D5,FD783EDD,299E4E3B,…*Checksum <CR><LF>`
	`$PMTK001,721,3,65,00021798*32<CR><LF>`
	`$PMTK001,721,3,66,00000000*34<CR><LF>`
	:
	`$PMTK001,721,3,88,00000000*34<CR><LF>`

The following figure is a partial content of GPS+GLONASS EPO data.

Address	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
00000000	d5	42	05	01	16	04	78	b4	72	de	9b	29	2b	0b	54	fe
00000010	14	39	f0	fe	13	de	9b	29	8e	35	7b	07	5b	86	31	8b
00000020	b9	8a	f1	07	41	93	0a	d4	de	0c	70	03	cc	69	03	a6
00000030	d2	9f	e0	92	4b	91	b1	20	44	c4	57	1c	1c	00	00	10
00000040	00	00	10	21	63	71	0d	d0	d5	42	05	02	26	03	78	af
00000050	69	ee	9b	29	45	21	d6	f1	52	09	b7	f0	ff	ee	9b	29

Figure 4. Partial Content of GPS+GLONASS EPO Data

The EPO data within the red frame is for GPS SV 1. The following uses PMTK721 to transmit EPO data to the GNSS module:

`$PMTK721,01,010542D5,B4780416,299BDE72,FE540B2B,FEF03914,299BDE13,077B358E,8B31865B,07F18AB9,D40A9341,03700CDE,A60369CC,92E09FD2,20B1914B,1C57C444,1000001C,21100000,D00D7163*1B<CR><LF>`

6.2.2 PMTK740 Reference UTC Time

Description:

Transmits the reference UTC time to the GNSS module. The deviation of the reference UTC time should be less than 3 seconds.

Data Field:

`PMTK740,YYYY,MM,DD,hh,mm,ss`

- **YYYY:** Year in 4 digits
- **MM:** Month (range: 1 – 12)
- **DD:** Day (range: 1 – 31)

- **hh:** Hour (range: 0 – 23)
- **mm:** Minute (range: 0 – 59)
- **ss:** Second (range: 0 – 59)

Example:

Command	Response
`$PMTK740,2019,5,7,5,20,22*0E<CR><LF>`	`$PMTK001,740,3,2019,5,7,5,20,22*0C<CR><LF>`

6.2.3 PMTK741 Reference Location

Description:

Transmits the reference location to the GNSS module. The deviation of the reference location should be less than 30 kilometers. The deviation of the reference UTC time should be less than 3 seconds.

Data Field:

`PMTK741,Latitude,Longitude,Altitude,YYYY,MM,DD,hh,mm,ss`

- **Latitude:** WGS84 geodetic latitude (unit: decimal degree, 6-decimal places; range: -90 to +90; +: North; -: South)
- **Longitude:** WGS84 geodetic longitude (unit: decimal degree, 6-decimal places; range: -180 to +180; +: West; -: East)
- **Altitude:** WGS84 ellipsoidal altitude (unit: meter, 6-decimal places)
- **YYYY:** Year in 4 digits
- **MM:** Month (range: 1 – 12)
- **DD:** Day (range: 1 – 31)
- **hh:** Hour (range: 0 – 23)
- **mm:** Minute (range: 0 – 59)
- **ss:** Second (range: 0 – 59)

Example:

Command	Response
`$PMTK741,23.096162,120.283768,44.81,2019, 5,7,5,20,22*3C<CR><LF>`	`$PMTK001,741,3,23.096162,120.283768,44.810000, 2019,5,7,5,20,22*3E<CR><LF>`

7 Flash Aiding

Flash aiding stores EPO data into the flash memory of the GNSS module. EPO data stored in the module won't be cleared when the module restarts.

The maximum period of EPO data can be stored in the GNSS module is 14 days for GPS-only EPO file, and 7 days for GPS+GLONASS EPO file. If a 30-day GPS-only EPO file is sent, only the first 14 days' EPO data will be stored. If a 30-day GPS+GLONASS EPO file is sent, only the first 7 days' EPO data will be stored.

7.1 EPO Binary Transfer Protocol

EPO data are packeted in EPO packets using MTK Binary Protocol and then transferred from the host to the GNSS module. At the beginning of the protocol, the host have to split the EPO file and encapsulate it into several EPO packets, and give each EPO packet a sequence number starting from zero. The sequence number is in order to make sure the EPO packets are transferred in the correct order and that no packet is missing. The host and the GNSS module then follow the EPO Data Transfer Protocol to transfer EPO data into the GNSS module.

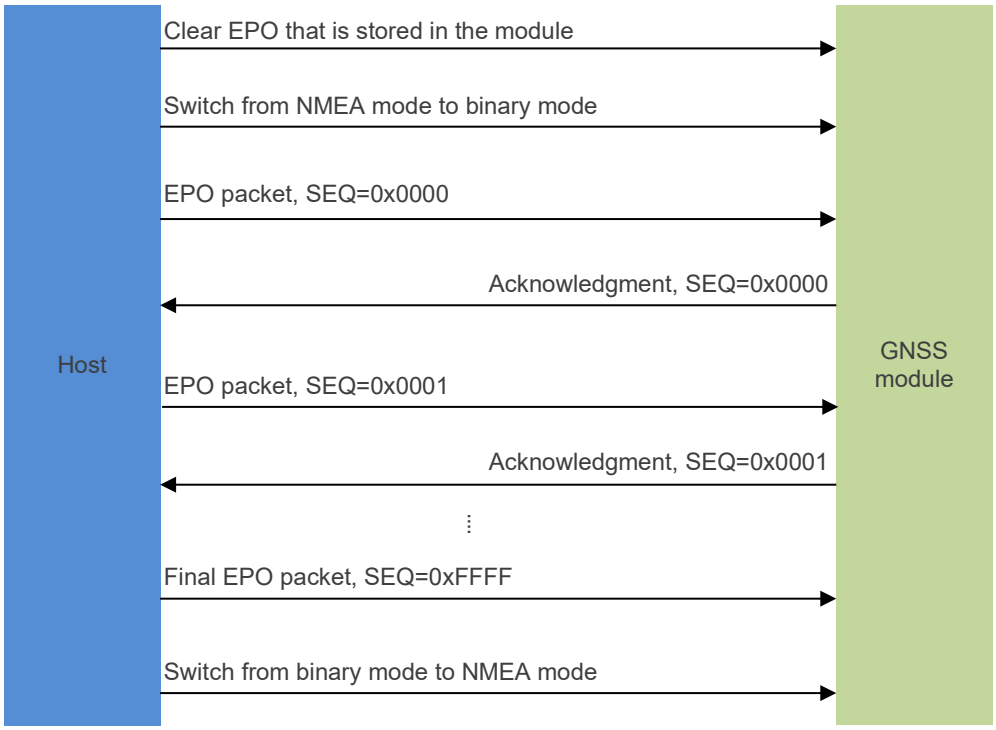

Figure 5. Sequence of Binary EPO Data Transfer

1. **Host:** Clear EPO that stored in GNSS module via PMTK127.
2. **Host:** Switch from NMEA mode to binary mode via PMTK253.
3. **Host:** Send one EPO packet, which contains 1 – 3 SAT data to the GNSS module. The sequence number in the packet starts from zero and will increase by one for each of the following EPO packets.
4. **GNSS module:** Receive EPO packet from host. Verify validity of EPO data in the packet. The GNSS module will then return an acknowledgment to indicate success or not.
5. **Host:** If the acknowledgment indicates success, the host prepares to send the next EPO packet; otherwise, exit the protocol.
6. Repeat steps 3 – 5 until all EPO data are transferred.
7. **Host:** Send a final EPO packet which contains sequence number 0xFFFF to indicate the end of the protocol. The 3 SAT data fields in the final EPO packet are filled with 0x00.
8. **Host:** Switch from binary mode to NMEA mode via PMTK253.

For GPS+GLONASS EPO files, send all SAT data of one SET before sending another SET. Otherwise, the GNSS receiver may receive data incorrectly.

Specifically, the right way of sending is: Send SET1's SAT1–SAT32 → Send SET1's SAT65–SAT88 → Send SET2's SAT1–SAT32 → Send SET2's SAT65–SAT88 → Send SET3's SAT1–SAT32, etc.

It is incorrect to send as follows: Send SET1's SAT1–SAT32 → Send SET2's SAT1–SAT32 → Send SET3's SAT1–SAT32 → … → Send SET1's SAT65–SAT88 → Send SET2's SAT65–SAT88, etc.

7.2 Completing the Transmission

If you want to check the EPO data is stored in the module, execute **$PMTK607*33<CR><LF>** to query EPO status.

The module will then return the EPO status acknowledgment. For example:

$PMTK707,12,2053,432000,2054,64800,2053,435600,2053,457200*25<CR><LF>

Note that:

1. EPO sets in the module are 12 for 3 days (28 for 7 days, 56 for 14 days).
2. GPS week and GPS time of week which the EPO set starts from (from the example above: 2053, 432000) needs to be converted to UTC.
3. GPS week and GPS time of week which the EPO set ends (from the example above: 2054, 64800) needs to be converted to UTC.

7.3 Error Handling

If any problem occurs in the protocol, you should stop the process and restart the protocol again. Every time the protocol starts, the EPO sequence number should be reset to zero to indicate to the GNSS module that a new transferring process has begun. The GNSS module then needs to prepare for the new process.

The interval of time between two continuous EPO packets should not be longer than 10 seconds. Otherwise, the GNSS module will determine that a problem has occurred and terminate the process.

7.4 EPO Packet

The command for transmitting EPO data to the GNSS module in flash aiding is in binary mode.

7.4.1 MTK Binary Protocol

The MTK binary protocol is as follows:

Preamble		Length	Command ID	Data	Checksum	End Word	
0x04	0x24					0x0D	0x0A
2 bytes		2 bytes	2 bytes	Variable	1 byte	2 bytes	

- Use one-byte alignment
- Use little endian
- Preamble (2-byte word): 0x2404
- Length: Total number of bytes in the packet from Preamble to End Word
- Maximum packet size: 256 bytes
- Packet type:
  - (1) 0 – 999: conform to PMTK ASCII Protocol
  - (2) 1000 – 65535: Reserved
- Checksum: The checksum is calculated by 8-bit XOR-ing all bytes between “Preamble” and “Checksum” (not including “Preamble” and “Checksum”).
- End Word (2-byte word): 0x0A0D

7.4.2 EPO Binary Packet Format

The EPO version 1 packet format is as follows:

Preamble		Length	Packet Type	Data Field				Checksum	End Word	
0x04	0x24	0x00BF	0x02D2	EPO SEQ	SAT data	SAT data	SAT data		0x0D	0x0A
2 bytes	2 bytes	2 bytes	2 bytes	2 bytes	60 bytes	60 bytes	60 bytes	1 byte	2 bytes	

- The packet type for transmitting EPO version 1 data to the GNSS module is 722.
- The EPO sequence of a packet is used for synchronization of EPO packets in EPO binary transfer protocol.
- The length of a SAT data is 60 in EPO version 1.
- The max number of SAT data is 3. If the number of SAT data to be sent is less than 3, you should fill the rest of SAT data with 0x00.
- The length of a packet is 191 bytes.

The EPO version 2 packet format is as follows:

Preamble		Length	Packet Type	Data Field				Checksum	End Word	
0x04	0x24	0x00E3	0x02D3	EPO SEQ	SAT data	SAT data	SAT data		0x0D	0x0A
2 bytes	2 bytes	2 bytes	2 bytes	2 bytes	72 bytes	72 bytes	72 bytes	1 byte	2 bytes	

- The packet type for transmitting EPO version 2 data to the GNSS module is 723.
- The EPO sequence of a packet is used for synchronization of EPO packets in EPO binary transfer protocol.
- The length of a SAT data is 72 in EPO version 2.
- The max number of SAT data is 3. If the number of SAT data to be sent is less than 3, you should fill the rest of SAT data with 0x00.
- The length of a packet is 227 bytes.

The GNSS module will send an acknowledgment when receiving an EPO packet. The result should be checked before sending the next EPO packet.

The acknowledgment format is as follows:

Preamble		Length	Packet Type	Data Field		Checksum	End Word	
0x04	0x24	0x000C	0x0002	EPO SEQ	Result		0x0D	0x0A
2 bytes		2 bytes	2 bytes	2 bytes	1 byte	1 byte	2 bytes	

- The packet type for acknowledgment is 2.
- The packet is used for EPO version 1 and 2 packet acknowledgment.
- The EPO sequence of a packet is used for synchronization of EPO packets in EPO binary transfer protocol.
- The length of a packet is 12 bytes.
- The result is used for indicating whether the received EPO packet is valid. Possible values are as follows:
  - "0" = the received EPO packet is invalid.
  - "1" = the received EPO packet is valid.

8 Appendix

8.1 PMTK Commands

8.1.1 PMTK253 Binary Mode

Description:

Enables MTK binary protocol and switches to binary mode.

Data Field:

`PMTK253,Enabled`

- **Enabled:**
  - '0' = Disable; switches to NMEA mode
  - '1' = Enable; switches to binary mode

Example:

Command	Response
`$PMTK253,1*2B<CR><LF>`	None

8.2 MTK Binary Protocols

8.2.1 PMTK001 Acknowledgment

Decription:

PMTK command acknowledgment.

Packet Length:

12 bytes

Format:

Preamble		Length	Packet Type	Data Field		Checksum	End Word	
0x04	0x24	0x000C	0x0001	Cmd	Flag		0x0D	0x0A
2 bytes		2 bytes	2 bytes	2 bytes	1 byte	1 byte	2 bytes	

Data Field:

- **Cmd:** The command / packet type that the acknowledgment corresponds to
- **Flag:**
  - '0' = Invalid command /packet
  - '1' = Unsupported command / packet type
  - '2' = Valid command / packet, but action failed
  - '3' = Valid command / packet, and action succeeded

Example:

`0x04 0x24 0x0C 0x00 0x01 0x00 0xFD 0x00 0x03 0xF3 0x0D 0x0A`

Preamble		Length	Packet Type	Data Field		Checksum	End Word	
0x04	0x24	0x000C	0x0001	0x00FD	0x03	0xF3	0x0D	0x0A

Preamble	Length	Packet Type	Data Field		Checksum	End Word
2 bytes	2 bytes	2 bytes	2 bytes	1 byte	1 byte	2 bytes

8.2.2 PMTK253 Binary Mode

Description:

Enables MTK binary protocol and switches to binary mode.

Packet Length:

14 bytes

Format:

Preamble		Length	Packet Type	Data Field		Checksum	End Word	
0x04	0x24	0x000E	0x00FD	Enabled	Baud rate		0x0D	0x0A
2 bytes		2 bytes	2 bytes	1 byte	4 bytes	1 byte	2 bytes	

Data Field:

- **Enabled:**
  - '0' = Disable; switches to NMEA mode
  - '1' = Enable; switches to binary mode
- **Baud rate:** Possible baud rate values: 4800, 9600, 14400, 19200, 38400, 57600, 115200, 460800, 921600
  - 0 = Default baud rate

Example:

```
0x04 0x24 0x0E 0x00 0xFD 0x00 0x00 0x00 0x00 0x00 0x00 0xF3 0x0D 0x0A
```

Preamble		Length	Packet Type	Data Field		Checksum	End Word	
0x04	0x24	0x000E	0x00FD	0x00	0x00000000	0xF3	0x0D	0x0A
2 bytes		2 bytes	2 bytes	1 byte	4 bytes	1 byte	2 bytes	

9 Reference Documents

[1] AirPrime GMM-G3, XA11xx and XM11xx Software User Guide
Reference number: 41111121

[2] AirPrime XA1110 and XM1110 Hardware Design Guide
Reference number: 41111116

[3] AirPrime XM1110 Product Technical Specification
Reference number: 41111059

10 Support

For direct clients: contact your Sierra Wireless FAE

For distributor clients: contact your distributor FAE

For distributors: contact your Sierra Wireless FAE

11 Document History

Level	Date	History
1.0	February 26, 2018	Creation
1.1	March 21, 2018	Updated Table 4: Message 723 Packet Format
2.0	May 20, 2019	Added: • 6 Host Aiding • 7 Flash Aiding • 8 Appendix
		Deleted: • UTC • Current Position
2.1	May 27, 2019	Updated: • 5.2 EPO Version 2 • 6.2.1 PMTK721 Transmit EPO Data
2.2	June 03, 2019	Fixed table formatting in section 7.4.2 EPO Binary Packet Format

12 Legal Notice