MC60 GNSS

Protocol Specification

GSM/GPRS/GNSS Module Series

Rev. MC60_GNSS_Protocol_Specification_V1.0

Date: 2016-07-22

Our aim is to provide customers with timely and comprehensive service. For any assistance, please contact our company headquarters:

Quectel Wireless Solutions Co., Ltd.
Office 501, Building 13, No.99, Tianzhou Road, Shanghai, China, 200233
Tel: +86 21 5108 6236
Email: info@quectel.com

Or our local office. For more information, please visit:
http://www.quectel.com/support/salesupport.aspx

For technical support, or to report documentation errors, please visit:
http://www.quectel.com/support/techsupport.aspx
Or email to: Support@quectel.com

GENERAL NOTES

QUECTEL OFFERS THE INFORMATION AS A SERVICE TO ITS CUSTOMERS. THE INFORMATION PROVIDED IS BASED UPON CUSTOMERS' REQUIREMENTS. QUECTEL MAKES EVERY EFFORT TO ENSURE THE QUALITY OF THE INFORMATION IT MAKES AVAILABLE. QUECTEL DOES NOT MAKE ANY WARRANTY AS TO THE INFORMATION CONTAINED HEREIN, AND DOES NOT ACCEPT ANY LIABILITY FOR ANY INJURY, LOSS OR DAMAGE OF ANY KIND INCURRED BY USE OF OR RELIANCE UPON THE INFORMATION. THE INFORMATION SUPPLIED HEREIN IS SUBJECT TO CHANGE WITHOUT PRIOR NOTICE.

COPYRIGHT

About the Document

History

Revision	Date	Author	Description
1.0	2016-07-22	Ziv LIAO	Initial

Contents

Table Index

1 Introduction

MC60 is a multi-purpose module which integrates a high performance GNSS engine and a quad-band GSM/GPRS engine. The GNSS engine is a single receiver integrating GPS and GLONASS systems. It supports multiple positioning and navigation systems including autonomous GPS, GLONASS, SBAS (including WAAS, EGNOS, MSAS and GAGAN), and QZSS. With the embedded GNSS function, MC20 can help customers get accurate coordinates, high-precision time, etc., and thus is ideal for use in wearable devices, vehicle and personnel tracking, and more fields.

This document describes the software aspects of MC60. MC60 supports NMEA 0183 standard commands, and also can be controlled and configured via MTK NMEA extended packet.

1.1. Differences between Two Application Modes of MC60

MC60' internal GSM and GNSS engines can work as a whole unit (**all-in-one** solution) or work independently (**stand-alone** solution) according to customer demands.

In all-in-one solution, the MC60 works as a whole unit. The GNSS Part can be regarded as a peripheral of the GSM Part. This allows for convenient communication between GSM and GNSS Parts, such as AT command sending for GNSS control, GNSS part firmware upgrading, and EPO data download.

In stand-alone solution, GSM and GNSS Parts work independently, and thus have to be controlled separately.

When working in all-in-one or stand-alone solution, there are some differences for MC60 to acquire NMEA output data, or send PMTK/SDK command. The details are listed below.

Table 1: Differences between All-in-one Solution and Stand-alone Solution

Item	All-in-one	Stand-alone
NMEA Output Data Acquisition	Acquire via sending **AT+QGNSSRD** command	Acquire directly
PMTK Command Sending	Send via **AT+QGNSSCMD** command	Send directly
SDK Command Sending	Supported	Send directly

NOTES

1. MC60 CS version does not support SDK commands in all-in-one solution, but MP version will support.
2. AT commands are effective only when the module is in all-in-one solution.

2 Standard NMEA Packet Protocol

MC60 supports standard NMEA 0183 messages, and the following tables show the structure of these messages.

2.1. --RMC

RMC-Recommended Minimum Position Data (including position, velocity and time).

Example: $GPRMC,015606.000,A,3150.7584,N,11712.0491,E,0.00,231.36,280715,,,A*67<CR><LF> $GNRMC,084629.000,A,3150.7822,N,11711.9323,E,0.00,119.00,240715,,,D*7C<CR><LF> $BDRMC,020547.000,A,3150.7813,N,11711.9212,E,0.37,229.71,280715,,,A*7C<CR><LF>	
Field	**Description**
$	Each NMEA message starts with '$'
--RMC	Message ID
UTC Time	Time in format 'hhmmss.sss'
Data Valid	'V'=Invalid 'A'=Valid
Latitude	Latitude in format 'ddmm.mmmm' (degrees and minutes)
N/S	'N'=North 'S'=South
Longitude	Longitude in format 'ddmm.mmmm' (degrees and minutes)
E/W	'E'=East 'W'=West
Speed	Speed over ground in knots
COG	Course over ground in degree
Date	Date in format 'ddmmyy'
Magnetic Variation	Magnetic variation in degree, not being output

E/W	Magnetic variation E/W indicator, not being output
Positioning Mode	'N'=No fix 'A'=Autonomous GNSS fix 'D'=Differential GNSS fix
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

NOTE

For the details, please refer to the Notes of ***Chapter 4.24***.

2.2. --VTG

VTG-Track Made Good and Ground Speed.

Example:
$GPVTG,227.15,T,,M,0.00,N,0.00,K,A*3E<CR><LF>
$GNVTG,19.11,T,,M,0.16,N,0.30,K,A*1F<CR><LF>
$BDVTG,229.71,T,,M,0.37,N,0.68,K,A*29<CR><LF>

Field	Description
$	Each NMEA message starts with '$'
--VTG	Message ID
COG (T)	Course over ground (true) in degree
T	Fixed field, true
COG(M)	Course over ground (magnetic), not being output
M	Fixed field, magnetic
Speed	Speed over ground in knots
N	Fixed field, knots
Speed	Speed over ground in km/h

K	Fixed field, km/h
Positioning Mode	'N'=No fix 'A'=Autonomous GNSS fix 'D'=Differential GNSS fix
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

NOTE

For the details, please refer to the Notes of ***Chapter 4.24***.

2.3. --GGA

GGA-Global Positioning System Fix Data, is the essential fix data which provides 3D location and accuracy data.

Example:
$GPGGA,015606.000,3150.7584,N,11712.0491,E,1,5,2.28,265.0,M,0.0,M,,*65<CR><LF>
$GNGGA,083354.000,3150.7790,N,11711.9289,E,1,8,2.85,53.2,M,0.0,M,,*4B<CR><LF>
$BDGGA,020547.000,3150.7813,N,11711.9212,E,1,3,3.65,55.3,M,0.0,M,,*4C<CR><LF>

Field	Description
$	Each NMEA message starts with '$'
--GGA	Message ID
UTC Time	Time in format 'hhmmss.sss'
Latitude	Latitude in format 'ddmm.mmmm' (degrees and minutes)
N/S	'N'=North 'S'=South
Longitude	Longitude in format 'ddmm.mmmm' (degrees and minutes)
E/W	'E'=East 'W'=West
Fix Status	'0'=Invalid '1'=GNSS fix

	'2'=DGPS fix '6'=Estimated (dead reckoning) Mode
Number of SV	Number of satellites being used (0~24)
HDOP	Horizontal Dilution of Precision
Altitude	Altitude in meters according to WGS84 ellipsoid
M	Fixed field, meter
Geoid Separation	Height of Geoid (means sea level) above WGS84 ellipsoid, meter
M	Fixed field, meter
DGPS Age	Age of DGPS data in seconds, empty if DGPS is not used
DGPS Station ID	DGPS station ID, empty if DGPS is not used
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

NOTE

For the details, please refer to the Notes of ***Chapter 4.24***.

2.4. --GSA

GSA-GNSS DOP and Active Satellites, which provides details on the fix and includes the number of satellites being used in the current solution and the DOP. At most the first 12 satellite IDs are output.

Example:
$GPGSA,A,3,03,17,11,23,193,,,,,,,,3.72,2.85,2.39*3C<CR><LF>
$GNGSA,A,3,23,09,17,03,01,193,,,,,,,1.23,0.74,0.99*28<CR><LF>
$BDGSA,A,2,10,11,07,,,,,,,,,,3.79,3.65,1.00*18<CR><LF>

Field	Description
$	Each NMEA message starts with '$'
--GSA	Message ID

Mode	Auto selection of 2D or 3D fix 'M'=Manual, forced to switch 2D/3D mode 'A'=Allowed to automatically switch 2D/3D mode
Fix Status	'1'=No fix '2'=2D fix '3'=3D fix
Satellite Used 1	Satellite used on channel 1
Satellite Used 2	Satellite used on channel 2
Satellite Used 3	Satellite used on channel 3
Satellite Used 4	Satellite used on channel 4
Satellite Used 5	Satellite used on channel 5
Satellite Used 6	Satellite used on channel 6
Satellite Used 7	Satellite used on channel 7
Satellite Used 8	Satellite used on channel 8
Satellite Used 9	Satellite used on channel 9
Satellite Used 10	Satellite used on channel 10
Satellite Used 11	Satellite used on channel 11
Satellite Used 12	Satellite used on channel 12
PDOP	Position Dilution of Precision
HDOP	Horizontal Dilution of Precision
VDOP	Vertical Dilution of Precision
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

NOTE

For the details, please refer to the Notes of ***Chapter 4.24***.

2.5. --GSV

GSV-GNSS Satellites in View. One GSV sentence can only provide data for at most 4 satellites, so several sentences might be required for full information. Since GSV includes satellites that are not used as part of the solution, GSV sentence contains more satellites than GGA does.

Example:
$GPGSV,3,1,11,193,69,099,30,17,62,354,36,06,47,272,,03,40,054,30*4E<CR><LF>
$GPGSV,3,2,11,02,13,255,,01,12,055,19,23,11,102,25,11,05,074,24*75<CR><LF>
$GPGSV,3,3,11,24,03,303,,47,,,,32,,,21*4D<CR><LF>
$GLGSV,3,1,11,69,48,142,39,68,43,058,51,83,40,049,51,84,40,334,43*64<CR><LF>
$GLGSV,3,2,11,74,30,271,15,73,17,218,19,75,13,324,30,70,07,184,*6E<CR><LF>
$GLGSV,3,3,11,85,06,296,34,82,02,092,21,67,02,023,*56<CR><LF>
$BDGSV,2,1,06,08,63,015,30,11,59,350,32,12,39,081,26,14,29,184,21*68<CR><LF>
$BDGSV,2,2,06,13,07,295,,01,,,26*50<CR><LF>

Field	Description
$	Each NMEA message starts with '$'
--GSV	Message ID
Number of Message	Number of messages, total number of GPGSV messages being output (1~3)
Sequence Number	Sequence number of this entry (1~3)
Satellites in View	Total satellites in view
Satellite ID 1	Satellite ID
Elevation 1	Elevation in degree (0~90)
Azimuth 1	Azimuth in degree (0~359)
SNR 1	Signal to Noise Ration in dB-Hz (0~99), empty if not tracking
Satellite ID 2	Satellite ID
Elevation 2	Elevation in degree (0~90)
Azimuth 2	Azimuth in degree (0~359)
SNR 2	Signal to Noise Ration in dB-Hz (0~99), empty if not tracking
Satellite ID 3	Satellite ID
Elevation 3	Elevation in degree (0~90)

Azimuth 3	Azimuth in degree (0~359)
SNR 3	Signal to Noise Ration in dB-Hz (0~99), empty if not tracking
Satellite ID 4	Satellite ID
Elevation 4	Elevation in degree (0~90)
Azimuth 4	Azimuth in degree (0~359)
SNR 4	Signal to Noise Ration in dB-Hz (0~99), empty if not tracking
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

NOTE

If the receiver is fixed by multi-GNSS, it will output GPGSV sentences of GPS satellites, GLGSV sentences of GLONASS satellites. For the details, please refer to the Notes of ***Chapter 4.24***.

2.6. --GLL

GLL-Geographic Latitude and Longitude, which contains position information, time of position fix and status.

Example:
$GPGLL,3150.7584,N,11712.0491,E,015606.000,A,A*5C<CR><LF>
$GNGLL,3150.7790,N,11711.9289,E,083354.000,A,A*4D<CR><LF>
$BDGLL,3150.7813,N,11711.9212,E,020547.000,A,A*49<CR><LF>

Field	Description
$	Each NMEA message starts with '$'
--GLL	Message ID
Latitude	Latitude in format 'ddmm.mmmm' (degrees and minutes)
N/S	'N'=North 'S'=South
Longitude	Longitude in format 'dddmm.mmmm' (degrees and minutes)

E/W	‘E’=East ‘W’=West
UTC Time	Time in format ‘hhmmss.sss’
Data Valid	‘V’=Invalid ‘A’=Valid
Positioning Mode	‘N’=No fix ‘A’=Autonomous GNSS fix ‘D’=Differential GNSS fix
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

NOTE

For the details, please refer to the Notes of ***Chapter 4.24***.

3 AT Commands for MC60 GNSS

The commands below are used to control or configure MC60's internal GNSS engine. These commands are effective only in all-in-one solution.

Table 2: Overview of AT Commands for MC60 GNSS

Command	Description
AT+QGNSSC	GNSS module power control
AT+QGNSSRD	Read GNSS navigation information
AT+QGNSSCMD	Send commands to GNSS module

3.1. AT+QGNSSC GNSS Module Power Control

The command is used to control the power supply of GNSS module.

AT+QGNSSC GNSS Module Power Control	
Test Command **AT+QGNSSC=?**	Response **+QGNSSC: (list of supported <mode>s)** **OK**
Read Command **AT+QGNSSC?**	Response **+QGNSSC: <mode>** **OK**
Write Command **AT+QGNSSC=<mode>**	Response **OK** If error is related to ME functionality: **+CME ERROR: <err>**

Parameter

<mode>	0	Power off GNSS module
	1	Power on GNSS module

Example

```
AT+QGNSSC?                  // Query GNSS power status
+QGNSSC: 0                  // GNSS powered off

OK
AT+QGNSSC=1                 // Power on GNSS

OK
```

NOTE

In stand-alone solution, the power supply of GNSS is controlled by an external circuit rather than the PIN GPS_VCC_EN. In such case, command **AT+QGNSSC** cannot be used and thus can be ignored.

3.2. AT+QGNSSRD Read GNSS Navigation Information

The command is used to read the GNSS navigation information.

AT+QGNSSRD Read GNSS Navigation Information	
Test Command **AT+QGNSSRD=?**	Response **+QGNSSRD: (list of supported <item>s)** **OK**
Read Command **AT+QGNSSRD?**	Response **+QGNSSRD: (information of all supported <item>s)** **OK**
Write Command **AT+QGNSSRD=<item>**	Response **+QGNSSRD: (information of <item>)** **OK** If error is related to ME functionality: **+CME ERROR: <err>**

Parameter

<item>	“NMEA/GGA”: Get GGA sentence “NMEA/GLL”: Get GLL sentence “NMEA/GSA”: Get GSA sentence “NMEA/GSV”: Get GSV sentence “NMEA/RMC”: Get RMC sentence “NMEA/VTG”: Get VTG sentence

Example

```
AT+QGNSSRD?                          // Inquire GNSS NMEA sentence
+QGNSSRD: $GNRMC,034035.000,A,3150.8617,N,11711.9038,E,3.02,183.45,240516,,,A*75
$GNVTG,183.45,T,,M,3.02,N,5.59,K,A*20
$GNGGA,034035.000,3150.8617,N,11711.9038,E,1,4,1.50,40.9,M,0.0,M,,*44
$GPGSA,A,3,26,21,,,,,,,,,,,1.75,1.50,0.91*0A
$GLGSA,A,3,82,70,,,,,,,,,,,1.75,1.50,0.91*1C
$GPGSV,3,1,12,16,67,308,,26,58,021,16,23,40,307,,31,40,088,*7F
$GPGSV,3,2,12,08,17,199,,09,14,320,,21,10,086,14,14,10,153,*73
$GPGSV,3,3,12,22,09,226,,193,06,165,,32,03,154,,29,01,034,*45
$GLGSV,3,1,09,81,44,073,,79,40,041,,82,38,145,15,80,36,323,*66
$GLGSV,3,2,09,70,30,290,16,69,26,225,,78,12,078,,88,09,027,*64
$GLGSV,3,3,09,71,05,334,*5B
$GNGLL,3150.8617,N,11711.9038,E,034035.000,A,A*4C

OK
AT+QGNSSRD=”NMEA/RMC”                // Inquire RMC information
+QGNSSRD: $GNRMC,034036.000,A,3150.8612,N,11711.9045,E,2.74,178.00,240516,,,A*7C

OK
AT+QGNSSRD=”NMEA/GSA”                // Inquire GSA information
+QGNSSRD: $GPGSA,A,3,26,21,,,,,,,,,,,1.76,1.50,0.91*09

OK
AT+QGNSSRD?                          // Inquire GNSS NMEA sentence
+QGNSSRD: $GNRMC,034039.000,A,3150.8596,N,11711.9049,E,2.13,194.12,240516,,,A*70
$GNVTG,194.12,T,,M,2.13,N,3.95,K,A*23
$GNGGA,034039.000,3150.8596,N,11711.9049,E,1,5,1.50,38.7,M,0.0,M,,*44
$GPGSA,A,3,22,26,21,,,,,,,,,,,1.75,1.50,0.91*0A
$GLGSA,A,3,82,70,,,,,,,,,,,1.75,1.50,0.91*1C
$GPGSV,3,1,12,16,67,308,,26,58,021,17,23,40,307,,31,40,088,*7E
$GPGSV,3,2,12,08,17,199,,09,14,320,,21,10,086,12,14,10,153,*75
$GPGSV,3,3,12,22,09,226,16,193,06,165,,32,03,154,,29,01,034,*42
$GLGSV,3,1,09,81,44,073,,79,40,041,,82,38,145,16,80,36,323,*65
```

```
$GLGSV,3,2,09,70,30,290,16,69,26,225,,78,12,078,,88,09,027,*64
$GLGSV,3,3,09,71,05,334,*5B
$GNGLL,3150.8596,N,11711.9049,E,034039.000,A,A*4C

OK
```

3.3. AT+QGNSSCMD Send Commands to GNSS Module

The command is used to send commands to GNSS module, which allows customers to optionally use some functions to meet application demands.

AT+QGNSSCMD Send Commands to GNSS Module	
Test Command **AT+QGNSSCMD=?**	Response **+QGNSSCMD: (0,1),"cmdString"** **OK**
Write Command **AT+QGNSSCMD=<cmdType>,<cmdString>**	Response **OK** If error is related to ME functionality: **+CME ERROR: <err>**

Parameter

<cmdType>	0	NMEA style command
	1	Hex style command
<cmdString>	Command string	

Example

```
AT+QGNSSCMD=0,"$PMTK605*31"          // Inquire GNSS version information
OK

+QGNSSCMD: $PMTK705,AXN_3.82_3333_16051101,0002,MC60-GNSS,1.0*2D
```

NOTE

Currently only **<cmdType>**=0 is supported.

4 MTK NMEA Packet Protocol

This chapter introduces the MTK NMEA packet protocol, which is a set of extension messages of standard NMEA packet protocol. These messages are used to control and configure MC60's internal GNSS engine. The following tables show the structure of MTK NMEA packet.

4.1. Packet Type: 010 PMTK_SYS_MSG

This message is used to automatically output system messages through GNSS module.

Data Field: None Example: $PMTK010,001*2E<CR><LF>	
Field	**Description**
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	010
Message	System message '0'=Unknown '1'=Startup '2'=Notification for the host aiding EPO '3'=Notification for the transition to normal mode is successfully done
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.2. Packet Type: 011 PMTK_TXT_MSG

This message is used to automatically output system messages through GNSS module.

Data Field: None Example: $PMTK011,MTKGPS*08<CR><LF>	
Field	**Description**
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	011
Message	MTKGPS
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.3. Packet Type: 001 PMTK_ACK

Acknowledgement of PMTK Command. In order to inform the sender whether the receiver has received the packet, and an acknowledge packet PMTK_ACK should be returned after the receiver receives a packet.

Some commands will cause the GNSS module to restart or change the baud rate. There is no PMTK_ACK for those commands as listed below.

- PMTK_CMD_HOT_START
- PMTK_CMD_WARM_START
- PMTK_CMD_COLD_START
- PMTK_CMD_FULL_COLD_START
- PMTK_SET_NMEA_BAUDRATE

Data Field: $PMTK001,Cmd,Flag Example:

$PMTK001,869,3*37<CR><LF>	
Field	**Description**
$	Each NMEA message starts with ‘$’
PMTK	MTK proprietary message
Packet Type	001
Cmd	The packet type that the acknowledge responds
Flag	‘0’=Invalid packet ‘1’=Unsupported packet type ‘2’=Valid packet, but action failed ‘3’=Valid packet, action succeeded
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.4. Packet Type: 101 PMTK_CMD_HOT_START

This message is used to achieve the hot start of the GNSS module (use all available data in the NV store). Normally hot start means the GNSS module was powered down for less than 3 hours (RTC must be alive) and its ephemeris is still valid. As there is no need for downloading ephemeris, it is the fastest startup method.

Data Field: None Example: $PMTK101*32<CR><LF>	
Field	**Description**
$	Each NMEA message starts with ‘$’
PMTK	MTK proprietary message
Packet Type	101
*	End character of data field
Checksum	Hexadecimal checksum

<CR><LF>	Each of the messages

4.5. Packet Type: 102 PMTK_CMD_WARM_START

This message is used to achieve the warm start of the GNSS module. Warm start means the GNSS module has approximate information on time, position and coarse data of satellite positions. But it needs to download ephemeris until it can get a fix. Using this message will force the GNSS warm restart without using the ephemeris data in NV.

Data Field: None Example: $PMTK102*31<CR><LF>	
Field	**Description**
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	102
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.6. Packet Type: 103 PMTK_CMD_COLD_START

This message is used to achieve cold start of the GNSS module. Using this message will force the GNSS cold restart without using any prior location information, which includes time, position, almanacs and ephemeris data.

Data Field: None Example: $PMTK103*30<CR><LF>	
Field	**Description**

$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	103
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.7. Packet Type: 104 PMTK_CMD_FULL_COLD_START

This message is essentially a cold restart, but additionally clear system and user configuration at re-start. That is, reset the GNSS module to the factory status. Full cold start means the GNSS module has no information on last location. It needs to search the full time and frequency space, and also all possible satellite numbers before it can get a fix.

Data Field: None Example: $PMTK104*37<CR><LF>	
Field	**Description**
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	104
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.8. Packet Type: 161 PMTK_CMD_STANDBY_MODE

This message is used to enter into standby mode for power saving.

Data Field: $PMTK161,Type Example: $PMTK161,0*28<CR><LF> Response: $PMTK001,161,3*36<CR><LF>	
Field	**Description**
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	161
Type	'0'=Stop mode
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.9. Packet Type: 183 PMTK_LOCUS_QUERY_STATUS

This message is used to query logging status and is not supported in all-in-one solution.

Data Field: None Example: $PMTK183*38<CR><LF> Response: $PMTK001,183,3*3A<CR><LF>	
Field	**Description**
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	183
*	End character of data field
Checksum	Hexadecimal checksum

<CR><LF>	Each of the messages

Return:

Example: $PMTKLOG,456,0,11,31,2,0,0,0,3769,46*48<CR><LF>	
Field	Description
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	LOG
Serial#	Logging serial number: 0~65535
Type	Logging type-0: Overlap, 1: Fullstop
Mode	Logging mode-0x08: Interval logger
Content	Logging contents of configuration
Interval	Logging interval setting (valid when interval mode is selected)
Distance	Logging distance setting (valid when distance mode is selected)
Speed	Logging speed setting (valid when speed mode is selected)
Status	Logging status-1: Stop logging, 0: Logging
Number	Logging number of data record
Percent	Logging life used percentage (0%~100%)
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.10. Packet Type: 184 PMTK_LOCUS_ERASE_FLASH

This message is used to erase logger flash and is not supported in all-in-one solution.

Data Field: $PMTK184,Type Example: $PMTK184,1*22<CR><LF> Response: $PMTK001,184,3*3D<CR><LF>	
Field	**Description**
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	184
Type	'1'=Erase all logger internal flash data
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.11. Packet Type: 185 PMTK_LOCUS_STOP_LOGGER

This message is used to stop or start logging data and is not supported in all-in-one solution.

Data Field: $PMTK185,Stutas Example: $PMTK185,1*23<CR><LF> Response: $PMTK001,185,3*3C<CR><LF>	
Field	**Description**
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	185
Status	'0'=Start logging '1'=Stop logging
*	End character of data field

Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.12. Packet Type: 622 PMTK_Q_LOCUS_DATA

This message is used to dump locus flash data and is not supported in all-in-one solution.

Data Field:
$PMTK622,Type
Example:
$PMTK622,1*29<CR><LF>
Response:
$PMTK001,622,3*36<CR><LF>

Field	Description
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	622
Type	'1'=Dump partial in used LOCUS flash data.
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.13. Packet Type: 220 PMTK_SET_POS_FIX

This message is used to set position fix interval.

Data Field:
$PMTK220, Interval
Example:
$PMTK220,1000*1F<CR><LF>
Response:
$PMTK001,220,3*30<CR><LF>

Field	Description
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	220
Interval	Position fix interval [msec]. [Range:100~10000]
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.14. Packet Type: 223 PMTK_SET_AL_DEE_CFG

This message is used to config DEE.

Data Field: $PMTK223,SV,SNR,Extension threshold,Extension gap Example: $PMTK223,1,30,180000,60000*3C<CR><LF> Response: $PMTK001,223,3*33<CR><LF>	
Field	**Description**
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	223
SV	Range: 1~4 (Default value: 1)
SNR	Range: 25~30 (Default value: 30)
Extension Threshold	Range: 40000~180000 (Default value: 180000)
Extension Gap	Range: 0~3600000 (Default value: 60000)
*	End character of data field

Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.15. Packet Type: 225 PMTK_SET_PERIODIC_MODE

This message is used to enter into periodic mode for power saving.

Data Field: $PMTK225,Type,Running time,Sleep time,Second running time,Second sleep time Example: $PMTK225,8*23<CR><LF> Response: $PMTK001,225,3*35<CR><LF>	
Field	**Description**
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	225
Type	'0'=Back to normal mode '1'=Periodic Backup mode '2'=Periodic Standby mode '4'=Perpetual Backup mode '8'=AlwaysLocate™ Standby mode '9'=AlwaysLocate™ Backup mode
Run Time	'0': Disable >='1000': Enable (Range: 1000~518400000)
Sleep Time	(Range: 1000~518400000)
Second Run Time	'0': Disable >='1000': Enable (Range: 1000~518400000)
Second Sleep Time	(Range: 1000~518400000)
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

NOTE

The unit of running time or sleep time is msec. The second running time should be more than the first running time when the first one is a non-zero value

4.16. Packet Type: 256 PMTK_SET_TIMING_PRODUCT

This message is used to enable or disable the timing of product mode. (Default off)

Data Field:
$PMTK256,Enable
Example:
$PMTK256,0*2F<CR><LF>
Response:
$PMTK001,256,3*31<CR><LF>

Field	Description
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	256
Enable	'0'=Disable '1'=Enable
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.17. Packet Type: 286 PMTK_SET_AIC_ENABLED

This message is used to enable or disable AIC function. It is suggested to set cold start command first and then PMTK command.

Data Field:
$PMTK286,Enable
Example:
$PMTK286,0*22<CR><LF>

Response: $PMTK001,286,3*3C<CR><LF>	
Field	**Description**
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	286
Enable	'0'=Disable '1'=Enable
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.18. Packet Type: 301 PMTK_API_SET_DGPS_MODE

This message is used to configure the source mode of DGPS correction data.

Data Field: $PMTK301,Mode Example: $PMTK301,2*2E<CR><LF> Response: $PMTK001,301,3*32<CR><LF>	
Field	**Description**
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	301
Mode	DGPS data source mode. '0'=No DGPS source '1'=RTCM(Not Support) '2'=SBAS (Including WAAS/EGNOS/GAGAN/MSAS)
*	End character of data field
Checksum	Hexadecimal checksum

<CR><LF>	Each of the messages

NOTE

Currently mode 1 is not supported.

4.19. Packet Type: 311 PMTK_API_SET_ELEV_MASK

This message is used to set satellite elevation mask.

Data Field:
$PMTK311,Type
Example:
$PMTK311,5*28<CR><LF>
Response:
$PMTK001,311,3*33<CR><LF>

Field	Description
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	311
Satellite Elevation Mask	(Range: 0~90°)
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

NOTE

The satellite elevation mask is recommended to be set not more than 10 degrees. As with the increase of satellite elevation mask, the number of satellites involved in positioning will decrease.

4.20. Packet Type: 313 PMTK_API_SET_SBAS_ENABLED

This message is used to enable or disable the searching of a SBAS satellite. SBAS (Satellite-Based Augmentation System) is a system that supports wide-area or regional augmentation through the use of geostationary satellite-broadcast messages. The geostationary satellite-broadcast GNSS integrity and correction data are composed of multiple ground stations which are located at accurately surveyed points.

Data Field: $PMTK313,Enable Example: $PMTK313,1*2E<CR><LF> Response: $PMTK001,313,3*31<CR><LF>	
Field	**Description**
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	313
Enable	'0'=Disable '1'=Enable
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.21. Packet Type: 314 PMTK_API_SET_NMEA_OUTPUT

This message is used to set NMEA sentence output frequencies. There are totally 19 data fields that present output frequencies for the 19 supported NMEA sentences individually.

Supported Frequency Settings:

0 - Disabled or not supported sentence
1 - Output once every one position fix
2 - Output once every two position fixes
3 - Output once every three position fixes
4 - Output once every four position fixes
5 - Output once every five position fixes

Data Field:
None
Example:
The module only output RMC once every one position fix.
$PMTK314,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0*29<CR><LF>
Response:
$PMTK001,314,3*36<CR><LF>

Field	Description
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	314
0 GLL	GLL interval - Geographic Position, Latitude and Longitude
1 RMC	RMC interval - Recommended Minimum Specific GNSS Sentence
2 VTG	VTG interval - Course Over Ground and Ground Speed
3 GGA	GGA interval - GNSS Fix Data
4 GSA	GSA interval - GNSS DOPS and Active Satellites
5 GSV	GSV interval - GNSS Satellites in View
6 GRS	GRS interval – GNSS Range Residuals
7 GST	GST interval – GNSS Pseudorange Error Statistics
8 Reserved	Always 0
9 Reserved	Always 0
10 Reserved	Always 0
11 Reserved	Always 0
12 Reserved	Always 0
13 Reserved	Always 0
14 Reserved	Always 0
15 Reserved	Always 0
16 Reserved	Always 0

17 ZDA	ZDA interval - Time and Date
18 MCHN	PMTKCHN interval - GNSS Channel Status
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

Use following messages to restore the system default setting:

Example:
$PMTK314,-1*04<CR><LF>

Field	Description
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	314
Restore	Always -1
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.22. Packet Type: 351 PMTK_API_SET_SUPPORT_QZSS_NMEA

The receiver supports new NMEA format for QZSS. The command allows users to enable or disable QZSS NMEA format. QZSS NMEA format is disabled by default and is not supported in all-in-one solution.

Data Field:
$PMTK351,Enable
Example:
$PMTK351,1*28<CR><LF>
Response:
$PMTK001,351,3*37<CR><LF>

Field	Description
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	351
QZSS_Enable	'0'=Disable '1'=Enable
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.23. Packet Type: 352 PMTK_API_SET_STOP_QZSS

QZSS is regional positioning service and is used to enable or disable QZSS function. QZSS function is enabled by default and is not supported in all-in-one solution.

Data Field: $PMTK352,Enable Example: $PMTK352,0*2A<CR><LF> Response: $PMTK001,352,3*34<CR><LF>	
Field	**Description**
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	352
QZSS_Enable	'0'=Enable '1'=Disable
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.24. Packet Type: 353 PMTK_API_SET_GNSS_SEARCH_MODE

This command is used to configure the receiver to start searching satellite system.

Data Field:
$PMTK353,GPS_Enable,GLONASS_Enable,GALILEO_Enable,GALILEO_FULL_Enable,BEIDOU_Enable
Example:
$PMTK353,1,1,0,0,0*2B<CR><LF>: Search GPS+GLONASS
Response:
$PMTK001,353,3,1,1,0,0,0,3*36<CR><LF>

Field	Description
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	353
GPS_Enable	'0'=Disable (DO NOT search GPS satellites) '1'or non-ZERO : search GPS satellites
GLONASS_Enable	'0'=Disable (DO NOT search GLONASS satellites) '1'or non-ZERO : search GLONASS satellites
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

NOTES

Actually GLONASS-only mode is only for testing purpose. Please use GPS+GLONASS in the real-application. In Europe, GPS+GLONASS will be used too. GLONASS can not be enabled at the same time.

When the firmware supports GPS and GLONASS systems, the NMEA Sentences output is as below:

1. If the receiver is fixed by GPS only, it will print GPRMC, GPVTG, GPGGA, GPGSA, GPGSV and GPGLL.
2. If the receiver is fixed by GPS only, and can also search QZSS satellite, it will print GPRMC, GPVTG, GPGGA, GPGSA, QZQSA, GPGSV, QZGSV and GPGLL.
3. If the receiver is fixed by GLONASS only, it will print GNRMC, GPVTG, GPGGA, GNGSA, GPGSV, GLGSV and GNGLL.
4. If the receiver is fixed by GPS and GLONASS, it will print GNRMC, GPVTG, GPGGA, GNGSA, GPGSV, GLGSV and GNGLL.

5. In the state of no satellite positioning, it will print initial state of NMEA, such as GPRMC, GPVTG, GPGGA, GPGSA, GPGSV and GPGLL. The time before satellite positioning after cold start, warm start or hot start belongs to this situation.

When the firmware supports GPS and BeiDou systems, the NMEA Sentences output is as below:

1. If the receiver is fixed by GPS only, it will print GPRMC, GPVTG, GPGGA, GPGSA, GPGSV and GPGLL.
2. In the state of no satellite positioning, it will print initial state of NMEA, such as GNRMC, GNVTG, GNGGA and GNGLL. The time before satellite positioning after cold start, warm start or hot start belongs to this situation.

When the firmware supports GPS, GLONASS systems, the NMEA Sentences output is as below:
If the receiver is fixed by GPS, GLONASS and GALILEO, it will print GNGGA, GPGSA, GLGSA, GPGSV, GLGSV, GNRMC, GNVTG and GNGLL.

4.25. Packet Type: 386 PMTK_API_SET_STATIC_NAV_THD

This message is used to set the speed threshold for static navigation. If the actual speed is below the threshold, the output position will keep the same and output speed will be zero. This function is disabled if the threshold is set to 0.

Data Field: $PMTK386,Speed_threshold Example: $PMTK386,0.3*3E<CR><LF> Response: $PMTK001,386,3*3D<CR><LF>	
Field	**Description**
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	386
Speed Threshold	0~2m/s
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.26. Packet Type: 400 PMTK_API_Q_FIX_CTL

This message is used to query the rate of position fixing activity.

Refer to PMTK_API_SET_FIX_CTL for setting the rate.

Refer to PMTK_DT_FIX_CTL for the result of the query.

Data Field: None Example: $PMTK400*36<CR><LF>	
Field	**Description**
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	400
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.27. Packet Type: 401 PMTK_API_Q_DGPS_MODE

This message is used to query the setting of DGPS mode.

Refer to PMTK_API_SET_DGPS_MODE for setting the DGPS mode.

Refer to PMTK_DT_DGPS_MODE for the result of the query.

Data Field: None Example: $PMTK401*37<CR><LF>	
Field	**Description**
$	Each NMEA message starts with '$'

PMTK	MTK proprietary message
Packet Type	401
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.28. Packet Type: 413 PMTK_API_Q_SBAS_ENABLED

This message is used to query the setting of SBAS.

Refer to PMTK_API_SET_SBAS_ENABLE for SBAS setting.

Refer to PMTK_DT_SBAS_ENABLED for the result of the query.

Data Field: None Example: $PMTK413*34<CR><LF>	
Field	**Description**
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	413
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.29. Packet Type: 414 PMTK_API_Q_NMEA_OUTPUT

This message is used to query the current NMEA sentence output frequencies.

Refer to PMTK_API_SET_NMEA_OUTPUT for the frequencies setting.

Refer to PMTK_DT_NMEA_OUTPUT for the result of the query.

Data Field: None Example: $PMTK414*33<CR><LF>	
Field	**Description**
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	414
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.30. Packet Type: 605 PMTK_Q_RELEASE

This message is used to query the firmware release information.

Refer to PMTK_DT_RELEASE for the result of the query.

Data Field: None Example: $PMTK605*31<CR><LF>	
Field	**Description**
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	605
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.31. Packet Type: 500 PMTK_DT_FIX_CTL

This message is the response to PMTK_API_Q_FIX_CTL.

Data Field: $PMTK500,Fix interval Example: $PMTK500,1000,0,0,0,0*1A<CR><LF>	
Field	**Description**
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	500
Fix Interval	Position fix interval [msec]. More than 100
Reserved	Always 0
Reserved	Always 0
Reserved	Always 0
Reserved	Always 0
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.32. Packet Type: 501 PMTK_DT_DGPS_MODE

This message is the response to PMTK_API_Q_DGPS_MODE.

Data Field: $PMTK501,Mode Example: $PMTK501,1*2B<CR><LF>	
Field	**Description**
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	501
Mode	DGPS data source mode '0'=No DGPS source '1'=RTCM '2'=SBAS
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.33. Packet Type: 513 PMTK_DT_SBAS_ENABLED

This message is the response to PMTK_API_Q_SBAS_ENABLED.

Data Field: $PMTK513,Enable Example: $PMTK513,1*28<CR><LF>	
Field	**Description**
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	513
Enable	'0'=Disable '1'=Enable
*	End character of data field
Checksum	Hexadecimal checksum

<CR><LF>	Each of the messages

4.34. Packet Type: 514 PMTK_DT_NMEA_OUTPUT

This message is the response to PMTK_API_Q_NMEA_OUTPUT.

Data Field:
None
Example:
$PMTK514,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0*33<CR><LF>

Field	Description
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	514
0 GLL	GLL interval - Geographic Position - Latitude Longitude
1 RMC	RMC interval - Recommended Minimum Specific GNSS Sentence
2 VTG	VTG interval - Course Over Ground and Ground Speed
3 GGA	GGA interval - GNSS Fix Data
4 GSA	GSA interval - GNSS DOPS and Active Satellites
5 GSV	GSV interval - GNSS Satellites in View
6 Reserved	GRS interval – GNSS Range Residuals
7 Reserved	GST interval – GNSS Pseudorange Error Statistics
8 Reserved	
9 Reserved	
10 Reserved	
11 Reserved	
12 Reserved	
13 Reserved	

14 Reserved	
15 Reserved	
16 Reserved	
17 ZDA	ZDA interval - Time and Date
18 Reserved	PMTKCHN interval - GNSS Channel Status
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.35. Packet Type: 705 PMTK_DT_RELEASE

This message is the response to PMTK_Q_RELEASE.

Data Field: $PMTK705, Release string, Build ID, Product Model (SDK Version) Example: $PMTK705,AXN_3.10_3333_12102201,0000,QUECTEL-L76,*18<CR><LF>	
Field	**Description**
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	705
Release String	Firmware release name and version 3318: Mcore_x.x 3329: AXN_x.x 3339: AXN_x.x 3333: AXN_x.x 3337: AXN_x.x
Build ID	Build ID set in CoreBuilder for firmware version control
Product Model	Product Model set in CoreBuilder for product identification
SDK Version (Optional)	Showing SDK version if the firmware is used for SDK

*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.36. Packet Type: 869 PMTK_EASY_ENABLE

This message is used to enable or disable EASY function, and query whether EASY is enabled or disabled.

Data Field:
$PMTK869,CmdType[, Enabled]
Example:
$PMTK869,1,1*35<CR><LF>
Response:
$PMTK001,869,3*37<CR><LF>

Field	Description
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	869
CmdType	'0'=Query '1'=Set '2'=Result for Query operation
Enabled	'0'=Disable '1'=Enable
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

NOTES

1. If EASY is disabled, the receiver returns:
   $PMTK869,2,0,0*2B<CR><LF>
2. If EASY is enabled and is not finished yet, the receiver may return:
   $PMTK869,2,1,0*2A<CR><LF>

3. If EASY is enabled and is finished 1-day extension, the receiver may return:
   $PMTK869,2,1,1*2B<CR><LF>
4. If EASY is enabled and is finished 2-day extension, the receiver may return:
   $PMTK869,2,1,2*28<CR><LF>
5. If EASY is enabled and is finished 3-day extension, the receiver may return:
   $PMTK869,2,1,3*29<CR><LF>

4.37. Packet Type: 875 PMTK_PMTKLSC_STN_OUTPUT

This message is used to enable or disable PMTKLSC sentence output and query whether PMTKLSC sentence output is enabled or disabled. This command is not supported in all-in-one solution.

Data Field:
$PMTK875,CmdType[,Enabled]
Example:
$PMTK875,1,1*38<CR><LF> : Enable PMTKLSC and PMTKLSCB sentence output
Response:
$PMTKLSC,Parameter1,Parameter2,Parameter3*CS
$PMTKLSB,Parameter1,Parameter2,Parameter3*CS
Where Parameter1 : current leap second
Parameter2 : leap indicator, 1 means updated from broadcast data
Parameter3 : next leap second

Field	Description
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	875
CmdType	'0'=Query '1'=Set '2'=Result for Query operation
Enabled	'0'=Disable '1'=Enable
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

4.38. Packet Type: 886 PMTK_FR_MODE

This message is used to set navigation mode.

Data Field: $PMTK886,CmdType Example: $PMTK886,3*2B<CR><LF> Response: $PMTK001,886,3*36	
Field	**Description**
$	Each NMEA message starts with '$'
PMTK	MTK proprietary message
Packet Type	886
CmdType	'0'=Normal mode: For general purpose '1'=Fitness mode: For running and walking purpose that the low-speed (<5m/s) movement will have more effect on the position calculation. '2'=Aviation mode: For high-dynamic purpose that the large-acceleration movement will have more effect on the position calculation. '3'=Balloon mode: For high-altitude balloon purpose that the vertical movement will have more effect on the position calculation.
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of the messages

5 SDK NMEA Packet Protocol

This chapter introduces the SDK NMEA packet protocol, which is a set of extension messages of standard NMEA packet protocol. These messages are used to control and configure MC60's internal GNSS engine. The following table shows the structure of SDK NMEA packet. The SDK NMEA packet is sent through AT commands.

5.1. PQEPE Enable/Disable PQEPE Sentence Output

The packet enables/disables output of the URC including EPE data, that is, estimated horizontal and vertical position errors.

PQEPE Enable/Disable PQEPE Sentence Output	
Write Command **$PQEPE,W,<mode>,<save>*Checksum<CR><LF>**	Response **$PQEPE,W,OK*Checksum<CR><LF>** If error **$PQEPE,W,ERROR*Checksum<CR><LF>**
URC Message	**$PQEPE,<EPE_hori>,<EPE_vert>*Checksum<CR><LF>** Parameter **<EPE_hori>** Estimated horizontal position error **<EPE_vert>** Estimated vertical position error Example **$PQEPE,5.3050,3.2000*53**
Reference	

Parameter

<mode>	Operation 0 Disable the URC including EPE data 1 Enable the URC including EPE data
<save>	Save operation 0 Parameter is not saved, ineffective after restart

1	Parameter is saved in flash, effective after restart

5.2. PQGLP Set the GNSS Module into GLP Mode

The packet is used to set the GNSS module into GLP mode.

PQGLP Set the GNSS Module into GLP Mode	
Write Command **$PQGLP,W,<mode>,<save>*ChkSum<CR><LF>**	Response **$PQGLP,W,OK*ChkSum<CR><LF>** If error **$PQGLP,W,ERROR*ChkSum<CR><LF>**
Read Command **$PQGLP,R*ChkSum<CR><LF>**	Response **$PQGLP,R,<mode>*ChkSum<CR><LF>**
Reference	

Parameter

<mode>	Module operation mode
	0 Normal mode
	1 GLP mode
<save>	Save operation
	0 Parameter is not saved, ineffective after restart
	1 Parameter is saved in flash, effective after restart

Example

```
$PQGLP,W,1,1*21          // Change to GLP mode
$PQGLP,W,OK*09           // Set OK

$PQGLP,R*24              // Read mode
$PQGLP,R,1*39            // Read OK, GLP mode enabled
```

NOTE

The command will be effective immediately after setting.

6 Default Configurations

Table 3: Default Configurations

Item	Default
NMEA Port Baud Rate	115200bps
Datum	WGS84
Rate of Position Fixing	1Hz
DGPS Mode	SBAS
SBAS	Enabled
NMEA Output Messages	GGA, RMC, GSA, GSV, VTG and GLL
AIC	On
EASY™	On

7 Appendix A References

Table 4: Related Documents

SN	Document Name	Remark
[1]	Quectel_MC60_Hardware_Design	MC60 Hardware Design
[2]	Quectel_MC60_GNSS_AT_Commands_Manual	MC60 GNSS AT Commands Manual
[3]	Quectel_MC60_Reference_Design	MC60 Reference Design
[4]	Quectel_MC60-TE-A_Kit_User_Guide	MC60-TE-A Kit User Guide
[5]	Quectel_GNSS_SDK_Commands_Manual	GNSS SDK Commands Manual

Table 5: Terms and Abbreviations

Abbreviation	Description
AGPS	Assisted Global Positioning System
AIC	Active Interference Cancellation
CS	Commercial Sample
DGPS	Differential Global Positioning System
EASY	Embedded Assist System
GBS	NMEA: GPS Satellite Fault Detection
GGA	NMEA: Global Positioning System Fix Data
GLL	NMEA: Geographic Latitude and Longitude
GNSS	Global Navigation Satellite System
GLONASS	Global Navigation Satellite System (The Russian GNSS)
GPS	Global Positioning System

GSA	NMEA: GNSS DOP and Active Satellites
GSV	NMEA: GNSS Satellites in View
HDOP	Horizontal Dilution of Precision
MP	Mass Production
NMEA	National Marine Electronics Association
PDOP	Position Dilution of Precision
PMTK	Private Protocol of MTK
RMC	NMEA: Recommended Minimum Position Data
SBAS	Satellite-Based Augmentation System
UTC	Universal Time Coordinated
VDOP	Vertical Dilution of Precision
VTG	NMEA: Track Made Good and Ground Speed
WAAS	Wide Area Augmentation System

Table 6: Structure of NMEA Message

Filed	Length (Bytes)	Description
$	1	Each NMEA message starts with '$'
Talker ID	1~2	Talker IDs can be 'GP', 'GN' and 'BD' when the message ID is RMC, VTG, GLL or GGA; Talker IDs can be 'GP', 'GN', 'BD' and 'GA' when the message ID is GSA; Talker IDs can be 'GP', 'GL', 'BD' and 'GA' when the message ID is GSV, and Talker IDs can be 'GN' when the message ID is GBS.
NMEA Message ID	3	NMEA message ID
Data Field	Variable, depends on the NMEA message type	Data fields, delimited by comma ','
*	1	End character of data field
Checksum	2	A hexadecimal number calculated by exclusive OR of all characters between '$' and '*'

<CR><LF>	2	Each NMEA message ends with ‘CR’ and ‘LF’

NOTE

The default output messages of MC60 are as following six sentences: RMC, VTG, GGA, GSA, GSV and GLL. But if the receiver is fixed by GPS, GLONASS and Galileo, the default output message of MC60 will be following six sentences: RMC, VTG, GGA, GSA, GSV and GBS.

Table 7: Structure of MTK NMEA Packet

Filed		Length (Bytes)	Description
$		1	Each NMEA message starts with ‘$’
Talker ID		1	‘P’ for proprietary message
NMEA	Data Type	3	Always ‘MTK’ to indicate MTK proprietary message
Data Filed	Packet Type	3	Packet type, from ‘000’ to ‘999’
	Packet Data	Variable, depends on the packet type	Data fields, delimited by comma ‘,’
*		1	End character of data field
Checksum		2	A hexadecimal number calculated by exclusive OR of all characters between ‘$’ and ‘*’
<CR><LF>		2	Each NMEA message ends with ‘CR’ and ‘LF’