LC29H (BS) GNSS Protocol Specification

GNSS Module Series

Version: 1.1

Date: 2023-05-18

Status: Released

At Quectel, our aim is to provide timely and comprehensive services to our customers. If you require any assistance, please contact our headquarters:

Quectel Wireless Solutions Co., Ltd.
Building 5, Shanghai Business Park Phase III (Area B), No.1016 Tianlin Road, Minhang District, Shanghai 200233, China
Tel: +86 21 5108 6236
Email: info@quectel.com

Or our local offices. For more information, please visit:
http://www.quectel.com/support/sales.htm.

For technical support, or to report documentation errors, please visit:
http://www.quectel.com/support/technical.htm.
Or email us at: support@quectel.com.

Legal Notices

We offer information as a service to you. The provided information is based on your requirements and we make every effort to ensure its quality. You agree that you are responsible for using independent analysis and evaluation in designing intended products, and we provide reference designs for illustrative purposes only. Before using any hardware, software or service guided by this document, please read this notice carefully. Even though we employ commercially reasonable efforts to provide the best possible experience, you hereby acknowledge and agree that this document and related services hereunder are provided to you on an “as available” basis. We may revise or restate this document from time to time at our sole discretion without any prior notice to you.

Use and Disclosure Restrictions

License Agreements

Documents and information provided by us shall be kept confidential, unless specific permission is granted. They shall not be accessed or used for any purpose except as expressly provided herein.

Copyright

Our and third-party products hereunder may contain copyrighted material. Such copyrighted material shall not be copied, reproduced, distributed, merged, published, translated, or modified without prior written consent. We and the third party have exclusive rights over copyrighted material. No license shall be granted or conveyed under any patents, copyrights, trademarks, or service mark rights. To avoid ambiguities, purchasing in any form cannot be deemed as granting a license other than the normal non-exclusive, royalty-free license to use the material. We reserve the right to take legal action for noncompliance with abovementioned requirements, unauthorized use, or other illegal or malicious use of the material.

Trademarks

Except as otherwise set forth herein, nothing in this document shall be construed as conferring any rights to use any trademark, trade name or name, abbreviation, or counterfeit product thereof owned by Quectel or any third party in advertising, publicity, or other aspects.

Third-Party Rights

This document may refer to hardware, software and/or documentation owned by one or more third parties ("third-party materials"). Use of such third-party materials shall be governed by all restrictions and obligations applicable thereto.

We make no warranty or representation, either express or implied, regarding the third-party materials, including but not limited to any implied or statutory, warranties of merchantability or fitness for a particular purpose, quiet enjoyment, system integration, information accuracy, and non-infringement of any third-party intellectual property rights with regard to the licensed technology or use thereof. Nothing herein constitutes a representation or warranty by us to either develop, enhance, modify, distribute, market, sell, offer for sale, or otherwise maintain production of any our products or any other hardware, software, device, tool, information, or product. We moreover disclaim any and all warranties arising from the course of dealing or usage of trade.

Privacy Policy

To implement module functionality, certain device data are uploaded to Quectel's or third-party's servers, including carriers, chipset suppliers or customer-designated servers. Quectel, strictly abiding by the relevant laws and regulations, shall retain, use, disclose or otherwise process relevant data for the purpose of performing the service only or as permitted by applicable laws. Before data interaction with third parties, please be informed of their privacy and data security policy.

Disclaimer

a) We acknowledge no liability for any injury or damage arising from the reliance upon the information.
b) We shall bear no liability resulting from any inaccuracies or omissions, or from the use of the information contained herein.
c) While we have made every effort to ensure that the functions and features under development are free from errors, it is possible that they could contain errors, inaccuracies, and omissions. Unless otherwise provided by valid agreement, we make no warranties of any kind, either implied or express, and exclude all liability for any loss or damage suffered in connection with the use of features and functions under development, to the maximum extent permitted by law, regardless of whether such loss or damage may have been foreseeable.
d) We are not responsible for the accessibility, safety, accuracy, availability, legality, or completeness of information, advertising, commercial offers, products, services, and materials on third-party websites and third-party resources.

About the Document

Document Information	
Title	**LC29H (BS) GNSS Protocol Specification**
Subtitle	GNSS Module Series
Document Type	GNSS Protocol Specification
Document Status	Released

Revision History

Version	Date	Description
-	2022-09-19	Creation of the document
1.0	2022-11-25	First official release
1.1	2023-05-18	1. Added the table of error codes (*Table 3*). 2. Added Sample Code for NMEA Checksum (*Chapter 2.1*). 3. Added a note about default value (*Chapter 2.2.1*). 4. Added PQTMCFGMSGRATE, PQTMEPE, PQTMSVINSTATUS and PAIR864 messages (*Chapters 2.2.5*, *2.2.6*, *2.2.7* and *2.3.8*).

Contents

Table Index

1 Introduction

Quectel LC29H (BS) GNSS module supports GPS, GLONASS, Galileo, BDS and QZSS constellations. Concurrent tracking of GPS L1 C/A, GLONASS L1, Galileo E1, BDS B1I, QZSS L1 C/A, GPS L5, Galileo E5a, BDS B2a and QZSS L5 frequency bands provides fast and accurate acquisition and makes this module an outstanding solution for a base station and for generating RTK corrections messages.

This document describes the software commands that are relevant to control and modify the module configuration. The software commands are NMEA proprietary commands defined by Quectel (PQTM commands) and the chipset supplier (PAIR messages). To report GNSS information, the module supports message outputting in RTCM format.

The Quectel LC29H (BS) is designed to be used as a GNSS base station. The module supports the following protocols:

Table 1: Supported Protocols

Protocol	Type
NMEA 0183 V3.10/V4.10	Input/output, ASCII, proprietary
RTCM 10403.3	Output, binary

NOTE

Quectel assumes no responsibility if commands other than the ones listed herein are used.

2 NMEA Protocol

2.1. Structure of NMEA Protocol Messages

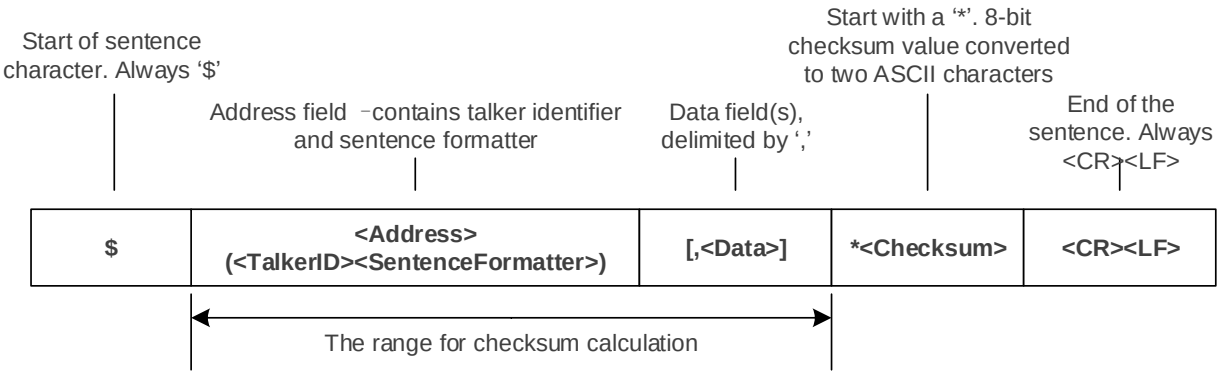

Figure 1: Structure of NMEA Protocol Messages

Table 2: Structure of NMEA Protocol Messages

Field	Description
$	Start of the sentence (Hex 0x24).
<Address>	**In Proprietary Messages:** In NMEA proprietary messages, this field consists of the proprietary character **P** followed by a three-character Manufacturer's Mnemonic Code, used to identify the TALKER issuing a proprietary sentence, and any additional characters as required.
<Data>	Data fields, delimited by data field delimiter **','**. Variable length (depends on the NMEA message type).
<Checksum>	Checksum field follows the checksum delimiter character **'*'**. Checksum is the 8-bit exclusive OR of all characters in the sentence, including the **','** the field delimiter, between but not including the **'$'** and the **'*'** delimiters.
<CR><LF>	End of the sentence (Hex 0x0D 0x0A).

NMEA Checksum Sample Code:

```
// pData  is the data array of which the checksum needs to be calculated:

unsigned char Ql_Check_XOR(const unsigned char *pData, unsigned int Length)
{
    unsigned char result = 0;
    unsigned int i = 0;

    if((NULL == pData) || (Length < 1))
    {
        return 0;
    }
    for(i = 0; i < Length; i++)
    {
        result ^= *(pData + i);
    }

    return result;
}
```

2.2. PQTM Messages

This chapter provides an explain on the **PQTM** messages (proprietary NMEA messages defined by Quectel) supported by LC29H (BS) GNSS module.

Table 3: Error Codes

Field	Format	Unit	Description
<ErrCode>	Numeric	-	Error code. 1 = Invaild parameters. 2 = Execution failed.

2.2.1. PQTMCFGSVIN

Sets/gets status of the survey-in feature. In order to work as a base station, the module external antenna should be mounted on a static point (try to mount it with a clear sky visibility) .This feature can determine the antenna location either by Survey-in mode or Fixed mode.

In order to operate as a base station, the module external antenna should be mounted on a fix point. The antenna accurate coordinate location can be acquired through a self-survey process. The Survey-in mode (**<Mode>** = 1) determines the receiver's position by building a weighted mean of all valid 3D positioning solutions. You can set values of **<MinDur>** and **<3D_AccLimit>** to define the minimum observation time and 3D position standard deviation used for the position estimation. The Fixed mode (**<Mode>** = 2) requires user to manually enter the receiver position coordinates. Any error in the base station position will translate directly into rover position error.

Type:

Set/Get

Synopsis:

```
//Set:
$PQTMCFGSVIN,W,<Mode>,<MinDur>,<3D_AccLimit>,<ECEF_X>,<ECEF_Y>,<ECEF_Z>*<Checksum><CR><LF>
//Get:
$PQTMCFGSVIN,R*<Checksum><CR><LF>
```

Parameter:

Field	Format	Unit	Description
<Mode>	Numeric	-	Configure the receiver mode. 0 = Disable 1 = Survey-in mode 2 = Fixed mode (ARP position is given in ECEF.)
<MinDur>	Numeric	Second	Survey-in minimum duration. Range: 0–86400. Default value: 43200.
<3D_AccLimit>	Numeric	Meter	Limit 3D positioning accuracy in survey-in mode. Default value: 15.0. If this field is 0, it means there is no limit.
<ECEF_X>	Numeric	Meter	WGS84 ECEF X coordinate. Default value: 0.0.
<ECEF_Y>	Numeric	Meter	WGS84 ECEF Y coordinate. Default value: 0.0.
<ECEF_Z>	Numeric	Meter	WGS84 ECEF Z coordinate. Default value: 0.0.

Result:

- If successful, the module returns:

```
//Response to Set command:
$PQTMCFGSVIN,OK*<Checksum><CR><LF>
//Response to Get command:
$PQTMCFGSVIN,OK,<Mode>,<MinDur>,<3D_AccLimit>,<ECEF_X>,<ECEF_Y>,<ECEF_Z>*<Checksum><CR><LF>
```

- If failed, the module returns:

```
$PQTMCFGSVIN,ERROR,<ErrCode>*<Checksum><CR><LF>
```

For details about **<ErrCode>**, see *Table 3: Error Codes*.

Example:

```
//Set:
$PQTMCFGSVIN,W,2,0,0.0,-2472446.4619,4828304.1363,3343730.2653*34
$PQTMCFGSVIN,OK*70

//Get:
$PQTMCFGSVIN,R*26
$PQTMCFGSVIN,OK,2,0,0.0,-2472446.4619,4828304.1363,3343730.2653*67
```

NOTE

If none of the settings has been overwritten by a user command, you can use the get command to review the default configuration. If the user overwrites the default configuration and you need to determine the default values, please contact Quectel Technical Support (support@quectel.com) to query the module default configuration.

2.2.2. PQTMSAVEPAR

Saves the configurations set via **$PQTM** commands to NVM.

Type:

Command

Synopsis:

```
$PQTMSAVEPAR*<Checksum><CR><LF>
```

Parameter:

None

Result:

- If successful, the module returns:

```
$PQTMSAVEPAR,OK*<Checksum><CR><LF>
```

- If failed, the module returns:

```
$PQTMSAVEPAR,ERROR,<ErrCode>*<Checksum><CR><LF>
```

For details about **<ErrCode>**, see *Table 3: Error Codes*.

Example:

```
$PQTMSAVEPAR*5A
$PQTMSAVEPAR,OK*72
```

2.2.3. PQTMRESTOREPAR

Restores all parameters set via **$PQTM** commands to their default values. Reset the module after executing this command.

Type:

Command

Synopsis:

```
$PQTMRESTOREPAR*<Checksum><CR><LF>
```

Parameter:

None

Result:

- If successful, the module returns:

```
$PQTMRESTOREPAR,OK*3B
```

- If failed, the module returns:

```
$PQTMRESTOREPAR,ERROR,<ErrCode>*<Checksum><CR><LF>
```

For details about **<ErrCode>**, see *Table 3: Error Codes*.

Example:

```
$PQTMRESTOREPAR*13
$PQTMRESTOREPAR,OK*3B
```

2.2.4. PQTMVERNO

Queries the firmware version information.

Type:

Command

Synopsis:

```
$PQTMVERNO*<Checksum><CR><LF>
```

Parameter:

None

Result:

- If successful, the module returns:

```
$PQTMVERNO,<VerStr>,<BuildDate>,<BuildTime>*<Checksum><CR><LF>
```

- If failed, the module returns:

```
$PQTMVERNO,ERROR,<ErrCode>*<Checksum><CR><LF>
```

Parameters included in the result:

Field	Format	Unit	Description
<VerStr>	String	-	Version string.
<BuildDate>	yyyy/mm/dd	-	Firmware build date.
<BuildTime>	hh:mm:ss	-	Firmware build time.

For details about **<ErrCode>**, see *Table 3: Error Codes*.

Example:

```
$PQTMVERNO*58
$PQTMVERNO,LC29HBSNR01A01S,2022/08/31,15:22:59*27
```

2.2.5. PQTMCFGMSGRATE

Configures the message output rate on the current port.

Type:

Set/Get

Synopsis:

```
//Set:
$PQTMCFGMSGRATE,W,<MsgName>,<Rate>,<MsgVer>*<Checksum><CR><LF>
//Get:
$PQTMCFGMSGRATE,R,<MsgName>,<MsgVer>*<Checksum><CR><LF>
```

Parameter:

Field	Format	Unit	Description
<MsgName>	String	-	Configuration message name. See *Table 4: Supported Messages* for details.
<Rate>	Numeric	-	Message output rate. 0 = Not output. N = Output once per every N position fix(es). Range of N see *Table 4: Supported Messages* for details.
<MsgVer>	Numeric	-	Message version.

Result:

- If successful, the module returns:

```
//Response to Set command:
$PQTMCFGMSGRATE,OK*<Checksum><CR><LF>
//Response to Get command:
$PQTMCFGMSGRATE,OK,<MsgName>,<Rate>,<MsgVer>*<Checksum><CR><LF>
```

- If failed, the module returns:

```
$PQTMCFGMSGRATE,ERROR,<ErrCode>*<Checksum><CR><LF>
```

For details about **<ErrCode>**, see *Table 3: Error Codes*.

Example:

```
//Set the output rate of $PQTMEPE message to once every position fix:
$PQTMCFGMSGRATE,W,PQTMEPE,1,2*1D
$PQTMCFGMSGRATE,OK*29

//Get the output rate of $PQTMEPE message:
$PQTMCFGMSGRATE,R,PQTMEPE,2*05
$PQTMCFGMSGRATE,OK,PQTMEPE,1,2*4E

//Set the output rate of $PQTMSVINSTATUS message to once every position fix:
$PQTMCFGMSGRATE,W,PQTMSVINSTATUS,1,1*58
$PQTMCFGMSGRATE,OK*29

//Get the output rate of $PQTMSVINSTATUS message:
$PQTMCFGMSGRATE,R,PQTMSVINSTATUS,1*40
$PQTMCFGMSGRATE,OK,PQTMSVINSTATUS,1,1*0B
```

Table 4: Supported Messages

Message Name	Message Output Rate Range (N)
$PQTMSVINSTATUS	1–20
$PQTMEPE	1–20

2.2.6. PQTMEPE

Outputs the estimated positioning error.

Type:

Output

Synopsis:

```
$PQTMEPE,<MsgVer>,<EPE_North>,<EPE_East>,<EPE_Down>,<EPE_2D>,<EPE_3D>*<Checksum>
<CR><LF>
```

Parameter:

Field	Format	Unit	Description
<MsgVer>	Numeric	-	Message version. 2 = Version 2 (Always 2 for this version.)
<EPE_North>	Numeric	Meter	Estimated north error.
<EPE_East>	Numeric	Meter	Estimated east error.
<EPE_Down>	Numeric	Meter	Estimated down error.
<EPE_2D>	Numeric	Meter	Estimated 2D position error.
<EPE_3D>	Numeric	Meter	Estimated 3D position error.

Example:

```
$PQTMEPE,2,3.259,3.303,15.440,4.640,16.122*5E
```

2.2.7. PQTMSVINSTATUS

Outputs the survey-in feature status.

Type:

Output

Synopsis:

```
$PQTMSVINSTATUS,<MsgVer>,<TOW>,<Valid>,<Res0>,<Res1>,<Obs>,<CfgDur>,<MeanX>,<MeanY>,<MeanZ>,<MeanAcc>*<Checksum><CR><LF>
```

Parameter:

Field	Format	Unit	Description
<MsgVer>	Numeric	-	Message version. 1 = Version 1 (Always 1 for this version.)
<TOW>	Numeric	Millisecond	GPS time of week.
<Valid>	Numeric	-	Survey-in position validity flag. 0 = Invalid 1 = In-progress 2 = Valid
<Res0>	Numeric	-	Reserved.

Field	Format	Unit	Description
<Res1>	Numeric	-	Reserved.
<Obs>	Numeric	-	Number of position observations used during survey-in process.
<CfgDur>	Numeric	-	Duration configured via the **<MinDur>** field of **$PQTMCFGSVIN** command.
<MeanX>	Numeric	Meter	Current survey-in mean position along X axis of ECEF coordinate system.
<MeanY>	Numeric	Meter	Current survey-in mean position along Y axis of ECEF coordinate system.
<MeanZ>	Numeric	Meter	Current survey-in mean position along Z axis of ECEF coordinate system.
<MeanAcc>	Numeric	Meter	Current survey-in mean position accuracy estimation.

Example:

```
$PQTMSVINSTATUS,1,2241,1,,01,538,43200,-2472436.0802,4828383.0026,3343698.4839,9.5*38
```

2.3. PAIR Messages

This chapter provides an explanation on the PAIR commands.

PAIR Message Format:

```
$PAIR<PacketType>[,<Data>]<Checksum><CR><LF>
```

Packet Type: Three-byte character string, from 000 to 999. An identifier for each PAIR message.
Data: This field can be omitted, or multiple fields can be delimited by a data field delimiter **‘,’**. Different commands correspond to different data. See the specific values below.

2.3.1. Packet Type: 001 PAIR_ACK

Acknowledges a PAIR command. An acknowledgement packet **$PAIR001** is returned to inform the sender that the receiver has received the packet.

Type:

Output

Synopsis:

```
$PAIR001,<CommandID>,<Result>*<Checksum><CR><LF>
```

Parameter:

Field	Format	Unit	Description
<CommandID>	Numeric	-	Type of command/packet to be acknowledged.
<Result>	Numeric	-	Result type. 0 = Command has been successfully sent. 1 = Command is being processed. Please wait for the result. 2 = Command sending failed. 3 = This Command ID is not supported. 4 = Command parameter error. Out of range/some parameters were lost/checksum error. 5 = MNL service is busy. You can try again soon.

Example:

```
$PAIR001,006,0*3D
```

2.3.2. Packet Type: 432 PAIR_RTCM_SET_OUTPUT_MODE

Sets RTCM output mode.

Type:

Set

Synopsis

```
$PAIR432,<Mode>*<Checksum><CR><LF>
```

Parameter:

Field	Format	Unit	Description
<Mode>	Numeric	-	RTCM output mode setting. -1 = Disable outputting RTCM-3 messages 0 = Enable outputting RTCM-3 MSM4 messages 1 = Enable outputting RTCM-3 MSM7 messages

Result:

Returns **$PAIR001** message.

Example:

```
$PAIR432,1*22
$PAIR001,432,0*3E
```

2.3.3. Packet Type: 433 PAIR_RTCM_GET_OUTPUT_MODE

Gets the RTCM output mode.

Type:

Get

Synopsis

```
$PAIR433*<Checksum><CR><LF>
```

Parameter:

None

Result:

Returns **$PAIR001** message and the query result.

Query result message format:

```
$PAIR433,<Mode>*<Checksum><CR><LF>
```

Parameter included in the result:

Field	Format	Unit	Description
<Mode>	Numeric	-	RTCM output mode setting. -1 = Disable outputting RTCM-3 messages 0 = Enable outputting RTCM-3 MSM4 messages 1 = Enable outputting RTCM-3 MSM7 messages

Example:

```
$PAIR433*3E
$PAIR001,433,0*3F
$PAIR433,-1*0E
```

2.3.4. Packet Type: 434 PAIR_RTCM_SET_OUTPUT_ANT_PNT

Enables/disables outputting stationary antenna reference point in RTCM format.

Type:

Set

Synopsis

```
$PAIR434,<Enable>*<Checksum><CR><LF>
```

Parameter:

Field	Format	Unit	Description
<Enable>	Numeric	-	Enable/disable outputting stationary RTK reference station ARP (message type 1005). 0 = Disable 1 = Enable

Result:

Returns **$PAIR001** message.

Example:

```
$PAIR434,1*24
$PAIR001,434,0*38
```

2.3.5. Packet Type: 435 PAIR_RTCM_GET_OUTPUT_ANT_PNT

Queries the setting of outputting stationary antenna reference point in RTCM format.

Type:

Get

Synopsis

```
$PAIR435*<Checksum><CR><LF>
```

Parameter:

None

Result:

Returns **$PAIR001** message and the query result.

Query result message format:

```
$PAIR435,<Enable>*<Checksum><CR><LF>
```

Parameter included in the result:

Field	Format	Unit	Description
<Enable>	Numeric	-	Status of outputting stationary RTK reference station ARP (message type 1005). 0 = Disabled 1 = Enabled

Example:

```
$PAIR435*38
$PAIR001,435,0*39
$PAIR435,1*25
```

2.3.6. Packet Type: 436 PAIR_RTCM_SET_OUTPUT_EPHEMERIS

Enables/disables outputting satellite ephemeris in RTCM format.

Type:

Set

Synopsis

```
$PAIR436,<Enable>*<Checksum><CR><LF>
```

Parameter:

Field	Format	Unit	Description
<Enable>	Numeric	-	Enable/disable outputting satellite ephemeris in RTCM format. 0 = Disable 1 = Enable

Result:

Returns **$PAIR001** message.

Example:

```
$PAIR436,1*26
$PAIR001,436,0*3A
```

2.3.7. Packet Type: 437 PAIR_RTCM_GET_OUTPUT_EPHEMERIS

Queries the status of satellite ephemeris in RTCM format.

Type:

Get

Synopsis

```
$PAIR437*<Checksum><CR><LF>
```

Parameter:

None

Result:

Returns **$PAIR001** message and the query result.

Query result message format:

```
$PAIR437,<Enable>*<Checksum><CR><LF>
```

Parameter included in the result:

Field	Format	Unit	Description
<Enable>	Numeric	-	Status of outputting satellite ephemeris in RTCM format. 0 = Disabled 1 = Enabled

Example:

```
$PAIR437*3A
$PAIR001,437,0*3B
$PAIR437,1*27
```

2.3.8. Packet Type: 864 PAIR_IO_SET_BAUDRATE

Sets the baud rate of UART interface (the default baud rate is 115200 bps).

Type:

Set

Synopsis:

```
$PAIR864,<PortType>,<PortIndex>,<Baudrate>*<Checksum><CR><LF>
```

Parameter:

Field	Format	Unit	Description
<PortType>	Numeric	-	Hardware port type. 0 = UART
<PortIndex>	Numeric	-	Hardware port index. 0 = UART1
<Baudrate>	Numeric	bps	Baud rate. 4800 9600 19200 38400

Field	Format	Unit	Description
			57600 115200 230400 460800 921600 3000000

Result:

Returns **$PAIR001** message.

Example:

```
$PAIR864,0,0,115200*1B
$PAIR001,864,0*31
```

NOTE

1. The module must be rebooted after changing the port baud rate, and the change will take effect after the reboot.
2. Messages may be lost when the output baud rate is lower than 115200 bps.

3 RTCM Protocol

The LC29H (BS) module supports the RTCM protocol according to *RTCM Standard 10403.3 Differential GNSS (Global Navigation Satellite Systems) Services - Version 3*. This protocol is used to transfer GNSS raw measurement data and is available from https://www.rtcm.org/.

Table 5: Supported RTCM3 Messages

Message Type	Mode	Message Name
1005	Output	Stationary RTK Reference Station ARP
1019	Output	GPS Ephemerides
1020	Output	GLONASS Ephemerides
1042	Output	BDS Satellite Ephemeris Data
1044	Output	QZSS Ephemerides
1046	Output	Galileo I/NAV Satellite Ephemeris Data
1074	Output	GPS MSM4
1077	Output	GPS MSM7
1084	Output	GLONASS MSM4
1087	Output	GLONASS MSM7
1094	Output	Galileo MSM4
1097	Output	Galileo MSM7
1114	Output	QZSS MSM4
1117	Output	QZSS MSM7
1124	Output	BDS MSM4
1127	Output	BDS MSM7

NOTE

1. The **$PAIR432** command can enable/disable MSM4/MSM7 (1074, 1077, 1084, 1087, 1094, 1097,1114, 1117, 1124 and 1127) messages if the corresponding constellation is enabled.
2. The **$PAIR434** command can enable/disable Stationary RTK Reference Station ARP (1005) message.
3. The **$PAIR436** command can enable/disable ephemeris (1019, 1020, 1042, 1044 and 1046) messages if the corresponding constellation is enabled.

4 Appendix A References

Table 6: Terms and Abbreviations

Abbreviation	Description
ACK	Acknowledgement
ARP	Antenna Reference Point
ASCII	American Standard Code for Information Interchange
BDS	BeiDou Navigation Satellite System
ECEF	Earth Centered Earth Fixed
Galileo	Galileo Satellite Navigation System (EU)
GLONASS	Global Navigation Satellite System (Russia)
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
MNL	MTK Navigation Lib
MSM	Multiple Signal Messages
NMEA	NMEA (National Marine Electronics Association) 0183 Interface Standard
NVM	Non-Volatile Memory
PAIR	Proprietary Protocol of MTK
QZSS	Quasi-Zenith Satellite System
RTCM	Radio Technical Commission for Maritime Services
RTK	Real-Time Kinematic
WGS84	World Geodetic System 1984

5 Appendix B Special Characters

Table 7: Special Characters

Special Character	Definition
<...>	Parameter name. Angle brackets do not appear in the message.
[...]	Optional field of a message. Square brackets do not appear in the message.
{…}	Repeated field of a message. Curly brackets do not appear in the message.
Underline	Default setting of a parameter.