# Application Note AN0037

# Binary Messages

# Of

# SkyTraq Phoenix GNSS Receiver

Ver 1.4.66

July 03, 2023

## Binary Message Protocol

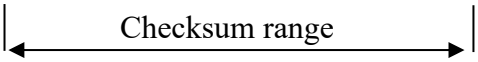
The SkyTraq binary message protocol manual provides the detailed descriptions on the SkyTraq binary protocol serving as a communicating interface between SkyTraq GNSS receivers and an external host such as PC, Notebook and mobile personal device. It is a standard protocol used by all SkyTraq devices and provides users a satisfactory control over the GNSS receivers.

The SkyTraq GNSS receiver outputs standard NMEA messages during normal operation. This NMEA messages may be a scheduled output at a specified rate subject to user's requests. The SkyTraq binary message protocol is designed with cares on reliable transmissions of data, ease & efficiency of implement, and payload independence mechanism which ensure users to retrieve data in a most effective & flexible way. The overall binary protocol messages can be categorized as input and output messages. Input messages provide the functionality to users to control the behavior of the GNSS receiver and to retrieve the detailed information of the GNSS status in real-time. Output messages, on the other hand, are information strings that GNSS receiver responses to requests from hosts and can optionally periodically reports the Position, Velocity and Time (PVT) via NMEA or binary messages.

## BINARY MESSAGE STRUCTURE

### Message Format

The following picture shows the structure of a binary message.




| Start of Sequence | Payload Length (PL) | Payload | | Checksum (CS) | End of Sequence |
|---|---|---|---|---|---|
| | | Message ID | Message Body | | |
| 0xA0, 0xA1 | Two bytes | Message ID: 1 bytes; Payload up to 65535 bytes | | One byte | 0x0D, 0x0A |

The syntax of the message is shown below.

<0xA0,0xA1><PL><Message ID><Message Body><CS><0x0D,0x0A>

### Start of Sequence

This field contains two bytes of values 0xA0, 0xA1 which indicate start of Messages.

### Payload Length

The payload length (PL) field contains 16 bits of value which indicates the length of payload.

### Payload

The payload field consists of 2 sub-fields, Message ID and Message Body. Message ID field defines the message ID.

| Sub-Field | Values |
|---|---|
| Message ID (ID) | 0x01~0xFF |
| Message Body | Data Bytes |

### Message Body

The Message Body may further consist of 2 sub-fields, Sub-Message ID (Sub-ID) and Sub-Message Body.

| Sub-Field | Values |
|---|---|

| | |
|---|---|
| Sub-Message ID(SID) | 0x01~0xFF |
| Sub-Message Body | Data Bytes |

## Checksum

Checksum (CS) field is transmitted in all messages. The checksum field is the last field in a message before the end of sequence field. The checksum is the 8-bit exclusive OR of only the payload bytes which start from Message ID until the last byte prior to the checksum byte. A reference to the calculation of CS is provided below,

CS = 0, N=PL;
For n = 0 to N
CS = CS ^ <Payload Byte # n>

## End of Sequence

This field contains two bytes of values 0x0D, 0x0A which indicate end of Messages.

## Data Byte Ordering

All payloads in binary protocol are transferred in big-endian format. The high order byte is transmitted first followed by the low order byte for data size larger than a byte (e.g. UINT32, DPFP).

## Data Type Definition

| | |
|---|---|
| UINT8 | 8 bit unsigned integer |
| UINT16 | 16 bit unsigned integer |
| UINT32 | 32 bit unsigned integer |
| SINT8 | 8 bit signed integer |
| SINT16 | 16 bit signed integer |
| SINT32 | 32 bit signed integer |
| SPFP | 32 bit single precision floating point number |
| DPFP | 64 bit double precision floating point number |

## MESSAGE FLOW

Host can perform actions to GNSS receiver by issuing a request or a set message. The message flow between Host and GNSS receiver is designed under the considerations of certain reliable transmission. SkyTraq binary message protocol requires an ACK response from the GNSS receiver upon receiving a successful input message and on the other hand, requires a NACK response from the receiver to a failed input message. Figure 1 shows a message flow that a host requests information from GNSS receiver and the GNSS receiver responses with an ACK and information respectively. Figure 2 shows a message flow with un-successful input message. Therefore, all requests (input messages) will have a corresponding ACK or NACK to be related with. However, output messages will not require the host to confirm by an ACK or NACK back in current design. A NACK may be caused by a request message with invalid length, invalid checksum, wrong input values, firmware message format changed or firmware not supported.

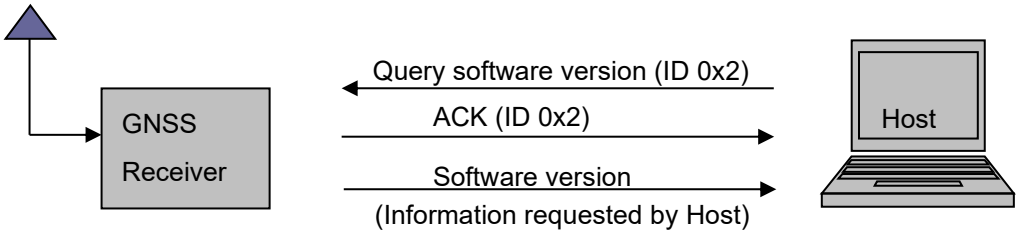



Figure 1

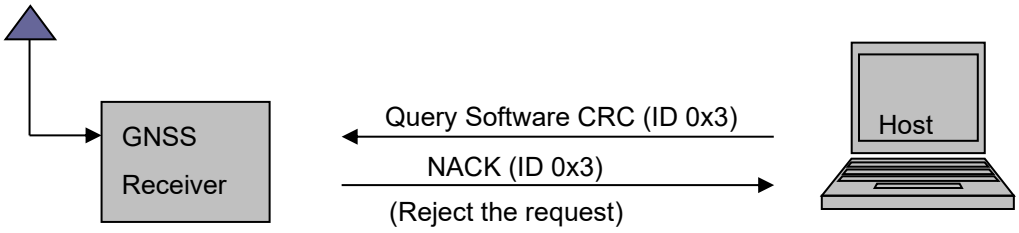



Figure 2

## MESSAGE LIST

This section provides brief information about available SkyTraq binary input, output and sub-id messages shown in a tabular list. All the messages are listed by Message ID. Full descriptions of input and output messages will be described in later sections.

| Input Messages | | | | |
|---|---|---|---|---|
| ID (Hex) | ID (Decimal) | Attribute | Name | Descriptions |
| 0x1 | 1 | Input | System Restart | Force system to restart |
| 0x2 | 2 | Input | Query Software version | Query revision information of software |
| 0x3 | 3 | Input | Query Software CRC | Query the CRC of the software |
| 0x4 | 4 | Input | Set Factory Defaults | Set system to factory default values |
| 0x5 | 5 | Input | Configure Serial Port | Set up serial port COM, baud rate, data bits, stop bits and parity |
| 0x9 | 9 | Input | Configure Message Type | Configure and select the output message type |
| 0xC | 12 | Input | Configure Power Mode | Set system power mode |
| 0xE | 14 | Input | Configure Position Update Rate | Configure the position update rate of GNSS system |
| 0x10 | 16 | Input | Query Position Update Rate | Query the position update rate of GNSS system |
| 0x15 | 21 | Input | Query Power Mode | Query the power mode status of GNSS receiver |
| 0x2A | 42 | Input | Configure DOP Mask | Configure values of DOP mask |
| 0x2B | 43 | Input | Configure Elevation and CNR Mask | Configure values of Elevation and CNR Mask |
| 0x2E | 46 | Input | Query DOP Mask | Query the information of DOP mask used by GNSS receiver |
| 0x2F | 47 | Input | Query Elevation and CNR Mask | Query the values of elevation mask and CNR mask used by GNSS receiver |
| 0x30 | 48 | Input | Get GPS Ephemeris | Retrieve GPS ephemeris data of the GNSS receiver |
| 0x39 | 57 | Input | Configure Position Pinning | Enable or disable position pinning of GNSS receiver |
| 0x3A[*1] | 58 | Input | Query Position Pinning | Query position pinning status of the GNSS receiver |

| | | | | |
|---|---|---|---|---|
| 0x3B*1 | 59 | Input | Configure Position Pinning Parameters | Set position pinning parameters of GNSS receiver |
| 0x41 | 65 | Input | Set GPS Ephemeris | Set GPS ephemeris data to the GNSS receiver |
| 0x44*1 | 68 | Input | Query 1PPS Timing | Query 1PPS timing of the GNSS receiver |
| 0x45 | 69 | Input | Configure 1PPS Cable Delay | Configure cable delay of 1PPS timing |
| 0x46 | 70 | Input | Query 1PPS Cable Delay | Query 1PPS cable delay of GNSS receiver |
| 0x4B | 75 | Input | Configure NMEA talker ID | Configure NMEA talker ID of GNSS receiver |
| 0x4F | 79 | Input | Query NMEA talk ID | Query NMEA talker ID of GNSS receiver |
| 0x54*1 | 84 | Input | Configure 1PPS Timing | Configure 1PPS timing of GNSS receiver |
| 0x55*1 | 85 | Input | Configure 1PPS Output Mode | Configure 1PPS Output Mode of GNSS receiver |
| 0x56*1 | 86 | Input | Query 1PPS Output Mode | Query 1PPS Output Mode of GNSS receiver |
| 0x5B | 91 | Input | Get GLONASS Ephemeris | Retrieve GLONASS ephemeris data of the GNSS receiver |
| 0x5C | 92 | Input | Set GLONASS Ephemeris | Set GLONASS ephemeris data to the GNSS receiver |
| 0x5F | 95 | Input | Get GLONASS Time Correction Parameters | Retrieve GLONASS time correction parameters $\tau_C$ and $\tau_{GPS}$ of the GNSS receiver |
| 0x60 | 96 | Input | Set GLONASS Time Correction Parameters | Set GLONASS time correction parameters $\tau_C$ and $\tau_{GPS}$ to the GNSS receiver |
| Messages with Sub-ID | | | | |
| ID/SubID (Hex) | ID/Sub ID (Decimal) | Attribute | Name | Descriptions |
| 0x62/0x1 | 98/1 | Input | Configure SBAS | Configure SBAS parameters of GNSS receiver |
| 0x62/0x2 | 98/2 | Input | Query SBAS Status | Query SBAS status of GNSS receiver |
| 0x62/0x3 | 98/3 | Input | Configure QZSS | Configure QZSS parameters of GNSS receiver |
| 0x62/0x4 | 98/4 | Input | Query QZSS Status | Query QZSS status of GNSS receiver |
| 0x62/0x5 | 98/5 | Input | Configure SBAS Advanced | Configure SBAS advanced functions of GNSS receiver |

| | | | | |
|---|---|---|---|---|
| 0x62/0x6 | 98/6 | Input | Query SBAS Advanced | Query SBAS advanced functions of GNSS receiver |
| 062/0x80 | 98/128 | Output | SBAS Status | SBAS status of GNSS receiver |
| 062/0x81 | 98/129 | Output | QZSS Status | QZSS status of GNSS receiver |
| 062/0x82 | 98/130 | Output | SBAS Advanced | SBAS advanced function of GNSS receiver |
| 0x63/0x1 | 99/1 | Input | Configure SAEE | Configure SAEE of GNSS receiver |
| 0x63/0x2 | 99/2 | Input | Query SAEE | Query SAEE of GNSS receiver |
| 0x63/0x80 | 99/128 | Output | SAEE status | SAEE status of GNSS receiver |
| 0x64/0x1 | 100/1 | Input | Query Boot Status | Query boot status of GNSS receiver |
| 0x64/0x2 | 100/2 | Input | Configure Extended NMEA Message Interval | Configure extended NMEA message interval of GNSS Receiver |
| 0x64/0x3 | 100/3 | Input | Query Extended NMEA Message Interval | Query extended NMEA message interval of GNSS receiver |
| 0x64/0x6 | 100/6 | Input | Configure Interference Detection | Configure interference detection of GNSS receiver |
| 0x64/0x7 | 100/7 | Input | Query Interference Detection Status | Query interference detection status of GNSS receiver |
| 0x64/0xB | 100/11 | Input | Query GNSS Parameter Search Engine Number | Query parameter search engine number of GNSS receiver |
| 0x64/0x11 | 100/17 | Input | Configure Position Fix Navigation Mask | Configure the position fix mask of GNSS receiver |
| 0x64/0x12 | 100/18 | Input | Query Position Fix Navigation Mask | Query the position fix of GNSS receiver |
| 0x64/0x15 | 100/21 | Input | Configure UTC Reference Time Sync to GPS Time | Configure UTC reference time to GNSS receiver to synchronize to GPS time |
| 0x64/0x16 | 100/22 | Input | Query UTC Reference Time Sync to GPS Time | Query the UTC reference time of GNSS receiver set to synchronize to GPS time |
| 0x64/0x17 | 100/23 | Input | Configure GNSS Navigation Mode | Configure the navigation mode of GNSS receiver |
| 0x64/0x18 | 100/24 | Input | Query GNSS Navigation Mode | Query the navigation mode of GNSS receiver |
| 0x64/0x19 | 100/25 | Input | Configure GNSS constellation type for navigation solution | Configure the GNSS constellation type used for navigation solution |
| 0x64/0x1A | 100/26 | Input | Query GNSS | Query the GNSS constellation type used |

| | | | | |
|---|---|---|---|---|
| | | | constellation type for navigation solution | for navigation solution |
| 0x64/0x1B | 100/27 | Input | Software Image Download Using ROM External Loader | Software image download using ROM external loader to system flash |
| 0x64/0x1C | 100/28 | Input | Configure GNSS Doze Mode | Configure the doze mode of GNSS receiver |
| 0x64/0x20 | 100/32 | Input | Query GPS Time | Query GPS time of GNSS receiver |
| 0x64/0x21 | 100/33 | Input | Configure PSTI Message Interval | Configure the PSTI message interval of GNSS receiver |
| 0x64/0x22 | 100/34 | Input | Query PSTI Message Interval | Query the PSTI message interval of GNSS receiver |
| 0x64/0x23 | 100/35 | Input | Query Requested PSTI Message Interval | Query message interval of the requested PSTI ID of GNSS receiver |
| 0x64/0x27 | 100/39 | Input | Configure GNSS Datum Index | Configure GNSS datum index of GNSS receiver |
| 0x64/0x28 | 100/40 | Input | Query GNSS Datum Index | Query GNSS datum index of GNSS receiver |
| 0x64/0x2D | 100/45 | Input | Configure GPS/UTC Leap Seconds in UTC | Configure GPS/UTC leap seconds in UTC of GNSS receiver |
| 0x64/0x2F | 100/47 | Input | Configure Navigation Data Message Interval | Configure the navigation output message interval of GNSS receiver |
| 0x64/0x30 | 100/48 | Input | Query Navigation Data Message Interval | Query the navigation output message interval of GNSS receiver |
| 0x64/0x34 | 100/52 | Input | Configure GNSS Geo-Fencing Data by Polygon | Configure geo-fencing data by polygon to GNSS receiver |
| 0x64/0x35 | 100/53 | Input | Query GNSS Geo-Fencing Data by Polygon | Query geo-fencing data by polygon of GNSS receiver |
| 0x64/0x36 | 100/54 | Input | Query GNSS Multi-Polygon Geo-Fencing Result | Query multi-polygon geo-fencing result of GNSS receiver |
| 0x64/0x3B | 100/59 | Input | Configure NMEA String Interval | Configure the NMEA String interval of GNSS receiver |
| 0x64/0x3C | 100/60 | Input | Query NMEA String Interval | Query the NMEA String interval of GNSS receiver |
| 0x64/0x40 | 100/64 | Input | Query Requested NMEA String Interval | Query message interval of the requested NMEA String of GNSS receiver |

| | | | | |
|---|---|---|---|---|
| 0x64/0x4E | 100/78 | Input | Software Image Download Using Internal Loader | Software image download using internal loader to system flash |
| 0x64/0x4F | 100/79 | Input | Software Image Download Using External Loader | Software image download using external loader to system flash |
| 0x64/0x7D | 100/125 | Input | Query Version Extension String | Query version extension string of GNSS receiver |
| 0x64/0x80 | 100/128 | Output | GNSS Boot Status | Boot status of the GNSS receiver |
| 0x64/0x81 | 100/129 | Output | Extended NMEA Message Interval | Extended NMEA message interval of GNSS receiver |
| 0x64/0x83 | 100/131 | Output | Interference Detection Status | Interference detection status of GNSS receiver |
| 0x64/0x85 | 100/133 | Output | GNSS Parameter search engine number | Parameter search engine number of GNSS receiver |
| 0x64/0x88 | 100/136 | Output | Position Fix Navigation Mask | Position fix navigation mask of GNSS receiver |
| 0x64/0x8A | 100/138 | Output | GPS UTC Reference Time | UTC reference time of GNSS receiver that synchronizes to GPS time |
| 0x64/0x8B | 100/139 | Output | GNSS Navigation Mode | Navigation mode of GNSS receiver |
| 0x64/0x8C | 100/140 | Output | GNSS Constellation Type for Navigation Solution | Replying the GNSS constellation type used for navigation solution |
| 0x64/0x8E | 100/142 | Output | GPS Time | GPS time of GNSS receiver |
| 0x64/0x8F | 100/143 | Output | PSTI Message Interval | PSTI message interval of GNSS receiver |
| 0x64/0x90 | 100/144 | Output | Requested PSTI Message Interval | PSTI message interval of requested PSTI ID of GNSS receiver |
| 0x64/0x92 | 100/146 | Output | GNSS Datum Index | Datum Index of GNSS receiver |
| 0x64/0x98 | 100/152 | Output | Navigation Data Message Interval | Navigation data message interval of GNSS receiver |
| 0x64/0x99 | 100/153 | Output | GNSS Geo-Fencing Data by Polygon | Geo-Fencing Data by Polygon of GNSS receiver |
| 0x64/0x9A | 100/154 | Output | GNSS Multi-Polygon Geo-Fencing Result | Multi-Polygon Geo-Fencing Result of GNSS receiver |
| 0x64/0x9D | 100/157 | Output | NMEA String Interval | NMEA string Interval of GNSS receiver |
| 0x64/0x9F | 100/159 | Output | Requested NMEA String Interval | Message interval of requested NMEA String of GNSS receiver |
| 0x64/0xFE | 100/254 | Output | Version Extension String | Version extension string of GNSS receiver |

| | | | | |
|---|---|---|---|---|
| 0x65/0x1 | 101/1 | Input | Configure 1PPS Pulse Width | Configure 1PPS pulse width of GNSS receiver |
| 0x65/0x2 | 101/2 | Input | Query 1PPS Pulse Width | Query 1PPS pulse width of GNSS receiver |
| 0x65/0x3 | 101/3 | Input | Configure PPS2 Frequency Output | Configure PPS2 frequency output of GNSS receiver |
| 0x65/0x4 | 101/4 | Input | Query PPS2 Frequency Output | Query PPS2 frequency output of GNSS receiver |
| 0x65/0x80 | 101/128 | Output | 1PPS Pulse Width | 1PPS pulse width of GNSS receiver |
| 0x65/0x81 | 101/129 | Output | GNSS PPS2 Frequency Output | PPS2 frequency output of GNSS receiver |
| 0x67/0x1 | 103/1 | Input | Set Beidou Ephemeris Data | Set BEIDOU ephemeris data to the GNSS receiver |
| 0x67/0x2 | 103/2 | Input | Get Beidou Ephemeris Data | Retrieve BEIDOU ephemeris data of the GNSS receiver |
| 0x67/0x80 | 103/128 | Output | Beidou Ephemeris Data | Beidou ephemeris data of the GNSS receiver |
| 0x6A/0x6*2 | 106/6 | Input | Configure RTK Mode and Operational Function | Configure Real Time Kinematic mode and operational function of GNSS receiver |
| 0x6A/0x7*2 | 106/7 | Input | Query RTK Mode and Operational Function | Query Real Time Kinematic mode and operational function of GNSS receiver |
| 0x6A/0xC*2 | 106/12 | Input | Configure RTK slave base serial port baud rate | Configure RTK slave base serial port baud rate |
| 0x6A/0xD*2 | 106/13 | Input | Query RTK slave base serial port baud rate | Query RTK slave base serial port baud rate |
| 0x6A/0x13*2 | 106/19 | Input | Configure RTK Precisely Kinematic Base serial port baud rate | Configure RTK precisely kinematic base serial port baud rate |
| 0x6A/0x14*2 | 106/20 | Input | Query RTK Precisely Kinematic Base serial port baud rate | Query RTK precisely kinematic base serial port baud rate |
| 0x6A/0x15*2 | 106/21 | Input | Configure RTK Rover Moving Base Heading and Pitch Offsets | Configure RTK rover moving base heading and pitch offsets of GNSS receiver |
| 0x6A/0x16*2 | 106/22 | Input | Query RTK Rover Moving Base Heading | Query RTK rover moving base heading and pitch offsets of GNSS receiver |

| | | | | |
|---|---|---|---|---|
| | | | and Pitch Offsets | |
| 0x6A/0x83[*2] | 106/131 | Output | RTK Mode and Operational Function | Real Time Kinematic mode and operational function of GNSS receiver |
| 0x6A/0x85[*2] | 106/133 | Output | RTK Slave Base serial port baud rate | RTK Slave Base Serial port baud rate |
| 0x6A/0x88[*2] | 106/136 | Output | RTK Precisely Kinematic Base serial port baud rate | RTK precisely kinematic Base Serial port baud rate |
| 0x6A/0x89[*2] | 106/137 | Output | Heading and Pitch Offsets of RTK Rover Moving Base | Heading and pitch offsets of RTK rover moving base GNSS receiver |
| 0x6E/0x1 | 110/1 | Input | Set GALILEO ephemeris | Set GALILEO ephemeris data to the GNSS receiver |
| 0x6E/0x2 | 110/2 | Input | Get GALILEO ephemeris | Retrieve GALILEO ephemeris data of the GNSS receiver |
| 0x6E/0x80 | 110/80 | Output | GALILEO ephemeris data | GALILEO ephemeris data of the GNSS receiver |
| 0x6F/0x3 | 106/3 | Input | Set IRNSS ephemeris | Set IRNSS ephemeris data to the GNSS receiver |
| 0x6F/0x4 | 106/4 | Input | Get IRNSS ephemeris | Retrieve IRNSS ephemeris data of the GNSS receiver |
| 0x6F/0x81 | 106/129 | Output | IRNSS ephemeris data | IRNSS ephemeris data of the GNSS receiver |
| Output Messages | | | | |
| ID (Hex) | ID (Decimal) | Attribute | Name | Descriptions |
| 0x80 | 128 | Output | Software Version | Software revision of the receiver |
| 0x81 | 129 | Output | Software CRC | Software CRC of the receiver |
| 0x82 | 130 | Output | Reserved | Reserved |
| 0x83 | 131 | Output | ACK | ACK to a successful input message |
| 0x84 | 132 | Output | NACK | Response to an unsuccessful input message |
| 0x86 | 134 | Output | Position Update Rate | Position update rate of GNSS system |
| 0x90 | 144 | Output | GLONASS Ephemeris Data | GLONASS ephemeris data of the GNSS receiver |
| 0x92 | 146 | Output | GLONASS Time Correction Parameters | GLONASS time correction parameters $\tau_C$ and $\tau_{GPS}$ |
| 0x93 | 147 | Output | GNSS NMEA Talker ID | NMEA Talker ID of GNSS receiver |
| 0xA8 | 168 | Output | Navigation Data | Output user navigation data in binary |

| | | | Message | format |
|---|---|---|---|---|
| 0xAF | 175 | Output | GNSS DOP Mask | DOP Mask used by the GNSS receiver |
| 0xB0 | 176 | Output | Elevation and CNR Mask | Elevation and CNR Mask used by the GNSS receiver |
| 0xB1 | 177 | Output | GPS Ephemeris Data | GPS ephemeris data of the GNSS receiver |
| 0xB4 | 180 | Output | GNSS Position Pinning Status | Position pinning status of the GNSS receiver |
| 0xB9 | 185 | Output | GNSS Power Mode Status | Power mode status of GNSS receiver |
| 0xBB | 187 | Output | GNSS 1PPS Cable Delay | 1PPS cable delay of the GNSS receiver |
| 0xC2[*1] | 194 | Output | GNSS 1PPS Timing | 1PPS timing information of the GNSS receiver |
| 0xC3[*1] | 195 | Output | GNSS 1PPS Output Mode | 1PPS output mode of the GNSS receiver |

| Messages with Sub-ID and Sub Sub-ID | | | | | |
|---|---|---|---|---|---|
| ID (Hex/ Decimal) | Sub-ID (Hex/Decimal) | Sub Sub-ID (Hex/Decimal) | Attribute | Name | Descriptions |
| 0x7A/122 | 0xE/14 | 0x1/1 | Input | Query PX1172RH Rover Moving Base SW Version | Query the software version of rover moving base receiver of PX1172RH GNSS receiver |
| 0x7A/122 | 0xE/14 | 0x2/2 | Input | Query PX1172RH Rover Moving Base SW CRC | Query the software CRC of rover moving base receiver of PX1172RH GNSS receiver |
| 0x7A/122 | 0xE/14 | 0x3/3 | Input | Query PX1172RH Rover Moving Base Position Update Rate | Query the position update rate of rover moving base receiver of PX1172RH GNSS receiver |
| 0x7A/122 | 0xE/14 | 0x4/4 | Input | Configure PX1172RH Rover Moving Base Heading and Pitch Offsets | Configure the heading and pitch offsets of rover moving base receiver of PX1172RH GNSS receiver |
| 0x7A/122 | 0xE/14 | 0x5/5 | Input | Query PX1172RH Rover Moving Base | Query the heading and pitch offsets of rover moving base receiver of |

| | | | | | |
|---|---|---|---|---|---|
| | | | | Heading and Pitch Offsets | PX1172RH GNSS receiver |
| 0x7A/122 | 0xE/14 | 0x80/128 | Output | Software Version of PX1172RH Rover Moving Base | Software version of rover moving base receiver of PX1172RH GNSS receiver |
| 0x7A/122 | 0xE/14 | 0x81/129 | Output | Software CRC of PX1172RH Rover Moving Base | Software CRC of rover moving base receiver of PX1172RH GNSS receiver |
| 0x7A/122 | 0xE/14 | 0x82/130 | Output | Rover Moving Base Position Update Rate of PX1172RH Rover Moving Base | Position update rate of rover moving base receiver of PX1172RH GNSS receiver |
| 0x7A/122 | 0xE/14 | 0x83/131 | Output | Heading and Pitch Offsets of PX1172RH Rover Moving Base | Heading and Pitch offsets of rover moving base receiver of PX1172RH GNSS receiver |

*1 supported only in timing mode receivers.

*2 supported only in RTK mode receivers

## INPUT MESSAGES

### *SYSTEM RESTART – Force System to restart (0x1)*

This is a request message which will reset and restart the GNSS receiver. This command is issued from the host to GNSS receiver and GNSS receiver should respond with an ACK or NACK. The payload length is 15 bytes.

Structure:

<0xA0,0xA1>< PL><01>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 0F 01 01 07 D8 0B 0E 08 2E 03 09 C4 30 70 00 64 16 0D 0A

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 01 | | UINT8 | - |
| 2 | Start Mode | 01 | 00 = Reserved<br>01 = System Reset, Hot start<br>02 = System Reset, Warm start<br>03 = System Reset, Cold start<br>04 = Reserved | UINT8 | |
| 3-4 | UTC Year | 07D8 | >= 1980 | UINT16 | |
| 5 | UTC Month | 0B | 1 ~ 12 | UINT8 | |
| 6 | UTC Day | 0E | 1 ~ 31 | UINT8 | |
| 7 | UTC Hour | 08 | 0 ~ 23 | UINT8 | |
| 8 | UTC Minute | 2E | 0 ~ 59 | UINT8 | |
| 9 | UTC Second | 03 | 0 ~ 59 | UINT8 | |
| 10-11 | Latitude | 09C4 | Between – 9000 and 9000<br>> 0: North Hemisphere<br>< 0: South Hemisphere | SINT16 | 1/100 degree |
| 12-13 | Longitude | 3070 | Between – 18000 and 18000<br>> 0: East Hemisphere<br>< 0: West Hemisphere | SINT16 | 1/100 degree |
| 14-15 | Altitude | 0064 | Between –1000 and 18300 | SINT16 | meter |
| Payload Length : 15 bytes | | | | | |

***QUERY SOFTWARE VERSION – Query revision information of loaded software (0x2)***

This is a request message which is issued from the host to GNSS receiver to retrieve loaded software version. The GNSS receiver should respond with an ACK along with information of software version, ***"SOFTWARE VERSION, ID: 0x80"***, when succeeded and should respond with an NACK when failed. The payload length is 2 bytes.

Structure:

<0xA0,0xA1>< PL><02>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 02 02 00 02 0D 0A

1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 02 | | UINT8 | |
| 2 | Software Type | 00 | 00 = Reserved<br>01 = System code | UINT8 | |
| Payload Length : 2 bytes | | | | | |

***QUERY SOFTWARE CRC – Query CRC information of loaded software (0x3)***

This is a request message which is issued from the host to GNSS receiver to retrieve loaded software CRC. The GNSS receiver should respond with an ACK along with information of software CRC, ***"SOFTWARE CRC, ID: 0x81"***, when succeeded and should respond with an NACK when failed. The payload length is 2 bytes.

Structure:

<0xA0,0xA1>< PL><03>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 02 03 00 03 0D 0A

1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 03 | | UINT8 | |
| 2 | Software Type | 00 | 00 = Reserved<br>01 = System code | UINT8 | |
| Payload Length : 2 bytes | | | | | |

***SET FACTORY DEFAULTS – Set the system to factory default values (0x4)***

This is a request message which is issued from the host to GNSS receiver. It will reset the GNSS receiver's internal parameters to factory default values. The GNSS receiver should respond with an ACK when succeeded and should respond with an NACK when failed. The user data will be erased and filled with factory default values. The payload length is 2 bytes.

Structure:
<0xA0,0xA1>< PL><04>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 02 04 01 05 0D 0A
1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 04 | | UINT8 | |
| 2 | Type | 00 | 00 = Reserved<br>01 = reboot after setting to factory defaults | UINT8 | |
| Payload Length : 2 bytes | | | | | |

***CONFIGURE SERIAL PORT – Set up serial port property (0x5)***

This is a request message which will configure the serial COM port, baud rate. This command is issued from the host to GNSS receiver and GNSS receiver should respond with an ACK or NACK. The payload length is 4 bytes.

Structure:

<0xA0,0xA1>< PL><05>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 04 05 00 00 00 05 0D 0A

1 2 3 4

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 05 | | UINT8 | |
| 2 | COM port | 00 | 00 = COM 1 | UINT8 | |
| 3 | Baud Rate | 00 | 0: 4800<br>1: 9600<br>2: 19200<br>3: 38400<br>4: 57600<br>5: 115200<br>6: 230400<br>7: 460800<br>8: 921600 | UINT8 | |
| 4 | Attributes | 00 | 0: update to SRAM<br>1: update to both SRAM & FLASH<br>2. temporarily | UINT8 | |
| Payload Length : 4 bytes | | | | | |

***CONFIGURE MESSAGE TYPE – Configure and select output message type (0x9)*** [*1]

This is a request message which will change the GNSS receiver output message type. This command is issued from the host to GNSS receiver and GNSS receiver should respond with an ACK or NACK. The payload length is 3 bytes.

Structure:

<0xA0,0xA1>< PL><09>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 03 09 00 00 09 0D 0A

1 2 3

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 09 | | UINT8 | |
| 2 | Type | 00 | 00 : No output<br>01 : NMEA message<br>02 : Binary Message | UINT8 | |
| 3 | Attributes | 00 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 3 bytes | | | | | |

*1: not supported in RTK receivers.

***CONFIGURE SYSTEM POWER MODE –Set the power mode of GNSS system (0xC)***

This is a request message which is issued from the host to GNSS receiver to configure the system power mode. By default power save mode is enabled, to reduce current consumption by the search engine. The GNSS receiver should respond with an ACK when succeeded and should respond with an NACK when failed. The payload length is 3 bytes.

Structure:
<0xA0,0xA1>< PL><0C>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 03 0C 00 00 0C 0D 0A
1 2 3

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 0C | | UINT8 | |
| 2 | Mode | 00 | 00 = Normal (disable)<br>01 = Power Save (enable) | UINT8 | |
| 3 | Attributes | 00 | 0: update to SRAM<br>1: update to both SRAM & FLASH<br>2: temporarily enabled | UINT8 | |
| Payload Length : 3 bytes | | | | | |

***CONFIGURE SYSTEM POSITION RATE – Configure the position update rate of GNSS system (0xE)***

This is a request message which is issued from the host to GNSS receiver to configure the system position update rate. Receivers with position rate 4 or higher needs to configure baud rate to 38400 or higher value. The GNSS receiver should respond with an ACK when succeeded and should respond with an NACK when failed. The payload length is 3 bytes.

Structure:
<0xA0,0xA1>< PL><0E>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 03 0E 01 00 0F 0D 0A
1 2 3

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 0E | | UINT8 | |
| 2 | Rate | 01 | Value with 1, 2, 4, 5, 8, 10, 20, 25, 40, 50<br>01: 1Hz update rate<br>Note: value with 4 ~10 should work with baud rate 38400 or higher, value with 20 should work with baud rate 115200 or higher, value with 40, 50 should work with 230400.<br>Note: Firmware with default baud rate at 115200 with multi-constellation and with/without dual frequency at rate 10 or higher may require baud rate 230400 or 460800. | UINT8 | |
| 3 | Attributes | 00 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 3 bytes | | | | | |

***QUERY POSITION UPDATE RATE – Query the position update rate of GNSS system (0x10)***

This is a request message which is issued from the host to GNSS receiver to query position update rate. The GNSS receiver should respond with an ACK along with information of position update rate, ***"POSITION UPDATE RATE, ID: 0x86"***, when succeeded and should respond with an NACK when failed. The payload length is 1 byte.

Structure:

<0xA0,0xA1>< PL><10>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 01 10 10 0D 0A

1

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 10 | | UINT8 | |
| Payload Length : 1 byte | | | | | |

***QUERY POWER MODE – Query status of power mode of GNSS receiver (0x15)***

This is a request message which is issued from the host to GNSS receiver to query power mode status. The GNSS receiver should respond with an ACK along with power mode status, ***"GNSS POWER MODE STATUS, ID: 0xB9"***, when succeeded and should respond with an NACK when failed. The payload length is 1 byte.

Structure:
<0xA0,0xA1>< PL><15>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 01 15 15 0D 0A
1

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 15 | | UINT8 | |
| Payload Length : 1 byte | | | | | |

***CONFIGURE DOP MASK – Configure values of DOP mask (0x2A)***

This is a request message which will set the GNSS receiver DOP mode and its corresponding mask. This command is issued from the host to GNSS receiver and GNSS receiver should respond with an ACK or NACK. If either value of PDOP, HDOP or GDOP is not valid, the GNSS receiver will respond with an NACK. The payload length is 9 bytes.

Structure:

<0xA0,0xA1>< PL><2A>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 09 2A 01 00 32 00 32 00 32 00 19 0D 0A

1 2 3 4 5 6 7 8 9

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 2A | | UINT8 | |
| 2 | DOP Mode Select | 01 | 00 : Disable<br>01 : Auto mode, PDOP when 3-D fix and HDOP when 2-D fix<br>02 : PDOP only<br>03 : HDOP only<br>04 : GDOP only | UINT8 | |
| 3-4 | PDOP Value | 0032 | Valid values between 0.5~30<br>Valid input value 5 ~ 300 | UINT16 | 0.1 |
| 5-6 | HDOP Value | 0032 | Valid values between 0.5~30<br>Valid input value 5 ~ 300 | UINT16 | 0.1 |
| 7-8 | GDOP Value | 0032 | Valid values between 0.5~30<br>Valid input value 5 ~ 300 | UINT16 | 0.1 |
| 9 | Attributes | 00 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 9 bytes | | | | | |

***CONFIGURE ELEVATION AND CNR MASK – Configure values of elevation and CNR mask (0x2B)***

This is a request message which will configure the satellite elevation and CNR mask of GNSS receiver. This command is issued from the host to GNSS receiver and GNSS receiver should respond with an ACK or NACK. If either value of elevation or CNR mask is not valid, the GNSS receiver will respond with an NACK. The payload length is 5 bytes.

Structure:
<0xA0,0xA1>< PL><2B>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 05 2B 01 05 0A 00 25 0D 0A
1 2 3 4 5

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 2B | | UINT8 | |
| 2 | Elevation and CNR Mode Select | 01 | 00 : Disable<br>01 : Elevation and CNR both<br>02 : Elevation only<br>03 : CNR only | UINT8 | |
| 3 | Elevation Mask | 05 | Valid values between 3~85 | UINT8 | degree |
| 4 | CNR Mask | 0A | Valid values between 0~40 | UINT8 | dB |
| 5 | Attributes | 00 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 5 bytes | | | | | |

***QUERY DOP MASK – Query information of DOP mask used by the GNSS receiver (0x2E)***

This is a request message which is issued from the host to GNSS receiver to retrieve information of DOP mask. The GNSS receiver should respond with an ACK along with DOP mask information, ***"GNSS DOP MASK, ID: 0xAF"***, when succeeded and should respond with an NACK when failed. The payload length is 1 byte.

Structure:

<0xA0,0xA1>< PL><2E>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 01 2E 2E 0D 0A

1

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 2E | | UINT8 | |
| Payload Length : 1 byte | | | | | |

***QUERY ELEVATION AND CNR MASK – Query elevation and CNR mask used by the GNSS receiver (0x2F)***

This is a request message which is issued from the host to GNSS receiver to retrieve information of elevation and CNR mask. The GNSS receiver should respond with an ACK along with elevation and CNR mask information, ***"GNSS ELEVATION AND CNR MASK, ID: 0xB0"***, when succeeded and should respond with an NACK when failed. The payload length is 1 byte.

Structure:

<0xA0,0xA1>< PL><2F>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 01 2F 2F 0D 0A

1

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 2F | | UINT8 | |
| Payload Length : 1 byte | | | | | |

***GET GPS EPHEMERIS – Get GPS ephemeris used of GNSS receiver (0x30)***

This is a request message which is issued from the host to GNSS receiver to retrieve GPS ephemeris data. The GNSS receiver should respond with an ACK along with information of ephemeris, ***"GPS EPHEMERIS DATA, ID: 0xB1"***, when succeeded and should respond with an NACK when failed. The payload length is 2 bytes.

Structure:

<0xA0,0xA1>< PL><30>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 02 30 00 30 0D 0A

1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 30 | | UINT8 | |
| 2 | SV # | 00 | 0: means all SVs<br>1~32 : mean for the particular SV | UINT8 | |
| Payload Length : 2 bytes | | | | | |

***CONFIGURE POSITION PINNING – Enable or disable position pinning of GNSS receiver (0x39)***

This is a request message which is issued from the host to GNSS receiver to configure the system position pinning. The GNSS receiver should respond with an ACK when succeeded and should respond with an NACK when failed. The payload length is 3 bytes.

Structure:

<0xA0,0xA1>< PL><39>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 03 39 01 01 39 0D 0A

1 2 3

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 39 | | UINT8 | |
| 2 | Position pinning | 01 | 0: default<br>1: enable<br>2: disable | UINT8 | |
| 3 | Attributes | 01 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 3 bytes | | | | | |

***QUERY POSITION PINNING – Query position pinning status of GNSS receiver (0x3A)***

This is a request message which is issued from the host to GNSS receiver to query position pinning status. The GNSS receiver should respond with an ACK along with position pinning status, ***"GNSS POSITION PINNING STATUS, ID: 0xB4"***, when succeeded and should respond with an NACK when failed. The payload length is 1 byte.

Structure:

<0xA0,0xA1>< PL><3A>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 01 3A 3A 0D 0A

1

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 3A | | UINT8 | |
| Payload Length : 1 byte | | | | | |

***CONFIGURE POSITION PINNING PARAMETERS – Set position pinning parameters of GNSS receiver (0x3B)***

This is a request message which is issued from the host to GNSS receiver to configure the system position pinning parameters. The GNSS receiver should respond with an ACK when succeeded and should respond with an NACK when failed. The payload length is 12 bytes.

Structure:

<0xA0,0xA1>< PL><3B>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 0C 3B 00 02 00 0A 00 08 00 2D 01 F4 01 E2 0D 0A

1 2 3 4 5 6 7 8 9 10 11 12

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 3B | | UINT8 | |
| 2-3 | Pinning speed | 0002 | | UINT16 | Km/Hr |
| 4-5 | Pinning cnt | 000A | | UINT16 | second |
| 6-7 | Unpinning speed | 0008 | | UINT16 | Km/Hr |
| 8-9 | Unpinning cnt | 002D | | UINT16 | second |
| 10-11 | Unpinning distance | 01F4 | | UINT16 | meter |
| 12 | Attributes | 01 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 12 bytes | | | | | |

***SET GPS EPHEMERIS – Set GPS ephemeris to GNSS receiver (0x41)***

This is a request message which is issued from the host to GNSS receiver to set GPS ephemeris data (open an ephemeris file) to GNSS receiver. The GNSS receiver should respond with an ACK when succeeded and should respond with an NACK when failed. The payload length is 87 bytes.

Structure:
<0xA0,0xA1>< PL><41>< message body><CS><0x0D,0x0A>

Example:

| A0 | A1 | 00 | 57 | 41 | 00 | 02 | 00 | 77 | 88 | 04 | 61 | 10 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | DB | DF | 59 | A6 | 00 | 00 | 1E |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 |

| 0A | 47 | 7C | 00 | 77 | 88 | 88 | DF | FD | 2E | 35 | A9 | CD | B0 | F0 | 9F | FD | A7 | 04 | 8E | CC | A8 | 10 | 2C | A1 | 0E | 22 | 31 | 59 | A6 | 74 | 00 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 |

| 77 | 89 | 0C | FF | A3 | 59 | 86 | C7 | 77 | FF | F8 | 26 | 97 | E3 | B9 | 1C | 60 | 59 | C3 | 07 | 44 | FF | A6 | 37 | DF | F0 | B0 | 2E | 0D | 0A |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 | 81 | 82 | 83 | 84 | 85 | 86 | 87 | | | |

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 41 | | UINT8 | |
| 2-3 | SV id | 0002 | Satellite id | UINT16 | |
| 4 | SubFrameData[0][0] | 77 | Eph data subframe 1 | UINT8 | |
| 5 | SubFrameData[0][1] | 88 | Eph data subframe 1 | UINT8 | |
| 6 | SubFrameData[0][2] | 04 | Eph data subframe 1 | UINT8 | |
| 7 | SubFrameData[0][3] | 61 | Eph data subframe 1 | UINT8 | |
| 8 | SubFrameData[0][4] | 10 | Eph data subframe 1 | UINT8 | |
| 9 | SubFrameData[0][5] | 00 | Eph data subframe 1 | UINT8 | |
| 10 | SubFrameData[0][6] | 00 | Eph data subframe 1 | UINT8 | |
| 11 | SubFrameData[0][7] | 00 | Eph data subframe 1 | UINT8 | |
| 12 | SubFrameData[0][8] | 00 | Eph data subframe 1 | UINT8 | |
| 13 | SubFrameData[0][9] | 00 | Eph data subframe 1 | UINT8 | |
| 14 | SubFrameData[0][10] | 00 | Eph data subframe 1 | UINT8 | |
| 15 | SubFrameData[0][11] | 00 | Eph data subframe 1 | UINT8 | |
| 16 | SubFrameData[0][12] | 00 | Eph data subframe 1 | UINT8 | |
| 17 | SubFrameData[0][13] | 00 | Eph data subframe 1 | UINT8 | |
| 18 | SubFrameData[0][14] | 00 | Eph data subframe 1 | UINT8 | |
| 19 | SubFrameData[0][15] | 00 | Eph data subframe 1 | UINT8 | |
| 20 | SubFrameData[0][16] | 00 | Eph data subframe 1 | UINT8 | |
| 21 | SubFrameData[0][17] | DB | Eph data subframe 1 | UINT8 | |

| | | | | | |
|---|---|---|---|---|---|
| 22 | SubFrameData[0][18] | DF | Eph data subframe 1 | UINT8 | |
| 23 | SubFrameData[0][19] | 59 | Eph data subframe 1 | UINT8 | |
| 24 | SubFrameData[0][20] | A6 | Eph data subframe 1 | UINT8 | |
| 25 | SubFrameData[0][21] | 00 | Eph data subframe 1 | UINT8 | |
| 26 | SubFrameData[0][22] | 00 | Eph data subframe 1 | UINT8 | |
| 27 | SubFrameData[0][23] | 1E | Eph data subframe 1 | UINT8 | |
| 28 | SubFrameData[0][24] | 0A | Eph data subframe 1 | UINT8 | |
| 29 | SubFrameData[0][25] | 47 | Eph data subframe 1 | UINT8 | |
| 30 | SubFrameData[0][26] | 7C | Eph data subframe 1 | UINT8 | |
| 31 | SubFrameData[0][27] | 00 | Eph data subframe 1 | UINT8 | |
| 32~59 | SubFrameData[1][0~27] | | Eph data subframe 2, same as field 4-31 | UINT8 | |
| 60-87 | SubFrameData[2][0~27] | | Eph data subframe 3, same as field 4-31 | UINT8 | |
| Payload Length : 87 bytes | | | | | |

***QUERY 1PPS TIMING – Query 1PPS timing of the GNSS receiver (0x44)*** [*1]

This is a request message which is issued from the host to GNSS receiver to query 1PPS timing information. The GNSS receiver should respond with an ACK along with information of 1PPS timing, **"*GNSS 1PPS TIMING, ID: 0xC2*"**, when succeeded and should respond with an NACK when failed. The payload length is 1 byte.

Structure:

<0xA0,0xA1>< PL><44>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 01 44 44 0D 0A

1

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 44 | | UINT8 | |
| Payload Length : 1 byte | | | | | |

*1: supported only in timing mode receivers.

***CONFIGURE 1PPS CABLE DELAY – Configure cable delay of 1PPS timing (0x45)***

This is a request message which will set the cable delay of 1PPS timing to the GNSS receiver. This command is issued from the host to GNSS receiver and GNSS receiver should respond with an ACK or NACK. If value of cable delay is not valid, the GNSS receiver will respond with an NACK. The payload length is 6 bytes.

Structure:

<0xA0,0xA1>< PL><45>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 06 45 00 00 00 00 00 45 0D 0A

1 2 3 4 5 6

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 45 | | UINT8 | - |
| 2-5 | Cable Delay | 00000000 | Cable delay adjustment for 1PPS<br>Valid input value -500000~+500000 | SINT32 | 1/100 ns |
| 6 | Attributes | 00 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 6 bytes | | | | | |

***QUERY 1PPS CABLE DELAY – Query 1PPS cable delay of the GNSS receiver (0x46)***

This is a request message which is issued from the host to GNSS receiver to query 1PPS cable delay. The GNSS receiver should respond with an ACK along with information of 1PPS cable delay, ***"GNSS 1PPS CABLE DELAY, ID: 0xBB"***, when succeeded and should respond with an NACK when failed. The payload length is 1 byte.

Structure:

<0xA0,0xA1>< PL><46>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 01 46 46 0D 0A

1

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 46 | | UINT8 | |
| Payload Length : 1 byte | | | | | |

***CONFIGURE NMEA TALKER ID – Configure NMEA talker ID of GNSS receiver (0x4B)***

This is a request message which will configure the type of talker ID (GP mode or GN mode) used in the NMEA output. This command is issued from the host to the receiver and the receiver should respond with an ACK or NACK. The payload length is 3 bytes.

Structure:

<0xA0,0xA1>< PL><4B>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 03 4B 01 01 4B 0D 0A

1 2 3

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 4B | | UINT8 | |
| 2 | Talker ID type | 01 | 0: GP mode<br>1: GN mode<br>2: Auto mode*1: according to NMEA 4.11 to combine GNSS system solution to output GN, GP, GL, GA, GB or GI appropriately. | UINT8 | |
| 3 | Attributes | 01 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 3 bytes | | | | | |

*1 supported only in NMEA version 4.11

GPS/GLONASS Receiver

| Mode 1 Talker ID GN | | Mode 2 Talker ID GP | |
|---|---|---|---|
| **$GNGGA** | Time, position, and fix related data of the receiver. | **$GPGGA** | Time, position, and fix related data of the receiver. |
| **$GNGLL** | Position, time and fix status. | **$GPGLL** | Position, time and fix status. |
| **$GNGSA**<br>**$GPGSA**<br>**$GLGSA** | Used to represent the ID's of satellites which are used for position fix. When both GPS and GLONASS satellites are used in position solution, a $GNGSA sentence is used for GPS satellites and another $GNGSA sentence is used for GLONASS satellites. When only GPS satellites are used for position fix, a single $GPGSA sentence is output. When only GLONASS satellites are used, a single $GLGSA sentence is output. | **$GPGSA**<br>**$GLGSA** | Used to represent the ID's of satellites which are used for position fix. When GPS satellites are used for position fix, $GPGSA sentence is output. When GLONASS satellites are used for position fix, $GLGSA sentence is output. |
| **$GPGSV**<br>**$GLGSV** | Satellite information about elevation, azimuth and CNR, $GPGSV is used for GPS satellites, while $GLGSV is used for GLONASS satellites | **$GPGSV**<br>**$GLGSV** | Satellite information about elevation, azimuth and CNR, $GPGSV is used for GPS satellites, while $GLGSV is used for GLONASS satellites |
| **$GNRMC** | Time, date, position, course and speed data. | **$GPRMC** | Time, date, position, course and speed data. |
| **$GNVTG** | Course and speed relative to the ground. | **$GPVTG** | Course and speed relative to the ground. |
| **$GNZDA** | UTC, day, month and year and time zone. | **$GPZDA** | UTC, day, month and year and time zone. |

GPS/Beidou Receiver

| Mode 1 Talker ID GN | | Mode 2 Talker ID GP | |
|---|---|---|---|
| **$GNGGA** | Time, position, and fix related data of the receiver. | **$GPGGA** | Time, position, and fix related data of the receiver. |
| **$GNGLL** | Position, time and fix status. | **$GPGLL** | Position, time and fix status. |
| **$GNGSA**<br>**$GPGSA**<br>**$BDGSA** | Used to represent the ID's of satellites which are used for position fix. When both GPS and Beidou satellites are used in position solution, a $GNGSA sentence is used for GPS satellites and another $GNGSA sentence is used for Beidou satellites. When only GPS satellites are used for position fix, a single $GPGSA sentence is output. When only Beidou satellites are used, a single $BDGSA sentence is output. | **$GPGSA**<br>**$BDGSA** | Used to represent the ID's of satellites which are used for position fix. When GPS satellites are used for position fix, $GPGSA sentence is output. When Beidou satellites are used for position fix, $BDGSA sentence is output. |
| **$GPGSV**<br>**$BDGSV** | Satellite information about elevation, azimuth and CNR, $GPGSV is used for GPS satellites, while $BDGSV is used for Beidou satellites | **$GPGSV**<br>**$BDGSV** | Satellite information about elevation, azimuth and CNR, $GPGSV is used for GPS satellites, while $BDGSV is used for Beidou satellites |
| **$GNRMC** | Time, date, position, course and speed data. | **$GPRMC** | Time, date, position, course and speed data. |
| **$GNVTG** | Course and speed relative to the ground. | **$GPVTG** | Course and speed relative to the ground. |
| **$GNZDA** | UTC, day, month and year and time zone. | **$GPZDA** | UTC, day, month and year and time zone. |

***QUERY NMEA TALKER ID – Query NMEA talker ID of GNSS receiver (0x4F)***

This is a request message which is issued from the host to GNSS receiver to query the talker ID. The GNSS receiver should respond with an ACK along with information of talker ID, ***"GNSS NMEA TALKER ID, ID: 0x93"***, when succeeded and should respond with an NACK when failed. The payload length is 1 byte.

Structure:

<0xA0,0xA1>< PL><4F>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 01 4F 4F 0D 0A

1

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 4F | | UINT8 | |
| Payload Length : 1 byte | | | | | |

***CONFIGURE 1PPS TIMING – Configure 1PPS timing of the GNSS receiver (0x54)*[1]**

This is a request message which will configure 1PPS timing of the GNSS receiver. This command is issued from the host to GNSS receiver and GNSS receiver should respond with an ACK or NACK. If value of survey length is not valid, the GPS receiver will respond with an NACK. The payload length is 31 bytes.

Structure:
<0xA0,0xA1>< PL><54>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 1F 54 00 00 00 07 D0 00 00 00 1E 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27

00 00 00 01 9C 0D 0A
28 29 30 31

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 54 | | UINT8 | - |
| 2 | Timing Mode | 00 | 00 = Timing PVT Mode<br>01 = Timing Survey Mode<br>02 = Timing Static Mode | UINT8 | |
| 3-6 | Survey Length | 000007D0 | Survey length when in Timing Survey Mode not used when in other mode.<br>Valid values between 60~1209600 | UINT32 | |
| 7-10 | Standard Deviation | 0000001E | Standard Deviation when in Timing Survey Mode not used when in other mode.<br>Valid values between 3~100 | UINT32 | |
| 11-18 | Latitude | 0000000000000000 | Latitude in double in Timing Static Mode not used when in other mode. | DPFP | |
| 19-26- | Longitude | 0000000000000000 | Longitude in double in Timing Static Mode not used when in other mode. | DPFP | |
| 27-30 | Altitude | 00000000 | Altitude in float in Timing Static Mode not used when in other mode. | SPFP | |
| 31 | Attributes | 01 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 31 bytes | | | | | |

*1: supported only in timing mode receivers.

**Remark:**

When using PVT Mode, precision 1PPS won't be generated with less than 4 satellites.
When using Survey Mode, survey length need to be given, the receiver will survey its location for number of specified points, and then change to Static Mode such that precision 1PPS will still be generated with 1 satellite in view.

Use Static Mode when location is known, latitude/longitude/altitude need to be set, and receiver will generate precision 1PPS output down to 1 satellite in view.

For attribute setting specifying "update to SRAM", it will make the setting take effect in the current session. Later if the receiver is turned off and SRAM / RTC backup supply source is still provided, then upon power up receiver will go into survey process if Survey Mode was chosen, or pinned to a fixed location if Static Mode was chosen. If without backup supply source and recycling power, the receiver will start in the default Survey Mode.

For attribute setting specifying "update to both SRAM and Flash", it will make the setting take effect in the current session. Later if the receiver is turned off then upon power up receiver will go into survey process if Survey Mode was chosen, or pinned to a fixed location if Static Mode was chosen.

***CONFIGURE 1PPS OUTPUT MODE – Configure 1PPS output mode of the GNSS receiver (0x55)*[1]**

This is a request message which will configure 1PPS output mode of the GNSS receiver. This command is issued from the host to GNSS receiver and GNSS receiver should respond with an ACK or NACK. The payload length is 4 bytes.

Structure:
<0xA0,0xA1>< PL><55>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 04 55 00 00 01 54 0D 0A
1 2 3 4

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 55 | | UINT8 | - |
| 2 | Output Mode | 00 | 00 = Reserved<br>01 = Output if GNSS time is available<br>02 = Output always and align to GNSS time automatically | UINT8 | |
| 3 | Align Source | 00 | 00 = Align to GNSS<br>01 = Align to UTC | UINT8 | |
| 4 | Attributes | 01 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 4 bytes | | | | | |

*1: supported only in timing mode receivers.

***QUERY 1PPS OUTPUT MODE – Query 1PPS output mode of the GNSS receiver (0x56)*** [*1]

This is a request message which is issued from the host to GNSS receiver to query 1PPS output mode. The GNSS receiver should respond with an ACK along with information of 1PPS output mode, **"*GNSS 1PPS OUTPUT MODE, ID: 0xC3***", when succeeded and should respond with an NACK when failed. The payload length is 1 byte.

Structure:

<0xA0,0xA1>< PL><56>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 01 56 56 0D 0A

1

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 56 | | UINT8 | |
| Payload Length : 1 byte | | | | | |

*1: supported only in timing mode receivers.

***GET GLONASS EPHEMERIS – GET GLONASS EPHEMERIS USED OF THE GNSS RECEIVER (0X5B)***

This is a request message which is issued from the host to the receiver to retrieve GLONASS ephemeris data. The receiver should respond with an ACK along with information of ephemeris, ***"GLONASS EPHEMERIS DATA, ID: 0x90"***, when succeeded and should respond with an NACK when failed. The payload length is 2 bytes.

Structure:
<0xA0,0xA1>< PL><5B>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 02 5B 01 5A 0D 0A
1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 5B | | UINT8 | |
| 2 | GLONASS SV slot number | 01 | 0: means all SVs<br>1~32 : mean for the particular SV | UINT8 | |
| Payload Length : 2 bytes | | | | | |

***SET GLONASS EPHEMERIS – Set GLONASS ephemeris to the GNSS receiver (0x5C)***

This is a request message which is issued from the host to the receiver to set GLONASS ephemeris data (open an ephemeris file) to the receiver. The receiver should respond with an ACK when succeeded and should respond with a NACK when failed. The payload length is 43 bytes.

Structure:
<0xA0,0xA1>< PL><5C>< message body><CS><0x0D,0x0A>

Example:

| A0 | A1 | 00 | 2B | 5C | 01 | 01 | 01 | 07 | 43 | 0F | AC | 06 | 89 | A2 | 01 | 9A | 02 | 17 | 60 | 28 | 75 | 47 | 01 | 16 | FE | B5 | 03 | 80 | 06 | 9C | CB |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 |

| CC | 92 | 6A | C0 | 42 | 04 | 09 | 94 | 79 | 20 | 00 | 00 | 20 | 11 | 85 | 2F | 0D | 0A |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | | | |

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 5C | | UINT8 | |
| 2 | Slot number | 01 | GLONASS SV slot number | UINT8 | |
| 3 | K number | 01 | GLONASS SV frequency number (-7 ~ +6) | SINT8 | |
| 4 | glo_eph_data0_byte0 | 01 | Stuffing zeros and bit 85 - bit 81 (LSB) of string 1 | UINT8 | |
| 5 | glo_eph_data0_byte1 | 07 | bit 80 (MSB)- bit 73 (LSB) of string 1 | UINT8 | |
| 6 | glo_eph_data0_byte2 | 43 | bit 72 (MSB)- bit 65 (LSB) of string 1 | UINT8 | |
| 7 | glo_eph_data0_byte3 | 0F | bit 64 (MSB)- bit 57 (LSB) of string 1 | UINT8 | |
| 8 | glo_eph_data0_byte4 | AC | bit 56 (MSB)- bit 49 (LSB) of string 1 | UINT8 | |
| 9 | glo_eph_data0_byte5 | 06 | bit 48 (MSB)- bit 41 (LSB) of string 1 | UINT8 | |
| 10 | glo_eph_data0_byte6 | 89 | bit 40 (MSB)- bit 33 (LSB) of string 1 | UINT8 | |
| 11 | glo_eph_data0_byte7 | A2 | bit 32 (MSB)- bit 25 (LSB) of string 1 | UINT8 | |
| 12 | glo_eph_data0_byte8 | 01 | bit 24 (MSB)- bit 17 (LSB) of string 1 | UINT8 | |
| 13 | glo_eph_data0_byte9 | 9A | bit 16 (MSB)- bit 09 (LSB) of string 1 | UINT8 | |
| 14 | glo_eph_data1_byte0 | 02 | Stuffing zeros and bit 85 - bit 81 (LSB) of string 2 | UINT8 | |
| 15 | glo_eph_data1_byte1 | 17 | bit 80 (MSB)- bit 73 (LSB) of string 2 | UINT8 | |
| 16 | glo_eph_data1_byte2 | 60 | bit 72 (MSB)- bit 65 (LSB) of string 2 | UINT8 | |
| 17 | glo_eph_data1_byte3 | 28 | bit 64 (MSB)- bit 57 (LSB) of string 2 | UINT8 | |
| 18 | glo_eph_data1_byte4 | 75 | bit 56 (MSB)- bit 49 (LSB) of string 2 | UINT8 | |

| | | | | | |
|---|---|---|---|---|---|
| 19 | glo_eph_data1_byte5 | 47 | bit 48 (MSB)- bit 41 (LSB) of string 2 | UINT8 | |
| 20 | glo_eph_data1_byte6 | 01 | bit 40 (MSB)- bit 33 (LSB) of string 2 | UINT8 | |
| 21 | glo_eph_data1_byte7 | 16 | bit 32 (MSB)- bit 25 (LSB) of string 2 | UINT8 | |
| 22 | glo_eph_data1_byte8 | FE | bit 24 (MSB)- bit 17 (LSB) of string 2 | UINT8 | |
| 23 | glo_eph_data1_byte9 | B5 | bit 16 (MSB)- bit 09 (LSB) of string 2 | UINT8 | |
| 24-33 | glo_eph_data2_byte0 - glo_eph_data2_byte9 | | Stuffing-zeros and bit 85 - bit 09 of string 3 | | |
| 34-43 | glo_eph_data3_byte0 – glo_eph_data3_byte9 | | Stuffing-zeros and bit 85 - bit 09 of string 4 | | |
| Payload Length : 43 bytes | | | | | |

***GET GLONASS TIME CORRECTION PARAMETERS – Get GLONASS time correction parameters (0x5F)***

This is a request message which is issued from the host to the receiver to retrieve GLONASS time correction data. The receiver should respond with an ACK along with information of time correction, ***"GLONASS TIME CORRECTION, ID: 0x92"***, when succeeded and should respond with an NACK when failed. The payload length is 1 byte.

Structure:
<0xA0,0xA1>< PL><5F>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 01 5F 5F 0D 0A
1

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 5F | | UINT8 | |
| Payload Length : 1 byte | | | | | |

***SET GLONASS TIME CORRECTION PARAMETERS – Set GLONASS time correction parameters to the GNSS receiver (0x60)***

This is a request message which is issued from the host to the receiver to set GLONASS time correction data ( $\tau_{GPS}$ and $\tau_C$ ) to the receiver. The receiver should respond with an ACK when succeeded and should respond with an NACK when failed. The payload length is 10 bytes.

Structure:
<0xA0,0xA1>< PL><60>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 0A 60 FF FF FF BF 00 00 00 14 00 34 0D 0A
1 2 3 4 5 6 7 8 9 10

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 60 | | UINT8 | |
| 2-5 | $\tau_C$ | FFFFFFBF | GLONASS time scale correction to UTC(SU) time | SINT32 | $2^{-31}$ sec |
| 6-9 | $\tau_{GPS}$ | 00000014 | Correction to GPS time relative to GLONASS time | SINT32 | $2^{-30}$ sec |
| 10 | attributes | 00 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 10 bytes | | | | | |

# MESSAGES WITH Sub-ID*1

*1: Message ID with range from 0x60~0x7A contains both input and output messages.

***CONFIGURE SBAS – Configure SBAS parameters of GNSS receiver (ID: 0x62, SID: 0x1)***

This is a request message which is issued from the host to GNSS receiver to configure SBAS parameters. The GNSS receiver should respond with an ACK when succeeded and should respond with an NACK when failed. The payload length is 9 bytes.

Structure:

<0xA0,0xA1>< PL><62><01>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 09 62 01 01 01 08 01 03 07 00 6E 0D 0A

1 2 3 4 5 6 7 8 9

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 62 | | UINT8 | |
| 2 | Message Sub-ID | 01 | | UINT8 | |
| 3 | Enable | 01 | 0: disable SBAS system<br>1: enable SBAS system | UINT8 | |
| 4 | Ranging | 01 | 0: do not use SBAS satellite for navigation<br>1: use SBAS satellite for navigation<br>2: auto mode determined by receiver whether ranging will use or not*1 | UINT8 | |
| 5 | Ranging URA Mask | 08 | Default:8, range 0~15 | UINT8 | |
| 6 | Correction | 01 | 0: disable the correction<br>1: enable the correction | UINT8 | |
| 7 | Number of tracking channels | 03 | Value: 0~3<br>Set how many channels are reserved for SBAS tracking | UINT8 | |
| 8 | Subsystem mask | 07 | Allows selectively enabling/disabling SBAS satellites<br>Bit0: WAAS, 1: enable; 0: disable<br>Bit1: EGNOS, 1: enable; 0: disable<br>Bit2: MSAS, 1: enable; 0: disable<br>Bit3: GAGAN, 1: enable; 0: disable | UINT8 | |

| | | | | | |
|---|---|---|---|---|---|
| | | | Bit4: SDCM, 1: enable; 0: disable<br>Bit5: BDSBAS, 1: enable; 0: disable<br>Bit7: All SBAS PRN 120~158 | | |
| 9 | Attributes | 00 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 9 bytes | | | | | |

***QUERY SBAS STATUS – Query SBAS status of GNSS receiver (ID: 0x62, SID: 0x2)***

This is a request message which is issued from the host to GNSS receiver to query SBAS status. The GNSS receiver should respond with an ACK along with SBAS status, ***"SBAS STATUS, ID: 0x62, SID: 0x80"***, when succeeded and should respond with an NACK when failed. The payload length is 2 bytes.

Structure:

<0xA0,0xA1>< PL><62><02>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 02 62 02 60 0D 0A

1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 62 | | UINT8 | |
| 2 | Message Sub-ID | 02 | | UINT8 | |
| Payload Length : 2 bytes | | | | | |

***CONFIGURE QZSS – Configure QZSS of GNSS receiver (ID: 0x62, SID: 0x3)***

This is a request message which is issued from the host to GNSS receiver to configure QZSS parameters. The GNSS receiver should respond with an ACK when succeeded and should respond with an NACK when failed. The payload length is 5 bytes.

Structure:

<0xA0,0xA1>< PL><62><03>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 05 62 03 01 03 00 63 0D 0A

1 2 3 4 5

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 62 | | UINT8 | |
| 2 | Message Sub-ID | 03 | | UINT8 | |
| 3 | Enable | 01 | 0: disable QZSS system<br>1: enable QZSS system | UINT8 | |
| 4 | Number of tracking channels | 03 | Value: 1~3<br>Set how many channels are used for QZSS tracking, default: 1 | UINT8 | |
| 5 | Attributes | 00 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 5 bytes | | | | | |

***QUERY QZSS STATUS – Query QZSS status of GNSS receiver (ID: 0x62, SID: 0x4)***

This is a request message which is issued from the host to GNSS receiver to query QZSS status. The GNSS receiver should respond with an ACK along with QZSS status, ***"QZSS STATUS, ID: 62, SID: 0x81"***, when succeeded and should respond with an NACK when failed. The payload length is 2 bytes.

Structure:

<0xA0,0xA1>< PL><62><04>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 02 62 04 66 0D 0A

1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 62 | | UINT8 | |
| 2 | Message Sub-ID | 04 | | UINT8 | |
| Payload Length : 2 bytes | | | | | |

***CONFIGURE SBAS ADVANCED – Configure SBAS advanced functions of GNSS receiver (ID: 0x62, SID: 0x5)***

This is a request message which is issued from the host to GNSS receiver to configure SBAS advanced functions. The GNSS receiver should respond with an ACK when succeeded and should respond with an NACK when failed. The payload length is 30 bytes.

Structure:

<0xA0,0xA1>< PL><62><05>< message body><CS><0x0D,0x0A>

Example:

| A0 | A1 | 00 | 1E | 62 | 05 | 01 | 02 | 08 | 01 | 02 | 7F | 83 | 85 | 87 | 7B | 88 | 00 | 89 | 00 | 00 | 7F | 80 | 84 | 7D | 8C | 8D | 82 | 8F | 90 | 7A |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 |
| 00 | 00 | 01 | 0A | 0D | 0A | | | | | | | | | | | | | | | | | | | | | | | | | |
| 28 | 29 | 30 | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 62 | | UINT8 | |
| 2 | Message Sub-ID | 05 | | UINT8 | |
| 3 | Enable | 01 | 0: disable SBAS system<br>1: enable SBAS system | UINT8 | |
| 4 | Ranging | 02 | 0: do not use SBAS satellite for navigation<br>1: use SBAS satellite for navigation<br>2: auto mode determined by receiver whether ranging will use or not | UINT8 | |
| 5 | Ranging URA Mask | 08 | Default:8, range 0~15 | UINT8 | |
| 6 | Correction | 01 | 0: disable the correction<br>1: enable the correction | UINT8 | |
| 7 | Number of tracking channels | 02 | Value: 0~3<br>Set how many channels are reserved for SBAS tracking | UINT8 | |
| 8 | Subsystem mask | 7F | Modified subsystem mask for fields 9~20<br>Bit0: WAAS, 1: enable; 0: disable<br>Bit1: EGNOS, 1: enable; 0: disable<br>Bit2: MSAS, 1: enable; 0: disable<br>Bit3: GAGAN, 1: enable; 0: disable<br>Bit4: SDCM, 1: enable; 0: disable<br>Bit5: BDSBAS, 1: enable; 0: disable | UINT8 | |

| | | | | | |
|---|---|---|---|---|---|
| | | | Bit6: SouthPAN, 1: enable; 0: disable<br>Bit7: SBAS ALL PRN 120~158, 1: enable all 0: use Bit0~Bit5 | | |
| 9~11 | WAAS PRN | 838587 | Modify WAAS PRN (Default: 131, 133, 135)<br>PRN 131: 0x83<br>PRN 133: 0x85<br>PRN 135: 0x87 | UINT8 | |
| 12~14 | EGNOS PRN | 7B8800 | Modify EGNOS PRN (Default: 123, 136)<br>PRN 123: 0x7B<br>PRN 136: 0x88 | UINT8 | |
| 15~17 | MSAS PRN | 890000 | Modify MSAS PRN (Default: 137)<br>PRN 137: 0x89 | UINT8 | |
| 18~20 | GAGAN PRN | 7F8084 | Modify GAGAN PRN (Default: 127, 128, 132)<br>PRN 127: 0x7F<br>PRN 128: 0x80<br>PRN 132: 0x84 | UINT8 | |
| 21~23 | SDCM PRN | 7D8C8D | Modify SDCM PRN (Default: 125, 140, 141)<br>PRN 125: 0x7D<br>PRN 140: 0x8C<br>PRN 141: 0x8D | UINT8 | |
| 24~26 | BDSBAS PRN | 828F90 | Modify Beidou SBAS PRN (Default: 130, 143, 144)<br>PRN 130: 0x82<br>PRN 143: 0x8F<br>PRN 144: 0x90 | UINT8 | |
| 27~29 | SouthPAN PRN | 7A0000 | Modify SouthPAN PRN (Default: 122)<br>PRN 122: 0x7a | UINT8 | |
| 30 | Attributes | 01 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 27 bytes if firmware not support SouthPAN PRN<br>30 bytes if firmware support SouthPAN PRN | | | | | |

***QUERY SBAS ADVANCED – Query SBAS advanced functions of GNSS receiver (ID: 0x62, SID: 0x6)***

This is a request message which is issued from the host to GNSS receiver to query SBAS advanced functions. The GNSS receiver should respond with an ACK along with SBAS status, ***"SBAS ADVANCED, ID: 0x62, SID: 0x82"***, when succeeded and should respond with an NACK when failed. The payload length is 2 bytes.

Structure:

<0xA0,0xA1>< PL><62><06>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 02 62 06 64 0D 0A

1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 62 | | UINT8 | |
| 2 | Message Sub-ID | 06 | | UINT8 | |
| Payload Length : 2 bytes | | | | | |

***SBAS STATUS – SBAS status of GNSS receiver (ID: 0x62, SID: 0x80)***

This is a response message to ***"QUERY SBAS STATUS, ID: 0x62, SID: 0x2"*** which provides the SBAS status of GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 8 bytes.

Structure:

<0xA0,0xA1>< PL><62><80>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 08 62 80 01 01 08 01 03 07 EF 0D 0A

1 2 3 4 5 6 7 8

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 62 | | UINT8 | |
| 2 | Message Sub-ID | 80 | | UINT8 | |
| 3 | Enable | 01 | 0: disable SBAS system<br>1: enable SBAS system | UINT8 | |
| 4 | Ranging | 01 | 0: do not use SBAS satellite for navigation<br>1: use SBAS satellite for navigation<br>2: auto mode determined by receiver*1 | UINT8 | |
| 5 | Ranging URA Mask | 08 | Range 0~15 default 8 | UINT8 | |
| 6 | Correction | 01 | 0: disable the correction<br>1: enable the correction | UINT8 | |
| 7 | Number of tracking channels | 03 | Value: 0~3<br>Set how many channels are reserved for SBAS tracking | UINT8 | |
| 8 | Subsystem mask | 07 | Allows selectively enabling/disabling SBAS satellites<br>Bit0: WAAS, 1: enable; 0: disable<br>Bit1: EGNOS, 1: enable; 0: disable<br>Bit2: MSAS, 1: enable; 0: disable<br>Bit3: GAGAN, 1: enable; 0: disable<br>Bit4: SDCM, 1: enable; 0: disable<br>Bit5: BDSBAS, 1: enable; 0: disable<br>Bit7: All SBAS PRN 120~158 | UINT8 | |
| Payload Length : 8 bytes | | | | | |

***QZSS STATUS – QZSS status of GNSS receiver (ID: 0x62, SID: 0x81)***

This is a response message to ***"QUERY QZSS STATUS, ID: 0x62, SID: 0x4"*** which provides the QZSS status of GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 4 bytes.

Structure:

<0xA0,0xA1>< PL><62><81>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 04 62 81 01 03 E1 0D 0A

1 2 3 4

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 62 | | UINT8 | |
| 2 | Message Sub-ID | 81 | | UINT8 | |
| 3 | Enable | 01 | 0: disable QZSS system<br>1: enable QZSS system | UINT8 | |
| 4 | Number of tracking channels | 03 | Value: 1~3<br>Set how many channels are used for QZSS tracking | UINT8 | |
| Payload Length : 4 bytes | | | | | |

***SBAS ADVANCED – SBAS advanced functions of GNSS receiver (ID: 0x62, SID: 0x82)***

This is a response message to ***"QUERY SBAS ADVANCED, ID: 0x62, SID: 0x5"*** which provides the SBAS advanced functions of GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 36 bytes.

Structure:

<0xA0,0xA1>< PL><62><82>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 24 62 82 01 02 08 01 02 7F 03 83 85 87 02 7B 88 00 01 89 00 00 03 7F 80 84 03 7D 8C

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27

8D 03 82 8F 90 01 7A 00 00 8E 0D 0A

28 29 30 31 32 33 34 35 35

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 62 | | UINT8 | |
| 2 | Message Sub-ID | 82 | | UINT8 | |
| 3 | Enable | 01 | 0: disable SBAS system<br>1: enable SBAS system | UINT8 | |
| 4 | Ranging | 01 | 0: do not use SBAS satellite for navigation<br>1: use SBAS satellite for navigation<br>2: auto mode determined by receiver whether ranging will use or not | UINT8 | |
| 5 | Ranging URA Mask | 08 | Default:8, range 0~15 | UINT8 | |
| 6 | Correction | 01 | 0: disable the correction<br>1: enable the correction | UINT8 | |
| 7 | Number of tracking channels | 02 | Value: 0~3<br>Set how many channels are reserved for SBAS tracking | UINT8 | |
| 8 | Subsystem mask | 3F | Modified subsystem mask for fields 9~24<br>Bit0: WAAS, 1: enable; 0: disable<br>Bit1: EGNOS, 1: enable; 0: disable<br>Bit2: MSAS, 1: enable; 0: disable<br>Bit3: GAGAN, 1: enable; 0: disable<br>Bit4: SDCM, 1: enable; 0: disable<br>Bit5: BDSBAS, 1: enable; 0: disable | UINT8 | |

| | | | | | |
|---|---|---|---|---|---|
| | | | Bit6: SouthPAN, 1: enable; 0: disable<br>Bit7: SBAS ALL PRN 120~158, 1:<br>enable all 0: use Bit0~Bit5 | | |
| 9 | WAAS PRN # | 03 | WAAS PRN number | UINT8 | |
| 10~12 | WAAS PRN | 838587 | Modified WAAS PRN<br>WAAS PRN 131: 0x83<br>WAAS PRN 133: 0x85<br>WAAS PRN 135: 0x87<br>000000: when Subsystem mask is<br>SBAS All or Bit 0 is disabled | UINT8 | |
| 13 | EGONS PRN # | 02 | EGNOS PRN number | UINT8 | |
| 14~16 | EGNOS PRN | 7B8800 | Modified EGNOS PRN<br>EGNOS PRN 123: 0x7B<br>EGNOS PRN 136: 0x88<br>000000: when Subsystem mask is<br>SBAS All or Bit 1 is disabled | UINT8 | |
| 17 | MSAS PRN # | 01 | MSAS PRN number | UINT8 | |
| 18~20 | MSAS PRN | 890000 | Modified MSAS PRN<br>EGNOS PRN 137: 0x89<br>000000: when Subsystem mask is<br>SBAS All or Bit 2 is disabled | UINT8 | |
| 21 | GAGAN PRN # | 03 | GAGAN PRN number | UINT8 | |
| 22~24 | GAGAN PRN | 7F8084 | Modify GAGAN PRN<br>GAGAN PRN 127: 0x7F<br>GAGAN PRN 128: 0x80<br>GAGAN PRN 132: 0x84<br>000000: when Subsystem mask is<br>SBAS All or Bit 3 is disabled | UINT8 | |
| 25 | SDCM PRN # | 03 | SDCM PRN number | UINT8 | |
| 26~28 | SDCM PRN | 7D8C8D | Modify SDCM PRN<br>SDCM PRN 125: 0x7D<br>SDCM PRN 140: 0x8C<br>SDCM PRN 141: 0x8D<br>000000: when Subsystem mask is<br>SBAS All or Bit 4 is disabled | UINT8 | |
| 29 | BDSBAS PRN # | 03 | BDSBAS PRN number | UINT8 | |
| 30~32 | BDSBAS PRN | 828F90 | Modify Beidou SBAS PRN<br>BDSBAS PRN 130: 0x82<br>BDSBAS PRN 143: 0x8F<br>BDSBAS PRN 144: 0x8D | UINT8 | |

| | | | | | |
|---|---|---|---|---|---|
| | | | 000000: when Subsystem mask is<br>SBAS All or Bit 5 is disabled | | |
| 33 | SouthPAN PRN # | 01 | SouthPAN PRN number | UINT8 | |
| 34~36 | SouthPAN PRN | 7A0000 | Modify SouthPAN PRN<br>SouthPAN PRN 122: 0x7A<br>000000: when Subsystem mask is<br>SBAS All or Bit 6 is disabled | UINT8 | |
| Payload Length : 32 bytes if firmware not support SouthPAN PRN<br>36 bytes if firmware support SouthPAN PRN | | | | | |

***CONFIGURE SAEE – configure SAEE of GNSS receiver (ID: 0x63, SID: 0x1)***

This is a request message which is issued from the host to GNSS receiver to configure enable or disable self-aided ephemeris estimation (SAEE). The GNSS receiver should respond with an ACK when succeeded and should respond with an NACK when failed. The payload length is 4 bytes.

Structure:
<0xA0,0xA1>< PL><63><01>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 04 63 01 01 01 62 0D 0A
1 2 3 4

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 63 | | UINT8 | |
| 2 | Message Sub-ID | 01 | | UINT8 | |
| 3 | Enable | 01 | 0: SAEE mode default<br>ROM version decided by HW power-on latch<br>1: SAEE enable<br>2: SAEE disable | UINT8 | |
| 4 | Attributes | 01 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 4 bytes | | | | | |

***QUERY SAEE STATUS – Query SAEE status of GNSS receiver (ID: 0x63, SID: 0x2)***

This is a request message which is issued from the host to GNSS receiver to query self-aided ephemeris estimation (SAEE) status. The GNSS receiver should respond with an ACK along with SAEE status, ***"SAEE STATUS, ID: 63, SID: 0x80"***, when succeeded and should respond with an NACK when failed. The payload length is 2 bytes.

Structure:

<0xA0,0xA1>< PL><63><0x2>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 02 63 02 61 0D 0A

1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 63 | | UINT8 | |
| 2 | Message Sub-ID | 02 | | UINT8 | |
| Payload Length : 2 bytes | | | | | |

***SAEE STATUS – SAEE status of GNSS receiver (ID: 0x63, SID: 0x80)***

This is a response message to ***"QUERY SAEE STATUS, ID: 0x63, SID: 0x2"*** which provides the self-aided ephemeris estimation (SAEE) status of GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 3 bytes.

Structure:

<0xA0,0xA1>< PL><63><80>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 03 63 80 01 E2 0D 0A

1 2 3

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 63 | | UINT8 | |
| 2 | Message Sub-ID | 80 | | UINT8 | |
| 3 | Status | 01 | 0: SAEE mode default<br>ROM version decided by HW power-on latch<br>1: SAEE enable<br>2: SAEE disable | UINT8 | |
| Payload Length : 3 bytes | | | | | |

***QUERY GNSS BOOT STATUS – Query boot status of GNSS receiver (ID: 0x64, SID: 0x1)***

This is a request message which is issued from the host to GNSS receiver to query boot status. The GNSS receiver should respond with an ACK along with boot status, ***"GNSS BOOT STATUS, ID: 64, SID: 0x80"***, when succeeded and should respond with an NACK when failed. The payload length is 2 bytes.

Structure:
<0xA0,0xA1>< PL><64><01>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 02 64 01 65 0D 0A
1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Message Sub-ID | 01 | | UINT8 | |
| Payload Length : 2 bytes | | | | | |

***CONFIGURE EXTENDED NMEA MESSAGE INTERVAL – Configure extended NMEA message Interval of GNSS receiver (ID: 0x64, SID: 0x2)***

This is a request message which will set NMEA message interval configuration. This command is issued from the host to GNSS receiver and GNSS receiver should respond with an ACK or NACK. The payload length is 15 bytes.

Structure:

<0xA0,0xA1>< PL><64><02>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 0F 64 02 01 01 03 01 01 01 01 00 00 00 00 00 01 64 0D 0A

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Message Sub-ID | 02 | | UINT8 | |
| 3 | GGA Interval | 01 | 0 ~255, 0: disable | UINT8 | second |
| 4 | GSA Interval | 01 | 0 ~255, 0: disable | UINT8 | second |
| 5 | GSV Interval | 03 | 0 ~255, 0: disable | UINT8 | second |
| 6 | GLL Interval | 01 | 0 ~255, 0: disable | UINT8 | second |
| 7 | RMC Interval | 01 | 0 ~255, 0: disable | UINT8 | second |
| 8 | VTG Interval | 01 | 0 ~255, 0: disable | UINT8 | second |
| 8 | ZDA Interval | 01 | 0 ~255, 0: disable | UINT8 | second |
| 10 | GNS Interval | 00 | 0 ~255, 0: disable | UINT8 | second |
| 11 | GBS Interval | 00 | 0 ~255, 0: disable | UINT8 | second |
| 12 | GRS Interval | 00 | 0 ~255, 0: disable | UINT8 | second |
| 13 | DTM Interval | 00 | 0 ~255, 0: disable | UINT8 | second |
| 14 | GST Interval | 00 | 0 ~255, 0: disable | UINT8 | second |
| 15 | Attributes | 01 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 15 bytes | | | | | |

***QUERY EXTENDED NMEA MESSAGE INTERVAL – Query extended NMEA message interval of GNSS receiver (ID: 0x64, SID: 0x3)***

This is a request message which is issued from the host to GNSS receiver to query extended nmea message interval. The GNSS receiver should respond with an ACK along with nmea message interval, ***"EXTENDED NMEA MESSAGE INTERVAL, ID: 0x64, SID: 0x81***", when succeeded and should respond with an NACK when failed. The payload length is 2 bytes.

Structure:
<0xA0,0xA1>< PL><64><03><CS><0x0D,0x0A>

Example:
A0 A1 00 02 64 03 67 0D 0A
1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Message Sub-ID | 03 | | UINT8 | |
| Payload Length : 2 bytes | | | | | |

***CONFIGURE INTERFERENCE DETECTION – Configure the interference detection of GNSS receiver (ID: 0x64, SID: 0x6)***

This is a request message which is issued from the host to GNSS receiver to configure interference detect control. The GNSS receiver should respond with an ACK when succeeded and should respond with an NACK when failed. The payload length is 4 bytes.

Structure:
<0xA0,0xA1>< PL><64><06>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 04 64 06 01 00 63 0D 0A
1 2 3 4

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Message Sub-ID | 06 | | UINT8 | |
| 3 | Interference Detect Control | 01 | 0: disable<br>1: enable | UINT8 | |
| 4 | Attributes | 00 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 4 bytes | | | | | |

***QUERY INTERFERENCE DETECTION STATUS – Query the status of interference detection of the GNSS receiver (ID: 0x64, SID: 0x7)***

This is a request message which is issued from the host to GNSS receiver to query interference detection status. The GNSS receiver should respond with an ACK along with information of interference detection status, ***"INTERFERENCE DETECTION STATUS, ID: 0x64, SID: 0x83"***, when succeeded and should respond with an NACK when failed. The payload length is 2 bytes.

Structure:
<0xA0,0xA1>< PL><64><07>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 02 64 07 63 0D 0A
1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Message Sub-ID | 07 | | UINT8 | |
| Payload Length : 2 bytes | | | | | |

***QUERY GNSS PARAMETER SEARCH ENGINE NUMBER – Query the parameter search engine number of the GPS receiver (ID: 0x64, SID: 0xB)***

This is a request message which is issued from the host to GNSS receiver to query parameter search engine number. The GNSS receiver should respond with an ACK along with information of GNSS parameter search engine number, **"GNSS *PARAMETER SEARCH ENGINE NUMBER, ID 0x64, SID 0x85*"**, when succeeded and should respond with an NACK when failed. The payload length is 2 bytes.

Structure:
<0xA0,0xA1>< PL><64><0B>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 02 64 0B 6F 0D 0A
1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Message Sub-ID | 0B | | UINT8 | |
| Payload Length : 2 bytes | | | | | |

***CONFIGURE POSITION FIX NAVIGATION MASK – Configure the position fix navigation mask of GNSS receiver (ID: 0x64, SID: 0x11)***

This is a request message which is issued from the host to GNSS receiver to configure the 2D or 3D position fix navigation mask. The GNSS receiver should respond with an ACK when succeeded and should respond with an NACK when failed. The payload length is 5 bytes.

Structure:
<0xA0,0xA1>< PL><64><11>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 05 64 11 00 00 00 75 0D 0A
1 2 3 4 5

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Sub ID | 11 | | UINT8 | |
| 3 | First fix navigation mask | 00 | 0: 3D<br>1: 2D | UINT8 | |
| 4 | Subsequent fix navigation mask | 00 | 0: 3D<br>1: 2D | UINT8 | |
| 5 | Attributes | 00 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 5 bytes | | | | | |

***QUERY POSITION FIX NAVIGATION MASK – Query the position fix navigation mask of GNSS receiver (ID: 0x64, SID: 0x12)***

This is a request message which is issued from the host to GNSS receiver to query position fix navigation mask. The GNSS receiver should respond with an ACK along with information of navigation mask ***"POSITION FIX NAVIGATION MASK, ID: 0x64, SID: 0x88"***, when succeeded and should respond with an NACK when failed. The payload length is 2 bytes.

Structure:
<0xA0,0xA1>< PL><64><12>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 02 64 12 76 0D 0A
1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Sub ID | 12 | | UINT8 | |
| Payload Length : 2 bytes | | | | | |

***CONFIGURE UTC REFERENCE TIME SYNC TO GPS TIME – Configure the UTC reference time to GNSS receiver to synchronize to GPS time (ID: 0x64, SID: 0x15)*** *[*1]*

This is a request message which is issued from the host to GNSS receiver to configure the UTC reference time that is used to synchronize to GPS time. The GNSS receiver should respond with an ACK when succeeded and should respond with an NACK when failed. The payload length is 8 bytes.

Structure:
<0xA0,0xA1>< PL><64><15>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 08 64 15 01 07 EC 01 01 01 9A 0D 0A
1 2 3 4 5 6 7 8

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Message Sub-ID | 15 | | UINT8 | |
| 3 | Enable | 01 | 0: Disable<br>1: Enable | UINT8 | |
| 4-5 | UTC Year | 07EC | UTC year: 2028 | UINT16 | |
| 6 | UTC Month | 01 | UTC month: 01 | UINT8 | |
| 7 | UTC Day | 01 | UTC day: 01 | UINT8 | |
| 8 | Attributes | 01 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 8 bytes | | | | | |

*1
The time of week is transmitted by GPS satellites, but only the bottom 10 bits of the week number are transmitted. This means valid range is from 0 to 1023, until it reaches 1023 after which it will "roll over" back to zero.
The 1st week rollover occurred in 1999 and the 2nd will be in 2019.

How to decide default week rollover times?

SkyTraq receivers solve this problem by assuming that all week numbers must be at least as large as a reference rollover week number. This reference rollover week number is hard-coded into the firmware at compile time and is normally set a few weeks before the software is completed, but it can be adjusted by command "*Configure UTC Reference Time Sync to GPS Time*".

For example :
User just input reasonable UTC time they want, SkyTraq receivers will transform this reference time to proper week

rollover times automatically.

It is important to set the reference rollover week number appropriately when supplying SkyTraq receivers with simulated signals, especially when the scenarios are in the past.

***QUERY UTC REFERENCE TIME SYNC TO GPS TIME – Query the UTC reference time of GNSS receiver set to synchronize to GPS time (ID: 0x64, SID: 0x16)***

This is a request message which is issued from the host to GPS receiver to query UTC reference time of GNSS receiver that set to synchronize to GPS time. The GNSS receiver should respond with an ACK along with GPS UTC reference time, ***"GPS UTC REFERENCE TIME, ID: 0x64, SID: 0x8A"***, when succeeded and should respond with an NACK when failed. The payload length is 2 bytes.

Structure:
<0xA0,0xA1>< PL><64><16>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 02 64 16 72 0D 0A
1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Message Sub-ID | 16 | | UINT8 | |
| Payload Length : 2 bytes | | | | | |

***CONFIGURE GNSS NAVIGATION MODE – Configure the navigation mode of GNSS receiver (ID: 0x64, SID: 0x17)***

This is a request message which is issued from the host to GNSS receiver to configure the system navigation mode. The GNSS receiver should respond with an ACK when succeeded and should respond with an NACK when failed. The payload length is 4 bytes.

Structure:
<0xA0,0xA1>< PL><64><17>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 04 64 17 00 00 73 0D 0A
1 2 3 4

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Message Sub-ID | 17 | | UINT8 | |
| 3 | Navigation mode | 00 | 0: auto<br>1: pedestrian<br>2: car<br>3: marine<br>4: balloon<br>5: airborne<br>6: reserved<br>7: quadcopter<br>8: reserved<br>9: SLR (Speed Lag Reduced) mode | UINT8 | |
| 4 | Attributes | 00 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 4 bytes | | | | | |

***QUERY GNSS NAVIGATION MODE – Query the navigation mode of GNSS receiver (ID: 0x64, SID: 0x18)***

This is a request message which is issued from the host to GNSS receiver to query navigation mode. The GNSS receiver should respond with an ACK along with navigation mode, ***"GNSS NAVIGATION MODE, ID: 0x64, SID: 0x8B"***, when succeeded and should respond with an NACK when failed. The payload length is 2 bytes.

Structure:

<0xA0,0xA1>< PL><64><18>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 02 64 18 7C 0D 0A

1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Message Sub-ID | 18 | | UINT8 | |
| Payload Length : 2 bytes | | | | | |

***CONFIGURE GNSS CONSTELLATION TYPE FOR NAVIGATION SOLUTION – Set the GNSS constellation type for navigation solution (ID: 0x64, SID: 0x19)***

This is a request message which is issued from the host to GNSS receiver to configure the GNSS constellation type for navigation solution. The GNSS receiver should respond with an ACK when succeeded and should respond with an NACK when failed. The payload length is 5 bytes.

Structure:
<0xA0,0xA1>< PL><64><19>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 05 64 19 00 09 00 74 0D 0A
1 2 3 4 5

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Sub ID | 19 | | UINT8 | |
| 3-4 | Constellation Type | 00 09 | Bit 0: GPS<br>Bit 1: Glonass<br>Bit 2: Galileo<br>Bit 3: Beidou<br>Bit 4: Navic | UINT16 | |
| 5 | Attributes | 00 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 5 bytes | | | | | |

***QUERY GNSS CONSTELLATION TYPE FOR NAVIGATION SOLUTION – Query the GNSS constellation type for navigation solution (ID: 0x64, SID: 0x1A)***

This is a request message which is issued from the host to GNSS receiver to query GNSS constellation type for navigation solution. The GNSS receiver should respond with an ACK along with constellation type, **"*GNSS CONSTELLATION TYPE FOR NAVIGATION SOLUTION, ID 0x64, SID 0x8C*"**, when succeeded and should respond with an NACK when failed. The payload length is 2 bytes.

Structure:
<0xA0,0xA1>< PL><64><1A>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 02 64 1A 7E 0D 0A
1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Sub ID | 1A | | UINT8 | |
| Payload Length : 2 bytes | | | | | |

***SOFTWARE IMAGE DOWNLOAD USING ROM EXTERNAL LOADER – Download software image to system flash using ROM external loader (ID: 0x64, SID: 0x1B)*** *1

This is a request message which is issued from the host to GNSS receiver to download image to system flash using ROM external loader when download from ROM to Flash. The GNSS receiver should respond with an ACK when succeeded and should respond with an NACK when failed. The payload length is 7 bytes.

Structure:
<0xA0,0xA1>< PL><64><1B>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 07 64 1B 07 00 00 00 00 78 0D 0A
1 2 3 4 5 6 7

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Sub ID | 1B | | UINT8 | |
| 3 | Baud | 07 | 0: 4800<br>1: 9600<br>2: 19200<br>3: 38400<br>4: 57600<br>5: 115200<br>6: 230400<br>7: 460800<br>8: 921600 | UINT8 | |
| 4 | Flash Type | 00 | 0: default, auto<br>1: QSPI Winbond<br>2. QSPI EON<br>3: Parallel Flash NUMONYX<br>4. Parallel Flash EON | UINT8 | |
| 5-6 | Flash ID | 0000 | If field 3 is not 0, then need to specify the flash ID | UINT16 | |
| 7 | Buffer Used Index | 00 | 0:8k<br>1:16K<br>2:24K<br>3:32K | UINT8 | |
| Payload Length : 7 bytes | | | | | |

*1 Please refer to SkyTraq software image download application notes and API.

***CONFIGURE GNSS DOZE MODE – Configure the doze mode of GNSS receiver (ID: 0x64, SID:0x1C)***

This is a request message which is issued from the host to GNSS receiver to configure the doze mode of GNSS receiver. The GNSS receiver should respond with an ACK when succeeded and should respond with an NACK when failed. The payload length is 2 bytes. When in doze mode, there is no NMEA output, and the GNSS receiver is in doze mode. To wake up from doze mode, issuing a cold start will bring GNSS receiver back to normal.

Structure:
<0xA0,0xA1>< PL><64><1C>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 02 64 1C 78 0D 0A
1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Sub ID | 1C | | UINT8 | |
| Payload Length : 2 bytes | | | | | |

***QUERY GPS TIME – Query GPS time of GNSS receiver (ID: 0x64, SID: 0x20)***

This is a request message which is issued from the host to GNSS receiver to query GPS time. The GNSS receiver should respond with an ACK along with GPS time, "***GPS TIME, ID: 0x64, SID: 0x8E***", when succeeded and should respond with an NACK when failed. The payload length is 2 bytes.

Structure:
<0xA0,0xA1>< PL><64><0x20>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 02 64 20 44 0D 0A
1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Message Sub ID | 20 | | UINT8 | |
| Payload Length : 2 bytes | | | | | |

***CONFIGURE PSTI MESSAGE INTERVAL – Configure PSTI message interval of GNSS receiver (ID: 0x64, SID: 0x21)***

This is a request message which will set PSTI message interval of certain PSTI message ID to the GNSS receiver. This command is issued from the host to GNSS receiver and GNSS receiver should respond with an ACK or NACK. The payload length is 5 bytes. On one condition that firmware does not support certain PSTI ID, the GNSS receiver will reply NACK.

Structure:

<0xA0,0xA1>< PL><64><21>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 05 64 21 1E 01 01 5B 0D 0A

1 2 3 4 5

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Sub ID | 21 | | UINT8 | |
| 3 | PSTI ID*1 | 1E | PSTI ID of SkyTraq proprietary message<br>Ex. A value equals 4 (hex, 0x04) corresponding to PSTI,004.<br>Ex. A value equals 30 (hex 0x1E) corresponding to PSTI,030 | UINT8 | |
| 4 | Message Interval | 01 | 0: disable<br>1~255: interval or enable | UINT8 | |
| 5 | Attributes | 01 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 5 bytes | | | | | |

*1 PSTI ID list

PSTI ID 5: time stamp proprietary message.

PSTI ID 7: geofencing proprietary message.

PSTI,ID 20: dead reckoning proprietary message.

PSTI ID 30, 32, 33: RTK proprietary messages.

***QUERY PSTI MESSAGE INTERVAL – Query PSTI message interval of GNSS receiver (ID: 0x64, SID: 0x22)***

This is a request message which is issued from the host to GNSS receiver to query PSTI message interval of certain PSTI message ID. The GNSS receiver should respond with an ACK along with PSTI message interval, "***PSTI MESSAGE INTERVAL, ID 0x64, SID 0x8F***", when succeeded and should respond with an NACK when failed. The payload length is 3 bytes.

Structure:

<0xA0,0xA1>< PL><64><22>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 03 64 22 1E 58 0D 0A

1 2 3

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Sub ID | 22 | | UINT8 | |
| 3 | Message ID*1* | 1E | SkyTraq proprietary message ID<br>Ex. A value equals 4 (hex, 0x04) corresponding to PSTI,004.<br>Ex. A value equals 30 (hex 0x1E) corresponding to PSTI,030 | UINT8 | |
| Payload Length : 3 bytes | | | | | |

*1 PSTI ID list

PSTI ID 5: time stamp proprietary message.

PSTI ID 7: geofencing proprietary message.

PSTI,ID 20: dead reckoning proprietary message.

PSTI ID 30, 32, 33: RTK proprietary messages.

***QUERY REQUESTED PSTI MESSAGE INTERVAL – Query the message interval of requested PSTI ID of GNSS receiver[*1] (ID: 0x64, SID: 0x23)***

This is a request message which is issued from the host to GNSS receiver to query PSTI message interval of certain PSTI message ID. The GNSS receiver should respond with an ACK along with requested PSTI message ID and its message interval, "***REQUESTED PSTI MESSAGE INTERVAL, ID 0x64, SID 0x90***", when succeeded and should respond with an NACK when failed. The payload length is 3 bytes.

Structure:
<0xA0,0xA1>< PL><64><23>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 03 64 23 1E 59 0D 0A
1 2 3

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Sub ID | 23 | | UINT8 | |
| 3 | Message ID[*2] | 1E | SkyTraq proprietary message ID<br>Ex. A value equals 4 (hex, 0x04) corresponding to PSTI,004.<br>Ex. A value equals 30 (hex 0x1E) corresponding to PSTI,030 | UINT8 | |
| Payload Length : 3 bytes | | | | | |

*1 The query message will get PSTI message interval plus requested PSTI ID in the response message format.

*2 PSTI ID list

PSTI ID 5: time stamp proprietary message.

PSTI ID 7: geofencing proprietary message.

PSTI,ID 20: dead reckoning proprietary message.

PSTI ID 30, 32, 33: RTK proprietary messages.

***CONFIGURE GNSS DATUM INDEX – Configure the datum index of GNSS receiver (ID: 0x64, SID: 0x27)***

This is a request message which is issued from the host to configure the datum index to GNSS receiver. The GNSS receiver should respond with an ACK when succeeded and should respond with an NACK when failed. The payload length is 5 bytes.

Structure:
<0xA0,0xA1>< PL><64><27>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 05 64 27 00 DC 01 9E 0D 0A
1 2 3 4 5

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Sub ID | 27 | | UINT8 | |
| 3-4 | Datum index | 00DC | Datum index, range 0~220. Please refer to Appendix B. | UINT16 | |
| 5 | Attributes | 01 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 5 bytes | | | | | |

***QUERY GNSS DATUM INDEX – Query the datum index of the GNSS receiver (ID: 0x64, SID: 0x28)***

This is a request message which is issued from the host to GNSS receiver to query datum index. The GNSS receiver should respond with an ACK along with information of datum index ***"GNSS DATUM INDEX, ID: 0x64, SID: 0x92"***, when succeeded and should respond with an NACK when failed. The payload length is 2 bytes.

Structure:

<0xA0,0xA1>< PL><64><28>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 02 64 28 4C 0D 0A

1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Sub ID | 28 | | UINT8 | |
| Payload Length : 2 bytes | | | | | |

***CONFIGURE GPS/UTC LEAP SECONDS IN UTC – Configure GPS/UTC leap seconds of GNSS receiver in UTC (ID: 0x64, SID:0x2D)***

This is a request message which is issued from the host to GNSS receiver to configure GPS/UTC leap seconds with UTC and leap second parameters. The GNSS receiver should respond with an ACK when succeeded and should respond with an NACK when failed. The payload length is 8 bytes.

Structure:
<0xA0,0xA1>< PL><64><2D>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 08 64 2D 07 DF 06 11 01 01 86 0D 0A
1 2 3 4 5 6 7 8

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Message Sub ID | 2D | | UINT8 | |
| 3-4 | UTC Year | 07DF | 2015 | UINT16 | |
| 5 | UTC Month | 06 | 6 or 12 | UINT8 | |
| 6 | Leap seconds | 11 | Leap seconds | SINT8 | |
| 7 | Insert second | 01 | +1 or -1 | SINT8 | |
| 8 | Attributes | 01 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 8 bytes | | | | | |

***CONFIGURE NAVIGATION DATA MESSAGE INTERVAL – Configure navigation data message Interval of GNSS receiver (ID: 0x64, SID: 0x2F)***

This is a request message which will set navigation data message (ID: 0xA8) interval of GNSS receiver. This command is issued from the host to GNSS receiver and GNSS receiver should respond with an ACK or NACK. The payload length is 4 bytes.

Structure:
<0xA0,0xA1>< PL><64><2F>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 04 64 2F 01 01 4B 0D 0A
1 2 3 4

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Message Sub-ID | 2F | | UINT8 | |
| 3 | Navigation Data Message Interval | 01 | 0 ~255, 0: disable | UINT8 | second |
| 4 | Attributes | 01 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 4 bytes | | | | | |

***QUERY NAVIGATION DATA MESSAGE INTERVAL – Query navigation data message interval of GNSS receiver (ID: 0x64, SID: 0x30)***

This is a request message which is issued from the host to GNSS receiver to query navigation data message ID: 0xA8) interval. The GNSS receiver should respond with an ACK along with navigation data message interval, ***"NAVIGATION DATA MESSAGE INTERVAL, ID: 0x64, SID: 0x98***", when succeeded and should respond with an NACK when failed. The payload length is 2 bytes.

Structure:
<0xA0,0xA1>< PL><64><30><CS><0x0D,0x0A>

Example:
A0 A1 00 02 64 30 54 0D 0A
1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Message Sub-ID | 30 | | UINT8 | |
| Payload Length : 2 bytes | | | | | |

***QUERY VERSION EXTENSION STRING – Query version extension string of GNSS receiver (ID: 0x64, SID: 0x7D)***

This is a request message which is issued from the host to GNSS receiver to query version extension string. The GNSS receiver should respond with an ACK along with version extension string, "***VERSION EXTENSION STRING, ID: 0x64, SID: 0xFE***", when succeeded and should respond with an NACK when failed. The payload length is 2 bytes.

Structure:

<0xA0,0xA1>< PL><64><0x7D>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 02 64 7D 19 0D 0A

1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Message Sub ID | 7D | | UINT8 | |
| Payload Length : 2 bytes | | | | | |

***CONFIGURE GNSS GEO-FENCING DATA BY POLYGON – Configure geo-fencing data by polygon to GNSS receiver (ID: 0x64, SID: 0x34)***

This is a request message which is issued from the host to configure the geo-fencing data by polygon to GNSS receiver. The GNSS receiver should respond with an ACK when succeeded and should respond with an NACK when failed. The payload length is maximum 261 bytes.

Structure:
<0xA0,0xA1>< PL><64><34>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 45 64 34 00 01 04 40 38 C8 E5 BF 18 FC 73 40 5E 40 90 38 79 65 94 40 38 C8 E9 C1 87 15
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28

D6 40 5E 40 92 D5 3D 3C 54 40 38 C8 F1 8D 47 37 07 40 5E 40 92 24 BC 08 40 40 38 C8 ED 64 06 8F
29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60

BC 40 5E 40 8F 70 E0 2B BE B9 0D 0A
61 62 63 64 65 66 67 68 69

| Field | Name | Example (hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Sub ID | 34 | | UINT8 | |
| 3 | Attributes | 00 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| 4 | Polygon | 01 | Polygon index, range: 1~4. | UINT8 | |
| 5 | Number of points | 04 | Number of points of a polygon<br>Maximum number: 16 | UINT8 | |
| 6-13 | Latitude | 4038C8E5<br>BF18FC73 | Latitude in double of polygon points #1 | DPFP | degree |
| 14-21 | Longitude | 405E4090<br>38796594 | Longitude in double of polygon points #1 | DPFP | degree |
| 22-29 | Latitude | 4038C8E9<br>C18715D6 | Latitude in double of polygon points #2 | DPFP | degree |
| 30-37 | Longitude | 405E4092<br>D53D3C54 | Longitude in double of polygon points #2 | DPFP | degree |
| 38-45 | Latitude | 4038C8F1 | Latitude in double of polygon points #3 | DPFP | degree |

| | | | | | |
|---|---|---|---|---|---|
| | | 8D473707 | | | |
| 46-53 | Longitude | 405E4092<br>24BC0840 | Longitude in double of polygon points #3 | DPFP | degree |
| 54-61 | | 4038C8ED<br>64068FBC | Latitude in double of polygon points #4 | DPFP | degree |
| 62-69 | | 405E408F<br>70E02BBE | Longitude in double of polygon points #4 | DPFP | degree |
| …. | | | | | |
| 6+((ndx-1)*16)<br>~<br>13+((ndx-1)*16) | Latitude | | Latitude in double of polygon points #ndx | DPFP | degree |
| 14+((ndx-1)*16)<br>~<br>21+((ndx-1)*16) | Longitude | | Longitude in double of polygon points #ndx | DPFP | degree |
| Payload Length : maximum 261 bytes, ndx = number of polygon points, maximum 16 points of each polygon | | | | | |

***QUERY GNSS GEO-FENCING DATA BY POLYGON – Query geo-fencing data by polygon of the GNSS receiver (ID: 0x64, SID: 0x35)***

This is a request message which is issued from the host to GNSS receiver to query geo-fencing data. The GNSS receiver should respond with an ACK along with geo-fencing data ***"GNSS GEO-FENCING DATA BY POLYGON, ID: 0x64, SID: 0x99"***, when succeeded and should respond with an NACK when failed. The payload length is 3 bytes.

Structure:

<0xA0,0xA1>< PL><64><35>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 03 64 35 01 50 0D 0A

1 2 3

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Sub ID | 35 | | UINT8 | |
| 3 | Polygon | 01 | Polygon index, range: 1~4 | UINT8 | |
| Payload Length : 3 bytes | | | | | |

***QUERY GNSS MULTI-POLYGON GEO-FENCING RESULT – Query multi-polygon geo-fencing result of the GNSS receiver (ID: 0x64, SID: 0x36)***

This is a request message which is issued from the host to GNSS receiver to query multi-polygon geo-fencing result. The GNSS receiver should respond with an ACK along with information of geo-fencing result ***"GNSS MULTI-POLYGON GEO-FENCING RESULT, ID: 0x64, SID: 0x9A"***, when succeeded and should respond with an NACK when failed. The payload length is 2 bytes.

Structure:
<0xA0,0xA1>< PL><64><36>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 02 64 36 52 0D 0A
1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Sub ID | 36 | | UINT8 | |
| Payload Length : 2 bytes | | | | | |

***CONFIGURE NMEA STRING INTERVAL – Configure NMEA string Interval of GNSS receiver (ID: 0x64, SID: 0x3B)***

This is a request message which will set NMEA string interval configuration. This command is issued from the host to GNSS receiver and GNSS receiver should respond with an ACK or NACK. The payload length is 7 bytes.

Structure:

<0xA0,0xA1>< PL><64><3B>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 07 64 3B 47 47 41 01 01 1E 0D 0A

1 2 3 4 5 6 7

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Message Sub-ID | 3B | | UINT8 | |
| 3~5 | NMEA String*1 | 474741 | String of NMEA, exclude Talker<br>Ex. GGA in binary format is<br>0x47,0x47,0x41 | UINT8 | |
| 6 | Interval | 01 | 0 ~255, 0: disable | UINT8 | second |
| 7 | Attributes | 01 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 7 bytes | | | | | |

*1 NMEA string supports: GGA, GNS, GSA, GSV, GLL, RMC, VTG, ZDA, DTM, GBS, GRS, GST, THS, HDT.

***QUERY NMEA STRING INTERVAL – Query NMEA string interval of GNSS receiver (ID: 0x64, SID: 0x3C)***

This is a request message which is issued from the host to GNSS receiver to query nmea string interval. The GNSS receiver should respond with an ACK along with nmea message interval, ***"NMEA STRING INTERVAL, ID: 0x64, SID: 0x9D***", when succeeded and should respond with an NACK when failed. The payload length is 5 bytes.

Structure:

<0xA0,0xA1>< PL><64><3C><CS><0x0D,0x0A>

Example:

A0 A1 00 05 64 3C 47 47 41 19 0D 0A

1 2 3 4 5

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Message Sub-ID | 3C | | UINT8 | |
| 3~5 | NMEA String*1 | 474741 | String of NMEA, exclude Talker<br>Ex. GGA in binary format is<br>0x47,0x47,0x41 | UINT8 | second |
| Payload Length : 5 bytes | | | | | |

*1 NMEA string supports: GGA, GNS, GSA, GSV, GLL, RMC, VTG, ZDA, DTM, GBS, GRS, GST, THS, HDT.

***QUERY REQUESTED NMEA STRING INTERVAL – Query requested NMEA string interval of GNSS receiver (ID: 0x64, SID: 0x40)***

This is a request message which is issued from the host to GNSS receiver to query the requested NMEA string interval. The GNSS receiver should respond with an ACK along with requested NMEA string and its message interval, ***"REQUESTED NMEA STRING INTERVAL, ID: 0x64, SID: 0x9F***", when succeeded and should respond with an NACK when failed. The payload length is 5 bytes.

Structure:
<0xA0,0xA1>< PL><64><40><CS><0x0D,0x0A>

Example:
A0 A1 00 05 64 40 47 47 41 65 0D 0A
1 2 3 4 5

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Message Sub-ID | 40 | | UINT8 | |
| 3~5 | NMEA String*1 | 474741 | String of NMEA, exclude Talker<br>Ex. GGA in binary format is<br>0x47,0x47,0x41 | UINT8 | second |
| Payload Length : 5 bytes | | | | | |

*1 NMEA string supports: GGA, GNS, GSA, GSV, GLL, RMC, VTG, ZDA, DTM, GBS, GRS, GST, THS, HDT.

***SOFTWARE IMAGE DOWNLOAD USING INTERNAL LOADER – Download software image to system flash using Internal loader (ID: 0x64, SID: 0x4E)*** *1

This is a request message which is issued from the host to GNSS receiver to download image to system flash using internal loader when download from flash mode to flash. The GNSS receiver should respond with an ACK when succeeded and should respond with an NACK when failed. The payload length is 7 bytes.

Structure:
<0xA0,0xA1>< PL><64><4E>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 07 64 4E 07 00 00 00 00 2D 0D 0A
1 2 3 4 5 6 7

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Sub ID | 4E | | UINT8 | |
| 3 | Baud | 07 | 0: 4800<br>1: 9600<br>2: 19200<br>3: 38400<br>4: 57600<br>5: 115200<br>6: 230400<br>7: 460800<br>8: 921600 | UINT8 | |
| 4 | Flash Type | 00 | 0: default, auto<br>1: QSPI Winbond<br>2. QSPI EON<br>3: Parallel Flash NUMONYX<br>4. Parallel Flash EON | UINT8 | |
| 5-6 | Flash ID | 0000 | If field 3 is not 0, then need to specify the flash ID | UINT16 | |
| 7 | Buffer Used Index | 00 | 0:8k<br>1:16K<br>2:24K<br>3:32K | UINT8 | |
| Payload Length : 7 bytes | | | | | |

*1 Please refer to SkyTraq software image download application notes and API

***SOFTWARE IMAGE DOWNLOAD USING EXTERNAL LOADER – Download software image to system flash using external loader (ID: 0x64, SID: 0x4F)*** *1

This is a request message which is issued from the host to GNSS receiver to download image to system flash using external loader when download from Flash to Flash. The GNSS receiver should respond with an ACK when succeeded and should respond with an NACK when failed. The payload length is 7 bytes.

Structure:
<0xA0,0xA1>< PL><64><4F>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 07 64 4F 07 00 00 00 00 2C 0D 0A
1 2 3 4 5 6 7

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Sub ID | 4F | | UINT8 | |
| 3 | Baud | 07 | 0: 4800<br>1: 9600<br>2: 19200<br>3: 38400<br>4: 57600<br>5: 115200<br>6: 230400<br>7: 460800<br>8: 921600 | UINT8 | |
| 4 | Flash Type | 00 | 0: default, auto<br>1: QSPI Winbond<br>2. QSPI EON<br>3: Parallel Flash NUMONYX<br>4. Parallel Flash EON | UINT8 | |
| 5-6 | Flash ID | 0000 | If field 3 is not 0, then need to specify the flash ID | UINT16 | |
| 7 | Buffer Used Index | 00 | 0:8k<br>1:16K<br>2:24K<br>3:32K | UINT8 | |
| Payload Length : 7 bytes | | | | | |

*1 Please refer to SkyTraq software image download application notes and API

***GNSS BOOT STATUS – Boot status of GNSS receiver (ID: 0x64, SID: 0x80)***

This is a response message to ***"QUERY GNSS BOOT STATUS, ID: 0x64, SID: 0x1"*** which provides the boot status of GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 4 bytes.

Structure:

<0xA0,0xA1>< PL><64><80>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 04 64 80 00 01 E5 0D 0A

1 2 3 4

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Message Sub-ID | 80 | | UINT8 | |
| 3 | Status | 00 | 0: Boot from flash OK<br>1: Boot from ROM due to flash boot failure | UINT8 | |
| 4 | Flash Type | 01 | 00: ROM<br>Bit 1: Winbond-type QSPI Flash<br>Bit 2: EON-type QSPI Flash<br>Bit 3: Parallel Flash | UINT8 | |
| Payload Length : 4 bytes | | | | | |

***EXTENDED NMEA MESSAGE INTERVAL– Extended NMEA message interval of the GNSS receiver (ID: 0x64, SID: 0x81)***

This is a response message to ***"QUERY EXTENDED NMEA MESSAGE INTERVAL, ID: 0x64, SID: 0x3"*** which provides the extended NMEA message interval of the GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 14 bytes.

Structure:
<0xA0,0xA1>< PL><64><81>< message body><CS><0x0D,0x0A>

Example:

| A0 | A1 | 00 | 0E | 64 | 81 | 01 | 01 | 03 | 01 | 01 | 01 | 01 | 00 | 00 | 00 | 00 | 00 | E6 | 0D | 0A |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | | | |

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Message Sub-ID | 81 | | UINT8 | |
| 3 | GGA Interval | 01 | 0 ~255, 0: disable | UINT8 | second |
| 4 | GSA Interval | 01 | 0 ~255, 0: disable | UINT8 | second |
| 5 | GSV Interval | 03 | 0 ~255, 0: disable | UINT8 | second |
| 6 | GLL Interval | 01 | 0 ~255, 0: disable | UINT8 | second |
| 7 | RMC Interval | 01 | 0 ~255, 0: disable | UINT8 | second |
| 8 | VTG Interval | 01 | 0 ~255, 0: disable | UINT8 | second |
| 9 | ZDA Interval | 01 | 0 ~255, 0: disable | UINT8 | second |
| 10 | GNS Interval | 00 | 0 ~255, 0: disable | UINT8 | second |
| 11 | GBS Interval | 00 | 0 ~255, 0: disable | UINT8 | second |
| 12 | GRS Interval | 00 | 0 ~255, 0: disable | UINT8 | second |
| 13 | DTM Interval | 00 | 0 ~255, 0: disable | UINT8 | second |
| 14 | GST Interval | 00 | 0 ~255, 0: disable | UINT8 | second |
| Payload Length : 14 bytes | | | | | |

***INTERFERENCE DETECTION STATUS – Interference detection status of GNSS receiver (ID: 0x64, SID: 0x83)***

This is a response message to ***"QUERY INTERFERENCE DETECTION STATUS, ID: 0x64, SID: 0x7"*** which provides the status of interference detection of the GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 4 bytes.

Structure:
<0xA0,0xA1>< PL><64><83>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 04 64 83 01 01 E7 0D 0A
1 2 3 4

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Message Sub-ID | 83 | | UINT8 | |
| 3 | Interference Detection Control | 01 | Interference detection control status<br>0: disable<br>1: enable | UINT8 | |
| 4 | Interference Status | 01 | 0: unknown<br>1: no interference<br>2: lite<br>3: critical | UINT8 | |
| Payload Length : 4 bytes | | | | | |

***GNSS PARAMETER SEARCH ENGINE NUMBER – Number of parameter search engine of GPS receiver (ID: 0x64, SID: 0x85)***

This is a response message to **"*QUERY GNSS PARAMETER SEARCH ENGINE NUMBER, ID: 0x64, SID: 0xB*"** which provides the number of parameter search engine of the GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 3 bytes.

Structure:
<0xA0,0xA1>< PL><64><85>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 03 64 85 01 E0 0D 0A
1 2 3

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Message Sub-ID | 0A | | UINT8 | |
| 3 | Search engine number | 01 | 0: PSE_MODE_DEFAULT<br>ROM version decided by HW power-on latch, FLASH version : by SW define<br>1: PSE_MODE_LOW (2 PSE)<br>2: PSE_MODE_MID (4 PSE)<br>3: PSE_MODE_HIGH (6 PSE)<br>4: PSE_MODE_FULL (8 PSE) | UINT8 | |
| Payload Length : 3 bytes | | | | | |

***POSITION FIX NAVIGATION MASK – Position fix navigation Mask of GNSS receiver (ID: 0x64, SID: 0x88)***

This is a response message to ***"QUERY POSITION FIX NAVIGATION MASK, ID: 0x64, SID: 0x12"***, which provides the position fix navigation mask of the GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 4 bytes.

Structure:
<0xA0,0xA1>< PL><64><88>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 04 64 88 00 00 EC 0D 0A
1 2 3 4

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Sub ID | 88 | | UINT8 | |
| 3 | First fix navigation mask | 00 | 0: 3D<br>1: 2D | UINT8 | |
| 4 | Subsequent fix navigation mask | 00 | 0: 3D<br>1: 2D | UINT8 | |
| Payload Length : 4 bytes | | | | | |

***GPS UTC REFERENCE TIME – UTC reference time of the GNSS receiver (ID: 0x64, SID: 0x8A)***

This is a response message to ***"QUERY GPS UTC REFERENCE TIME, ID: 0x64, SID: 0x16"*** which provides the UTC reference time of the GNSS receiver that synchronizes to GPS time. This message is sent from the GNSS receiver to host. The payload length is 7 bytes.

Structure:

<0xA0,0xA1>< PL><64><8A>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 07 64 8A 01 07 EC 01 01 04 0D 0A

1 2 3 4 5 6 7

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Message Sub-ID | 8A | | UINT8 | |
| 3 | Enable | 01 | 0: enable<br>1: disable | UINT8 | |
| 4-5 | UTC Year | 07EC | UTC year: 2028 | UINT16 | |
| 6 | UTC Month | 01 | UTC month: 01 | UINT8 | |
| 7 | UTC Day | 01 | UTC day: 01 | UINT8 | |
| Payload Length : 7 bytes | | | | | |

***GNSS NAVIGATION MODE – Navigation mode of the GNSS receiver (ID: 0x64, SID: 0x8B)***

This is a response message to ***"QUERY GNSS NAVIGATION MODE, ID: 0x64, SID: 0x18"*** which provides the navigation mode of the GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 3 bytes.

Structure:
<0xA0,0xA1>< PL><64><8B>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 03 64 8B 00 EF 0D 0A
1 2 3

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Message Sub-ID | 8B | | UINT8 | |
| 3 | Navigation mode | 00 | 0: auto<br>1: prdestrian<br>2: car<br>3: marine<br>4: balloon<br>5: airborne<br>6: reserved<br>7: quadcopter<br>8: reserved<br>9: SLR (Speed Lag Reduced) mode | UINT8 | |
| Payload Length : 3 bytes | | | | | |

***GNSS CONSTELLATION TYPE FOR NAVIGATION SOLUTION – GNSS constellation type for navigation solution (ID: 0x64, SID: 0x8C)***

This is a response message to ***"QUERY GNSS CONSTELLATION TYPE FOR NAVIGATION SOLUTION, ID 0x64, SID 0x1A"*** which provides the GNSS constellation type of the GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 4 bytes.

Structure:

<0xA0,0xA1>< PL><64><8C>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 04 64 8C 00 09 E1 0D 0A

1 2 3 4

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Sub ID | 8C | | UINT8 | |
| 3-4 | Navigation type | 00 09 | Bit 0: GPS<br>Bit 1: Glonass<br>Bit 2: Galileo<br>Bit 3: Beidou<br>Bit 4: Navic | UINT16 | |
| Payload Length : 4 bytes | | | | | |

***GPS TIME – GPS time of GNSS receiver (ID: 0x64, SID: 0x8E)***

This is a response message to "***QUERY GPS TIME, ID: 0x64, SID: 0x20",*** which provides the GPS time of the GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 15 bytes.

Structure:

<0xA0,0xA1>< PL><64><8E>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 0F 64 8E 1B 27 5A DD 00 0B B2 3D 06 F7 10 10 03 27 0D 0A

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Sub ID | 8E | | UINT8 | |
| 3-6 | Time of week | 1B275ADD | Time of week in unit of millisecond | UINT32 | ms |
| 7-10 | Sub time of week | 000BB23D | Millisecond fraction of tow in unit of nanosecond | UINT32 | ns |
| 11-12 | Week number | 06F7 | Week number | UINT16 | |
| 13 | Default leap seconds | 10 | Default GPS/UTC leap seconds | SINT08 | second |
| 14 | Current leap seconds | 10 | Current GPS/UTC leap seconds | SINT08 | second |
| 15 | Valid | 03 | BIT0: GPS time of week, 1: valid; 0: invalid<br>BIT1: GPS week number, 1: valid; 0: invalid<br>BIT2: GPS leap seconds from subfram4 page 18, 1: valid; 0: invalid | UINT08 | |
| Payload Length : 15 bytes | | | | | |

***PSTI MESSAGE INTERVAL – PSTI message interval of GNSS receiver (ID: 0x64, SID: 0x8F)***

This is a response message to **"*QUERY PSTI MESSAGE INTERVAL, ID 0x64, SID 0x22",*** which provides the PSTI message interval of the GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 3 bytes.

Structure:

<0xA0,0xA1>< PL><64><8F>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 03 64 8F 01 EA 0D 0A

1 2 3

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Sub ID | 8F | | UINT8 | |
| 3 | Message Interval | 01 | PSTI message interval<br>0: disable<br>1~255: interval or enable | UINT8 | |
| Payload Length : 3 bytes | | | | | |

***REQUESTED PSTI MESSAGE INTERVAL – Message interval of requested PSTI message ID interval of GNSS receiver (ID: 0x64, SID: 0x90)***

This is a response message to **"*QUERY REQUESTED PSTI MESSAGE INTERVAL, ID 0x64, SID 0x23",*** which provides the PSTI message interval of requested PSTI ID of the GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 4 bytes.

Structure:
<0xA0,0xA1>< PL><64><90>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 03 64 90 1E 01 EB 0D 0A
1 2 3

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Sub ID | 90 | | UINT8 | |
| 3 | Requested PSTI Message ID | 1E | SkyTraq proprietary message ID<br>Ex. A value equals 4 (hex, 0x04) corresponding to PSTI,004.<br>Ex. A value equals 30 (hex 0x1E) corresponding to PSTI,030 | UINT8 | |
| 4 | Message Interval | 01 | PSTI message interval<br>0: disable<br>1~255: interval or enable | UINT8 | |
| Payload Length : 4 bytes | | | | | |

***GNSS DATUM INDEX – Datum index of GNSS receiver (ID: 0x64, SID: 0x92)***

This is a response message to **"*QUERY GNSS DATUM INDEX, ID: 0x64, SID: 0x28",*** which provides the datum index of the GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 4 bytes.

Structure:
<0xA0,0xA1>< PL><64><92>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 04 64 92 00 00 F6 0D 0A
1 2 3 4

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Sub ID | 92 | | UINT8 | |
| 3-4 | Datum index | 0000 | Datum index, range 0-220. Please refer to Appendix B | UINT16 | |
| Payload Length : 4 bytes | | | | | |

***NAVIGATION DATA MESSAGE INTERVAL– Navigation data message interval of the GNSS receiver (ID: 0x64, SID: 0x90)***

This is a response message to ***"QUERY NAVIGATION DATA MESSAGE INTERVAL, ID: 0x64, SID: 0x30"*** which provides the navigation data message interval of the GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 3 bytes.

Structure:
<0xA0,0xA1>< PL><64><98>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 03 64 98 01 FD 0D 0A
1 2 3

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Message Sub-ID | 98 | | UINT8 | |
| 3 | Navigation Data Message Interval | 01 | 0 ~255, 0: disable | UINT8 | second |
| Payload Length : 3 bytes | | | | | |

***GNSS GEO-FENCING DATA BY POLYGON – Geo-fencing data by polygon of GNSS receiver (ID: 0x64, SID: 0x99)***

This is a response message to **"*QUERY GNSS GEO-FENCING DATA BY POLYGON, ID: 0x64, SID: 0x35",*** which provides the geo-fencing data by polygon of GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is maximum 260 bytes..

Structure:
<0xA0,0xA1>< PL><64><99>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 44 64 99 01 04 40 38 C8 E5 BF 18 FC 73 40 5E 40 90 38 79 65 94 40 38 C8 E9 C1 87 15 D6
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28

40 5E 40 92 D5 3D 3C 54 40 38 C8 F1 8D 47 37 07 40 5E 40 92 24 BC 08 40 40 38 C8 ED 64 06 8F BC
29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60

40 5E 40 8F 70 E0 2B BE 14 0D 0A
61 62 63 64 65 66 67 68

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Sub ID | 99 | | UINT8 | |
| 3 | Polygon | 01 | Polygon index, range: 1~4 | UINT8 | |
| 4 | Number of points | 04 | Number of points of a polygon<br>Maximum number: 16 | UINT8 | |
| 5-12 | Latitude | 4038C8E5<br>BF18FC73 | Latitude in double of polygon points #1 | DPFP | degree |
| 13-20 | Longitude | 405E4090<br>38796594 | Longitude in double of polygon points #1 | DPFP | degree |
| 21-28 | Latitude | 4038C8E9<br>C18715D6 | Latitude in double of polygon points #2 | DPFP | degree |
| 29-36 | Longitude | 405E4092<br>D53D3C54 | Longitude in double of polygon points #2 | DPFP | degree |
| 37-44 | Latitude | 4038C8F1<br>8D473707 | Latitude in double of polygon points #3 | DPFP | degree |
| 45-52 | Longitude | 405E4092<br>24BC0840 | Longitude in double of polygon points #3 | DPFP | degree |
| 53-60 | Latitude | 4038C8ED | Latitude in double of polygon points | DPFP | degree |

| | | | | | |
|---|---|---|---|---|---|
| | | 64068FBC | #4 | | |
| 61-68 | Longitude | 405E408F<br>70E02BBE | Longitude in double of polygon points #4 | DPFP | degree |
| ….. | | | | | |
| 5+((ndx-1)*16)<br>~<br>12+((ndx-1)*16) | Latitude | | Latitude in double of polygon points #ndx | DPFP | degree |
| 13+((ndx-1)*16)<br>~<br>20+((ndx-1)*16) | Longitude | | Longitude in double of polygon points #ndx | DPFP | degree |
| Payload Length : maximum 260 bytes, ndx = number of polygon points, maximum 16 points of each polygon | | | | | |

***GNSS MULTI-POLYGON GEO-FENCING RESULT – Multi-Polygon geo-fencing result of GNSS receiver (ID: 0x64, SID: 0x9A)*** [*1]

This is a response message to **"*QUERY GNSS MULTI-POLYGON GEO-FENCING RESULT, ID: 0x64, SID: 0x36*",** which provides the geo-fencing result of GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 22 bytes..

Structure:
<0xA0,0xA1>< PL><64><9A>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 16 64 9A 00 00 00 00 40 38 C8 FD C1 61 5E C0 40 5E 40 AB C5 15 48 67 87 0D 0A
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Sub ID | 9A | | UINT8 | |
| 3 | Result | 00 | 0: current GNSS position fix is out of configured polygon #1<br>1: current GNSS position fix within configured polygon #1 | UINT8 | |
| 4 | Result | 00 | 0: current GNSS position fix is out of configured polygon #2<br>1: current GNSS position fix within configured polygon #2 | UINT8 | |
| 5 | Result | 00 | 0: current GNSS position fix is out of configured polygon #3<br>1: current GNSS position fix within configured polygon #3 | UINT8 | |
| 6 | Result | 00 | 0: current GNSS position fix is out of configured polygon #4<br>1: current GNSS position fix within configured polygon #4 | UINT8 | |
| 7-14 | Latitude | 4038C8FD C1615EC0 | Latitude in double of current GNSS position fix | DPFP | degree |
| 15-22 | Longitude | 405E40AB C5154867 | Longitude in double of current GNSS position fix | DPFP | degree |
| Payload Length : 22 bytes | | | | | |

***NMEA STRING INTERVAL– NMEA string interval of the GNSS receiver (ID: 0x64, SID: 0x9D)***

This is a response message to ***"QUERY NMEA STRING INTERVAL, ID: 0x64, SID: 0x3C"*** which provides the NMEA string interval of the GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 3 bytes.

Structure:

<0xA0,0xA1>< PL><64><9D>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 03 64 9D 01 F8 0D 0A

1 2 3

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Message Sub-ID | 9D | | UINT8 | |
| 3 | String Interval | 01 | 0 ~255, 0: disable | UINT8 | second |
| Payload Length : 3 bytes | | | | | |

***REQUESTED NMEA STRING INTERVAL– Message interval of requested NMEA string of the GNSS receiver (ID: 0x64, SID: 0x9F)***

This is a response message to ***"QUERY REQUESTED NMEA STRING INTERVAL, ID: 0x64, SID: 0x40"*** which provides the requested NMEA string interval of the GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 6 bytes.

Structure:
<0xA0,0xA1>< PL><64><9F>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 03 64 9F 47 47 41 01 BB 0D 0A
1 2 3 4 5 6

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Message Sub-ID | 9F | | UINT8 | |
| 3~5 | Requested NMEA String | 474741 | String of NMEA, exclude Talker<br>Ex. GGA in binary format is 0x47,0x47,0x41 | UINT8 | second |
| 6 | String Interval | 01 | 0 ~255, 0: disable | UINT8 | second |
| Payload Length : 6 bytes | | | | | |

***VERSION EXTENSION STRING – Version extension string of GNSS receiver (ID: 0x64, SID: 0xFE)***

This is a response message to **"*QUERY VERSION EXTENSION STRING, ID: 0x64, SID: 0x7D"*** which provides the version extension string of the GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 34 bytes.

Structure:
<0xA0,0xA1>< PL><64><FE>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 22 64 FE 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29

00 00 00 00 00 9A 0D 0A
30 31 32 33 34

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 64 | | UINT8 | |
| 2 | Message Sub-ID | FE | | UINT8 | |
| 3~34 | Version extension string | 00~00 | Version extension string, 00 when end of string.<br>If the firmware is an official release, the version string is all 00.<br>If the firmware is under developed, the version string is "-dev-".<br>If the firmware is a release candidate, the version string is "-rc-". | UINT8 | |
| Payload Length : 34 bytes | | | | | |

***CONFIGURE 1PPS PULSE WIDTH – Configure 1PPS pulse width of GNSS receiver (ID: 0x65, SID: 0x1)***

This is a request message which will set the pulse width of 1PPS timing to the GNSS receiver. This command is issued from the host to GNSS receiver and GNSS receiver should respond with an ACK or NACK. If value of pulse width is not valid, the GNSS receiver will respond with an NACK. The payload length is 7 bytes.

Structure:

<0xA0,0xA1>< PL><65><01>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 07 65 01 00 00 00 01 00 65 0D 0A

1 2 3 4 5 6 7

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 65 | | UINT8 | - |
| 2 | Message Sub-ID | 01 | | UINT8 | |
| 3-6 | Pulse Width | 00 00 00 01 | Pulse width of 1PPS timing<br>Valid value between 1~100000 | UINT32 | us |
| 7 | Attributes | 00 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 7 bytes | | | | | |

***QUERY 1PPS PULSE WIDTH – Query 1PPS pulse width of GNSS receiver (ID: 0x65, SID: 0x2)***

This is a request message which is issued from the host to GNSS receiver to query 1PPS pulse width. The GNSS receiver should respond with an ACK along with information of 1PPS pulse width, ***"1PPS PULSE WIDTH, ID: 0x65, SID: 0x80"***, when succeeded and should respond with an NACK when failed. The payload length is 2 bytes.

Structure:

<0xA0,0xA1>< PL><65><02>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 02 65 02 67 0D 0A

1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 65 | | UINT8 | |
| 2 | Message Sub-ID | 02 | | UINT8 | |
| Payload Length : 2 bytes | | | | | |

***CONFIGURE PPS2 FREQUENCY OUTPUT – Configure frequency output of PPS2 (ID: 0x65, SID: 0x3)***

This is a request message which will set the frequency output of PPS2 to the GNSS receiver. This command is issued from the host to GNSS receiver and GNSS receiver should respond with an ACK or NACK. If value of frequency is not valid, the GNSS receiver will respond with an NACK. The payload length is 7 bytes.

Structure:

<0xA0,0xA1>< PL><65><03>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 07 65 03 00 00 00 01 01 66 0D 0A

1 2 3 4 5 6 7

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 65 | | UINT8 | - |
| 2 | Sub ID | 03 | | UINT8 | |
| 3-6 | Frequency output | 00000001 | Frequency output of PPS2<br>Valid value between 0~19200000 | UINT32 | Hz |
| 7 | Attributes | 01 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 7 bytes | | | | | |

***QUERY PPS2 FREQUENCY OUTPUT – Query PPS2 frequency output of the GNSS receiver (ID: 0x65, SID: 0x4)***

This is a request message which is issued from the host to GNSS receiver to query PPS2 frequency. The GNSS receiver should respond with an ACK along with information of PPS2 frequency, ***"PPS2 FREQUENCY OUTPUT ID 0x65, SID 0x81"***, when succeeded and should respond with an NACK when failed. The payload length is 2 bytes.

Structure:

<0xA0,0xA1>< PL><65><04>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 02 65 04 61 0D 0A

1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 65 | | UINT8 | |
| 2 | Sub ID | 04 | | UINT8 | |
| Payload Length : 2 bytes | | | | | |

***1PPS PULSE WIDTH – 1PPS pulse width of GNSS receiver (ID: 0x65, SID: 0x80)***

This is a response message to ***"QUERY 1PPS PULSE WIDTH, ID: 0x65, SID: 0x2"*** which provides the 1PPS pulse width of the GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 6 bytes.

Structure:

<0xA0,0xA1>< PL><65><80>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 06 65 80 00 00 00 01 E4 0D 0A

1 2 3 4 5 6

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 65 | | UINT8 | - |
| 2 | Message Sub-ID | 80 | | UINT8 | |
| 3-6 | Pulse Width | 00 00 00 01 | Pulse Width of 1PPS timing mode | UINT32 | us |
| Payload Length : 6 bytes | | | | | |

***PPS2 FREQUENCY OUTPUT – PPS2 frequency of the GNSS receiver (ID: 0x65, SID: 0x81)***

This is a response message to **"*QUERY PPS2 FREQUENCY OUTPUT, ID: 0x65, SID: 0x4",*** which provides the PPS2 frequency of the GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 6 bytes.

Structure:

<0xA0,0xA1>< PL><65><81>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 06 65 81 00 00 00 01 E5 0D 0A

1 2 3 4 5 6

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 65 | | UINT8 | - |
| 2 | Sub ID | 81 | | UINT8 | |
| 3-6 | Frequency | 00000001 | Frequency of PPS2 | UINT32 | Hz |
| Payload Length : 6 bytes | | | | | |

***SET BEIDOU EPHEMERIS – Set Beidou ephemeris to GNSS receiver (ID: 0x67, SID: 0x01)***

This is a request message which is issued from the host to GNSS receiver to set Beidou ephemeris data (open an ephemeris file) to GNSS receiver. The GNSS receiver should respond with an ACK when succeeded and should respond with an NACK when failed. There are 2 types of ephemeris corresponding to 2 types of Beidou satellites, GEO satellite and MEO/IGSO satellite. The GEO payload length is 126 bytes and the MEO/IGSO payload length is 87 bytes.

Structure:
<0xA0,0xA1>< PL><67><01>< message body><CS><0x0D,0x0A>

Example for GEO type of satellites:

a0 a1 00 7E 67 01 00 04 00 01 00 00 00 01 07 2C C1 60 F4 00 00 00 00 00 00 00 00 00 00 00 00 00
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28

00 00 00 00 00 00 00 00 00 00 1C B1 40 00 00 00 1C 3C 50 00 04 37 F8 0A 08 00 00 00 58 80 4E D4
29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60

4D 22 34 60 04 00 00 00 5C B0 EC C8 BA D8 C0 00 00 00 00 00 B0 FF F5 E4 78 2C 04 E9 28 00 00 00
61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92

24 FF 07 A8 04 7F FC FF E9 00 00 00 E4 E3 CE 8C C5 2C 24 4A 1C 00 00 00 08 30 C8 00 00 00 00 00
93 94 95 96 97 98 99 100101102103104105106107108109110111112113114115116117118119120121122123124

00 00 5B 0D 0A
125126

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 67 | | UINT8 | |
| 2 | Sub ID | 01 | | UINT8 | |
| 3-4 | SV id | 0004 | Satellite id | UINT16 | |
| 5 | Type | 00 | 0: GEO satellite<br>1: MEO/IGSO satellite | | |
| 6 | valid | 01 | 0: not valid<br>1: valid | UINT8 | |
| 7 | SubFrameData[0][0] | | Eph data subframe 1 | UINT8 | |
| 8 | SubFrameData[0][1] | | Eph data subframe 1 | UINT8 | |
| 9 | SubFrameData[0][2] | | Eph data subframe 1 | UINT8 | |
| 10 | SubFrameData[0][3] | | Eph data subframe 1 | UINT8 | |

| | | | | | |
|---|---|---|---|---|---|
| 11 | SubFrameData[0][4] | | Eph data subframe 1 | UINT8 | |
| 12 | SubFrameData[0][5] | | Eph data subframe 1 | UINT8 | |
| 13 | SubFrameData[0][6] | | Eph data subframe 1 | UINT8 | |
| 14 | SubFrameData[0][7] | | Eph data subframe 1 | UINT8 | |
| 15 | SubFrameData[0][8] | | Eph data subframe 1 | UINT8 | |
| 16 | SubFrameData[0][9] | | Eph data subframe 1 | UINT8 | |
| 17 | SubFrameData[0][10] | | Eph data subframe 1 | UINT8 | |
| 18 | SubFrameData[0][11] | | Eph data subframe 1 | UINT8 | |
| 19~30 | SubFrameData[1][0~11] | | Eph data subframe 2, same as field 7-18 for GEO satellite | UINT8 | |
| 31-42 | SubFrameData[2][0~11] | | Eph data subframe 2, same as field 7-18 for GEO satellite | UINT8 | |
| 43-54 | SubFrameData[3][0~11] | | Eph data subframe 3, same as field 7-18 for GEO satellite | UINT8 | |
| 55-66 | SubFrameData[4][0~11] | | Eph data subframe 4, same as field 7-18 for GEO satellite | UINT8 | |
| 67-78 | SubFrameData[5][0~11] | | Eph data subframe 5, same as field 7-18 for GEO satellite | UINT8 | |
| 79-90 | SubFrameData[6][0~11] | | Eph data subframe 6, same as field 7-18 for GEO satellite | UINT8 | |
| 91-102 | SubFrameData[7][0~11] | | Eph data subframe 7, same as field 7-18 for GEO satellite | UINT8 | |
| 103-114 | SubFrameData[8][0~11] | | Eph data subframe 8, same as field 7-18 for GEO satellite | UINT8 | |
| 115-126 | SubFrameData[9][0~11] | | Eph data subframe 9, same as field 7-18 for GEO satellite | UINT8 | |
| Payload Length : 126 bytes for GEO type of satellites | | | | | |

Example of MEI/IGSO type of satellites:

A0 A1 00 57 67 01 00 06 01 01 00 00 00 10 72 F0 16 18 C0 00 00 00 00 00 00 00 00 00 01 FF D8 63
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28

FF FC B9 FC 84 00 01 44 FF F3 70 81 B8 20 B2 30 20 7C 75 C4 0B 6A 80 FF FF F0 FE B2 B8 A9 E9 54
29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60

00 07 80 B1 34 6C FF FF FC FF FF FC FF FF FC FF F8 D0 98 74 8C 16 52 CC AD 98 C8 55 0D 0A
61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 67 | | UINT8 | |
| 2 | Sub ID | 01 | | UINT8 | |
| 3-4 | SV id | 0006 | Satellite id | UINT16 | |
| 5 | Type | 01 | 0: GEO satellite<br>1: MEO/IGSO satellite | | |
| 6 | valid | 01 | 0: not valid<br>1: valid | UINT8 | |
| 7 | SubFrameData[0][0] | | Eph data subframe 1 | UINT8 | |
| 8 | SubFrameData[0][1] | | Eph data subframe 1 | UINT8 | |
| 9 | SubFrameData[0][2] | | Eph data subframe 1 | UINT8 | |
| 10 | SubFrameData[0][3] | | Eph data subframe 1 | UINT8 | |
| 11 | SubFrameData[0][4] | | Eph data subframe 1 | UINT8 | |
| 12 | SubFrameData[0][5] | | Eph data subframe 1 | UINT8 | |
| 13 | SubFrameData[0][6] | | Eph data subframe 1 | UINT8 | |
| 14 | SubFrameData[0][7] | | Eph data subframe 1 | UINT8 | |
| 15 | SubFrameData[0][8] | | Eph data subframe 1 | UINT8 | |
| 16 | SubFrameData[0][9] | | Eph data subframe 1 | UINT8 | |
| 17 | SubFrameData[0][10] | | Eph data subframe 1 | UINT8 | |
| 18 | SubFrameData[0][11] | | Eph data subframe 1 | UINT8 | |
| 19 | SubFrameData[0][12] | | Eph data subframe 1 | UINT8 | |
| 20 | SubFrameData[0][13] | | Eph data subframe 1 | UINT8 | |
| 21 | SubFrameData[0][14] | | Eph data subframe 1 | UINT8 | |
| 22 | SubFrameData[0][15] | | Eph data subframe 1 | UINT8 | |
| 23 | SubFrameData[0][16] | | Eph data subframe 1 | UINT8 | |
| 24 | SubFrameData[0][17] | | Eph data subframe 1 | UINT8 | |
| 25 | SubFrameData[0][18] | | Eph data subframe 1 | UINT8 | |
| 26 | SubFrameData[0][19] | | Eph data subframe 1 | UINT8 | |
| 27 | SubFrameData[0][20] | | Eph data subframe 1 | UINT8 | |
| 28 | SubFrameData[0][21] | | Eph data subframe 1 | UINT8 | |
| 29 | SubFrameData[0][22] | | Eph data subframe 1 | UINT8 | |
| 30 | SubFrameData[0][23] | | Eph data subframe 1 | UINT8 | |
| 31 | SubFrameData[0][24] | | Eph data subframe 1 | UINT8 | |
| 32 | SubFrameData[0][25] | | Eph data subframe 1 | UINT8 | |
| 33 | SubFrameData[0][26] | | Eph data subframe 1 | UINT8 | |
| 34~60 | SubFrameData[1][0~26] | | Eph data subframe 2, same as field 7-33 for MEO/IGSO satellite | UINT8 | |
| 61-87 | SubFrameData[2][0~26] | | Eph data subframe 3, same as field 7-33 for MEO/IGSO satellite | UINT8 | |

| Payload Length : 87 bytes for MEO/IGSO type of satellites |
|---|

***GET BEUDOU EPHEMERIS – Get Beidou ephemeris data used of GNSS receiver (ID: 0x67, SID: 0x02)***

This is a request message which is issued from the host to GNSS receiver to retrieve ephemeris data. The GNSS receiver should respond with an ACK along with information of ephemeris, **"*BEIDOU EPHEMERIS DATA, ID 0x67, SID 0x80*"**, when succeeded and should respond with an NACK when failed. The payload length is 3 bytes.

Structure:

<0xA0,0xA1>< PL><67><02>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 03 67 02 00 65 0D 0A

1 2 3

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 67 | | UINT8 | |
| 2 | Sub ID | 02 | | UINT8 | |
| 3 | SV # | 00 | 0: means all SVs<br>1~32 : mean for the particular SV | UINT8 | |
| Payload Length : 3 bytes | | | | | |

***BEIDOU EPHEMERIS DATA – Beidou ephemeris data of the GNSS receiver (ID: 0x67, SID: 0x80)***

This is a response message to ***"GET BEIDOU EPHEMERIS, ID 0x67, SID 0x02",*** which provides the Beidou ephemeris data of the GNSS receiver to host. The host will save the ephemeris data as an ephemeris file. This message is sent from the GNSS receiver to host. There are 2 types of ephemeris corresponding to 2 types of Beidou satellites, GEO satellite and MEO/IGSO satellite. The GEO payload length is 126 bytes and the MEO/IGSO payload length is 87 bytes.

Structure:
<0xA0,0xA1>< PL><67><80>< message body><CS><0x0D,0x0A>

Example of GEO type of satellites:

A0 A1 00 7E 67 80 00 01 00 01 00 00 00 01 07 2C C1 62 38 00 00 00 00 00 00 00 00 00 00 00 00 00
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28

00 00 00 00 00 00 00 00 3F F5 DC 7A 60 00 00 00 24 F9 90 00 04 1B E4 FE 80 00 00 00 58 50 21 B4
29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60

4D 20 EC 00 00 00 00 00 94 B0 9E C8 BA FE 70 FF FC 00 00 00 90 FF EE 34 78 2C 04 AE 54 00 00 00
61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92

5C FF 12 BC FF 32 40 00 19 00 00 00 EC D6 B2 9C 48 2E 50 77 F0 00 00 00 B8 2D C8 00 00 00 00 00
93 94 95 96 97 98 99 100101102103104105106107108109110111112113114115116117118119120121122123124

00 00 B5 0D 0A
125126

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 67 | | UINT8 | |
| 2 | Sub ID | 80 | | UINT8 | |
| 3-4 | SV id | 0001 | Satellite id | UINT16 | |
| 5 | Type | 00 | 0: GEO satellite<br>1: MEO/IGSO satellite | UINT8 | |
| 6 | valid | 01 | 0: not valid<br>1: valid | UINT8 | |
| 7 | SubFrameData[0][0] | | Eph data subframe 1 | UINT8 | |
| 8 | SubFrameData[0][1] | | Eph data subframe 1 | UINT8 | |
| 9 | SubFrameData[0][2] | | Eph data subframe 1 | UINT8 | |
| 10 | SubFrameData[0][3] | | Eph data subframe 1 | UINT8 | |

| | | | | | |
|---|---|---|---|---|---|
| 11 | SubFrameData[0][4] | | Eph data subframe 1 | UINT8 | |
| 12 | SubFrameData[0][5] | | Eph data subframe 1 | UINT8 | |
| 13 | SubFrameData[0][6] | | Eph data subframe 1 | UINT8 | |
| 14 | SubFrameData[0][7] | | Eph data subframe 1 | UINT8 | |
| 15 | SubFrameData[0][8] | | Eph data subframe 1 | UINT8 | |
| 16 | SubFrameData[0][9] | | Eph data subframe 1 | UINT8 | |
| 17 | SubFrameData[0][10] | | Eph data subframe 1 | UINT8 | |
| 18 | SubFrameData[0][11] | | Eph data subframe 1 | UINT8 | |
| 19~30 | SubFrameData[1][0~11] | | Eph data subframe 2, same as field 7-18 for GEO satellite | UINT8 | |
| 31-42 | SubFrameData[2][0~11] | | Eph data subframe 2, same as field 7-18 for GEO satellite | UINT8 | |
| 43-54 | SubFrameData[3][0~11] | | Eph data subframe 3, same as field 67-18 for GEO satellite | UINT8 | |
| 55-66 | SubFrameData[4][0~11] | | Eph data subframe 4, same as field 7-18 for GEO satellite | UINT8 | |
| 67-78 | SubFrameData[5][0~11] | | Eph data subframe 5, same as field 7-18 for GEO satellite | | |
| 79-90 | SubFrameData[6][0~11] | | Eph data subframe 6, same as field 7-18 for GEO satellite | | |
| 91-102 | SubFrameData[7][0~11] | | Eph data subframe 7, same as field 7-18 for GEO satellite | | |
| 103-114 | SubFrameData[8][0~11] | | Eph data subframe 8, same as field 7-18 for GEO satellite | | |
| 115-126 | SubFrameData[9][0~11] | | Eph data subframe 9, same as field 7-18 for GEO satellite | | |
| Payload Length : 126 bytes for GEO type of satellites | | | | | |

Example for MEI/IGSO type of satellites:

A0 A1 00 57 67 80 00 06 01 01 00 00 00 10 72 F0 16 18 C0 00 00 00 00 00 00 00 00 00 01 FF D8 63
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28

FF FC B9 FC 84 00 01 44 FF F3 70 81 B8 20 B2 30 20 7C 75 C4 0B 6A 80 FF FF F0 FE B2 B8 A9 E9 54
29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60

00 07 80 B1 34 6C FF FF FC FF FF FC FF FF FC FF F8 D0 98 74 8C 16 52 CC AD 98 C8 D4 0D 0A
61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|

| | | | | | |
|---|---|---|---|---|---|
| 1 | Message ID | 67 | | UINT8 | |
| 2 | Sub ID | 80 | | UINT8 | |
| 3-4 | SV id | 0006 | Satellite id | UINT16 | |
| 5 | Type | 01 | 0: GEO satellite<br>1: MEO/IGSO satellite | UINT8 | |
| 6 | valid | 01 | 0: not valid<br>1: valid | UINT8 | |
| 7 | SubFrameData[0][0] | | Eph data subframe 1 | UINT8 | |
| 8 | SubFrameData[0][1] | | Eph data subframe 1 | UINT8 | |
| 9 | SubFrameData[0][2] | | Eph data subframe 1 | UINT8 | |
| 10 | SubFrameData[0][3] | | Eph data subframe 1 | UINT8 | |
| 11 | SubFrameData[0][4] | | Eph data subframe 1 | UINT8 | |
| 12 | SubFrameData[0][5] | | Eph data subframe 1 | UINT8 | |
| 13 | SubFrameData[0][6] | | Eph data subframe 1 | UINT8 | |
| 14 | SubFrameData[0][7] | | Eph data subframe 1 | UINT8 | |
| 15 | SubFrameData[0][8] | | Eph data subframe 1 | UINT8 | |
| 16 | SubFrameData[0][9] | | Eph data subframe 1 | UINT8 | |
| 17 | SubFrameData[0][10] | | Eph data subframe 1 | UINT8 | |
| 18 | SubFrameData[0][11] | | Eph data subframe 1 | UINT8 | |
| 19 | SubFrameData[0][12] | | Eph data subframe 1 | UINT8 | |
| 20 | SubFrameData[0][13] | | Eph data subframe 1 | UINT8 | |
| 21 | SubFrameData[0][14] | | Eph data subframe 1 | UINT8 | |
| 22 | SubFrameData[0][15] | | Eph data subframe 1 | UINT8 | |
| 23 | SubFrameData[0][16] | | Eph data subframe 1 | UINT8 | |
| 24 | SubFrameData[0][17] | | Eph data subframe 1 | UINT8 | |
| 25 | SubFrameData[0][18] | | Eph data subframe 1 | UINT8 | |
| 26 | SubFrameData[0][19] | | Eph data subframe 1 | UINT8 | |
| 27 | SubFrameData[0][20] | | Eph data subframe 1 | UINT8 | |
| 28 | SubFrameData[0][21] | | Eph data subframe 1 | UINT8 | |
| 29 | SubFrameData[0][22] | | Eph data subframe 1 | UINT8 | |
| 30 | SubFrameData[0][23] | | Eph data subframe 1 | UINT8 | |
| 31 | SubFrameData[0][24] | | Eph data subframe 1 | UINT8 | |
| 32 | SubFrameData[0][25] | | Eph data subframe 1 | UINT8 | |
| 33 | SubFrameData[0][26] | | Eph data subframe 1 | UINT8 | |
| 34~60 | SubFrameData[1][0~26] | | Eph data subframe 2, same as field 7-33 for MEO/IGSO satellite | UINT8 | |
| 61-87 | SubFrameData[2][0~26] | | Eph data subframe 3, same as field 7-33 for MEO/IGSO satellite | UINT8 | |
| Payload Length : 87 bytes for MEO/IGSO type of satellites | | | | | |

***CONFIGURE RTK MODE AND OPERATIONAL FUNCTION – Configure Real Time Kinematic mode and operational function of GNSS receiver (ID: 0x6A, SID: 0x6)*** [*1]

This is a request message which will set Real Time Kinematic mode and operational function to the GNSS receiver. This command is issued from the host to GNSS receiver and GNSS receiver should respond with an ACK or NACK. The payload length is 37 bytes.

Structure:

<0xA0,0xA1>< PL><6A><06>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 25 6A 06 00 00 00 00 07 D0 00 00 00 1E 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29

00 00 00 00 00 00 00 00 A5 0D 0A

30 31 32 33 34 35 36 37

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 6A | | UINT8 | - |
| 2 | Message Sub-ID | 06 | | UINT8 | |
| 3 | RTK Mode | 00 | 0: RTK rover mode<br>1: RTK base mode[*2]<br>2: RTK precisely kinematic base mode | UINT8 | |
| 4 | RTK Operational Function | 00 | If field 3 RTK Mode is RTK rover mode<br>0: Normal<br>1: Float<br>2: Moving base<br>If field 3 RTK Mode is RTK base mode<br>0: Kinematic<br>1: Survey<br>2: Static<br>If field 3 RTK Mode is RTK precisely kinematic base mode<br>0: Normal<br>1: Float | UINT8 | |
| 5-8 | Survey Length | 000007D0 | Used when RTK base mode.<br>Survey length when in RTK survey operational function, not used when in other operational function.<br>Valid values between 60~1209600 | UINT32 | second |

| | | | | | |
|---|---|---|---|---|---|
| 9-12 | Standard Deviation | 0000001E | Used when RTK base mode.<br>Standard Deviation when in RTK survey operational function, not used when in other operational function.<br>Valid values between 3~100 | UINT32 | meter |
| 13-20 | Latitude | 0000000000000000 | Used when RTK base mode.<br>Latitude in double in RTK static operational function, not used when in other operational function. | DPFP | degree |
| 21-28- | Longitude | 0000000000000000 | Used when RTK base mode.<br>Longitude in double in RTK static operational function, not used when in other operational function. | DPFP | degree |
| 29-32 | Altitude | 00000000 | Used when RTK base mode.<br>Altitude in float in RTK Static operational function, not used when in other operational function. | SPFP | meter |
| 33-36 | Baseline length constraint | 00000000 | Used for moving base mode when the baseline length is fixed and known to centimeter-level accuracy<br>Input 0 if baseline length is unknown or floating | SPFP | meter |
| 37 | Attributes | 00 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 37 bytes | | | | | |

*1 supported only in RTK mode receivers

*2 Please refer to AN0030, AN0039 for raw measurement data output format when in RTK base mode.

***QUERY RTK MODE AND OPERATIONAL FUNCTION – Query Real Time Kinematic mode and operational function of GNSS receiver (ID: 0x6A, SID: 0x7)*** [*1]

This is a request message which is issued from the host to GNSS receiver to query Real Time Kinematic mode and its operational function. The GNSS receiver should respond with an ACK along with mode of RTK, ***"RTK MODE AND OPERATION FUNCTION, ID: 0x6A, SID: 0x83"***, when succeeded and should respond with an NACK when failed. The payload length is 2 bytes.

Structure:
<0xA0,0xA1>< PL><6A><07>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 02 6A 07 6D 0D 0A
1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 6A | | UINT8 | |
| 2 | Message Sub-ID | 07 | | UINT8 | |
| Payload Length : 2 bytes | | | | | |

*1 supported only in RTK mode receivers

***CONFIGURE RTK SLAVE BASE SERIAL PORT BAUD RATE – Configure RTK Slave Base Serial Port Baud rate of GNSS receiver (ID: 0x6A, SID: 0xC)*** [*1]

This is a request message which will set RTK slave base serial port baud rate to the GNSS receiver. This command is issued from the host to GNSS receiver and GNSS receiver should respond with an ACK or NACK. The payload length is 4 bytes.

Structure:
<0xA0,0xA1>< PL><6A><0C>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 04 6A 0C 05 01 62 0D 0A
1 2 3 4

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 6A | | UINT8 | - |
| 2 | Message Sub-ID | 0C | | UINT8 | |
| 3 | Baud Rate | 05 | 0: 4800<br>1: 9600<br>2: 19200<br>3: 38400<br>4: 57600<br>5: 115200<br>6: 230400<br>7: 460800<br>8: 921600 | UINT8 | |
| 4 | Attributes | 00 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 4 bytes | | | | | |

*1 supported only in RTK mode receivers

***QUERY RTK SLAVE BASE SERIAL PORT BAUD RATE – Query RTK Base Serial Port Baud Rate of GNSS receiver (ID: 0x6A, SID: 0xD)*** [*1]

This is a request message which is issued from the host to GNSS receiver to query RTK slave base serial port baud rate. The GNSS receiver should respond with an ACK along with RTK slave base serial port baud rate, ***"RTK SLAVE BASE SERIAL PORT BAUD RATE, ID: 0x6A, SID: 0x85"***, when succeeded and should respond with an NACK when failed. The payload length is 2 bytes.

Structure:

<0xA0,0xA1>< PL><6A><0D>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 02 6A 0D 67 0D 0A

1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 6A | | UINT8 | |
| 2 | Message Sub-ID | 0D | | UINT8 | |
| Payload Length : 2 bytes | | | | | |

*1 supported only in RTK mode receivers

***CONFIGURE RTK PRECISELY KINEMATIC BASE SERIAL PORT BAUD RATE – Configure RTK Precisely Kinematic Base Serial Port Baud rate of GNSS receiver (ID: 0x6A, SID: 0x13)*** [*1]

This is a request message which will set RTK precisely kinematic base serial port baud rate to the GNSS receiver. This command is issued from the host to GNSS receiver and GNSS receiver should respond with an ACK or NACK. The payload length is 4 bytes.

Structure:

<0xA0,0xA1>< PL><6A><13>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 04 6A 13 06 01 7E 0D 0A

1 2 3 4

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 6A | | UINT8 | - |
| 2 | Message Sub-ID | 13 | | UINT8 | |
| 3 | Baud Rate | 06 | 0: 4800<br>1: 9600<br>2: 19200<br>3: 38400<br>4: 57600<br>5: 115200<br>6: 230400<br>7: 460800<br>8: 921600 | UINT8 | |
| 4 | Attributes | 00 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 4 bytes | | | | | |

*1 supported only in RTK mode receivers

***QUERY RTK PRECISELY KINEMATIC BASE SERIAL PORT BAUD RATE – Query RTK Precisely Kinematic Base Serial Port Baud Rate of GNSS receiver (ID: 0x6A, SID: 0x14)*** [*1]

This is a request message which is issued from the host to GNSS receiver to query RTK precisely kinematic base serial port baud rate. The GNSS receiver should respond with an ACK along with RTK precisely kinematic base serial port baud rate, ***"RTK PRECISELY KINEMATIC BASE SERIAL PORT BAUD RATE, ID: 0x6A, SID: 0x88"***, when succeeded and should respond with an NACK when failed. The payload length is 2 bytes.

Structure:
<0xA0,0xA1>< PL><6A><14>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 02 6A 14 7E 0D 0A
1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 6A | | UINT8 | |
| 2 | Message Sub-ID | 14 | | UINT8 | |
| Payload Length : 2 bytes | | | | | |

*1 supported only in RTK mode receivers

***CONFIGURE RTK ROVER MOVING BASE HEADING AND PITCH OFFSETS – Configure heading and pitch offsets of RTK rover moving base GNSS receiver (ID: 0x6A, SID: 0x15)*** *[1]*

This is a request message which will set heading and pitch offsets of the RTK rover moving base GNSS receiver. This command is issued from the host to GNSS receiver and GNSS receiver should respond with an ACK or NACK. The payload length is 15 bytes.

Structure:

<0xA0,0xA1>< PL><6A><15>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 0F 6A 15 41 70 00 00 00 00 00 00 00 00 00 00 01 4F 0D 0A

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 6A | | UINT8 | - |
| 2 | Message Sub-ID | 15 | | UINT8 | |
| 3-6 | Heading offset | 41700000 | Heading offset in degree<br>-180.00 ~ +180.00, default value = 0.00 | F32 | |
| 7-10 | Pitch offset | 00000000 | Pitch offset in degree<br>-90.0 ~ +90.0, default value = 0.0 | F32 | |
| 11-14 | reserved | 00000000 | reserved | | |
| 15 | Attributes | 01 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 15 bytes | | | | | |

*1 supported only in RTK mode receivers

***QUERY RTK ROVER MOVING BASE HEADING AND PITCH OFFSETS – Query heading and pitch offsets of RTK ROVER MOVING BASE GNSS receiver (ID: 0x6A, SID: 0x16)*** [*1]

This is a request message which is issued from the host to GNSS receiver to query heading and pitch offsets of RTK rover moving base GNSS receiver. The GNSS receiver should respond with an ACK along with heading and pitch offsets, ***"HEADING AND PITCH OFFSETS OF RTK ROVER MOVING BASE, ID: 0x6A, SID: 0x89"***, when succeeded and should respond with an NACK when failed. The payload length is 2 bytes.

Structure:
<0xA0,0xA1>< PL><6A><16>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 02 6A 16 7C 0D 0A
1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 6A | | UINT8 | |
| 2 | Message Sub-ID | 16 | | UINT8 | |
| Payload Length : 2 bytes | | | | | |

*1 supported only in RTK mode receivers

***RTK MODE AND OPERATIONAL FUNCTION – Real Time Kinematic mode and operational function of the GNSS receiver (ID: 0x6A, SID: 0x83)*** [*1]

This is a response message to **"*QUERY RTK MODE AND OPERATIONAL FUNCTION, ID: 0x6A, SID: 0x7",*** which provides all information of the Real Time Kinematic operational mode of the GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 41 bytes.

Structure:

<0xA0,0xA1>< PL><6A><83>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 29 6A 83 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29

00 00 00 00 00 00 00 00 00 00 00 00 E9 0D 0A

30 31 32 33 34 35 36 37 38 39 40 41

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 6A | | UINT8 | - |
| 2 | Message Sub-ID | 83 | | UINT8 | |
| 3 | RTK Mode | 00 | 0: RTK rover mode<br>1: RTK base mode[*2]<br>2: RTK precisely kinematic base mode | UINT8 | |
| 4 | RTK Operational Function | 00 | When field 3 RTK Mode is RTK rover mode<br>0: Normal<br>1: Float<br>2: Moving base<br>When field 3 RTK Mode is RTK base mode<br>0: Kinematic<br>1: Survey<br>2: Static<br>When field 3 RTK Mode is RTK precisely kinematic base mode<br>0: Normal<br>1: Float<br>Value saved in SRAM/Flash by request command | UINT8 | |
| 5-8 | Saved Survey Length | 00000000 | Used when in RTK base mode.<br>Survey length used when in RTK | UINT32 | second |

| | | | | | |
|---|---|---|---|---|---|
| | | | survey operational function, not used when in other operational function. | | |
| 9-12 | Standard deviation | 00000000 | Used when in RTK base mode<br>Standard Deviation when in RTK survey operational function, not used when in other operational function.<br>Valid values between 3~100 | UINT32 | meter |
| 13-20 | Latitude | 0000000000000000 | Used when in RTK base mode<br>1. Saved latitude in double in RTK static operational function<br>or<br>2. Run-time latitude in double when in RTK survey operational function and run-time operational function (field 33) is static operational function | DPFP | degree |
| 21-28 | Longitude | 0000000000000000 | Used when in RTK base mode<br>1. Saved longitude in double in RTK static operational function<br>or<br>2. Run-time longitude in double when in RTK survey operational function and run-time operational function (field 33) is static operational function | DPFP | degree |
| 29-32 | Altitude | 00000000 | Used when in RTK base mode<br>1. Saved altitude in float in RTK static operational function<br>or<br>2. Run-time altitude in float when in RTK survey operational function and run-time operational function (field 33) is static operational function | SPFP | meter |
| 33-36 | Baseline length constraint | 00000000 | Used for moving base mode when the baseline length is fixed and known to centimeter-level accuracy | SPFP | meter |
| 37 | Run-time Operational Function | 00 | Used when in RTK base mode<br>00 = Normal<br>01 = Survey | UINT8 | |

| | | | | | |
|---|---|---|---|---|---|
| | | | 02 = Static<br>Value currently used and not saved in SRAM/Flash | | |
| 38-41 | Run-time Survey Length | 00000000 | Used when in RTK base mode<br>Survey length used when in “Run-time Survey Operational Function”.<br>Value currently used and not saved in SRAM/Flash | UINT32 | second |
| Payload Length : 41 bytes | | | | | |

*1 supported only in RTK mode receivers

*2 Please refer to AN0030, AN0039 for raw measurement data output format when in RTK base mode.

***RTK SLAVE BASE SERIAL PORT BAUD RATE – RTK Slave Base Serial Port Baud Rate of the GNSS receiver (ID: 0x6A, SID: 0x85)*** [*1]

This is a response message to **"*QUERY RTK SLAVE BASE SERIAL PORT BAUD RATE, ID: 0x6A, SID: 0xD",*** which provides the RTK slave base serial port baud rate of the RTK GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 3 bytes.

Structure:

<0xA0,0xA1>< PL><6A><85>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 03 6A 85 05 EA 0D 0A

1 2 3

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 6A | | UINT8 | - |
| 2 | Sub ID | 85 | | UINT8 | |
| 3 | Baud Rate | 00 | 0: 4800<br>1: 9600<br>2: 19200<br>3: 38400<br>4: 57600<br>5: 115200<br>6: 230400<br>7: 460800<br>8: 921600 | UINT8 | |
| Payload Length : 3 bytes | | | | | |

*1 supported only in RTK mode receivers

***RTK PRECISELY KINEMATIC BASE SERIAL PORT BAUD RATE – RTK Precisely Kinematic Base Serial Port Baud Rate of the GNSS receiver (ID: 0x6A, SID: 0x88)*** [*1]

This is a response message to **"*QUERY RTK PRECISELY KINEMATIC BASE SERIAL PORT BAUD RATE, ID: 0x6A, SID: 0x14",*** which provides the RTK precisely kinematic base serial port baud rate of the RTK GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 3 bytes.

Structure:

<0xA0,0xA1>< PL><6A><88>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 03 6A 88 06 E4 0D 0A

1 2 3

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 6A | | UINT8 | - |
| 2 | Sub ID | 88 | | UINT8 | |
| 3 | Baud Rate | 06 | 0: 4800<br>1: 9600<br>2: 19200<br>3: 38400<br>4: 57600<br>5: 115200<br>6: 230400<br>7: 460800<br>8: 921600 | UINT8 | |
| Payload Length : 3 bytes | | | | | |

*1 supported only in RTK mode receivers

***HEADING AND PITCH OFFSETS OF RTK ROVER MOVING BASE – Heading and pitch offsets of RTK rover moving base of the GNSS receiver (ID: 0x6A, SID: 0x89)*** *1

This is a response message to **"*QUERY HEADING AND PITCH OFFSETS of RTK ROVER MOVING BASE, ID: 0x6A, SID: 0x16",*** which provides heading and pitch offsets of RTK rover moving base of the GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 14 bytes.

Structure:

<0xA0,0xA1>< PL><6A><89>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 0E 6A 89 41 70 00 00 00 00 00 00 00 00 00 00 D2 0D 0A

1 2 3 4 5 6 7 8 9 10 11 12 13 14

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 6A | | UINT8 | - |
| 2 | Message Sub-ID | 89 | | UINT8 | |
| 3-6 | Heading Offset | 41700000 | Heading offset in degree | F32 | |
| 7-10 | Pitch Offset | 00000000 | Pitch offset in degree | F32 | |
| 11-14 | Reserved | 00000000 | Reserved | | |
| Payload Length : 14 bytes | | | | | |

*1 supported only in RTK mode receivers

***SET GALILEO EPHEMERIS – Set GALILEO ephemeris to GNSS receiver (ID: 0x6E, SID: 0x01)***

This is a request message which is issued from the host to GNSS receiver to set Galileo E1-B ephemeris data (open an ephemeris file) to GNSS receiver. The GNSS receiver should respond with an ACK when succeeded and should respond with an NACK when failed. The payload length is 85 bytes.

Structure:

<0xA0,0xA1>< PL><6E><01>< message body><CS><0x0D,0x0A>

Example:

| A0 | A1 | 00 | 55 | 6E | 01 | 00 | 02 | 01 | 04 | 50 | 5C | 83 | FE | 0B | 1B | F4 | 00 | F1 | 53 | 0A | A8 | 13 | 79 | 48 | 08 | 50 | 47 | D1 | D3 | C6 | 27 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 |

| AF | E6 | DF | 13 | 6C | 6B | 9B | 3F | B1 | 0C | 50 | FF | C5 | 2D | 1E | 28 | 09 | 47 | 10 | 8E | 16 | 2D | 0B | AD | 6B | 10 | 50 | 0B | FF | 08 | 00 | C1 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 |

| 72 | 00 | 09 | 58 | C5 | 00 | 00 | 00 | 00 | 14 | 00 | 00 | 00 | 00 | 01 | FE | 7F | 20 | 24 | C2 | B7 | 0C | 00 | 00 | 00 | 9D | 0D | 0A |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 | 81 | 82 | 83 | 84 | 85 | | | |

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 6E | | UINT8 | |
| 2 | Sub ID | 01 | | UINT8 | |
| 3-4 | SV id | 0001 | Satellite id | UINT16 | |
| 5 | Valid | 01 | 0: not valid<br>1: valid | UINT8 | |
| 6-9 | SubFrameData[0][0] | | Eph data subframe word type 1 | UINT32 | |
| 10-13 | SubFrameData[0][1] | | Eph data subframe word type 1 | UINT32 | |
| 14-17 | SubFrameData[0][2] | | Eph data subframe word type 1 | UINT32 | |
| 18-21 | SubFrameData[0][3] | | Eph data subframe word type 1 | UINT32 | |
| 22-25 | SubFrameData[1][0] | | Eph data subframe word type 2 | UINT32 | |
| 26-29 | SubFrameData[1][1] | | Eph data subframe word type 2 | UINT32 | |
| 30-33 | SubFrameData[1][2] | | Eph data subframe word type 2 | UINT32 | |
| 34-37 | SubFrameData[1][3] | | Eph data subframe word type 2 | UINT32 | |
| 38-41 | SubFrameData[2][0] | | Eph data subframe word type 3 | UINT32 | |
| 42~45 | SubFrameData[2][1] | | Eph data subframe word type 3 | UINT32 | |
| 46-49 | SubFrameData[2][2] | | Eph data subframe word type 3 | UINT32 | |
| 50~53 | SubFrameData[2][3] | | Eph data subframe word type 3 | UINT32 | |
| 54-57 | SubFrameData[3][0] | | Eph data subframe word type 4 | UINT32 | |
| 58-61 | SubFrameData[3][1] | | Eph data subframe word type 4 | UINT32 | |

| | | | | | |
|---|---|---|---|---|---|
| 62-65 | SubFrameData[3][2] | | Eph data subframe word type 4 | UINT32 | |
| 66-69 | SubFrameData[3][3] | | Eph data subframe word type 4 | UINT32 | |
| 70-73 | SubFrameData[4][0] | | Eph data subframe word type 5 | UINT32 | |
| 74-77 | SubFrameData[4][1] | | Eph data subframe word type 5 | UINT32 | |
| 78-81 | SubFrameData[4][2] | | Eph data subframe word type 5 | UINT32 | |
| 82-85 | SubFrameData[4][3] | | Eph data subframe word type 5 | UINT32 | |
| Payload Length : 85 bytes | | | | | |

***GET GALILEO EPHEMERIS – Get GALILEO ephemeris data used of GNSS receiver (ID: 0x6E, SID: 0x02)***

This is a request message which is issued from the host to GNSS receiver to retrieve GALILEO E1-B ephemeris data. The GNSS receiver should respond with an ACK along with information of ephemeris, **"*GALILEO EPHEMERIS DATA, ID 0x6E, SID 0x80***", when succeeded and should respond with an NACK when failed. The payload length is 3 bytes.

Structure:

<0xA0,0xA1>< PL><6E><02>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 03 6E 02 00 6C 0D 0A

1 2 3

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 6E | | UINT8 | |
| 2 | Sub ID | 02 | | UINT8 | |
| 3 | SV # | 00 | 0: means all SVs<br>1~36 : mean for the particular SV | UINT8 | |
| Payload Length : 3 bytes | | | | | |

***GALILEO EPHEMERIS – GALILEO ephemeris of GNSS receiver (ID: 0x6E, SID: 0x80)***

This is a response message to ***"GET GALILEO EPHEMERIS, ID 0x6E, SID 0x02",*** which provides the GALILEO E1-B ephemeris data of the GNSS receiver to host. The host will save the ephemeris data as an ephemeris file. This message is sent from the GNSS receiver to host. The payload length is 85 bytes.

Structure:

<0xA0,0xA1>< PL><6E><80>< message body><CS><0x0D,0x0A>

Example:

| a0 | a1 | 00 | 55 | 6E | 80 | 00 | 02 | 01 | 04 | 50 | 5C | 83 | FE | 0B | 1B | F4 | 00 | F1 | 53 | 0A | A8 | 13 | 79 | 48 | 08 | 50 | 47 | D1 | D3 | C6 | 27 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 |

| AF | E6 | DF | 13 | 6C | 6B | 9B | 3F | B1 | 0C | 50 | FF | C5 | 2D | 1E | 28 | 09 | 47 | 10 | 8E | 16 | 2D | 0B | AD | 6B | 10 | 50 | 0B | FF | 08 | 00 | C1 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 |

| 72 | 00 | 09 | 58 | C5 | 00 | 00 | 00 | 00 | 14 | 00 | 00 | 00 | 00 | 01 | FE | 7F | 20 | 24 | C2 | B7 | 0C | 00 | 00 | 00 | 1C | 0D | 0A |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 | 81 | 82 | 83 | 84 | 85 | | | |

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 6E | | UINT8 | |
| 2 | Sub ID | 80 | | UINT8 | |
| 3-4 | SV id | 0001 | Satellite id | UINT16 | |
| 5 | Valid | 01 | 0: not valid<br>1: valid | UINT8 | |
| 6-9 | SubFrameData[0][0] | | Eph data subframe word type 1 | UINT32 | |
| 10-13 | SubFrameData[0][1] | | Eph data subframe word type 1 | UINT32 | |
| 14-17 | SubFrameData[0][2] | | Eph data subframe word type 1 | UINT32 | |
| 18-21 | SubFrameData[0][3] | | Eph data subframe word type 1 | UINT32 | |
| 22-25 | SubFrameData[1][0] | | Eph data subframe word type 2 | UINT32 | |
| 26-29 | SubFrameData[1][1] | | Eph data subframe word type 2 | UINT32 | |
| 30-33 | SubFrameData[1][2] | | Eph data subframe word type 2 | UINT32 | |
| 34-37 | SubFrameData[1][3] | | Eph data subframe word type 2 | UINT32 | |
| 42~45 | SubFrameData[2][1] | | Eph data subframe word type 3 | UINT32 | |
| 46-49 | SubFrameData[2][2] | | Eph data subframe word type 3 | UINT32 | |
| 50~53 | SubFrameData[2][3] | | Eph data subframe word type 3 | UINT32 | |
| 54-57 | SubFrameData[3][0] | | Eph data subframe word type 4 | UINT32 | |
| 58-61 | SubFrameData[3][1] | | Eph data subframe word type 4 | UINT32 | |
| 62-65 | SubFrameData[3][2] | | Eph data subframe word type 4 | UINT32 | |

| | | | | | |
|---|---|---|---|---|---|
| 66-69 | SubFrameData[3][3] | | Eph data subframe word type 4 | UINT32 | |
| 70-73 | SubFrameData[4][0] | | Eph data subframe word type 5 | UINT32 | |
| 74-77 | SubFrameData[4][1] | | Eph data subframe word type 5 | UINT32 | |
| 78-81 | SubFrameData[4][2] | | Eph data subframe word type 5 | UINT32 | |
| 82-85 | SubFrameData[4][3] | | Eph data subframe word type 5 | UINT32 | |
| Payload Length : 85 bytes | | | | | |

***SET IRNSS EPHEMERIS – Set IRNSS ephemeris to GNSS receiver (ID: 0x6F, SID: 0x03)***

This is a request message which is issued from the host to GNSS receiver to set IRNSS ephemeris data (open an ephemeris file) to GNSS receiver. The GNSS receiver should respond with an ACK when succeeded and should respond with an NACK when failed. The payload length is 77 bytes.

Structure:
<0xA0,0xA1>< PL><6F><03>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 4D 6F 03 00 02 01 8B 00 00 00 30 21 D1 88 03 78 00 0B 01 FF 00 0E E5 A9 00 0D EF DA BC
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28

FF C7 7F EB 56 33 84 8F C6 90 00 00 00 8B 00 00 09 C5 D0 70 4C B0 1C 03 E3 B5 6F 2B A9 7E 89 40
29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60

BD 01 85 F5 53 BE B7 FF 1A 81 46 FD 1F A0 00 00 00 8E 0D 0A
61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 6F | | UINT8 | |
| 2 | Sub ID | 03 | | UINT8 | |
| 3-4 | SV id | 0002 | Satellite id | UINT16 | |
| 5 | Valid | 01 | 0: not valid<br>1: valid | UINT8 | |
| 6-9 | SubFrameData[0][0] | | Eph data subframe 1 | UINT32 | |
| 10-13 | SubFrameData[0][1] | | Eph data subframe 1 | UINT32 | |
| 14-17 | SubFrameData[0][2] | | Eph data subframe 1 | UINT32 | |
| 18-21 | SubFrameData[0][3] | | Eph data subframe 1 | UINT32 | |
| 22-25 | SubFrameData[0][4] | | Eph data subframe 1 | UINT32 | |
| 26-29 | SubFrameData[0][5] | | Eph data subframe 1 | UINT32 | |
| 30-33 | SubFrameData[0][6] | | Eph data subframe 1 | UINT32 | |
| 34-37 | SubFrameData[0][7] | | Eph data subframe 1 | UINT32 | |
| 38-41 | SubFrameData[0][8] | | Eph data subframe 1 | UINT32 | |
| 42~77 | SubFrameData[1][0~8] | | Eph data subframe 2 | UINT32 | |
| Payload Length : 77 bytes | | | | | |

***GET IRNSS EPHEMERIS – Get IRNSS ephemeris data used of GNSS receiver (ID: 0x6F, SID: 0x04)***

This is a request message which is issued from the host to GNSS receiver to retrieve NAVIC ephemeris data. The GNSS receiver should respond with an ACK along with information of ephemeris, **"*IRNSS EPHEMERIS DATA, ID 0x6F, SID 0x81***", when succeeded and should respond with an NACK when failed. The payload length is 3 bytes.

Structure:

<0xA0,0xA1>< PL><6F><04>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 03 6F 04 00 6B 0D 0A

1 2 3

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 6F | | UINT8 | |
| 2 | Sub ID | 04 | | UINT8 | |
| 3 | SV # | 00 | 0: means all SVs<br>1~14 : mean for the particular SV | UINT8 | |
| Payload Length : 3 bytes | | | | | |

***IRNSS EPHEMERIS – IRNSS ephemeris of GNSS receiver (ID: 0x6F, SID: 0x81)***

This is a response message to ***"GET IRNSS EPHEMERIS, ID 0x6F, SID 0x04",*** which provides the IRNSS ephemeris data of the GNSS receiver to host. The host will save the ephemeris data as an ephemeris file. This message is sent from the GNSS receiver to host. The payload length is 77 bytes.

Structure:

<0xA0,0xA1>< PL><6F><81>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 4D 6F 81 00 02 01 8B 00 00 00 30 21 D1 88 03 78 00 0B 01 FF 00 0E E5 A9 00 0D EF DA BC
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28

FF C7 7F EB 56 33 84 8F C6 90 00 00 00 8B 00 00 09 C5 D0 70 4C B0 1C 03 E3 B5 6F 2B A9 7E 89 40
29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60

BD 01 85 F5 53 BE B7 FF 1A 81 46 FD 1F A0 00 00 00 0C 0D 0A
61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 6F | | UINT8 | |
| 2 | Sub ID | 81 | | UINT8 | |
| 3-4 | SV id | 0002 | Satellite id | UINT16 | |
| 5 | Valid | 01 | 0: not valid<br>1: valid | UINT8 | |
| 6-9 | SubFrameData[0][0] | | Eph data subframe 1 | UINT32 | |
| 10-13 | SubFrameData[0][1] | | Eph data subframe 1 | UINT32 | |
| 14-17 | SubFrameData[0][2] | | Eph data subframe 1 | UINT32 | |
| 18-21 | SubFrameData[0][3] | | Eph data subframe 1 | UINT32 | |
| 22-25 | SubFrameData[0][4] | | Eph data subframe 1 | UINT32 | |
| 26-29 | SubFrameData[0][5] | | Eph data subframe 1 | UINT32 | |
| 30-33 | SubFrameData[0][6] | | Eph data subframe 1 | UINT32 | |
| 34-37 | SubFrameData[0][7] | | Eph data subframe 1 | UINT32 | |
| 38-41 | SubFrameData[0][8] | | Eph data subframe 1 | UINT32 | |
| 42~77 | SubFrameData[1][0~8] | | Eph data subframe 2 | UINT32 | |
| Payload Length : 77 bytes | | | | | |

# MESSAGES WITH Sub-ID AND Sub Sub-ID

***QUERY PX1172RH ROVER MOVING BASE SOFTWARE VERSION – Query the software version of rover moving base of PX1172RH GNSS receiver (ID: 0x7A, Sub ID: 0xE, Sub Sub-ID: 0x1)***

This is a request message which is issued from the host to PX1172RH GNSS receiver to query the software version of PX1172RH rover moving base receiver. The PX1172RH GNSS receiver should respond with an ACK along with PX1172RH rover moving base software version, "***SOFTWARE VERSION OF PX1172RH ROVER MOVING BASE, ID 0x7A, Sub ID: 0xE, Sub Sub-ID 0x80***", when succeeded and should respond with an NACK when failed. The payload length is 3 bytes.

Structure:
<0xA0,0xA1>< PL><7A><0E><01><CS><0x0D,0x0A>

Example:
A0 A1 00 03 7A 0E 01 75 0D 0A
1 2 3

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 7A | | UINT8 | |
| 2 | Sub ID | 0E | PX1172RH | UINT8 | |
| 3 | Sub Sub-ID | 01 | | UINT8 | |
| Payload Length : 3 bytes | | | | | |

***QUERY PX1172RH ROVER MOVING BASE SOFTWARE CRC – Query the software CRC of rover moving base of PX1172RH GNSS receiver (ID: 0x7A, Sub ID: 0xE, Sub Sub-ID: 0x2)***

This is a request message which is issued from the host to PX1172RH GNSS receiver to query the software CRC of PX1172RH rover moving base receiver. The PX1172RH GNSS receiver should respond with an ACK along with PX1172RH rover moving base software CRC, "***SOFTWARE CRC OF PX1172RH ROVER MOVING BASE, ID 0x7A, Sub ID: 0xE, Sub Sub-ID 0x81***", when succeeded and should respond with an NACK when failed. The payload length is 3 bytes.

Structure:

<0xA0,0xA1>< PL><7A><0E><02><CS><0x0D,0x0A>

Example:

A0 A1 00 03 7A 0E 02 76 0D 0A

1 2 3

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 7A | | UINT8 | |
| 2 | Sub ID | 0E | PX1172RH | UINT8 | |
| 3 | Sub Sub-ID | 02 | | UINT8 | |
| Payload Length : 3 bytes | | | | | |

***QUERY PX1172RH ROVER MOVING BASE POSITION UPDATE RATE – Query the position update rate of rover moving base of PX1172RH GNSS receiver (ID: 0x7A, Sub ID: 0xE, Sub Sub-ID: 0x3)***

This is a request message which is issued from the host to PX1172RH GNSS receiver to query the position update rate of rover moving base receiver. The PX1172RH GNSS receiver should respond with an ACK along with rover moving base position update rate, "***POSITION UPDATE RATE OF ROVER MOVING BASE OF PX1172RH, ID 0x7A, Sub ID: 0xE, Sub Sub-ID 0x82***", when succeeded and should respond with an NACK when failed. The payload length is 3 bytes.

Structure:
<0xA0,0xA1>< PL><7A><0E><03><CS><0x0D,0x0A>

Example:
A0 A1 00 03 7A 0E 03 77 0D 0A
1 2 3

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 7A | | UINT8 | |
| 2 | Sub ID | 0E | PX1172RH | UINT8 | |
| 3 | Sub Sub-ID | 03 | | UINT8 | |
| Payload Length : 3 bytes | | | | | |

***CONFIGURE PX1172RH ROVER MOVING BASE HEADING AND PITCH OFFSETS – Configure heading and pitch offsets of rover moving base of PX1172RH GNSS receiver (ID: 0x7A, Sub ID: 0xE, Sub Sub-D: 0x4)***

This is a request message which will set rover moving base receiver heading and pitch offsets of the PX1172RH GNSS receiver. This command is issued from the host to PX1172RH GNSS receiver and PX1172RH GNSS receiver should respond with an ACK or NACK. The payload length is 16 bytes.

Structure:
<0xA0,0xA1>< PL><7A><0E><04>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 10 7A 4E 04 41 70 00 00 00 00 00 00 00 00 00 01 00 0D 0A
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 7A | | UINT8 | |
| 2 | Sub ID | 0E | PX1172RH | UINT8 | |
| 3 | Sub Sub-ID | 04 | | UINT8 | |
| 4-7 | Heading offset | 41700000 | Heading offset in degree<br>-180.00 ~ +180.00, default value = 0.00 | F32 | |
| 8-11 | Pitch offset | 00000000 | Pitch offset in degree<br>-90.0 ~ +90.0, default value = 0.0 | F32 | |
| 12-15 | reserved | 00000000 | reserved | | |
| 16 | Attributes | 01 | 0: update to SRAM<br>1: update to both SRAM & FLASH | UINT8 | |
| Payload Length : 16 bytes | | | | | |

***QUERY PX1172RH ROVER MOVING BASE HEADING AND PITCH OFFSETS – Query heading and pitch offsets of rover moving base of PX1172RH GNSS receiver (ID: 0x7A, Sub ID: 0xE, Sub Sub-D: 0x5)***

This is a request message which is issued from the host to PX1172RH GNSS receiver to query the heading and pitch offsets of rover moving base receiver. The PX1172RH GNSS receiver should respond with an ACK along with heading and pitch offsets, ***"HEADING AND PITCH OFFSETS OF ROVER MOVING BASE OF PX1172RH, ID 0x7A, Sub ID: 0xE, Sub Sub-ID 0x83"***, when succeeded and should respond with an NACK when failed. The payload length is 3 bytes.

Structure:
<0xA0,0xA1>< PL><7A><0E><05>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 03 7A 0E 05 71 0D 0A
1 2 3

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 7A | | UINT8 | |
| 2 | Sub ID | 0E | PX1172RH | UINT8 | |
| 3 | Sub Sub-ID | 05 | | UINT8 | |
| Payload Length : 3 bytes | | | | | |

***SOFTWARE VERSION OF PX1172RH ROVER MOVING BASE – Software version of rover moving base receiver of PX1172RH GNSS receiver (ID: 0x7A, Sub ID: 0xE, Sub Sub-ID: 0x80)***

This is a response message to **"*QUERY PX1172RH ROVER MOVING BASE SOFTWARE VERSION, ID 0x7A, Sub ID: 0xE, Sub Sub-ID 0x1",*** which provides the software version of rover moving base receiver of the PX1172RH GNSS receiver. This message is sent from the PX1172RH GNSS receiver to host. The payload length is 16 bytes.

Structure:

<0xA0,0xA1>< PL><7A><0E><80>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 10 7A 0E 80 01 00 03 00 01 00 0E 07 21 00 15 04 08 C6 0D 0A

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 7A | | UINT8 | |
| 2 | Sub ID | 0E | PX1172RH | UINT8 | |
| 3 | Sub Sub-ID | 80 | | UINT8 | |
| 4 | Software Type | 01 | 0: Reserved<br>1: System code | UINT8 | |
| 5-8 | Kernel Version | 00030001 | X1.Y1.Z1 = SkyTraq Kernel Version<br>Ex. X1=03, Y1=00, Z1=01 (3.0.1) | UINT32 | |
| 9-12 | ODM version | 000E0721 | X1.Y1.Z1 = SkyTraq Version<br>Ex. X1=0E, Y1=07, Z1=21 (14.7.33) | UINT32 | |
| 13-16 | Revision | 00150408 | YYMMDD = SkyTraq Revision<br>Ex. YY=15, MM=04, DD=08 (210408) | UINT32 | |
| Payload Length : 16 bytes | | | | | |

***SOFTWARE CRC OF PX1172RH ROVER MOVING BASE – Software CRC of rover moving base receiver of PX1172RH GNSS receiver (ID: 0x7A, Sub ID: 0xE, Sub Sub-ID: 0x81)***

This is a response message to **"*QUERY PX1172RH ROVER MOVING BASE SOFTWARE CRC, ID 0x7A, Sub ID: 0xE, Sub Sub-ID 0x2",*** which provides the software CRC of rover moving base receiver of the PX1172RH GNSS receiver. This message is sent from the PX1172RH GNSS receiver to host. The payload length is 6 bytes.

Structure:

<0xA0,0xA1>< PL><7A><0E><81>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 06 7A 0E 81 01 A1 03 56 0D 0A

1 2 3 4 5 6

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 7A | | UINT8 | |
| 2 | Sub ID | 0E | PX1172RH | UINT8 | |
| 3 | Sub Sub-ID | 81 | | UINT8 | |
| 4 | Software Type | 01 | 0: Reserved<br>1: System code | UINT8 | |
| 5-6 | CRC | A103 | CRC value | UINT16 | |
| Payload Length : 6 bytes | | | | | |

***POSITION UPDATE RATE OF PX1172RH ROVER MOVING BASE – Position update rate of rover moving base receiver of PX1172RH GNSS receiver (ID: 0x7A, Sub ID: 0xE, Sub Sub-ID: 0x82)***

This is a response message to **"*QUERY PX1172RH ROVER MOVING BASE POSITION UPDATE RATE, ID 0x7A, Sub ID: 0xE, Sub Sub-ID 0x3",*** which provides the position update rate of rover moving base receiver of the PX1172RH GNSS receiver. This message is sent from the PX1172RH GNSS receiver to host. The payload length is 4 bytes.

Structure:

<0xA0,0xA1>< PL><7A><0E><82>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 04 7A 0E 82 01 F7 0D 0A

1 2 3 4

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 7A | | UINT8 | |
| 2 | Sub ID | 0E | PX1172RH | UINT8 | |
| 3 | Sub Sub-ID | 82 | | UINT8 | |
| 4 | Position Update Rate | 01 | 01: 1Hz<br>Value with 1, 2, 4, 5, or 8 Hz | UINT8 | |
| Payload Length : 4 bytes | | | | | |

***HEADING AND PITCH OFFSETS OF PX1172RH ROVER MOVING BASE – Heading and pitch offsets of rover moving base receiver of PX1172RH GNSS receiver (ID: 0x7A, Sub ID: 0xE, Sib Sib-ID: 0x83)***

This is a response message to **"*QUERY HEADING AND PITCH OFFSETS OF PX1172RH ROVER MOVING BASE, , ID 0x7A, Sub ID: 0xE, Sub Sub-ID 0x5",*** which provides heading and pitch offsets of the rover moving base receiver of the PX1172RH GNSS receiver. This message is sent from the PX1172RH GNSS receiver to host. The payload length is 15 bytes.

Structure:
<0xA0,0xA1>< PL><7A><0E><83>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 0F 7A 0E 83 41 70 00 00 00 00 00 00 00 00 00 00 C6 0D 0A
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 7A | | UINT8 | |
| 2 | Sub ID | 0E | PX1172RH | UINT8 | |
| 3 | Sub Sub-ID | 83 | | UINT8 | |
| 4-7 | Heading Offset | 41700000 | Heading offset in degree | F32 | |
| 8-11 | Pitch Offset | 00000000 | Pitch offset in degree | F32 | |
| 12-15 | Reserved | 00000000 | Reserved | | |
| Payload Length : 15 bytes | | | | | |

# OUTPUT MESSAGES

***SOFTWARE VERSION – Software version of the GNSS receiver (0x80)***

This is a response message to ***"QUERY SOFTWARE VERSION, ID: 0x2"*** which provides the software version of the GNSS receiver. This message is sent from the GNSS receiver to host. The example below output the SkyTraq software version as 01.01.01-01.03.14-07.01.18 on System image. The payload length is 14 bytes.

Structure:

<0xA0,0xA1>< PL><80>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 0E 80 01 00 01 01 01 00 01 03 0E 00 07 01 12 98 0D 0A

1 2 3 4 5 6 7 8 9 10 11 12 13 14

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 80 | | UINT8 | |
| 2 | Software Type | 00 | 0: Reserved<br>1: System code | UINT8 | |
| 3-6 | Kernel Version | 00010101 | X1.Y1.Z1 = SkyTraq Kernel Version<br>Ex. X1=01, Y1=00, Z1=01 (1.0.1) | UINT32 | |
| 7-10 | ODM version | 0001030E | X1.Y1.Z1 = SkyTraq Version<br>Ex. X1=01, Y1=03, Z1=0E (1.3.14) | UINT32 | |
| 11-14 | Revision | 00070112 | YYMMDD = SkyTraq Revision<br>Ex. YY=07, MM=01, DD=12 (070118) | UINT32 | |
| Payload Length : 14 bytes | | | | | |

***SOFTWARE CRC – Software CRC of the GNSS receiver (0x81)***

This is a response message to ***"QUERY SOFTWARE CRC, ID: 0x3"*** which provides the software CRC of the GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 4 bytes.

Structure:

<0xA0,0xA1>< PL><81>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 04 81 01 98 76 6E 0D 0A

1 2 3 4

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 81 | | UINT8 | |
| 2 | Software Type | 00 | 0: Reserved<br>1: System code | UINT8 | |
| 3-4 | CRC | 9876 | CRC value | UINT16 | |
| Payload Length : 4 bytes | | | | | |

***ACK – Acknowledgement to a Request Message (0x83)***

This is a response message which is an acknowledgement to a request message. The payload length is 2 bytes

Structure:

<0xA0,0xA1>< PL><83>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 02 83 02 81 0D 0A

1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 83 | | UINT8 | |
| 2 | ACK ID*1* | 02 | Message ID of the request message | UINT8 | |
| Payload Length : 2 bytes | | | | | |

*1: ACK ID may further consist of message ID and message sub-ID which will become 3 bytes of ACK message.

***NACK – Response to an unsuccessful request message (0x84)***

This is a response message which is a response to an unsuccessful request message. This is used to notify the Host that the request message has been rejected. The payload length is 2 bytes

Structure:

<0xA0,0xA1>< PL><84>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 02 84 01 85 0D 0A

1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 84 | | UINT8 | |
| 2 | NACK ID*1* | 01 | Message ID of the request message | UINT8 | |
| Payload Length : 2 bytes | | | | | |

*1: NACK ID may further consist of message ID and message sub-ID which will become 3 bytes of NACK message.

*2: A NACK may be caused by a request message with invalid length, invalid checksum, wrong input values, firmware message format changed or firmware not supported.

***POSITON UPDATE RATE – Position Update rate of the GNSS system (0x86)***

This is a response message to ***"QUERY POSITION UPDATE RATE, ID: 0x10"*** which provides the position update rate of the GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 2 bytes.

Structure:

<0xA0,0xA1>< PL><86>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 02 86 01 87 0D 0A

1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 86 | | UINT8 | |
| 2 | Update Rate | 01 | 01: 1Hz | UINT8 | |
| Payload Length : 2 bytes | | | | | |

***GLONASS EPHEMERIS DATA – GLONASS ephemeris data of the GLONASS/GPS receiver (0x90)***

This is a response message to ***"GET GLONASS EPHEMERIS, ID: 0x5B"*** which provides the GLONASS ephemeris data of the receiver to the host. The host may save the ephemeris data as an ephemeris file. This message is sent from the receiver to host. The payload length is 43 bytes.

Structure:
<0xA0,0xA1>< PL><90>< message body><CS><0x0D,0x0A>

Example:

| A0 | A1 | 00 | 2B | 90 | 01 | 01 | 01 | 07 | 43 | 0F | AC | 06 | 89 | A2 | 01 | 9A | 02 | 17 | 60 | 28 | 75 | 47 | 01 | 16 | FE | B5 | 03 | 80 | 06 | 9C | CB |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 |

| CC | 92 | 6A | C0 | 42 | 04 | 09 | 94 | 79 | 20 | 00 | 00 | 20 | 11 | 85 | E3 | 0D | 0A |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | | | |

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 90 | | UINT8 | |
| 2 | Slot number | 01 | GLONASS SV slot number | UINT8 | |
| 3 | K number | 01 | GLONASS SV frequency number (-7 ~ +6) | SINT8 | |
| 4 | glo_eph_data0_byte0 | 01 | Stuffing zeros and bit 85 - bit 81 (LSB) of string 1 | UINT8 | |
| 5 | glo_eph_data0_byte1 | 07 | bit 80 (MSB)- bit 73 (LSB) of string 1 | UINT8 | |
| 6 | glo_eph_data0_byte2 | 43 | bit 72 (MSB)- bit 65 (LSB) of string 1 | UINT8 | |
| 7 | glo_eph_data0_byte3 | 0F | bit 64 (MSB)- bit 57 (LSB) of string 1 | UINT8 | |
| 8 | glo_eph_data0_byte4 | AC | bit 56 (MSB)- bit 49 (LSB) of string 1 | UINT8 | |
| 9 | glo_eph_data0_byte5 | 06 | bit 48 (MSB)- bit 41 (LSB) of string 1 | UINT8 | |
| 10 | glo_eph_data0_byte6 | 89 | bit 40 (MSB)- bit 33 (LSB) of string 1 | UINT8 | |
| 11 | glo_eph_data0_byte7 | A2 | bit 32 (MSB)- bit 25 (LSB) of string 1 | UINT8 | |
| 12 | glo_eph_data0_byte8 | 01 | bit 24 (MSB)- bit 17 (LSB) of string 1 | UINT8 | |
| 13 | glo_eph_data0_byte9 | 9A | bit 16 (MSB)- bit 09 (LSB) of string 1 | UINT8 | |
| 14 | glo_eph_data1_byte0 | 02 | Stuffing zeros and bit 85 - bit 81 (LSB) of string 2 | UINT8 | |
| 15 | glo_eph_data1_byte1 | 17 | bit 80 (MSB)- bit 73 (LSB) of string 2 | UINT8 | |
| 16 | glo_eph_data1_byte2 | 60 | bit 72 (MSB)- bit 65 (LSB) of string 2 | UINT8 | |
| 17 | glo_eph_data1_byte3 | 28 | bit 64 (MSB)- bit 57 (LSB) of string 2 | UINT8 | |
| 18 | glo_eph_data1_byte4 | 75 | bit 56 (MSB)- bit 49 (LSB) of string 2 | UINT8 | |
| 19 | glo_eph_data1_byte5 | 47 | bit 48 (MSB)- bit 41 (LSB) of string 2 | UINT8 | |

| | | | | | |
|---|---|---|---|---|---|
| 20 | glo_eph_data1_byte6 | 01 | bit 40 (MSB)- bit 33 (LSB) of string 2 | UINT8 | |
| 21 | glo_eph_data1_byte7 | 16 | bit 32 (MSB)- bit 25 (LSB) of string 2 | UINT8 | |
| 22 | glo_eph_data1_byte8 | FE | bit 24 (MSB)- bit 17 (LSB) of string 2 | UINT8 | |
| 23 | glo_eph_data1_byte9 | B5 | bit 16 (MSB)- bit 09 (LSB) of string 2 | UINT8 | |
| 24-33 | glo_eph_data2_byte0 - glo_eph_data2_byte9 | | Stuffing-zeros and bit 85 - bit 09 of string 3 | | |
| 34-43 | glo_eph_data3_byte0 – glo_eph_data3_byte9 | | Stuffing-zeros and bit 85 - bit 09 of string 4 | | |
| Payload Length : 43 bytes | | | | | |

***GLONASS TIME CORRECTION PARAMETERS – GLONASS time correction parameters (0x92)***

This is a response message to ***"GET GLONASS TIME CORRECTION, ID: 0x5F"*** which provides the GLONASS time correction data of the receiver to the host. The host may save the data as a file. This message is sent from the receiver to host. The payload length is 9 bytes.

Structure:

<0xA0,0xA1>< PL><92>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 09 92 FF FF FF BF 00 00 00 14 C6 0D 0A

1 2 3 4 5 6 7 8 9

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 92 | | UINT8 | |
| 2-5 | $\tau_c$ | FFFFBF | GLONASS time scale correction to UTC(SU) time | SINT32 | $2^{-31}$ sec |
| 6-9 | $\tau_{GPS}$ | 00000014 | Correction to GPS time relative to GLONASS time | SINT32 | $2^{-30}$ sec |
| Payload Length : 9 bytes | | | | | |

***GNSS NMEA TALKER ID – NMEA talker ID of GNSS receiver (0x93)***

This is a response message to "***QUERY NMEA TALKER ID, ID: 0x4F"*** which provides the type of NMEA talker id of GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 2 bytes.

Structure:

<0xA0,0xA1>< PL><93>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 02 93 01 92 0D 0A

1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | 93 | | UINT8 | |
| 2 | Talker ID type | 01 | 0: GP mode<br>1: GN mode<br>2: Auto mode*1: according to NMEA 4.11 to combine GNSS system solution to output GN, GP, GL, GA, GB or GI appropriately. | UINT8 | |
| Payload Length : 2 bytes | | | | | |

*1 supported only in NMEA version 4.11

***NAVIGATION DATA MESSAGE – Message of user navigation data in binary format (0xA8)***

This is a response message which provides data of user navigation solution in binary format. This message is sent from the GNSS receiver to host. The payload length is 59 bytes

Structure:
<0xA0,0xA1>< PL><A8>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 3B A8 02 08 06 04 02 32 18 18 0E C5 E1 99 48 20 78 ED 00 00 2E 3B 00 00 26 93 00 93 00 93
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29
00 93 00 93 00 93 EE 35 4D 30 1D 99 AA 37 0F D7 0B 74 00 00 00 00 00 00 00 00 00 00 00 00 F5 0D 0A
30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | A8 | | UINT8 | |
| 2 | Fix Mode | 02 | Quality of fix<br>0: no fix<br>1: 2D<br>2: 3D<br>3: 3D+DGNSS | UINT8 | |
| 3 | Number of SV in fix | 08 | Number of SV in fix | UINT8 | |
| 4-5 | GNSS Week | 0604 | GNSS week number | UINT16 | |
| 6-9 | TOW | 02321818 | GNSS time of week<br>Scaling 0.01 | UINT32 | 1/100 sec |
| 10-13 | Latitude | 0EC5E199 | > 0: North Hemisphere<br>< 0: South Hemisphere<br>Scaling 1e-7 | SINT32 | 1/1e-7 degree |
| 14-17 | Longitude | 482078ED | > 0: East Hemisphere<br>< 0: West Hemisphere | SINT32 | 1/1e-7 degree |
| 18-21 | ellipsoid altitude, | 00002E3B | height above ellipsoid<br>Scaling 0.01 | SINT32 | 1/100 meter |
| 22-25 | mean sea level altitude | 00002693 | height above mean sea level<br>Scaling 0.01 | SINT32 | 1/100 meter |
| 26-27 | GDOP | 0093 | Geometric dilution of precision<br>Scaling 0.01 | UINT16 | 1/100 |
| 28-29 | PDOP | 0093 | Position dilution of precision<br>Scaling 0.01 | UINT16 | 1/100 |
| 30-31 | HDOP | 0093 | Horizontal dilution of precision | UINT16 | 1/100 |

| | | | | | |
|---|---|---|---|---|---|
| | | | Scaling 0.01 | | |
| 32-33 | VDOP | 0093 | Vertical dilution of precision<br>Scaling 0.01 | UINT16 | 1/100 |
| 34-35 | TDOP | 0093 | Time dilution of precision<br>Scaling 0.01 | UINT16 | 1/100 |
| 36-39 | ECEF-X | EE354D30 | ECEF X coordinate<br>Scaling 0.01 | SINT32 | 1/100<br>meter |
| 40-43 | ECEF-Y | 1D99AA37 | ECEF Y coordinate<br>Scaling 0.01 | SINT32 | 1/100<br>meter |
| 44-47 | ECEF-Z | 0FD70B74 | ECEF Z coordinate<br>Scaling 0.01 | SINT32 | 1/100<br>meter |
| 48-51 | ECEF-VX | 00000000 | ECEF X Veolcity<br>Scaling 0.01 | SINT32 | 1/100<br>m/s |
| 52-55 | ECEF-VY | 00000000 | ECEF Y Veolcity<br>Scaling 0.01 | SINT32 | 1/100<br>m/s |
| 56-59 | ECEF-VZ | 00000000 | ECEF Z Veolcity<br>Scaling 0.01 | SINT32 | 1/100<br>m/s |
| Payload Length : 59 bytes | | | | | |

***GNSS DOP MASK – DOP Mask used by the GNSS receiver (0xAF)***

This is a response message to ***"QUERY DOP MASK, ID: 0x2E"*** which provides the information of DOP masks of the GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 8 bytes.

Structure:

<0xA0,0xA1>< PL><AF>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 08 AF 01 00 32 00 32 00 32 9C 0D 0A

1 2 3 4 5 6 7 8

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | AF | | UINT8 | |
| 2 | DOP Mode Select | 01 | 00 : Disable<br>01 : Auto mode, PDOP when 3-D fix and HDOP when 2-D fix<br>02 : PDOP only<br>03 : HDOP only<br>04 : GDOP only | UINT8 | |
| 3-4 | PDOP Value | 0032 | Valid values between 0.5~30<br>Valid output value 5 ~ 300 | UINT16 | 1/10 |
| 5-6 | HDOP Value | 0032 | Valid values between 0.5~30<br>Valid output value 5 ~ 300 | UINT16 | 1/10 |
| 7-8 | GDOP Value | 0032 | Valid values between 0.5~30<br>Valid output value 5 ~ 300 | UINT16 | 1/10 |
| Payload Length : 8 bytes | | | | | |

***GNSS ELEVATION AND CNR MASK – Elevation and CNR mask used by the GNSS receiver (0xB0)***

This is a response message to "***QUERY ELEVATION AND CNR MASK, ID: 0x2F"*** which provides the information of elevation and CNR masks of the GNSS receiver. When enabled, satellite with elevation angle above the elevation mask value and tracked signal with CNR above the CNR mask value will be used for position fix. This message is sent from the GNSS receiver to host. The payload length is 4 bytes.

Structure:
<0xA0,0xA1>< PL><B0>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 04 B0 01 05 00 B4 0D 0A
1 2 3 4

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | B0 | | UINT8 | |
| 2 | Elevation and CNR Mask Select | 01 | 00 : Disable<br>01 : Elevation and CNR both<br>02 : Elevation only<br>03 : CNR only | UINT8 | |
| 3 | Elevation Mask | 05 | Value of elevation mask | UINT8 | degree |
| 4 | CNR Mask | 00 | Value of CNR mask | UINT8 | dB |
| Payload Length : 4 bytes | | | | | |

***GPS EPHEMERIS DATA – GPS ephemeris data of the GNSS receiver (0xB1)***

This is a response message to ***"GET GPS EPHEMERIS, ID: 0x30"*** which provides the GPS ephemeris data of the GNSS receiver to host. The host will save the ephemeris data as an ephemeris file. This message is sent from the GNSS receiver to host. The payload length is 87 bytes.

Structure:

<0xA0,0xA1>< PL><B1>< message body><CS><0x0D,0x0A>

Example:

| A0 | A1 | 00 | 57 | B1 | 00 | 02 | 00 | 77 | 88 | 04 | 61 | 10 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | DB | DF | 59 | A6 | 00 | 00 | 1E |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 |

| 0A | 47 | 7C | 00 | 77 | 88 | 88 | DF | FD | 2E | 35 | A9 | CD | B0 | F0 | 9F | FD | A7 | 04 | 8E | CC | A8 | 10 | 2C | A1 | 0E | 22 | 31 | 59 | A6 | 74 | 00 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 |

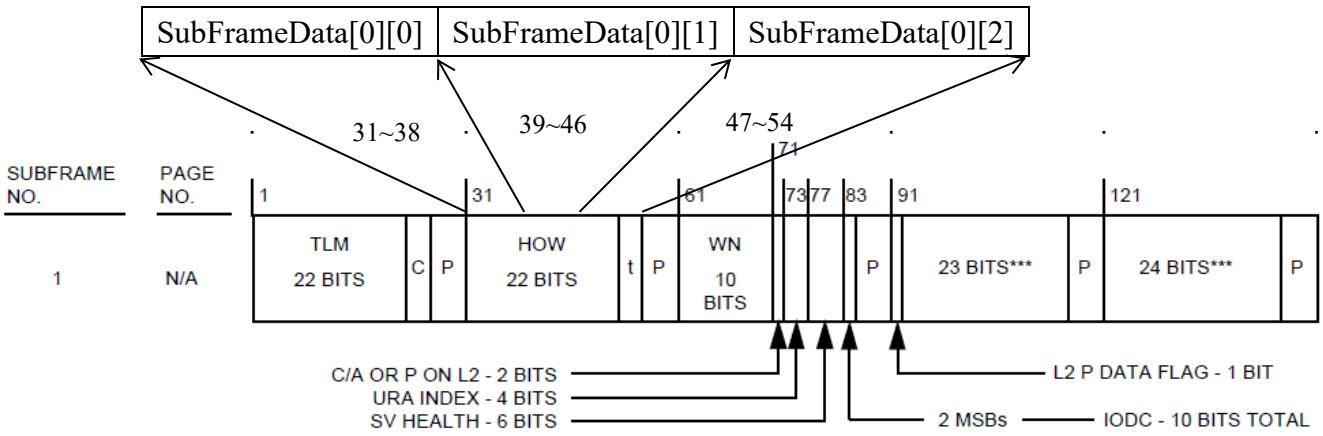
| 77 | 89 | 0C | FF | A3 | 59 | 86 | C7 | 77 | FF | F8 | 26 | 97 | E3 | B9 | 1C | 60 | 59 | C3 | 07 | 44 | FF | A6 | 37 | DF | F0 | B0 | 5E | 0D | 0A |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 | 81 | 82 | 83 | 84 | 85 | 86 | 87 | | | |

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | B1 | | UINT8 | |
| 2-3 | SV ID | 0x1 | Satellite id | UINT16 | |
| 4 | Reserved | 00 | Reserved | UINT8 | |
| 5 | SubFrameData[0][0] | 00 | 30~23 bits of eph data word 2 of subframe 1 | UINT8 | |
| 6 | SubFrameData[0][1] | 00 | 22~15 bits of eph data word 2 of subframe 1 | UINT8 | |
| 7 | SubFrameData[0][2] | 00 | 14~7 bits of eph data word 2 of subframe 1 | UINT8 | |
| 8 | SubFrameData[0][3] | 00 | 30~23 bits of eph data word 3 of subframe 1 | UINT8 | |
| 9 | SubFrameData[0][4] | 00 | 22~15 bits of eph data word 3 of subframe 1 | UINT8 | |
| 10 | SubFrameData[0][5] | 00 | 14~7 bits of eph data word 3 of subframe 1 | UINT8 | |
| 11 | SubFrameData[0][6] | 00 | 30~23 bits of eph data word 4 of subframe 1 | UINT8 | |
| 12 | SubFrameData[0][7] | 00 | 22~15 bits of eph data word 4 of subframe 1 | UINT8 | |
| 13 | SubFrameData[0][8] | 00 | 14~7 bits of eph data word 4 of | UINT8 | |

| | | | | | |
|---|---|---|---|---|---|
| | | | subframe 1 | | |
| 14 | SubFrameData[0][9] | 00 | 30~23 bits of eph data word 5 of subframe 1 | UINT8 | |
| 15 | SubFrameData[0][10] | 00 | 22~15 bits of eph data word 5 of subframe 1 | UINT8 | |
| 16 | SubFrameData[0][11] | 00 | 14~7 bits of eph data word 5 of subframe 1 | UINT8 | |
| 17 | SubFrameData[0][12] | 00 | 30~23 bits of eph data word 6 of subframe 1 | UINT8 | |
| 18 | SubFrameData[0][13] | 00 | 22~15 bits of eph data word 6 of subframe 1 | UINT8 | |
| 19 | SubFrameData[0][14] | 00 | 14~7 bits of eph data word 6 of subframe 1 | UINT8 | |
| 20 | SubFrameData[0][15] | 00 | 30~23 bits of eph data word 7 of subframe 1 | UINT8 | |
| 21 | SubFrameData[0][16] | 00 | 22~15 bits of eph data word 7 of subframe 1 | UINT8 | |
| 22 | SubFrameData[0][17] | 00 | 14~7 bits of eph data word 7 of subframe 1 | UINT8 | |
| 23 | SubFrameData[0][18] | 00 | 30~23 bits of eph data word 8 of subframe 1 | UINT8 | |
| 24 | SubFrameData[0][19] | 00 | 22~15 bits of eph data word 8 of subframe 1 | UINT8 | |
| 25 | SubFrameData[0][20] | 00 | 14~7 bits of eph data word 8 of subframe 1 | UINT8 | |
| 26 | SubFrameData[0][21] | 00 | 30~23 bits of eph data word 9 of subframe 1 | UINT8 | |
| 27 | SubFrameData[0][22] | 00 | 22~15 bits of eph data word 9 of subframe 1 | UINT8 | |
| 28 | SubFrameData[0][23] | 00 | 14~7 bits of eph data word 9 of subframe 1 | UINT8 | |
| 29 | SubFrameData[0][24] | 00 | 30~23 bits of eph data word 10 of subframe 1 | UINT8 | |
| 30 | SubFrameData[0][25] | 00 | 22~15 bits of eph data word 10 of subframe 1 | UINT8 | |
| 31 | SubFrameData[0][26] | 00 | 14~7 bits of eph data word 10 of subframe 1 | UINT8 | |
| 32 | Reserved | 00 | Reserved | UINT8 | |
| 33~59 | SubFrameData[1][0~26] | 00 | Eph data subframe 2, same as field 5-31 | UINT8 | |

| | | | | | |
|---|---|---|---|---|---|
| 60 | Reserved | 00 | Reserved | UINT8 | |
| 61-87 | SubFrameData[2][0~26] | 00 | Eph data subframe 3, same as field 5-31 | UINT8 | |
| Payload Length : 87 bytes | | | | | |

Each sub-frame data consists of word 2 to word 10. Each word is 24 bits without parity bits. For example, sub-frame data SubFrameData[0][0], SubFrameData[0][1], SubFrameData[0][2] are from sub-frame NO1 word 2 of picture below.



***GNSS POSITON PINNING STATUS – Position pinning status of the GNSS receiver (0xB4)***

This is a response message to ***"QUERY POSITION PINNING, ID 0x3A"*** which provides the position pinning status and position pinning parameters of the GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 12 bytes.

Structure:

<0xA0,0xA1>< PL><B4>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 0C B4 02 00 02 00 0A 00 08 00 2D 01 F4 6E 0D 0A

1 2 3 4 5 6 7 8 9 10 11 12

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | B4 | | UINT8 | |
| 2 | status | 02 | 0: default<br>1: enable<br>2: disable | UINT8 | |
| 3-4 | Pinning speed | 0002 | Be effective when status is enable | UINT16 | Km/Hr |
| 5-6 | Pinning cnt | 000A | Be effective when status is enable | UINT16 | second |
| 7-8 | Unpinning speed | 0008 | Be effective when status is enable | UINT16 | Km/Hr |
| 9-10 | Unpinning cnt | 002D | Be effective when status is enable | UINT16 | second |
| 11-12 | Unpinning distance | 01F4 | Be effective when status is enable | UINT16 | meter |
| Payload Length : 12 bytes | | | | | |

***GNSS POWER MODE STATUS – Power mode status of the GNSS receiver (0xB9)***

This is a response message to ***"QUERY POWER MODE, ID: 0x15"*** which provides the power mode status of the GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 2 bytes.

Structure:

<0xA0,0xA1>< PL><B9>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 02 B9 00 B9 0D 0A

1 2

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | B9 | | UINT8 | |
| 2 | Mode | 00 | 00 = Normal (disable power save)<br>01 = Power Save (enable power save) | UINT8 | |
| Payload Length : 2 bytes | | | | | |

***GNSS 1PPS CABLE DELAY – 1PPS cable delay of the GNSS receiver (0xBB)***

This is a response message to ***"QUERY 1PPS CABLE DELAY, ID: 0x46"*** which provides the 1PPS cable delay of the GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 5 bytes.

Structure:

<0xA0,0xA1>< PL><BB>< message body><CS><0x0D,0x0A>

Example:

A0 A1 00 05 BB 00 00 00 00 BB 0D 0A

1 2 3 4 5

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | BB | | UINT8 | - |
| 2-5 | Cable Delay | 00000000 | Cable delay of 1PPS timing mode<br>Return value is in unit of 1/100 ns. Ex. If 100 is the cable delay, it's of value 1ns. | SINT32 | 1/100 ns |
| Payload Length : 5 bytes | | | | | |

***GNSS 1PPS TIMING – 1PPS timing information of the GNSS receiver (0xC2)[*1]***

This is a response message to **"*QUERY 1PPS TIMING, ID: 0x44"*** which provides the information of 1PPS timing of the GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 35 bytes.

Structure:
<0xA0,0xA1>< PL><C2>< message body><CS><0x0D,0x0A>

Example:

| A0 | A1 | 00 | 23 | C2 | 00 | 00 | 00 | 07 | D0 | 00 | 00 | 00 | 1E | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 |

| 00 | 00 | 00 | 01 | 00 | 00 | 07 | D0 | DD | 0D | 0A |
|---|---|---|---|---|---|---|---|---|---|---|
| 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | | | |

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | C2 | | UINT8 | - |
| 2 | Saved Timing Mode | 00 | 00 = Timing PVT Mode<br>01 = Timing Survey Mode<br>02 = Timing Static Mode<br>Value saved in SRAM/Flash by request command, QUERY 1PPS TIMING, id 0x44 with attribute 1 or 2 | UINT8 | |
| 3-6 | Saved Survey Length | 000007D0 | Survey length used when in "Saved Timing Survey Mode".<br>Value saved in SRAM/Flash by request command, QUERY 1PPS TIMING, id 0x44 with attribute 1 or 2 | UINT32 | |
| 7-10 | Standard deviation | 0000001E | Standard Deviation when in Timing Survey Mode not used when in other mode.<br>Valid values between 3~100 | UINT32 | |
| 11-18 | Saved Latitude | 0000000000000000 | Latitude in double in Timing Static Mode | DPFP | |
| 19-26- | Saved Longitude | 0000000000000000 | Longitude in double in Timing Static Mode | DPFP | |
| 27-30 | Saved Altitude | 00000000 | Altitude in float in Timing Static Mode | SPFP | |
| 31 | Run-time Timing Mode | 00 | 00 = Timing Normal Mode<br>01 = Timing Survey Mode<br>02 = Timing Static Mode | UINT8 | |

| | | | | | |
|---|---|---|---|---|---|
| | | | Value currently used and not saved in SRAM/Flash by QUERY 1PPS TIMING, id 0x44 with attribute 0 | | |
| 32-35 | Run-time Survey Length | 000007D0 | Survey length used when in “Run-time Timing Survey Mode”.<br>Value currently used and not saved in SRAM/Flash by QUERY 1PPS TIMING, id 0x44 with attribute 0 | UINT32 | |
| Payload Length : 35 bytes | | | | | |

*1: supported only in timing mode receivers.

***GNSS 1PPS OUTPUT MODE – 1PPS output mode of the GNSS receiver (0xC3)*[1]**

This is a response message to **"*QUERY 1PPS OUTPUT MODE, ID: 0x56"*** which provides the information of 1PPS output mode of the GNSS receiver. This message is sent from the GNSS receiver to host. The payload length is 3 bytes.

Structure:
<0xA0,0xA1>< PL><C3>< message body><CS><0x0D,0x0A>

Example:
A0 A1 00 03 C3 00 00 C3 0D 0A
1 2 3

| Field | Name | Example(hex) | Description | Type | Unit |
|---|---|---|---|---|---|
| 1 | Message ID | C3 | | UINT8 | - |
| 2 | Output Mode | 00 | 00 = Reserved<br>01 = Output if GNSS time is available<br>02 = Output always and align to GNSS time automatically | UINT8 | |
| 3 | Align Source | 00 | 00 = Align to GNSS<br>01 = Align to UTC | UINT8 | |
| Payload Length : 3 bytes | | | | | |

*1: supported only in timing mode receivers

.

## A. Ellipsoid List

| Ellipsoid Index | Ellipsoid | Semi-major axis (a) | Inversed Flattening (1/f) |
|---|---|---|---|
| 1 | Airy 1830 | 6377563.396 | 299.3249646 |
| 2 | Modified Airy | 6377340.189 | 299.3249646 |
| 3 | Australian National | 6378160 | 298.25 |
| 4 | Bessel 1841 (Namibia) | 6377483.865 | 299.1528128 |
| 5 | Bessel 1841 | 6377397.155 | 299.1528128 |
| 6 | Clarke 1866 | 6378206.4 | 294.9786982 |
| 7 | Clarke 1880 | 6378249.145 | 293.465 |
| 8 | Everest (India 1830) | 6377276.345 | 300.8017 |
| 9 | Everest (Sabah Sarawak) | 6377298.556 | 300.8017 |
| 10 | Everest (India 1956) | 6377301.243 | 300.8017 |
| 11 | Everest (Malaysia 1969) | 6377295.664 | 300.8017 |
| 12 | Everest (Malay. & Sing) | 6377304.063 | 300.8017 |
| 13 | Everest (Pakistan) | 6377309.613 | 300.8017 |
| 14 | Modified Fischer 1960 | 6378155 | 298.3 |
| 15 | Helmert 1906 | 6378200 | 298.3 |
| 16 | Hough 1960 | 6378270 | 297 |
| 17 | Indonesian 1974 | 6378160 | 298.247 |
| 18 | International 1924 | 6378388 | 297 |
| 19 | Krassovsky 1940 | 6378245 | 298.3 |
| 20 | GRS 80 | 6378137 | 298.257222101 |
| 21 | South American 1969 | 6378160 | 298.25 |
| 22 | WGS 72 | 6378135 | 298.26 |
| 23 | WGS 84 | 6378137 | 298.257223563 |
| 24 | PZ-90 | 6378136 | 298.257839303*1 |
| 25 | ITRF | 6378137 | 298.257222101*1 |

*1 supported only in Configure GNSS Datum ((ID: 0x64, SID: 0x27).

## B. Datum Reference List

| Datum index | Datum Name | Delta X | Delta Y | Delta Z | Ellipsoid | Ellipsoid Index | Region of Use |
|---|---|---|---|---|---|---|---|
| 0 | WGS-84 | 0 | 0 | 0 | WGS 84 | 23 | Global |
| 1 | Adindan | -118 | -14 | 218 | Clarke 1880 | 7 | Burkina Faso |
| 2 | Adindan | -134 | -2 | 210 | Clarke 1880 | 7 | Cameroon |
| 3 | Adindan | -165 | -11 | 206 | Clarke 1880 | 7 | Ethiopia |
| 4 | Adindan | -123 | -20 | 220 | Clarke 1880 | 7 | Mali |
| 5 | Adindan | -166 | -15 | 204 | Clarke 1880 | 7 | MEAN FOR Ethiopia; Sudan |
| 6 | Adindan | -128 | -18 | 224 | Clarke 1880 | 7 | Senegal |
| 7 | Adindan | -161 | -14 | 205 | Clarke 1880 | 7 | Sudan |
| 8 | Afgooye | -43 | -163 | 45 | Krassovsky 1940 | 19 | Somalia |
| 9 | Ain el Abd 1970 | -150 | -250 | -1 | International 1924 | 18 | Bahrain |
| 10 | Ain el Abd 1970 | -143 | -236 | 7 | International 1924 | 18 | Saudi Arabia |
| 11 | American Samoa 1962 | -115 | 118 | 426 | Clarke 1866 | 6 | American Samoa Islands |
| 12 | Anna 1 Astro 1965 | -491 | -22 | 435 | Australian National | 3 | Cocos Islands |
| 13 | Antigua Island Astro 1943 | -270 | 13 | 62 | Clarke 1880 | 7 | Antigua (Leeward Islands) |
| 14 | Arc 1950 | -138 | -105 | -289 | Clarke 1880 | 7 | Botswana |
| 15 | Arc 1950 | -153 | -5 | -292 | Clarke 1880 | 7 | Burundi |
| 16 | Arc 1950 | -125 | -108 | -295 | Clarke 1880 | 7 | Lesotho |
| 17 | Arc 1950 | -161 | -73 | -317 | Clarke 1880 | 7 | Malawi |
| 18 | Arc 1950 | -143 | -90 | -294 | Clarke 1880 | 7 | MEAN FOR Botswana; Lesotho; Malawi; Swaziland; Zaire; Zambia; Zimbabwe |
| 19 | Arc 1950 | -134 | -105 | -295 | Clarke 1880 | 7 | Swaziland |
| 20 | Arc 1950 | -169 | -19 | -278 | Clarke 1880 | 7 | Zaire |
| 21 | Arc 1950 | -147 | -74 | -283 | Clarke 1880 | 7 | Zambia |
| 22 | Arc 1950 | -142 | -96 | -293 | Clarke 1880 | 7 | Zimbabwe |
| 23 | Arc 1960 | -160 | -6 | -302 | Clarke 1880 | 7 | MEAN FOR Kenya; Tanzania |
| 24 | Arc 1960 | -157 | -2 | -299 | Clarke 1880 | 7 | Kenya |
| 25 | Arc 1960 | -175 | -23 | -303 | Clarke 1880 | 7 | Taanzania |
| 26 | Ascension Island 1958 | -205 | 107 | 53 | International 1924 | 18 | Ascension Island |
| 27 | Astro Beacon E 1945 | 145 | 75 | -272 | International 1924 | 18 | Iwo Jima |
| 28 | Astro DOS 71/4 | -320 | 550 | -494 | International 1924 | 18 | St Helena Island |
| 29 | Astro Tern Island (FRIG) 1961 | 114 | -116 | -333 | International 1924 | 18 | Tern Island |
| 30 | Astronomical Station 1952 | 124 | -234 | -25 | International 1924 | 18 | Marcus Island |
| 31 | Australian Geodetic 1966 | -133 | -48 | 148 | Australian National | 3 | Australia; Tasmania |
| 32 | Australian Geodetic 1984 | -134 | -48 | 149 | Australian National | 3 | Australia; Tasmania |
| 33 | Ayabelle Lighthouse | -79 | -129 | 145 | Clarke 1880 | 7 | Djibouti |
| 34 | Bellevue (IGN) | -127 | -769 | 472 | International 1924 | 18 | Efate & Erromango Islands |
| 35 | Bermuda 1957 | -73 | 213 | 296 | Clarke 1866 | 6 | Bermuda |
| 36 | Bissau | -173 | 253 | 27 | International 1924 | 18 | Guinea-Bissau |
| 37 | Bogota Observatory | 307 | 304 | -318 | International 1924 | 18 | Colombia |
| 38 | Bukit Rimpah | -384 | 664 | -48 | Bessel 1841 | 5 | Indonesia (Bangka & Belitung Ids) |
| 39 | Camp Area Astro | -104 | -129 | 239 | International 1924 | 18 | Antarctica (McMurdo Camp Area) |
| 40 | Campo Inchauspe | -148 | 136 | 90 | International 1924 | 18 | Argentina |
| 41 | Canton Astro 1966 | 298 | -304 | -375 | International 1924 | 18 | Phoenix Islands |
| 42 | Cape | -136 | -108 | -292 | Clarke 1880 | 7 | South Africa |
| 43 | Cape Canaveral | -2 | 151 | 181 | Clarke 1866 | 6 | Bahamas; Florida |
| 44 | Carthage | -263 | 6 | 431 | Clarke 1880 | 7 | Tunisia |
| 45 | Chatham Island Astro 1971 | 175 | -38 | 113 | International 1924 | 18 | New Zealand (Chatham Island) |
| 46 | Chua Astro | -134 | 229 | -29 | International 1924 | 18 | Paraguay |
| 47 | Corrego Alegre | -206 | 172 | -6 | International 1924 | 18 | Brazil |
| 48 | Dabola | -83 | 37 | 124 | Clarke 1880 | 7 | Guinea |
| 49 | Deception Island | 260 | 12 | -147 | Clarke 1880 | 7 | Deception Island; Antarctia |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 50 | Djakarta (Batavia) | -377 | 681 | -50 | Bessel 1841 | 5 | Indonesia (Sumatra) |
| 51 | DOS 1968 | 230 | -199 | -752 | International 1924 | 18 | New Georgia Islands (Gizo Island) |
| 52 | Easter Island 1967 | 211 | 147 | 111 | International 1924 | 18 | Easter Island |
| 53 | Estonia; Coordinate System 1937 | 374 | 150 | 588 | Bessel 1841 | 5 | Estonia |
| 54 | European 1950 | -104 | -101 | -140 | International 1924 | 18 | Cyprus |
| 55 | European 1950 | -130 | -117 | -151 | International 1924 | 18 | Egypt |
| 56 | European 1950 | -86 | -96 | -120 | International 1924 | 18 | England; Channel Islands; Scotland; Shetland Islands |
| 57 | European 1950 | -86 | -96 | -120 | International 1924 | 18 | England; Ireland; Scotland; Shetland Islands |
| 58 | European 1950 | -87 | -95 | -120 | International 1924 | 18 | Finland; Norway |
| 59 | European 1950 | -84 | -95 | -130 | International 1924 | 18 | Greece |
| 60 | European 1950 | -117 | -132 | -164 | International 1924 | 18 | Iran |
| 61 | European 1950 | -97 | -103 | -120 | International 1924 | 18 | Italy (Sardinia) |
| 62 | European 1950 | -97 | -88 | -135 | International 1924 | 18 | Italy (Sicily) |
| 63 | European 1950 | -107 | -88 | -149 | International 1924 | 18 | Malta |
| 64 | European 1950 | -87 | -98 | -121 | International 1924 | 18 | MEAN FOR Austria; Belgium; Denmark; Finland; France; W Germany; Gibraltar; Greece; Italy; Luxembourg; Netherlands; Norway; Portugal; Spain; Sweden; Switzerland |
| 65 | European 1950 | -87 | -96 | -120 | International 1924 | 18 | MEAN FOR Austria; Denmark; France; W Germany; Netherlands; Switzerland |
| 66 | European 1950 | -103 | -106 | -141 | International 1924 | 18 | MEAN FOR Iraq; Israel; Jordan; Lebanon; Kuwait; Saudi Arabia; Syria |
| 67 | European 1950 | -84 | -107 | -120 | International 1924 | 18 | Portugal; Spain |
| 68 | European 1950 | -112 | -77 | -145 | International 1924 | 18 | Tunisia |
| 69 | European 1979 | -86 | -98 | -119 | International 1924 | 18 | MEAN FOR Austria; Finland; Netherlands; Norway; Spain; Sweden; Switzerland |
| 70 | Fort Thomas 1955 | -7 | 215 | 225 | Clarke 1880 | 7 | Nevis; St. Kitts (Leeward Islands) |
| 71 | Gan 1970 | -133 | -321 | 50 | International 1924 | 18 | Republic of Maldives |
| 72 | Geodetic Datum 1949 | 84 | -22 | 209 | International 1924 | 18 | New Zealand |
| 73 | Graciosa Base SW 1948 | -104 | 167 | -38 | International 1924 | 18 | Azores (Faial; Graciosa; Pico; Sao Jorge; Terceira) |
| 74 | Guam 1963 | -100 | -248 | 259 | Clarke 1866 | 6 | Guam |
| 75 | Gunung Segara | -403 | 684 | 41 | Bessel 1841 | 5 | Indonesia (Kalimantan) |
| 76 | GUX 1 Astro | 252 | -209 | -751 | International 1924 | 18 | Guadalcanal Island |
| 77 | Herat North | -333 | -222 | 114 | International 1924 | 18 | Afghanistan |
| 78 | Hermannskogel Datum | 653 | -212 | 449 | Bessel 1841 (Namibia) | 4 | Croatia -Serbia, Bosnia-Herzegovina |
| 79 | Hjorsey 1955 | -73 | 46 | -86 | International 1924 | 18 | Iceland |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 80 | Hong Kong 1963 | -156 | -271 | -189 | International 1924 | 18 | Hong Kong |
| 81 | Hu-Tzu-Shan | -637 | -549 | -203 | International 1924 | 18 | Taiwan |
| 82 | Indian | 282 | 726 | 254 | Everest (India 1830) | 8 | Bangladesh |
| 83 | Indian | 295 | 736 | 257 | Everest (India 1956) | 10 | India; Nepal |
| 84 | Indian | 283 | 682 | 231 | Everest (Pakistan) | 13 | Pakistan |
| 85 | Indian 1954 | 217 | 823 | 299 | Everest (India 1830) | 8 | Thailand |
| 86 | Indian 1960 | 182 | 915 | 344 | Everest (India 1830) | 8 | Vietnam (Con Son Island) |
| 87 | Indian 1960 | 198 | 881 | 317 | Everest (India 1830) | 8 | Vietnam (Near 16øN)) |
| 88 | Indian 1975 | 210 | 814 | 289 | Everest (India 1830) | 8 | Thailand |
| 89 | Indonesian 1974 | -24 | -15 | 5 | Indonesian 1974 | 17 | Indonesia |
| 90 | Ireland 1965 | 506 | -122 | 611 | Modified Airy | 2 | Ireland |
| 91 | ISTS 061 Astro 1968 | -794 | 119 | -298 | International 1924 | 18 | South Georgia Islands |
| 92 | ISTS 073 Astro 1969 | 208 | -435 | -229 | International 1924 | 18 | Diego Garcia |
| 93 | Johnston Island 1961 | 189 | -79 | -202 | International 1924 | 18 | Johnston Island |
| 94 | Kandawala | -97 | 787 | 86 | Everest (India 1830) | 8 | Sri Lanka |
| 95 | Kerguelen Island 1949 | 145 | -187 | 103 | International 1924 | 18 | Kerguelen Island |
| 96 | Kertau 1948 | -11 | 851 | 5 | Everest (Malay. & Sing) | 12 | West Malaysia & Singapore |
| 97 | Kusaie Astro 1951 | 647 | 1777 | -1124 | International 1924 | 18 | Caroline Islands |
| 98 | Korean Geodetic System | 0 | 0 | 0 | GRS 80 | 20 | South Korea |
| 99 | L. C. 5 Astro 1961 | 42 | 124 | 147 | Clarke 1866 | 6 | Cayman Brac Island |
| 100 | Leigon | -130 | 29 | 364 | Clarke 1880 | 7 | Ghana |
| 101 | Liberia 1964 | -90 | 40 | 88 | Clarke 1880 | 7 | Liberia |
| 102 | Luzon | -133 | -77 | -51 | Clarke 1866 | 6 | Philippines (Excluding Mindanao) |
| 103 | Luzon | -133 | -79 | -72 | Clarke 1866 | 6 | Philippines (Mindanao) |
| 104 | M'Poraloko | -74 | -130 | 42 | Clarke 1880 | 7 | Gabon |
| 105 | Mahe 1971 | 41 | -220 | -134 | Clarke 1880 | 7 | Mahe Island |
| 106 | Massawa | 639 | 405 | 60 | Bessel 1841 | 5 | Ethiopia (Eritrea) |
| 107 | Merchich | 31 | 146 | 47 | Clarke 1880 | 7 | Morocco |
| 108 | Midway Astro 1961 | 912 | -58 | 1227 | International 1924 | 18 | Midway Islands |
| 109 | Minna | -81 | -84 | 115 | Clarke 1880 | 7 | Cameroon |
| 110 | Minna | -92 | -93 | 122 | Clarke 1880 | 7 | Nigeria |
| 111 | Montserrat Island Astro 1958 | 174 | 359 | 365 | Clarke 1880 | 7 | Montserrat (Leeward Islands) |
| 112 | Nahrwan | -247 | -148 | 369 | Clarke 1880 | 7 | Oman (Masirah Island) |
| 113 | Nahrwan | -243 | -192 | 477 | Clarke 1880 | 7 | Saudi Arabia |
| 114 | Nahrwan | -249 | -156 | 381 | Clarke 1880 | 7 | United Arab Emirates |
| 115 | Naparima BWI | -10 | 375 | 165 | International 1924 | 18 | Trinidad & Tobago |
| 116 | North American 1927 | -5 | 135 | 172 | Clarke 1866 | 6 | Alaska (Excluding Aleutian Ids) |
| 117 | North American 1927 | -2 | 152 | 149 | Clarke 1866 | 6 | Alaska (Aleutian Ids East of 180øW) |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 118 | North American 1927 | 2 | 204 | 105 | Clarke 1866 | 6 | Alaska (Aleutian Ids West of 180øW) |
| 119 | North American 1927 | -4 | 154 | 178 | Clarke 1866 | 6 | Bahamas (Except San Salvador Id) |
| 120 | North American 1927 | 1 | 140 | 165 | Clarke 1866 | 6 | Bahamas (San Salvador Island) |
| 121 | North American 1927 | -7 | 162 | 188 | Clarke 1866 | 6 | Canada (Alberta; British Columbia) |
| 122 | North American 1927 | -9 | 157 | 184 | Clarke 1866 | 6 | Canada (Manitoba; Ontario) |
| 123 | North American 1927 | -22 | 160 | 190 | Clarke 1866 | 6 | Canada (New Brunswick; Newfoundland; Nova Scotia; Quebec) |
| 124 | North American 1927 | 4 | 159 | 188 | Clarke 1866 | 6 | Canada (Northwest Territories; Saskatchewan) |
| 125 | North American 1927 | -7 | 139 | 181 | Clarke 1866 | 6 | Canada (Yukon) |
| 126 | North American 1927 | 0 | 125 | 201 | Clarke 1866 | 6 | Canal Zone |
| 127 | North American 1927 | -9 | 152 | 178 | Clarke 1866 | 6 | Cuba |
| 128 | North American 1927 | 11 | 114 | 195 | Clarke 1866 | 6 | Greenland (Hayes Peninsula) |
| 129 | North American 1927 | -3 | 142 | 183 | Clarke 1866 | 6 | MEAN FOR Antigua; Barbados; Barbuda; Caicos Islands; Cuba; Dominican Republic; Grand Cayman; Jamaica; Turks Islands |
| 130 | North American 1927 | 0 | 125 | 194 | Clarke 1866 | 6 | MEAN FOR Belize; Costa Rica; El Salvador; Guatemala; Honduras; Nicaragua |
| 131 | North American 1927 | -10 | 158 | 187 | Clarke 1866 | 6 | MEAN FOR Canada |
| 132 | North American 1927 | -8 | 160 | 176 | Clarke 1866 | 6 | MEAN FOR CONUS |
| 133 | North American 1927 | -9 | 161 | 179 | Clarke 1866 | 6 | MEAN FOR CONUS (East of Mississippi; River Including Louisiana; Missouri; Minnesota) |
| 134 | North American 1927 | -8 | 159 | 175 | Clarke 1866 | 6 | MEAN FOR CONUS (West of Mississippi; River Excluding Louisiana; Minnesota; Missouri) |
| 135 | North American 1927 | -12 | 130 | 190 | Clarke 1866 | 6 | Mexico |
| 136 | North American 1983 | 0 | 0 | 0 | GRS 80 | 20 | Alaska (Excluding Aleutian Ids) |
| 137 | North American 1983 | -2 | 0 | 4 | GRS 80 | 20 | Aleutian Ids |
| 138 | North American 1983 | 0 | 0 | 0 | GRS 80 | 20 | Canada |
| 139 | North American 1983 | 0 | 0 | 0 | GRS 80 | 20 | CONUS |
| 140 | North American 1983 | 1 | 1 | -1 | GRS 80 | 20 | Hawaii |
| 141 | North American 1983 | 0 | 0 | 0 | GRS 80 | 20 | Mexico; Central America |
| 142 | North Sahara 1959 | -186 | -93 | 310 | Clarke 1880 | 7 | Algeria |
| 143 | Observatorio Meteorologico 1939 | -425 | -169 | 81 | International 1924 | 18 | Azores (Corvo & Flores Islands) |
| 144 | Old Egyptian 1907 | -130 | 110 | -13 | Helmert 1906 | 15 | Egypt |
| 145 | Old Hawaiian | 89 | -279 | -183 | Clarke 1866 | 6 | Hawaii |
| 146 | Old Hawaiian | 45 | -290 | -172 | Clarke 1866 | 6 | Kauai |
| 147 | Old Hawaiian | 65 | -290 | -190 | Clarke 1866 | 6 | Maui |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 148 | Old Hawaiian | 61 | -285 | -181 | Clarke 1866 | 6 | MEAN FOR Hawaii; Kauai; Maui; Oahu |
| 149 | Old Hawaiian | 58 | -283 | -182 | Clarke 1866 | 6 | Oahu |
| 150 | Oman | -346 | -1 | 224 | Clarke 1880 | 7 | Oman |
| 151 | Ordnance Survey Great Britain 1936 | 371 | -112 | 434 | Airy 1830 | 1 | England |
| 152 | Ordnance Survey Great Britain 1936 | 371 | -111 | 434 | Airy 1830 | 1 | England; Isle of Man; Wales |
| 153 | Ordnance Survey Great Britain 1936 | 375 | -111 | 431 | Airy 1830 | 1 | MEAN FOR England; Isle of Man; Scotland; Shetland Islands; Wales |
| 154 | Ordnance Survey Great Britain 1936 | 384 | -111 | 425 | Airy 1830 | 1 | Scotland; Shetland Islands |
| 155 | Ordnance Survey Great Britain 1936 | 370 | -108 | 434 | Airy 1830 | 1 | Wales |
| 156 | Pico de las Nieves | -307 | -92 | 127 | International 1924 | 18 | Canary Islands |
| 157 | Pitcairn Astro 1967 | 185 | 165 | 42 | International 1924 | 18 | Pitcairn Island |
| 158 | Point 58 | -106 | -129 | 165 | Clarke 1880 | 7 | MEAN FOR Burkina Faso & Niger |
| 159 | Pointe Noire 1948 | -148 | 51 | -291 | Clarke 1880 | 7 | Congo |
| 160 | Porto Santo 1936 | -499 | -249 | 314 | International 1924 | 18 | Porto Santo; Madeira Islands |
| 161 | Provisional South American 1956 | -270 | 188 | -388 | International 1924 | 18 | Bolivia |
| 162 | Provisional South American 1956 | -270 | 183 | -390 | International 1924 | 18 | Chile (Northern; Near 19 øS) |
| 163 | Provisional South American 1956 | -305 | 243 | -442 | International 1924 | 18 | Chile (Southern; Near 43 øS) |
| 164 | Provisional South American 1956 | -282 | 169 | -371 | International 1924 | 18 | Colombia |
| 165 | Provisional South American 1956 | -278 | 171 | -367 | International 1924 | 18 | Ecuador |
| 166 | Provisional South American 1956 | -298 | 159 | -369 | International 1924 | 18 | Guyana |
| 167 | Provisional South American 1956 | -288 | 175 | -376 | International 1924 | 18 | MEAN FOR Bolivia; Chile; Colombia; Ecuador; Guyana; Peru; Venezuela |
| 168 | Provisional South American 1956 | -279 | 175 | -379 | International 1924 | 18 | Peru |
| 169 | Provisional South American 1956 | -295 | 173 | -371 | International 1924 | 18 | Venezuela |
| 170 | Provisional South Chilean 1963 | 16 | 196 | 93 | International 1924 | 18 | Chile (Near 53 øS) (Hito XVIII) |
| 171 | Puerto Rico | 11 | 72 | -101 | Clarke 1866 | 6 | Puerto Rico; Virgin Islands |
| 172 | Pulkovo 1942 | 28 | -130 | -95 | Krassovsky 1940 | 19 | Russia |
| 173 | Qatar National | -128 | -283 | 22 | International 1924 | 18 | Qatar |
| 174 | Qornoq | 164 | 138 | -189 | International 1924 | 18 | Greenland (South) |
| 175 | Reunion | 94 | -948 | -1262 | International 1924 | 18 | Mascarene Islands |
| 176 | Rome 1940 | -225 | -65 | 9 | International 1924 | 18 | Italy (Sardinia) |
| 177 | S-42 (Pulkovo 1942) | 28 | -121 | -77 | Krassovsky 1940 | 19 | Hungary |
| 178 | S-42 (Pulkovo 1942) | 23 | -124 | -82 | Krassovsky 1940 | 19 | Poland |
| 179 | S-42 (Pulkovo 1942) | 26 | -121 | -78 | Krassovsky 1940 | 19 | Czechoslavakia |
| 180 | S-42 (Pulkovo 1942) | 24 | -124 | -82 | Krassovsky 1940 | 19 | Latvia |
| 181 | S-42 (Pulkovo 1942) | 15 | -130 | -84 | Krassovsky 1940 | 19 | Kazakhstan |
| 182 | S-42 (Pulkovo 1942) | 24 | -130 | -92 | Krassovsky 1940 | 19 | Albania |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 183 | S-42 (Pulkovo 1942) | 28 | -121 | -77 | Krassovsky 1940 | 19 | Romania |
| 184 | S-JTSK | 589 | 76 | 480 | Bessel 1841 | 5 | Czechoslavakia (Prior 1 JAN 1993) |
| 185 | Santo (DOS) 1965 | 170 | 42 | 84 | International 1924 | 18 | Espirito Santo Island |
| 186 | Sao Braz | -203 | 141 | 53 | International 1924 | 18 | Azores (Sao Miguel; Santa Maria Ids) |
| 187 | Sapper Hill 1943 | -355 | 21 | 72 | International 1924 | 18 | East Falkland Island |
| 188 | Schwarzeck | 616 | 97 | -251 | Bessel 1841 (Namibia) | 4 | Namibia |
| 189 | Selvagem Grande 1938 | -289 | -124 | 60 | International 1924 | 18 | Salvage Islands |
| 190 | Sierra Leone 1960 | -88 | 4 | 101 | Clarke 1880 | 7 | Sierra Leone |
| 191 | South American 1969 | -62 | -1 | -37 | South American 1969 | 21 | Argentina |
| 192 | South American 1969, | -61 | 2 | -48 | South American 1969 | 21 | Bolivia |
| 193 | South American 1969, | -60 | -2 | -41 | South American 1969 | 21 | Brazil |
| 194 | South American 1969, | -75 | -1 | -44 | South American 1969 | 21 | Chile |
| 195 | South American 1969, | -44 | 6 | -36 | South American 1969 | 21 | Colombia |
| 196 | South American 1969, | -48 | 3 | -44 | South American 1969 | 21 | Ecuador |
| 197 | South American 1969, | -47 | 26 | -42 | South American 1969 | 21 | Ecuador (Baltra; Galapagos) |
| 198 | South American 1969, | -53 | 3 | -47 | South American 1969 | 21 | Guyana |
| 199 | South American 1969, | -57 | 1 | -41 | South American 1969 | 21 | MEAN FOR Argentina; Bolivia; Brazil; Chile; Colombia; Ecuador; Guyana; Paraguay; Peru; Trinidad & Tobago; Venezuela |
| 200 | South American 1969, | -61 | 2 | -33 | South American 1969 | 21 | Paraguay |
| 201 | South American 1969, | -58 | 0 | -44 | South American 1969 | 21 | Peru |
| 202 | South American 1969, | -45 | 12 | -33 | South American 1969 | 21 | Trinidad & Tobago |
| 203 | South American 1969, | -45 | 8 | -33 | South American 1969 | 21 | Venezuela |
| 204 | South Asia | 7 | -10 | -26 | Modified Fischer 1960 | 14 | Singapore |
| 205 | Tananarive Observatory 1925 | -189 | -242 | -91 | International 1924 | 18 | Madagascar |
| 206 | Timbalai 1948 | -679 | 669 | -48 | Everest (Sabah Sarawak) | 9 | Brunei; E. Malaysia (Sabah Sarawak) |
| 207 | Tokyo | -148 | 507 | 685 | Bessel 1841 | 5 | Japan |
| 208 | Tokyo | -148 | 507 | 685 | Bessel 1841 | 5 | MEAN FOR Japan; South Korea; Okinawa |
| 209 | Tokyo | -158 | 507 | 676 | Bessel 1841 | 5 | Okinawa |
| 210 | Tokyo | -147 | 506 | 687 | Bessel 1841 | 5 | South Korea |
| 211 | Tristan Astro 1968 | -632 | 438 | -609 | International 1924 | 18 | Tristan da Cunha |
| 212 | Viti Levu 1916 | 51 | 391 | -36 | Clarke 1880 | 7 | Fiji (Viti Levu Island) |
| 213 | Voirol 1960 | -123 | -206 | 219 | Clarke 1880 | 7 | Algeria |
| 214 | Wake Island Astro 1952 | 276 | -57 | 149 | International 1924 | 18 | Wake Atoll |
| 215 | Wake-Eniwetok 1960 | 102 | 52 | -38 | Hough 1960 | 16 | Marshall Islands |
| 216 | WGS 1972 | 0 | 0 | 0 | WGS 72 | 22 | Global Definition |
| 217 | Yacare | -155 | 171 | 37 | International 1924 | 18 | Uruguay |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 218 | Zanderij | -265 | 120 | -358 | International 1924 | 18 | Suriname |

| Datum index | Datum Name | Delta X | Delta Y | Delta Z | Rotation X | Rotation Y | Rotation Z | Scale Factor | Ellipsoid | Ellipsoid Index | Region of Use |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 219*1 | Pulkovo 1995 | 24.82 | -131.21 | -82.66 | 0.000 | 0.000 | -0.160 | -82.66 | Krassovsky 1940 | 19 | Russia |
| 220*1 | PZ-90 | 0.00 | 0.00 | 1.50 | 0.000 | 0.000 | -0.076 | 0 | PZ-90 | 24 | Global |
| 221*1 | CSCS2000 | -0.0048 | -0.0026 | 0.0332 | 0 | 0 | -0.0006 | -0.00292 | ITRF | 25 | China |

*1 supported only in Configure GNSS Datum ((ID: 0x64, SID: 0x27).

## Change Log

Ver 1.4.66 July 03 2023

1. Modify “Configure GNSS Navigation Mode: ID: 0x64, SID: 0x17” and “GNSS Navigation Mode: ID: 0x64, SID: 0x8B”, to add quadcopter mode and SLR mode.

Ver 1.4.65 April 13 2023

1. Add “Query Requested NMEA String Interval, ID: 0x64, SID: 0x40”, and “Requested NMEA String Interval, ID: 0x64, SID: 0x9F” 2 messages.
2. Modify “Configure SBAS Advanced, ID: 0x62, SID: 0x05” and “SBAS Advanced, ID: 0x62, SID: 0x82” 2 messages to support SouthPAN PRN.

Ver 1.4.64 July 05 2022

1. Add “Query Requested PSTI Message Interval, ID: 0x64, SID: 0x23”, and “Requested PSTI Message Interval, ID: 0x64, SID: 0x90” 2 messages.

Ver 1.4.63 July 05 2022

1. Update “RTK Mode and Operational Function, ID: 0x6A, SID: 0x83” fields 33~41, move back “Baseline Length Constraint” to fields 33~36 Before “Run-time Operational Function” and “Run-time Survey Length”.

Ver 1.4.62 May 26 2022

1. Remove “Configure GPS Parameter Search Engine Number, ID: 0x64, SID: 0xA” due to parameter search engine number is fixed and non-configurable.
2. Modify “Configure 1PPS Output Mode, ID: 0x55” and “1PPS Output Mode, ID: 0xC3” 3 messages output mode field not supporting “No output”.
3. Add “Set Galileo Ephemeris, ID: 0x6E, SID: 0x01”, “Get Galileo Ephemeris, ID: 0x6E, SID: 0x02”, “and “Galileo Ephemeris, ID: 0x6E, SID: 0x80” 3 messages.
4. Modify “Configure SBAS Advanced, ID: 0x62, SID: 0x05” and “SBAS Advanced, ID: 0x62, SID: 0x82” 2 messages to support SDCM and BDSBAS.
5. Modify “Configure SBAS, ID: 0x62, SID: 0x01” and “SBAS Status, ID: 0x62, SID: 0x80” 2 messages to support SDCM and BDSBAS.

Ver 1.4.61 March 07 2022

1. Fixed incorrect example checksum.

Ver 1.4.60 Feb. 09 2022

1. Update examples of “Configure UTC reference time sync to GPS time, ID: 0x64, SID: 0x15” and “GPS UTC reference time, ID: 0x64, SID: 0x8A”.
2. Update “RTK Mode and Operational Function, ID: 0x6A, SID: 0x83” fields 33~41, move “Baseline Length Constraint” to fields 38~41 after “Run-time Operational Function” and “Run-time Survey Length”. It was a mistake in Ver 1.4.57 update.

Ver 1.4.59 Dec. 08, 2021

1. Add “Configure SBAS Advanced, ID: 0x62, SID: 0x05”, “Query SBAS Advanced, ID: 0x62, SID: 0x06”, “and “SBAS Advanced, ID: 0x62, SID: 0x82” 3 messages

Ver 1.4.58 Dec. 06, 2021

1. Remove “Configure GPS/UTC leap seconds, ID: 0x64, SID: 0x1F” and replaced by “Configure GPS/UTC leap seconds in UTC, ID: 0x64, SID: 0x2D”.

Ver 1.4.57 Oct. 06, 2021

1. Update “RTK Mode and Operational Function, ID: 0x6A, SID: 0x83” fields 33~41, move “Baseline Length Constraint” to fields 33~36 before “Run-time Operational Function” and “Run-time Survey Length”.

Ver 1.4.56 July. 23, 2021

1. Update “GNSS DOP Mask, ID: 0xAF” the DOP Mode Select field.

Ver 1.4.55 July. 16, 2021

1. Rename “Configure 1PPS Frequency Output, ID: 0x65, SID: 0x3”, “Query 1PPS Frequency Output, ID: 0x65, SID: 0x4”, “1PPS Frequency Output, ID: 0x65, SID: 0x81” 3 messages to “Configure PPS2 Frequency Output, ID: 0x65, SID: 0x3”, “Query PPS2 Frequency Output, ID: 0x65, SID: 0x4”, “PPS2 Frequency Output, ID: 0x65, SID: 0x81” respectively

Ver 1.4.54 June. 08, 2021

1. Add “Query RTK Rover Moving Base Heading and Pitch Offsets, ID: 0x6A, SID: 0x16”, “Configure RTK Rover Moving Base Heading and Pitch Offsets, ID: 0x6A, SID: 0x15”, “and “Heading and Pitch Offsets of RTK Rover Moving Base, ID: 0x6A, SID: 0x89” 3 messages.
2. Add “Query PX1172RH Rover Moving Base Heading and Pitch Offsets, ID: 0x7A, Sub ID: 0xE, Sub Sub-ID: 0x4”, “Configure PX1172RH Rover Moving Base Heading and Pitch Offsets, ID: 0x7A, Sub ID: 0xE, Sub Sub-ID: 0x05”, “and “Heading and Pitch Offsets of PX1172RH Rover Moving Base, ID: 0x7A, Sub ID: 0xE, Sub Sub-ID: 0x83” 3 messages.

Ver 1.4.53 April. 21, 2021

1. Add “Query PX1172RH Rover Moving Base Software Version, ID: 0x7A, Sub ID: 0xE, Sub Sub-ID: 0x1”, “Query PX1172RH Rover Moving Base Software CRC, ID: 0x7A, Sub ID: 0xE, Sub Sub-ID: 0x2”, “Query PX1172RH Rover Moving Base Position Update Rate, ID: 0x7A, Sub ID: 0xE, Sub Sub-ID: 0x3”, “Software Version of PX1172RH Rover Moving Base, ID: 0x7A, Sub ID: 0xE, Sub Sub-ID: 0x80”, “Software CRC of PX1172RH Rover Moving Base, ID: 0x7A, Sub ID: 0xE, Sub Sub-ID: 0x81”, “Position Update Rate of PX1172RH Rover Moving Base, ID: 0x7A, Sub ID: 0xE, Sub Sub-ID: 0x82” messages. Those are messages with ID, Sub ID and Sub Sub-ID.

Ver 1.4.52 Mar. 19, 2021

1. Update “Configure RTK Mode and Operational Function, ID: 0x6A, SID: 0x6”, fields 5~32 descriptions.
2. Update “RTK Mode and Operational Function, ID: 0x6A, SID: 0x83” fields 5~37, descriptions, and change field 33 “name” from “Run-time timing mode” to “Run-time Operational Function”.

Ver 1.4.51 Mar. 08, 2021

1. Update “GPS Ephemeris Data, ID: 0xB1” the ephemeris data to include reserve field.
2. Add “Configure Navigation Data Message Interval, ID: 0x64, SID: 0x2F”, “Query Navigation Data Message Interval, ID: 0x64, SID: 0x30”, and “Navigation Data Message Interval, ID: 0x64, SID: 0x98” 3 messages.
3. Add “Navigation Data Message, ID: 0xA8” message.
4. Add “Configure GNSS Doze Mode, ID: 0x64, SID: 0x1C” message.

Ver 1.4.50 Feb. 20, 2021

1. Update “RTK Mode and Operational Function, ID: 0x6A, SID: 0x83” field 13~32, latitude, longitude and altitude.

Ver 1.4.49 Dec. 30, 2020

1. Add “Configure RTK Slave Base Serial Port Baud Rate, ID: 0x6A, SID: 0xC”, “Query RTK Slave Base Serial Port Baud Rate, ID: 0x6A, SID: 0xD”, and “RTK Slave Base Serial Port Baud Rate, ID: 0x6A, SID: 0x85” 3 messages
2. Add “Configure RTK Precisely Kinematic Base Serial Port Baud Rate, ID: 0x6A, SID: 0x13”, “Query RTK Precisely Kinematic Base Serial Port Baud Rate, ID: 0x6A, SID: 0x14”, and “RTK Precisely Kinematic Base Serial Port Baud Rate, ID: 0x6A, SID: 0x88” 3 messages.
3. Add “Configure NMEA String Interval, ID: 0x64, SID: 0x3B”, “Query NMEA String Interval, ID: 0x64, SID: 0x3C”, and “NMEA String Interval, ID: 0x64, SID: 0x9D” 3 messages
4. Add “Configure 1PPS Output Mode, ID: 0x55”, “Query 1PPS Output Mode, ID: 0x56” and “1PPS Output Mode, ID: 0xC3” 3 messages.
5. Update “Configure NMEA Talker ID, ID: 0x4B” and “NMEA Talker ID, ID: 0x93” to add auto mode.

Ver 1.4.48, Aug. 14, 2020

1. Update appendix A, Ellipsoid List and appendix B, Datum Reference List to add CGCS2000 coordinate datum index.

Ver 1.4.47, April 14, 2020

1. Initial release based on AN0028 1.4.46.

SkyTraq Technology, Inc.
5F, No.26, Minsiang Street, Hsinchu, Taiwan, 300
Phone: +886 3 5678650
Fax: +886 3 5678680
Email: info@skytraq.com.tw