Teseo-LIV4F GNSS Module - Software manual

Introduction

The Teseo-LIV4F module family is an easy to use Global Navigation Satellite System (GNSS) stand-alone modules, embedding Teseo single die stand-alone positioning receiver IC working on multiple constellations (GPS/Galileo/Glonass/BeiDou/QZSS).

The modules are designed for top performance in a minimal space and it has been optimized for cost sensitive applications without quality compromise. It allows, at competitive costs, an easy integration and migration from existing designs of products such as trackers, telematics, portable, tablets, marine and sports accessories.

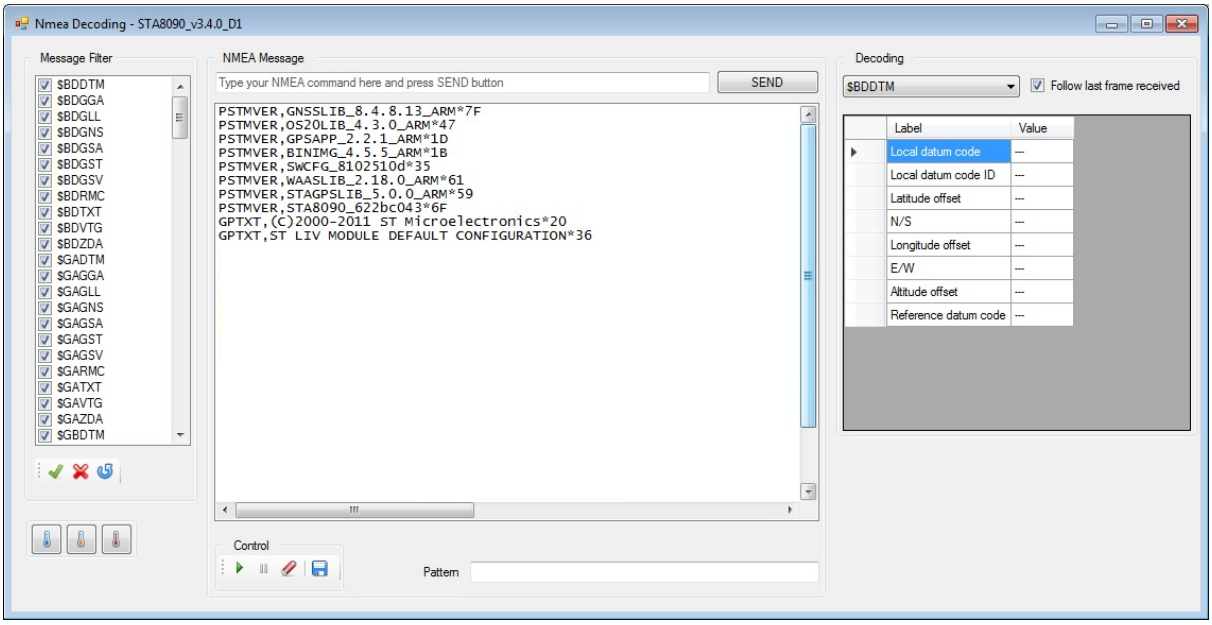
Within its compact size, Teseo-LIV4F is offering superior accuracy thanks to the on board Temperature Compensated Crystal Oscillator (TCXO) and a reduced Time To First Fix (TTFF) relying on its dedicated Real Time Clock (RTC) oscillator.

The devices are offered with a complete GNSS firmware which performs all GNSS operations including acquisition, tracking and navigation and data output with no need of external memories.

1 Firmware version

The firmware version defines which set of messages the receiver is able to manage.
The command `[$PSTMGETSWVER]` returns the firmware and all software versions in string format.
While booting Teseo Module reports on the serial port the current configuration as showed in the following figure:

Figure 1. Teseo Module booting message from UART



Each entry of the following table identifies a specific Teseo Module firmware subsystem version.

Table 1. Teseo Module firmware subsystem version

Entry	Description
PSTMVER,FreeRTOS*35	FreeRTOS configuration Version
GPTXT,(C)2000-2011 ST Microelectronics*20	Log message
GPTXT,ST LIV MODULE DEFAULT CONFIGURATION*36	Log message

The Binary Image Version covers all the firmware subsystem, therefore on every firmware subsystem update the Binary Image Version updates as well.

1.1 Firmware update algorithm protocol

Teseo Module supports the firmware upgrade.

Both the Host and Teseo Module have to follow a well-defined protocol.

Caution: take care that during the whole firmware upgrade procedure the Voltage VCC and VCC_IO must remain applied and stable; a power outage, during the firmware upgrade procedure, could force Teseo Module in an unrecoverable state.

Firmware upgrade has a preliminary phase to synchronize the Host and the Teseo Module.

Just after the synchronization with the device, the Host must send the binary image options. These options are packed inside a structure; below there's the description and the specific values:

```
struct ImageOptions
{
unsigned char reserved_0;
unsigned char reserved_1;
unsigned char chunk_size;
unsigned char reserved_2;
unsigned int firmwareSize;
unsigned int firmwareCRC;
unsigned int reserved_3;
unsigned int reserved_4;
} img_option = {
  .reserved_0 = 1,
  .reserved_1= 0,
  .chunk_size = <CHUNK_SIZE>,
  .reserved_2= 1,
  .firmwareSize = <FIRMWARE_SIZE>,
  .firmwareCRC = <FIRMWARE_CRC>,
  .reserved_3 = 0x00100000,
  .reserved_4 = 0x00100000,
};
```

The Host has to specify the chunk size, the firmware size and the firmware CRC in the related fields.

The chunk size can be selected setting the .chunk_size field in the structure ImageOptions; chunk size selection is described in the Table 2:

Table 2. Chunk_size bit field description

Chunk-size bit field	Description
[7:4]	Reserved must be zero
[3:0]	Set the chunk size value: 0: 16 Kbytes; 1: 1 Kbytes; 2: 2 Kbytes; 3: 3 Kbytes; 4: 4 Kbytes; 5: 5 Kbytes; 6: 6 Kbytes; 7: 7 Kbytes; 8: 8 Kbytes; 9: 9 Kbytes; 0: 10 Kbytes; 11: 11 Kbytes; 12: 12 Kbytes; 13: 13 Kbytes; 14: 14 Kbytes;

Chunk-size bit field	Description
	15: 15 Kbytes;

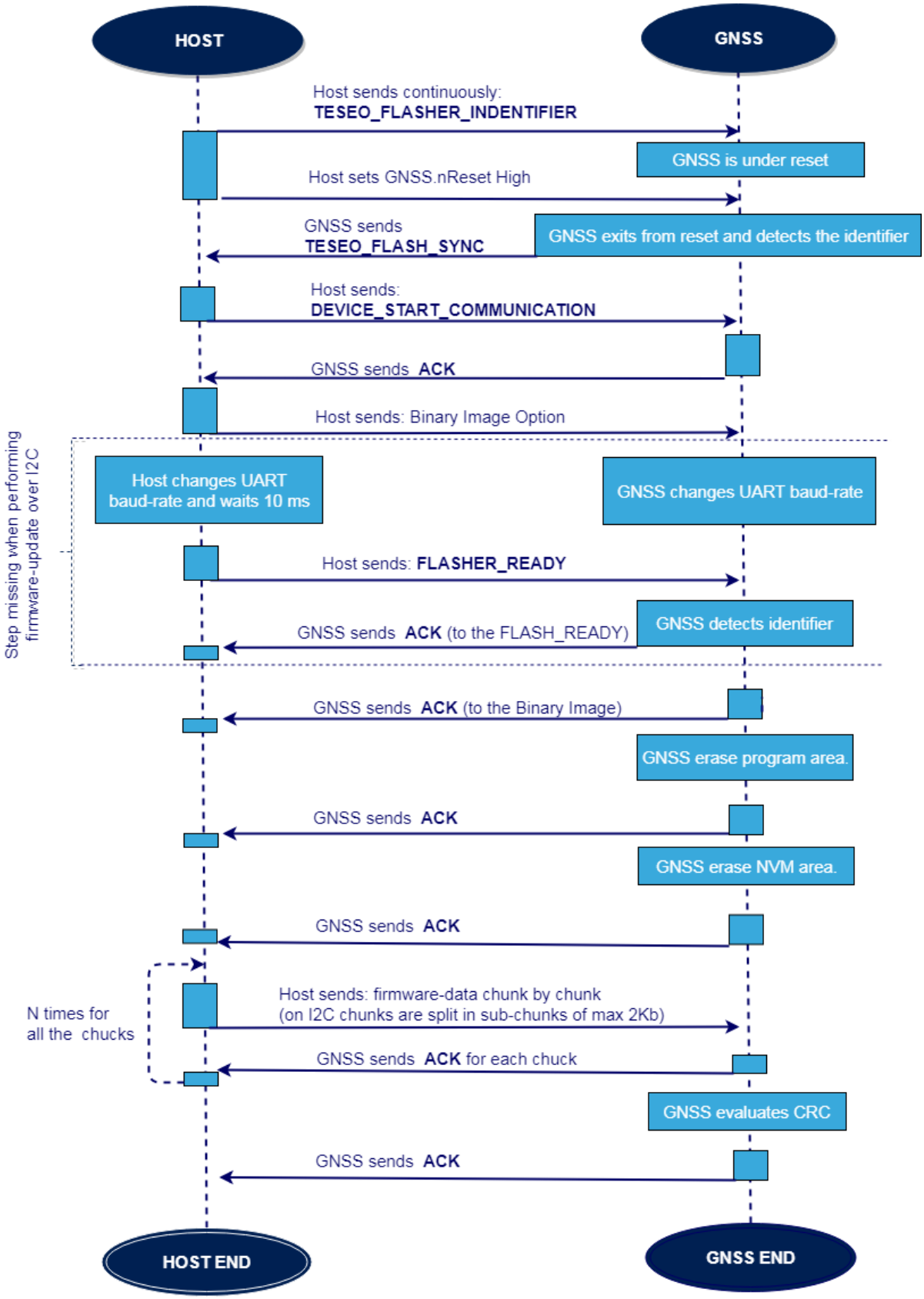
While sending the firmware, data host has to split the binary image in a N chunks with the selected chunk-size (in the ImageOptions structure); last chunk size must be equal to the remaining bytes number.

Each data chunk will be acknowledged with "ACK" response from Teseo Module.

When all the chunks are sent, Teseo Module performs a CRC error check on the image data received by the Host; if the check is passed an "ACK" response is sent back to the Host and the new downloaded firmware is validated. Otherwise if the check failed a "NAK" response is sent. In both cases Teseo Module device resets itself.

The firmware upgrade procedure is shown in Figure 2.

Figure 2. **Teseo Module firmware upgrade procedure**



Firmware upgrade procedure, on Teseo Module, uses the following constants:

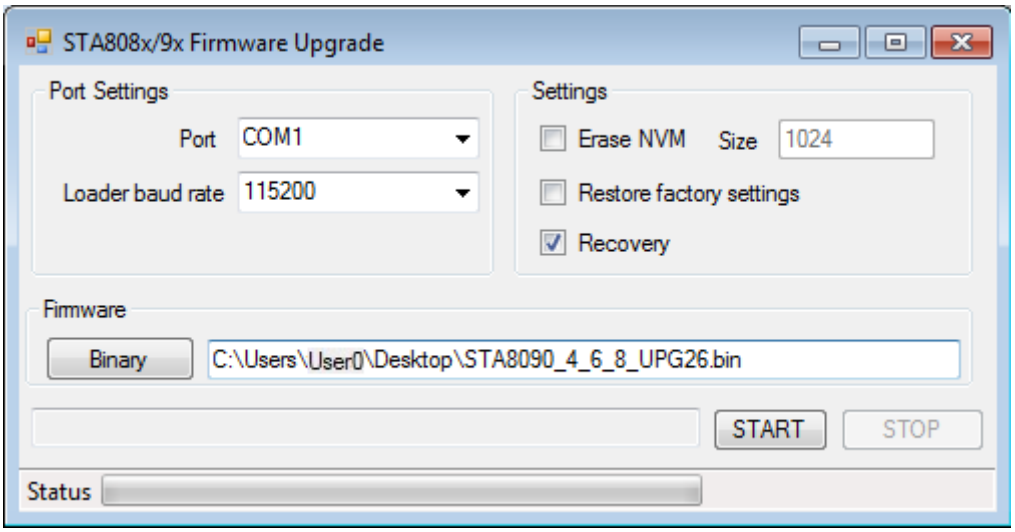
Table 3. Firmware upgrade constants

Constants	Value
TESEO_FLASHER_IDENTIFIER	0xBCD501F4
TESEO_FLASHER_SYNC	0x83984073
DEVICE_START_COMMUNICATION	0xA3
FLASHER_READY	0x4A
ACK	0xCC

1.2 Firmware update software tool

Teseo Module firmware update is supported through the ST Firmware Upgrade tool.

Figure 3. Firmware Upgrade tool



ST Firmware Upgrade tool is provided with the ST Teseo Suite Light program.

When the user wants to update the Teseo Module, Teseo Module must be powered and under reset; the user must configure the tool, select binary image and start firmware upgrade process. Only when the process has started, user must take Teseo Module out of reset.

1.2.1 Port settings option

When user clicks on UART mode button all COM ports available on your PC will be listed in the Output port box; also, two boxes for the baud rate selection are now selectable.

- **NMEA baud rate**: this is the UART baud rate used to send the FW Upgrade command and start the update process. When "Auto" is selected, the tool tries to automatically detect the baud rate of the select port and use it to send the command;
- **Baud rate**: this is the UART baud rate used to download the new firmware.
- **Output port**: COM port used to update the firmware;

USB mode is related to other ST-GNSS solutions and it doesn't have to be enabled in case of Teseo Module.

1.2.2 Firmware options

- **Erase NVM**: check this flag if you want to erase ST proprietary NVM during firmware upgrade process. If this flag is checked, the size of NVM can be entered using the related text box. Value is expressed in KB. The default value is 1024;
- **Program only**: check this flag if you don't want to erase program memory before writing new firmware.
- **Dump**: not available;
- **Recovery**: check this flag if you want to update the Teseo Module;

On this window there are two boxes where information about firmware size and CRC code is displayed; these fields are read-only. After configuring all options, the load button can be pushed in order to upload the firmware binary image

1.2.3 Upgrade process

When all preliminary steps described above are completed, the firmware upgrade process can be executed by clicking on Start button. A progress bar will be displayed in the status bar. The update process can be stopped by clicking on Stop button.

When upgrade has finished a confirmation message is displayed. If the process failed or was stopped by the user, no backup firmware can be executed; the only way to re-install a working firmware is to reset the hardware and start a new upgrade process.

2 Receiver Description

2.1 Receiver Configuration

The Teseo Module binary image supports the firmware configuration facility. It allows changing some application parameters in order to address most of the specific HW constraints and/or the final product functionality requirements.

The firmware configuration management supports the "Factory Setting", embedded in the binary code, and the "Customized Setting", stored in the GNSS backup memory (NVM). The "Factory Setting" can be changed and saved at run-time using specific NMEA commands.

Teseo Module binary image software is released with the ST defined default setting (Factory Setting).

2.1.1 Configuration Concept

All configuration parameters are grouped in a data block. Each field is addressed by a unique ID. The IDs are made by three digits: the most significant one represents the parameter type and the others are used to identify different parameters of the same type.

Default setting of configuration data block is hard coded into the binary image file.

When the system is running, it could be possible to have up to three different configuration blocks:

- Current configuration: it is placed in RAM memory and it includes the current configuration of each parameter. At start-up, the current configuration block is loaded from NVM (if a stored data block is available) or it is loaded from the default one embedded in the code (factory settings).
- Default configuration: it is generally placed in the flash/rom memory. It includes the factory setting for each parameter. This configuration is used at system startup if there is no configuration data into the NVM memory.
- NVM stored configuration: it is available in the NVM backup. It includes all parameters modified and stored by the user. At system startup the SW configuration management checks if a valid configuration block is available in the NVM backup memory. In case the stored configuration is available, it will be used for system configuration. If not available the default setting will be used.

The receiver always uses only the Current Configuration.

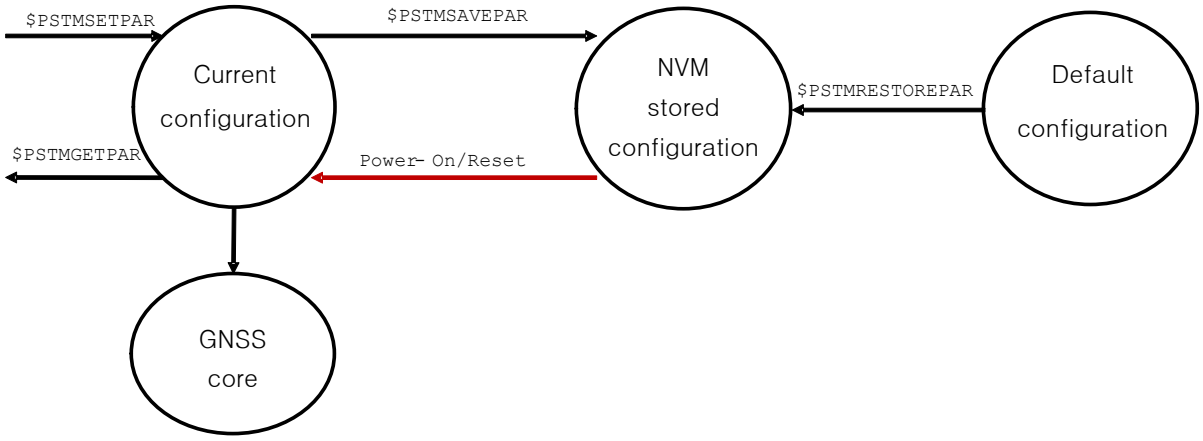
Current Configuration will be lost when there is:

- a power cycle
- a hardware reset
- a software reset

The Current Configuration can be made permanent (stored in a non-volatile memory) by saving it to the "NVM stored configuration".

On NMEA protocol the run-time configuration parameters can be read, changed and stored (in NVM) using the system configuration commands: $PSTMSETPAR, $PSTMGETPAR and $PSTMSAVEPAR. There is also a command to restore the factory setting parameters: $PSTMRESTOREPAR.

Figure 4. **Custom Configuration using NMEA Protocol**



For example if the UART baud rate would change, the following commands should be sent by the Host:

1. `$PSTMSETPAR,3102,0x9`
2. `$PSTMSAVEPAR`
3. `$PSTMSRR`

Where:

1. `$PSTMSETPAR` changes the UART's baudrate;
2. `$PSTMSAVEPAR` saves the whole configuration;
3. `$PSTMSRR` restarts the Teseo Module to guarantee that the change made is effective;

2.1.2 Configuration Data Blocks (CDB)

The configuration is divided into several sub-sections.

The IDs are made by three digits: the most significant one represents the parameter type and the others are used to identify different parameters of the same type.

2.2 Communication channels

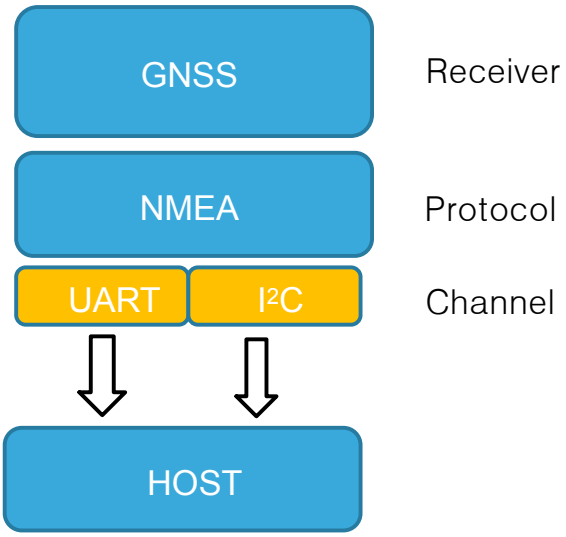
On the Teseo Module the NMEA Protocol is communication channel independent.

Users can select the channel based on their needs.

Teseo Module receiver supports the following communication channels:

- UART channel;
- I^2C channel;

Figure 5. **Teseo Module protocol routing over the available ports**



2.2.1 Communication over UART Port

Teseo Module receiver and Host are connected by serial port. Communication parameters are the following:

- 8 data bits
- No parity
- 1 stop bit
- 115200 bauds

In both directions, communication is based on the frames described in next sections.

From Teseo Module receiver to Host frames can be:

- Unsolicited: For instance, periodical frame reporting position
- Data Responses: Teseo Module Receiver returns data requested by Host
- ACK: in case no data need to be returned to Host (e.g. on a reset request), simple ACK is sent
- NACK: if request contains wrong parameters, NACK is returned to Host.

From Host to Teseo Module receiver frames can be:

- Read Requests;
- Write reset, initialization Requests

2.2.2 Communication over I²C Port

I²C is a two-wire communication interface invented by Philips Semiconductor.

Unlike all other interfaces, I²C is not able to communicate in full-duplex mode; it uses only two bidirectional open-drain lines, Serial Data Line (SDA) and Serial Clock Line (SCL), pulled up with resistors.

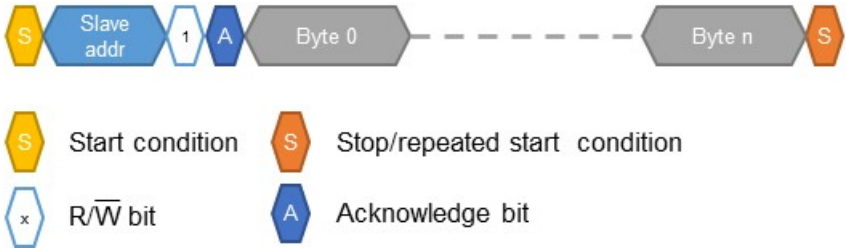
Teseo Module always acts as slave and it cannot initiate data on the bus; Host has to periodically pull the receiver to check about data availability. Default I²C slave address is 0x3A.

2.2.2.1 *I²C Read Access*

When the Host wants to read NMEA sentences from I²C, it must start a read operation over I²C, providing configured slave address.

After the acknowledge bit, a stream of bytes will be sent by Teseo Module up to the stop/repeated start condition.

Figure 6. I²C Read operation description



The format of the bytes is ASCII. When Teseo Module does not have any character to send, a dummy 0xFF byte is sent.

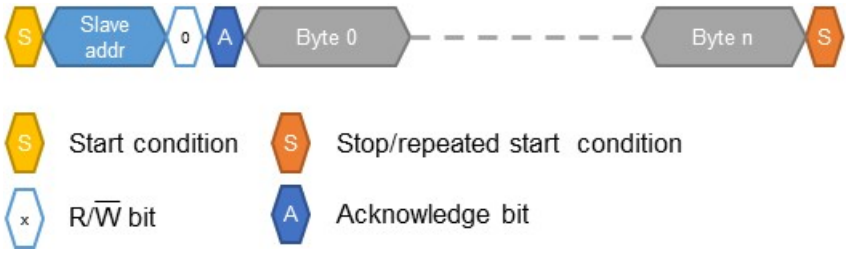
The Host can parse the data received as defined in ST GNSS NMEA specification and commands document.

2.2.2.2 *I²C Write Access*

When the Host wants to send commands to Teseo Module through I²C, it must start a write operation over I²C, providing configured slave address.

After the acknowledge bit, Teseo Module will receive any character coming from the Host up to the stop/repeated start condition.

Figure 7. I²C Write operation description



The format of the commands is defined in ST GNSS NMEA specification and commands document.

2.2.2.3 *I²C Register description*

On I²C communication channel Teseo Module allows 256 addressable registers.

On write operation:

- the first data byte is the register index while the following bytes are the register value;
- every write operation with less than 5 bytes is discarded;
- write operation with more than 5 bytes all the extra bytes not required are discarded;

Figure 8. I²C register write operation

On read operation:

- the register index is the last one indexed in a write operation;
- GNSS will send the 4 bytes register values;
- If the Host doesn't close the i2c-read-transaction after 4 bytes, extra bytes will be filled with 0x0 (zero) by the Teseo Module;

Figure 9. I²C register read operation

Teseo Module doesn't support auto-increment register index, this means, each register has to be addressed by the Host to access.

Registers from 0x0 to 0xFE are currently reserved for future use, every read or write operation on these registers can provide unpredictable operation on Teseo Module.

Teseo Module has a special register, the register 0xFF reports, as it is, the NMEA stream where the Host can perform read and write operations as a standard UART port.

On PowerOrReset the default register index value is 0xFF in this way every Host can read the NMEA stream directly just raising a simple i2c-read-operation on Teseo Module.

Table 4. I^2C registers map

Register id	Operation	Size	Description
0x00	-	32 bits	Reserved
…	-	-	-
0xFE	-	-	Reserved
0xFF	R/W	No-Limit	NMEA stream

3 Assisted GNSS

GNSS Teseo Module needs accurate satellite position data from at least 4 satellites to produce a position fix (FIX).

After that time a Teseo Module must download new ephemeris data.

Ephemeris download can take from dozens of seconds to several minutes, hours or can fail.

Assisted-GNSS is a mechanism to provide ephemeris assistance from external source, this reduces considerably the time to get a FIX especially in critical environments when the ephemeris download time could be very long.

Teseo-LIV4F supports one type of Assisted GNSS: RealTime GNSS.

3.1 RealTime AGNSS

The Real-Time AGNSS is able to provide the approximate current time, the ephemerides, the almanacs and optionally the approximate position to the GNSS engine in a time frame less than the usual time (about 30 seconds) needed to download real ephemeris from the sky. This reduces considerably the time to get fixed especially in critical environments when the ephemeris download time could be very long.

Real-time AGNSS requires a network connection to download assistance data from the server. Assistance data include the current time (if not available, for instance, from RTC), the ephemerides, the almanacs and optionally the rough position.

All the assistance data can be injected into the device backup memory using a few NMEA commands.

Once those data have been downloaded from the server, refer to the guidelines reported in the Application Note "AN5160: RxNetworks Assisted GNSS Server Interface Specification" to access the RxNetwork Service. The first thing to do is to inject the current time into the device (if the device has no RTC, or if it is set to a wrong time). This can be done either using the $PSTMINITTIME command or, if also the approximate position is available, then both current time and position can be injected using the $PSTMINITGPS command.

3.1.1 Password generation

As mentioned in the previous section, in order to access the RxNetworks servers, the user has to provide a set of parameters which are used in generating the HTTP request. These parameters are used to generate a password string (up to 41 characters in length) that is required by the HTTP request string.

GNSS device provides the $PSTMSTAGPS8PASSGEN NMEA command that performs the password generation. The user must supply three parameters to this command that it will be used to generate a unique password.

In order to generate the password the user must pass the following parameters:

- The vendor id string
- The current time expressed as GPS seconds (i.e., the number of seconds since midnight 06-Jan-1980)

The vendor id and device id strings will be provided by RxNetworks. The current time will be calculated by the software creating the HTTP request string.

3.1.2 Real-time assistance data uploading procedure

The real-time AGNSS performances depend on the availability of a network connection in order to download assistance data, which include:

- the current time (if not available, from instance, from RTC)
- the ephemerides
- the almanacs
- the rough position (optional)

Once those data have been downloaded from the server, the first thing to do is to inject the current time into the device (if the device has no RTC, or if it is set to a wrong time). This can be done either using the $PSTMINITTIME command or, if also the approximate position is available, then both current time and position can be injected using the $PSTMINITGPS command.

Then the ephemerides can be injected into the device using the $PSTMEPHEM command for each satellite (between two consecutive commands there must be at least a 20 millisecond delay).

Then the almanacs can be injected into the device using the $PSTMALMANAC command for each satellite (between two consecutive commands there must be at least a 20 millisecond delay).

Now the device will be capable of achieving the fix very quickly, if enough satellites are in view.

4 Communication interface

Communication between a host processor and the ST GNSS Teseo Module can be established in different ways, depending on the implementation of the Baseband Processor as a stand-alone unit or as an integrated subsystem on a "System on Chip".

For simplicity reasons this document will refer to "Stand-alone Processors" only and the interface described in the examples is a UART.

All information contained in this document is related to the "NMEA port" of the Baseband Processor. STMicroelectronics GNSS Teseo Module may contain an additional "Debug port" but the data exchanged on the "Debug Port" is not within the scope of this document.

4.1 Commands

A Command is a defined Data Packet which is sent from a host processor to the GPS-Baseband Controller in order to control the GPS system behaviour. The regular structure of a command is:

`command-ID,<parameters>*<checksum><cr><lf>`

In order to receive the commands, the GNSS Teseo Module is connected to the PC via the NMEA port (make sure that the serial cable is the right one, sometimes it is necessary to use a cross-cable). The user interaction can be achieved through the use of a PC terminal emulator that is connected to the appropriate COM port with settings in the following table:

Table 5. Default UART port configuration

Baudrate	Parity bits	Stop Bit	Data bits
115200	0	1	8

The NMEA default value baud rate is automatically set at the system start-up.

It can be modified at system runtime using the appropriate command.

The simplest way to send a command to the device is to write the command string in a text file and send it using the "send file" capability of the terminal emulator. For this reason, it is required that the terminal emulator (or production test program) running on the PC is capable of sending text files down the RS232 link to the GNSS Teseo Module.

Once the command is executed, the device replies with messages according to what specified in this document; after the message, the command is sent back to the host as final confirmation of the execution. This functionality can be configured according to what specified in the Firmware Configuration document.

4.2 Messages

A Message is a defined set of data sent from the GNSS Teseo Module to a host processor using the same interface which is used to transfer commands to the system. Messages may not be enabled by default but can be switched on and off using a command at run-time. The basic structure of a message is:

`message-ID,<parameters>*<checksum><cr><lf>`

There are two basic sets of messages implemented.

4.2.1 Standard NMEA messages

Standard NMEA Messages are defined in the "NMEA 0183" Standard, issued by the "National Marine Electronics Association".

To get an overview on the supported by ST's GNSS Teseo Module please refer to Standard NMEA messages specification.

Standard NMEA messages start the "message-ID" with:

`$<TalkerID>`

Supported talker IDs are: “GP”, “GL”, “GA”, “BD”, “QZ” and “GN” for standard NMEA sentences.

4.2.2 Proprietary messages

The STMicroelectronics GNSS Teseo Module can provide additional messages with more detailed data content. This is required to transmit GNSS and System information content which is not defined in the NMEA standard output.

Proprietary Messages from STMicroelectronics start with:

```
$PSTM...
```

To get an overview on the proprietary messages defined by STMicroelectronics please refer to ST NMEA messages specification.

5 Commands

ST NMEA proprietary command can modify the internal Teseo Module status, if not explicitly declared, all modifications of the status of the parameters, are not saved in the backup memory. For this reason, any changes of the parameters are replaced by the previous values after system reset or system power cycling.

5.1 ST NMEA command specification

5.1.1 $PSTMINITGPS

Initialize GPS position and time using UTC format. This command must be issued after a cold reset or it fails. The date issued with parameters Day, Month and Year must be later than January 2015, this threshold can be changed using the configuration options.

Synopsis:

```
$PSTMINITGPS,<Lat>,<LatRef>,<Lon>,<LonRef>,<Alt>,<Day>,<Month>,
<Year>,<Hour>,<Minute>,<Second>*<checksum><cr><lf>
```

Arguments:

Table 6. $PSTMINITGPS field description

Parameter	Format	Description
Lat	DDMM.MMM	Latitude (Degree-Minute.Minute decimals)
LatRef	'N' or 'S'	Latitude direction (North or South)
Lon	DDDMM.MMM	Longitude (Degree-Minute.Minute decimals)
LonRef	'E' or 'W'	Longitude Direction (East or West)
Alt	dddd – Decimal,4 digits	Altitude in meters (-1500 to 100000)
Day	dd – Decimal, 2 digits	Day of month (01 to 31)
Month	mm – Decimal, 2 digits	Month (01 to 12)
Year	YYYY – Decimal, 4 digits	Year (2015 - 2200)
Hour	HH – Decimal, 2 digits	Hour (00 to 23)
Minute	MM – Decimal, 2 digits	Minute (00 to 59)
Second	SS – Decimal, 2 digits	Second (00 to 59)

Results:

- The position and time will be initialized
- In case of no errors, the $PSTMINITGPSOK message is returned
- In case of errors, the error message $PSTMINITGPSERROR is returned

Example:

```
$PSTMINITGPS,4811.365,N,01164.123,E,0530,23,02,2015,09,44,12
```

5.1.2 $PSTMINITTIME

Initialize GPS time using UTC format. The date issued with parameters Day, Month and Year must be later than January 2015, this threshold can be changed using the configuration options.

Synopsis:

```
$PSTMINITTIME,<Day>,<Month>,<Year>,<Hour>,<Minute>,<Second>*<checksum><cr><lf>
```

Arguments:

Table 7. $PSTMINITTIME field description

Parameter	Format	Description
Day	dd – Decimal, 2 digits	Day of month (01 to 31)
Month	mm – Decimal, 2 digits	Month (01 to 12)
Year	YYYY – Decimal, 4 digits	Year (2015 - 2200)
Hour	HH – Decimal, 2 digits	Hour (00 to 23)
Minute	MM – Decimal, 2 digits	Minute (00 to 59)
Second	SS – Decimal, 2 digits	Second (00 to 59)

Results:

- The position and time will be initialized
- In case of no errors, the $PSTMINITTIMEOK message is returned
- In case of errors, the error message $PSTMINITTIMEERROR is returned

Example:

```
$PSTMINITTIME,23,02,2015,09,44,12
```

5.1.3 $PSTMINITFRQ

Initialize the centre frequency. This command can be used to set the local oscillator frequency offset.

Synopsis:

```
$PSTMINITFRQ,<offset>*<checksum><cr><lf>
```

Arguments:

Table 8. $PSTMINITFRQ field description

Parameter	Format	Description
offset	Decimal, 6 digits	Frequency offset in Hz

Results:

- The center frequency will be initialized

Example:

```
$PSTMINITFRQ,-47000*<checksum><cr><lf>
```

5.1.4 $PSTMDUMPEPHEMS

This command sends out all ephemeris stored in the backup memory.

Synopsis:

```
$PSTMDUMPEPHEMS*<checksum><cr><lf>
```

Arguments:

None.

Results:

- GNSS replies with the $PSTMEPHEM messages

Example:

```
$PSTMDUMPEPHEMS
$PSTMEPHEM,1,64,0f06bc34bc345f5f5f84f400dea4ff00f9f63c239f0a35f81400fbff33420000ee632f27698ef
001afa50da16cfcfa22e0b65a3e7a3cee27d700f7ffc616fe03*57
$PSTMEPHEM,2,64,0f06bc34bc344f4f4f78110019a5ff00b004fa1d1e0e3f04c8ffcaff1937000033515726556ba
9048eae0da1b6c346bd8f985c93ade10c76db001d00f8c7c503*58
$PSTMEPHEM,4,64,0f06bb34bb344b4b4b98050038a4ff000005351e110eea041b00b8ffd037000020b84e26b5138
b0425580ca16b211030e68b1a949cac9615f30066ffea92f603*06
$PSTMEPHEM,9,64,0f06bc34bc341818189c0a0069aaff005f06eb249a09ca0477ff6c00f72e00005131d827592b9
50a91010da1c7af88538e7ca1122fb9be3df4001300c4a0c203*52
```

5.1.5 $PSTMEPHEM

This command allows the user to load the ephemeris data into backup memory.

If more than one $PSTMEPHEM command needs to be issued, between two consecutive commands there must be at least a 20 ms delay.

Synopsis:

```
$PSTMEPHEM,<sat_id>,<N>,<byte1>,...,<byteN>*<checksum><cr><lf>
```

Arguments:

Table 9. $PSTMEPHEM field description

Parameter	Format	Description
sat_id	Decimal, 2 digits	Satellite number
N	Decimal, 1 digit	Number of the ephemeris data bytes
byte1	Hexadecimal, 2 digits	First byte of the ephemeris data
byteN	Hexadecimal, 2 digits	Last byte of the ephemeris data

The N Bytes that are in the parameters are the dump of structures that contain all the information of the ephemeris.

Data format is constellation dependent.

Table 10. $PSTMEPHEM field description for GPS constellation

Bits	Structure Member	Description
16	week	Week number of the Issue of Data
16	toe	Time of week for ephemeris epoch
16	toc	Time of week for clock epoch
8	iode1	Issue of data 1
8	iode2	Issue of data 2
10	iodc	Issue of data clock
14	i_dot	Rate of inclination angle
8	RESERVED	
24	omega_dot	Rate of right ascension
8	RESERVED	Must be 0
16	crs	Amplitude of the sine harmonic correction to the orbit radius
16	crc	Amplitude of the cosine harmonic correction to the orbit radius
16	cus	Amplitude of the sine harmonic correction to the argument of latitude
16	cuc	Amplitude of the cosine harmonic correction to the argument of latitude
16	cis	Amplitude of the sine harmonic correction to the angle of inclination

Bits	Structure Member	Description
16	cic	Amplitude of the cosine harmonic correction to the angle of inclination
16	motion_difference	Mean motion difference from computed value
16	RESERVED	Must be 0
32	inclination	Inclination angle at reference time
32	e	Eccentricity
32	root_A	Square root of major axis
32	mean_anomaly	Mean anomaly at reference time.
32	omega_zero	Longitude of ascending node of orbit plane at weekly epoch
32	perigee	Argument of perigee
8	time_group_delay	Estimated group delay differential
8	af2	Second order clock correction
16	af1	First order clock correction
22	af0	Constant clock correction
1	RESERVED	RESERVED for use by GNSS library – must be 1
1	RESERVED	RESERVED for use by GNSS library – must be 1
1	RESERVED	RESERVED for use by GNSS library – must be 1
1	available	Contains 1 if ephemeris is available, 0 if not
1	health	Contains 1 if the satellite is unhealthy, 0 if healthy
1	RESERVED	Must be 0
4	accuracy	Accuracy

Table 11. $PSTMEPHEM field description for GLONASS constellation

Bits	Structure Member	Description
16	week	Week number of the Issue of Data
16	toe	Time of week for ephemeris epoch
4	toe_lsb	Time of week for ephemeris epoch (LBS)
11	NA	Calendar day number within the four-year period since the beginning of last leap year (almanac)
7	tb	Time of ephemeris index
2	M	Type of satellite 00=GLONASS 01=GLONASS-M
2	P1	Time interval between two adjacent tb parameters
1	P3	Number of satellites for which almanac is transmitted within this frame 0=4 1=5
1	P2	Flag of oddness ("1") or evenness ("0") of the value of tb
1	P4	Flag to show that ephemeris parameters are present
2	KP	Notification on forthcoming leap second correction of UTC
1	RESERVED	
27	xn	Satellite PZ-90 x coordinate at epoch tb
5	xn_dot_dot	Satellite PZ-90 x velocity at epoch tb
24	xn_dot	Satellite PZ-90 x acceleration component at epoch tb
5	n	Slot number (1\xc9 24)
3	Bn	Healthy flags

Bits	Structure Member	Description
27	yn	Satellite PZ-90 y coordinate at epoch tb
5	yn_dot_dot	Satellite PZ-90 y acceleration component at epoch tb
24	yn_dot	Satellite PZ-90 y velocity at epoch tb
8	age_h	Age of predicted ephemeris (hours)
27	zn	Satellite PZ-90 z coordinate at epoch tb
5	zn_dot_dot	Satellite PZ-90 z acceleration component at epoch tb
24	zn_dot	Satellite PZ-90 z velocity at epoch tb
8	RESERVED	Must be 0
11	gamma_n	Satellite clock frequency drift at epoch tb
5	E_n	Age of the ephemeris information
4	freq_id	Frequency ID
12	RESERVED	
22	tau_n	Satellite clock correction at epoch tb
10	RESERVED	Must be 0
32	tau_c	GLONASS to UTC(SU) time correction
22	tau_GPS	GLONASS to GPS system time correction
10	RESERVED	
11	NT	Calendar day number of ephemeris within the four-year period since the beginning of last leap year
5	N4	Four-year interval number starting from 1996
12	tk	Satellite time referenced to the beginning of the frame
4	FT	Predicted satellite user range accuracy at time tb
32	RESERVED	
5	m_available	Must be 0x1F
1	nvm_reliable	Must be 1
26	spare	
25	RESERVED	
1	available	Contains 1 if ephemeris is available, 0 if not
1	health	Contains 1 if the satellite is unhealthy, 0 if healthy
1	RESERVED	Must be 0
4	RESERVED	

Table 12. $PSTMEPHEM field description for Galileo constellation

Bits	Structure Member	Description
16	week	Week number of the Issue of Data
14	toe	Time of week for ephemeris epoch
2	RESERVED	
16	toc	Time of week for clock epoch
10	iod_nav	Issue of data
8	SISA	Signal In Space Accuracy
10	RESERVED	Must be 0

Bits	Structure Member	Description
10	BGD_E1_E5a	E1-E5a Broadcast Group Delay
10	BGD_E1_E5b	E1-E5b Broadcast Group Delay
2	E1BHS	E1-B Signal Health Status
32	inclination	Inclination angle at reference time
32	eccentricity	Eccentricity
32	root_a	Square root of major axis
32	mean_anomaly	Mean anomaly at reference time
32	omega_zero	Longitude of ascending node of orbit plane at weekly epoch
32	perigee	Argument of perigee
14	i_dot	Rate of inclination angle
1	available	Contains 1 if ephemeris is available, 0 if not
1	health	Contains 1 if the satellite is unhealthy, 0 if healthy
16	motion_difference	Mean motion difference from computed value
16	crs	Amplitude of the sine harmonic correction to the orbit radius
16	crc	Amplitude of the cosine harmonic correction to the orbit radius
16	cus	Amplitude of the sine harmonic correction to the argument of latitude
16	cuc	Amplitude of the cosine harmonic correction to the argument of latitude
16	cis	Amplitude of the sine harmonic correction to the angle of inclination
16	cic	Amplitude of the cosine harmonic correction to the angle of inclination
24	omega_dot	Rate of right ascension
6	SVID	Satellite Identification
1	E1BDVS	E1-B Data Validity Status
1	RESERVED	Must be 0
8	RESERVED	Must be 0
16	RESERVED	Must be 0
6	af2	Second order clock correction
21	af1	First order clock correction
5	word_available	Must be 0x1F
31	af0	Constant clock correction
1	RESERVED	
6	RESERVED	Must be 0
26	RESERVED	RESERVED for use by GNSS library – must be 1
1	RESERVED	Must be 0

Table 13. $PSTMEPHEM field description for BEIDOU constellation

Bits	Structure Member	Description
32	inclination	Inclination angle at reference time
32	eccentricity	Eccentricity
32	root_a	Square root of major axis
32	mean_anomaly	Mean anomaly at reference time

Bits	Structure Member	Description
32	omega_zero	Longitude of ascending node of orbit plane at weekly epoch
32	perigee	Argument of perigee
17	toe	Time of week for ephemeris epoch
10	time_group_delay	Estimated group delay differential
5	aode	Issue of data, ephemeris
24	omega_dot	Rate of right ascension.
8	A0	Ionospheric Delay Model Parameter α_0
24	af0	Constant clock correction.
8	A1	Ionospheric Delay Model Parameter α_1
20	sow	Seconds of week
11	af2	Second order clock correction
1	is_geo	1 for Geostationary satellites, otherwise 0
22	af1	First order clock correction
10	subframe_avail	Must be 0x3FF
16	motion_difference	Mean motion difference from computed value
8	A2	Ionospheric Delay Model Parameter α_2
8	A3	Ionospheric Delay Model Parameter α_3
18	crs	Amplitude of the sine harmonic correction to the orbit radius
8	B2	Ionospheric Delay Model Parameter β_2
4	urai	User range accuracy index
2	RESERVED	Must be 0
18	crc	Amplitude of the cosine harmonic correction to the orbit radius
8	B3	Ionospheric Delay Model Parameter β_3
5	aodc	Issue of data, clock
1	spare	
18	cus	Amplitude of the sine harmonic correction to the argument of latitude
14	i_dot	Rate of inclination angle
18	cuc	Amplitude of the cosine harmonic correction to the argument of latitude
8	B0	Ionospheric Delay Model Parameter β_0
6	spare	
18	cis	Amplitude of the sine harmonic correction to the angle of inclination
8	B1	Ionospheric Delay Model Parameter β_1
6	RESERVED	Must be 0
18	cic	Amplitude of the cosine harmonic correction to the angle of inclination
1	nvm_reliable	Must be 1
11	RESERVED	Must be 0
2	spare	
17	toc	Time of week for clock epoch
13	week	Week number of the Issue of Data
1	available	Contains 1 if ephemeris is available, 0 if not

Bits	Structure Member	Description
1	health	Contains 1 if the satellite is unhealthy, 0 if healthy

Results:

- The ephemeris will be stored into backup RAM.
- In case of no errors, the $PSTMEPHEMOK message is returned.
- In case of errors, the error message $PSTMEPHEMERROR is returned.

Example:

```
$PSTMEPHEM,12,64,0f06bc34bc3437373790f40045a7ff00fcf5d522480b4bf71b00fbff8931000096126f271f869101c3870ca107afce79a763e13e360a1ce8e7003100380ff903*36
```

5.1.6 $PSTMCOLD

Perform a COLD start.

Synopsis:

```
$PSTMCOLD,<Mask>*<checksum><cr><lf>
```

Arguments:

Table 14. $PSTMCOLD field description

Parameter	Format	Description
Mask	Integer	Optional parameter to invalidate time, position, ephemeris and almanac : 0x1 – clear almanac 0x2 – clear ephemeris 0x4 – clear position 0x8 – clear time

Results:

- Coldstart initialization and system restart(1).
- If Mask parameter is used, only the selected GPS data is invalidated for this actual Coldstart. Multiple selects are supported (i.e. 0xD).
- If Mask parameter is not used, default is 0xE (clear ephemeris, time and position).

1. *The GPS engine will be reset. It is not a system reboot.*

Example:

```
$PSTMCOLD,6
```

5.1.7 $PSTMWARM

Perform a WARM start.

Synopsis:

```
$PSTMWARM*<checksum><cr><lf>
```

Arguments:

None.

Results:

- Warm start initialization and system restart(1).

1. *The GPS engine will be reset. It is not a system reboot.*

Example:

```
$PSTMWARM*<checksum><cr><lf>
```

5.1.8 $PSTMHOT

Perform a HOT start.

Synopsis:

```
$PSTMHOT*<checksum><cr><lf>
```

Arguments:

None.

Results:

- The system restarts(1).

1. *The GPS engine will be reset. It is not a system reboot.*

Example:

```
$PSTMHOT*<checksum><cr><lf>
```

5.1.9 $PSTMSRR

Executes a system reset. The GNSS firmware is rebooted.

Synopsis:

```
$PSTMSRR*<checksum><cr><lf>
```

Arguments:

None.

Results:

- The GNSS firmware reboots
- No message will be sent as a reply

Example:

```
$PSTMSRR*<checksum><cr><lf>
```

5.1.10 $PSTMGPSRESET

Reset the GNSS Teseo engine.

Synopsis:

```
$PSTMGPSRESET*<checksum><cr><lf>
```

Arguments:

None.

Results:

- The GNSS Teseo engine will be reset
- No message will be sent as a reply

Note: Using this command the GNSS module won't reboot.

Example:

```
$PSTMGPSRESET*<checksum><cr><lf>
```

5.1.11 $PSTMGETSWVER

Get the version string of the libraries embedded in the software application.

Synopsis:

```
$PSTMGETSWVER,<id>*<checksum><cr><lf>
```

Arguments:

Table 15. $PSTMGETSWVER field description

Parameter	Format	Description
id	Integer	Depending on the value of the <lib_id> parameter, the following version numbering is delivered by the command: 1 = OS20 Version 2 = SDK App Version 11 = SW configuration ID 254 = configuration data block 255 = all versions strings (as reported at the NMEA startup).

Results:

- GNSS replies with $PSTMVER message

5.1.12 $PSTMRFTESTON

Enable the RF test mode for production line tests.

Synopsis:

```
$PSTMRFTESTON,<sat_id>*<checksum><cr><lf>
```

Arguments:

Table 16. $PSTMRFTESTON field description

Parameter	Format	Description
sat_id	Decimal, 2 digits	Satellite number

Results:

- The GPS engine will restart in the RF test modality. This RF test forces the GPS to acquire the process only on the provided satellite's id. It could be useful to reduce the RF testing time in the production line where generally a single channel simulator is present

Example:

```
$PSTMRFTESTON,24*<checksum><cr><lf>
```

5.1.13 $PSTMRFTESTOFF

Disable the RF test mode for production line tests.

Synopsis:

```
$PSTMRFTESTOFF*<checksum><cr><lf>
```

Arguments:

None.

Results:

- The RF test modality will be disabled and the GNSS engine will be restarted.

Note: The RF test mode can be disabled also resetting the GNSS module.

Example:

```
$PSTMRFTESTOFF*<checksum><cr><lf>
```

5.1.14 $PSTMNMEAREQUEST

Send a set of NMEA messages according to the input message list as specified in the FW Configuration document.

Synopsis:

```
$PSTMNMEAREQUEST,<msglist_l>,<msglist_h>*<checksum><cr><lf>
```

Arguments:

Table 17. $PSTMNMEAREQUEST field description

Parameter	Format	Description
msglist_l	Hexadecimal, 1 Digit	First 32 bits of 64 bits message list (low). Each bit is used to enable/disable a specific message. 0 = disabled 1 = enabled
msglist_h	Hexadecimal, 1 Digit	Second 32 bits of 64 bits message list (high). Each bit is used to enable/disable a specific message. 0 = disabled 1 = enabled

Results:

A set of NMEA messages is sent according to the input message list.

Note: The order of NMEA messages in the message list is the same as for the periodic NMEA output messages.

5.1.15 $PSTMFORCESTANDBY

Force the platform to go in standby mode.

Synopsis:

```
$PSTMFORCESTANDBY,<duration>*<checksum><cr><lf>
```

Arguments:

Table 18. $PSTMFORCESTANDBY field description

Parameter	Format	Description
duration	Decimal, 5 digits	Duration of the standby time in seconds

Results:

- In case of no errors, the $PSTMFORCESTANDBYOK message is returned
- In case of errors, the error message $PSTMFORCESTANDBYERROR is returned

5.1.16 $PSTMGETUCODE

This command reads the unique code from the secondary boot flash memory partition.

Synopsis:

```
$PSTMGETUCODE*<checksum><cr><lf>
```

Arguments:

None

Results:

- In case of no error the $PSTMGETUCODEOK message is sent
- In case of error the $PSTMGETUCODEERROR message is sent

5.2 ST system configuration commands

The GNSS Software utilizes a "Configuration Data Block" that holds the working parameters for the system. The parameters can be set, read or stored (in NVM) using the system configuration commands: $PSTMSETPAR, $PSTMGETPAR and $PSTMSAVEPAR. There is also a command to restore the factory setting parameters: $PSTMRESTOREPAR.

At run-time it could be possible to have up to three different configuration blocks:

- Current configuration: it is placed in the RAM memory and it includes the current configuration of each parameter. This configuration block can be modified with the $PSTMSETPAR command. The $PSTMSAVEPAR command stores the current configuration data block into the NVM memory. At startup the current configuration block is loaded from NVM (if a stored data block is available) or it is loaded from the default one embedded in the code (factory settings).
- Default configuration: it is generally placed in the flash/rom memory. It includes the factory setting for each parameter. This configuration is used at system startup if there is no configuration data into the NVM memory.
- NVM stored configuration: it is available in the NVM backup memory as soon as the $PSTMSAVEPAR command is executed. It includes all parameters modified and stored by the user. At system startup the SW configuration management checks if a valid configuration block is available in the NVM backup memory. In case the stored configuration is available, it will be used for system configuration. If not available the default setting will be used.

Note: Other "Configuration Data Block" parameters not documented in this manual must be considered as RESERVED and must not be modified. Modifying any other parameter intentionally or unintentionally may stop the system from working and/or degrade the system performance.

5.2.1 $PSTMSETPAR

This command sets the defined parameter (indicated by "ID") to the value provided as "param_value" in the commands parameter.

Synopsis:

```
$PSTMSETPAR,<ConfigBlock><ID>,<param_value>[,<mode>]*<checksum><cr><lf>
```

Arguments:

Table 19. $PSTMSETPAR field description

Parameter	Format	Description
ConfigBlock	Decimal,1 digit	Indicates one of the configuration blocks: 1=Current Configuration, 2 = Default Configuration, 3 = NVM Stored configuration.
ID	Decimal, 3 digits	ID - Identifier (see Configuration Data Block as described in FW Configuration document)

Parameter	Format	Description
param_value	1 up to 80 bytes	Parameter to be set, see "Allowed values" as described in FW Configuration document.
mode	Decimal, 1 digit	This parameter is optional. It allows to perform bit-to-bit "OR" or "AND" operations between the selected parameter in the configuration block and the param_value in input. It has the following meaning: 0: the parameter in the configuration block is overwritten by the param_value. This is the default action as in the case mode is omitted. 1: the parameter in the configuration block is the result of bit-to-bit "OR" between old value and the param_value.This is useful for bit mask setting. 2: the parameter in the configuration block is the result of bit-to-bit "AND" between old value and NOT(param_value). This is useful for bit mask resettimg.

Results:

- In case of no errors, the $PSTMSETPAROK message is returned
- In case of errors, the error message $PSTMSETPARERROR is returned

Example:

Issuing the command:

```
$PSTMSETPAR,1121,10*<checksum><cr><lf>
```

You could have this answer:

```
$PSTMSETPAROK,1121*<checksum><cr><lf>
```

Note: The configuration block parameter is ignored by the "SET" command because only the current configuration, stored in the RAM memory, can be written. It is used only to keep the same syntax as for the "GET" command. The configuration block stored in NVM will be overwritten by the current configuration after the $PSTMSAVEPAR command.

There is no comma and no space between ConfigBlock and ID parameters.

The input param_value must be expressed in hexadecimal format without "0x" prefix for any integer value except DOP configuration. It must be decimal for any not integer value and DOP setting.

5.2.2 $PSTMGETPAR

This command reads the defined parameter (indicated by "ID") from the "Configuration Data Block" and returns it as a specific message.

Synopsis:

```
$PSTMGETPAR,<ConfigBlock><ID>*<checksum><cr><lf>
```

Arguments:

Table 20. $PSTMGETPAR field description

Parameter	Format	Description
ConfigBlock	Decima1, 1 digit	Indicates one of configuration blocks: 1 = Current Configuration, 2 = Default Configuration, 3 = NVM Stored configuration.
ID	Decimal, 3 digits	ID - Identifier (see Configuration Data Block)

Results:

- In case of no errors, $PSTMSETPAR message is sent
- In case of errors, the error message $PSTMGETPARERROR is returned

Example:
Issuing the command:

```
$PSTMGETPAR,1403*<checksum><cr><lf>
```

You could have this answer:

```
$PSTMSETPAR,1403,15,12,12,18*<checksum><cr><lf>
```

Note: There is no comma and no space between ConfigBlock and ID parameters.
In case of no errors the answer is deliberately $PSTMSET and not $PSTMGET.

5.2.3 $PSTMSAVEPAR

Save current configuration data block into the backup memory.
Synopsis:

```
$PSTMSAVEPAR*<checksum><cr><lf>
```

Arguments:
None.
Results:

- The current configuration data block, including changed parameters, will be stored into the backup memory (NVM).
- In case of no errors, the $PSTMSAVEPAROK message is returned
- In case of errors, the error message $PSTMSAVEPARERROR is returned

Note: The factory setting parameters can be restored using the $PSTMRESTOREPAR command.
Example:

```
$PSTMSAVEPAR*<checksum><cr><lf>
```

5.2.4 $PSTMRESTOREPAR

Restore the factory setting parameters. The configuration data block stored in NVM, if present, will be invalidated. Any changed parameter will be lost.
Synopsis:

```
$PSTMRESTOREPAR*<checksum><cr><lf>
```

Arguments:
None.
Results:

- The factory setting parameters will be restored and the configuration block in the backup memory will be lost. A system reboot is needed to complete the factory reset restoring and to get system working with default setting.
- In case of no errors, the $PSTMRESTOREPAROK message is returned
- In case of errors, the error message $PSTMRESTOREPARERROR is returned

Example:

```
$PSTMRESTOREPAR*<checksum><cr><lf>
```

5.3 Real Time AGNSS NMEA commands

5.3.1 $PSTMSTAGPS8PASSGEN

Request the generation of a password to access the Real-Time AGPS server to the device.

Synopsis:

```
$PSTMSTAGPS8PASSGEN,<time>,<VendorID>,<ModelID>*<checksum><cr><lf>
```

Arguments:

Table 21. $PSTMSTAGPS8PASSGEN field description

Parameter	Description
<time>	GPS time in seconds (i.e.: the current time expressed in the number of seconds since midnight 06-Jan-1980).
<VendorID>	Unique Vendor ID
<ModelID>	Model identifier

Results:

ST GNSS Teseo Module returns the password in the message $PSTMSTAGPS8PASSRTN.

6 Messages

6.1 Standard NMEA messages list

Table 22. Standard NMEA messages list

Syntax	Default	Description
$--GNS	ON	NMEA: Global Position System Fix Data
$GPGGA	ON	NMEA: Global Position System Fix Data
$GPGLL	OFF	NMEA: Geographic Position Latitude/Longitude
$--GSA	ON	NMEA: GPS DOP and Active Satellites. "GP", "GL" and "GN" talker ID are supported according to the software configuration.
$--GSV	ON	NMEA: GPS Satellites in View. "GP", "GL" and "GN" talker ID are supported according to the software configuration.
$GPRMC	ON	NMEA: Recommended Minimum Specific GNSS Data
$GPVTG	OFF	NMEA: Track made good and ground speed
$GPZDA	OFF	NMEA: Time and Date
$GPGST	ON	NMEA: GNSS Pseudorange Noise Statistics

6.2 Preliminary notes about satellites' PRN ranges

The satellite PRN is an ID used to identify satellites. In NMEA 0183 Rev 4.10, PRN was not described for new constellation.

Table 23. Satellite PRNs for each NMEA version

	GPS	SBAS	GLONASS	BAIDEU	QZSS	GALILEO
NMEA 4.10	from 1 to 32	from 33 to 64	from 65 to 99	from 1 to 32	from 1 to 32	from 1 to 36

6.3 Standard NMEA messages specification

These messages are defined within the "NMEA 0183" Specification.

6.3.1 $--GGA

Global Positioning System Fixed data

NMEA message list bitmask (64 bits): 0000 0000 0000 0002

Synopsis:

```
$<TalkerID>GGA,<Timestamp>,<Lat>,<N/S>,<Long>,<E/W>,<GPSQual>,<Sats>,<HDOP>,<Alt>,<AltVal>,<GeoSep>,<GeoVal>,<DGPSAge>,<DGPSRef>*<checksum><cr><lf>
```

Arguments:

Table 24. $--GGA message field description

Parameter	Format	Description
TalkerID	String, 2 characters	The talker ID (fixed two characters). GP: If system works in GPS only mode GL: If system works in GLONASS only mode GA: If system works in GALILEO only mode BD: If system works in BEIDOU only mode QZ: If system works in QZSS only mode GN: If system works in multiconstellation mode.
Timestamp	hhmmss.sss	UTC Time of GPS Sample: hh: hours (Fixed two digits) mm: minutes (Fixed two digits) ss: seconds (Fixed two digits) .sss: decimal fraction of seconds (variable length) Note that decimal fraction assumes non zero values when the fix rate is bigger than 1 Hz.
Lat	DDMM.MMMMM	Latitude as degrees: DD: Degree (Fixed two digits) MM: Minutes (Fixed two digits) .MMMMM: Decimal fraction of minutes (Variable)
N/S	"N" or "S"	Latitude direction: North or South Note that for Rev 4.10 this field is empty in case of invalid value
Long	DDMM.MMMMM	Longitude as degrees: DD: Degree (Fixed two digits) MM: Minutes (Fixed two digits) .MMMMM: Decimal fraction of minutes (Variable)
E/W	"E" or "W"	Longitude direction: East or West
GPSQual	Decimal, 1digit	0 = Fix not available or invalid 1 = GPS, SPS Mode, fix valid 2 = Differential GPS, SPS Mode, fix valid 6 = Estimated (dead reckoning) mode
Sats	Decimal, 2 digits	Satellites in use: example: 8
HDOP	Decimal, 3 digits	Horizontal Dilution of Precision, max: 99.0
Alt	Decimal, 6 digits	Height above mean sea level, max: 100000m
AltVal	"M"	Reference Unit for Altitude ("M" = meters)
GeoSep	Decimal, 4 digits	Geoidal Separation measure in "M" = meters
GeoVal	"M"	Reference Unit for GeoSep ("M" = meters)
DGPSAge	Empty	Not supported
DGPSRef	Empty	Not supported
Checksum	Hexadecimal, 2 digits	Checksum of the message bytes without *<checksum><cr><lf> characters.

Example:

```
$GPGGA,183417.000,04814.03970,N,01128.52205,E,0,00,99.0,495.53,M,47.6,M*53
```

6.3.2 $--GLL

Geographic Positioning Latitude / Longitude
NMEA message list bitmask (64 bits): 0000 0000 0010 0000
Synopsis:

```
$<TalkerID>GLL,<Lat>,<N/S>,<Long>,<E/W>,<Timestamp>,<Status>,<mode
indicator>*<checksum><cr><lf>
```

Arguments:

Table 25. $--GLL message field description

Parameter	Format	Description
TalkerID	String, 2 characters	The talker ID (Fixed two characters). GP: If system works in GPS only mode GL: If system works in GLONASS only mode GA: If system works in GALILEO only mode BD: If system works in BEIDOU only mode QZ: If system works in QZSS only mode GN: If system works in multi-constellation mode.
Lat	DDMM.MMMMM	Latitude as degrees: DD: Degree (Fixed two digits) MM: Minutes (Fixed two digits) .MMMMM: Decimal fraction of minutes (Variable)
N/S	"N" or "S"	Latitude direction: North or South
Long	DDDMM.MMMMM	Longitude as degrees: DDD: Degree (Fixed two digits) MM: Minutes (Fixed two digits) .MMMMM: Decimal fraction of minutes (Variable)
E/W	"E" or "W"	Longitude direction: East or West
Timestamp	hhmmss.sss	UTC Time of GGL Sample, example: 160836 ".sss" is the fraction of seconds; it assumes non zero values when the fix rate is bigger than 1Hz.
Status	"A" or "V"	Validity of Data "A" = valid, "V" = invalid
Mode indicator	"D", "A", "N" or "E"	Positioning system Mode Indicator: "D" = Differential mode "A" = Autonomous mode "N" = data not valid "E" = Estimated (dead reckoning) mode
checksum	Hexadecimal,2 digits	Checksum of the message bytes without *<checksum><cr><lf> characters.

Example:

```
$GPGLL,4055.04673,N,01416.54941,E,110505.000,A,A*54
```

6.3.3 $--GSA

GNSS DOP and Active Satellites. Satellites from different constellations are sent on separate messages.
In case of multi-constellation mode, the talker ID is always GN.

NMEA message list bitmask (64 bits): 0000 0000 0000 0004

Synopsis:

```
$--GSA,<Mode>,<CurrentMode>,<SatPRN1>,...,<SatPRNN>,<PDOP>,<HDOP>,<VDOP>,<SystemID>*<checksum
><cr><lf>
```

Arguments:

Table 26. $--GSA message field description

Parameter	Format	Description
TalkerID	String, 2 characters	The talker ID (Fixed two characters). GP: If system works in GPS only mode GL: If system works in GLONASS only mode GA: If system works in GALILEO only mode BD: If system works in BEIDOU only mode QZ: If system works in QZSS only mode GN: If system works in multi-constellation mode.
CurrentMode	Decimal, 1 digit	Current Mode: 1 = Fix not available or invalid 2 = GPS, SPS Mode, fix valid 3 = Differential GPS, SPS Mode, fix valid
SatPRN(1 to 12)	Decimal, 2 or 3 digits	Satellites list used for positioning. See Preliminary notes about satellites' PRN ranges for more info about available values.
PDOP	Decimal, 3 digits	Position Dilution of Precision, max: 99.0
HDOP	Decimal, 3 digits	Horizontal Dilution of Precision, max: 99.0
VDOP	Decimal, 3 digits	Vertical Dilution of Precision, max: 99.0
SystemID	Hexadecimal, 1 digit	The system ID of this message: 1 = GPS 2 = GLONASS 3 = GALILEO 4 = BEIDOU 5 = QZSS
checksum	Hexadecimal, 2 digits	Checksum of the message bytes without *<checksum><cr><lf> characters.

Example:

```
$GNGSA,A,3,23,03,22,09,01,19,17,06,31,11,,,1.1,0.6,0.9,1*3E
```

```
$GNGSA,A,3,67,66,81,65,88,75,82,74,,,,, 1.1,0.6,0.9,2*3D
```

```
$GNGSA,A,3,03,05,22,08,30,16,12,,,,,, 1.1,0.6,0.9,3*32
```

6.3.4 $--GSV

GNSS Satellites in View.

Usually GSV messages are organized per constellation and each message carries information about up to 4 satellites in view. Thus, in certain cases, to describe all the satellites in view from a constellation more than a message is needed. This set of message is printed once per each constellation with talker ID related to described constellation.

NMEA message list bitmask (64 bits): 0000 0000 0008 0000

Synopsis:

```
$--GSV,<GSVAmount>,<GSVNumber>,<TotSats>,<Sat1PRN>,<Sat1Elev>,<Sat1Azim>,<Sat1CN0>,…,<Sat4PRN>,<Sat4Elev>,<Sat4Azim>,<Sat4CN0>,<SignalID>*<checksum><cr><lf>
```

Arguments:

Table 27. $--GSV message field description

Parameter	Format	Description
TalkerID	String, 2 characters	The talker ID (Fixed two characters). GP: If system works in GPS only mode GL: If system works in GLONASS only mode GA: If system works in GALILEO only mode BD: If system works in BEIDOU only mode QZ: If system works in QZSS only mode GN: If system works in multi-constellation mode.
GSVAmount	Decimal, 1 digit	Total amount of GSV messages
GSVNumber	Decimal, 1 digit	Continued GSV number of this message
TotSats	Decimal, 2 digits	Total Number of Satellites in view, max. 32
SatxPRN	Decimal, 2 digits	Satellites list used for positioning.
SatxElev	Decimal, 2 digits	Elevation of satellite x in Degree, 0 \xc9 90
SatxAzim	Decimal, 3 digits	Azimuth of satellite x in degree, ref. "North", 000 \xc9 359
SatxCN0	Decimal, 2 digits	Carrier to Noise Ratio for satellite x in dB, 00 \xc9 99
SignalID	Decimal, 1 digits	An identifier to indicate the signal in use. Currently it is 1 for GPS, GLONASS, 2 for BEIDOU and QZSS 6 for GALILEO
checksum	Hexadecimal, 2 digits	Checksum of the message bytes without *<checksum><cr><lf> characters.

Example:

```
$GPGSV,3,1,09,30,68,039,49,05,61,266,50,28,52,137,47,07,38,052,48,01*5C
```

```
$GPGSV,3,2,09,13,37,301,45,09,17,105,43,15,07,297,40,08,06,056,41,01*56
```

```
$GPGSV,3,3,09,20,,,41,,,,,,,,,,,,01*5A
```

```
$GLGSV,2,1,06,68,86,031,43,78,78,013,46,79,51,226,43,69,33,325,38,01*43
```

```
$GLGSV,2,2,06,67,33,139,41,77,26,035,36,,,,,,,,,01*46
```

```
$GAGSV,2,1,05,08,76,129,44,02,65,057,46,30,56,205,45,07,48,311,44,06*4F
```

```
$GAGSV,2,2,05,03,22,129,40,,,,,,,,,,,,06*7D
```

6.3.5 $--RMC

Recommended Minimum Specific GPS/Transit data. Time, date, position and speed data provided by the GNSS Teseo. This sentence is transmitted at intervals not exceeding 2 seconds and is always accompanied by RMB when destination way point is active.

- NMEA message list bitmask (64 bits): 0000 0000 0000 0040

Synopsis:

```
$<TalkerID>RMC,<Timestamp>,<Status>,<Lat>,<N/S>,<Long>,<E/
W>,<Speed>,<Trackgood>,<Date>,<MagVar>,<MagVarDir>,<mode>,<Nav_status>*<checksum><cr><lf>
```

Arguments:

Table 28. $--RMC message field description

Parameter	Format	Description
TalkerID	String, 2 characters	The talker ID (Fixed two characters). GP: If system works in GPS only mode GL: If system works in GLONASS only mode GA: If system works in GALILEO only mode BD: If system works in BEIDOU only mode QZ: If system works in QZSS only mode GN: If system works in multi-constellation mode.
Timestamp	hhmmss.sss	UTC Time of GPS Sample: hh: hours (Fixed two digits) mm: minutes (Fixed two digits) ss: seconds (Fixed two digits) .sss: decimal fraction of seconds (Variable length) Note that decimal fraction assumes non zero values when the fix rate is bigger than 1 Hz.
Status	"A" or "V"	Teseo warning: "A" = valid, "V" = Warning Note that "V" is reported in NO FIX conditions and "A" is reported in 2D and 3D fix conditions.
Lat	DDDMM.MMMMM	Latitude as degrees: DDD: Degree (Fixed three digits) MM: Minutes (Fixed two digits) .MMMMM: Decimal fraction of minutes (Variable)
N/S	"N" or "S"	Latitude direction: North or South
Long	DDDMM.MMMMM	Longitude as degrees: DDD: Degree (Fixed three digits) MM: Minutes (Fixed two digits) .MMMMM: Decimal fraction of minutes (Variable)
E/W	"E" or "W"	Longitude direction: East or West
Speed	ddd.d	Speed over ground in knots
Trackgood	Decimal, 4 digits	Course made good, max. 999.9
Date	Decimal, 6 digits	Date of Fix: ddmmyy
MagVar	Decimal, 4 digits	Magnetic Variation, max.: 090.0
MagVarDir	"E" or "W"	Magnetic Variation Direction
Mode	"D", "A", "N" or "E"	Positioning system Mode Indicator:

Parameter	Format	Description
		"D" = Differential mode "A" = Autonomous mode "N" = data not valid "E" = Estimated (dead reckoning) mode
Nav_status	"S", "C", "U" or "V"	Navigational status indicator: "S" = Safe "C" = Caution "U" = Unsafe "V" = Not valid
checksum	Hexadecimal, 2 digits	Checksum of the message bytes without *<checksum><cr><lf> characters.

Example:

```
$GNRMC,,V,,,,,,,,,,N,V*37
```

```
or
```

```
$GNRMC,202340.000,A,4045.53297,N,01447.20361,E,0.2,0.0,291117,,,A,C*18
```

6.3.6 $--VTG

Course over ground and ground speed, this message provides the actual course and speed relative to ground.

- NMEA message list bitmask (64 bits): 0000 0000 0000 0010

Synopsis:

```
$<TalkerID>VTG,<TMGT>,T,<TMGM>,M,<SoGN>,N,<SoGK>,K,D*<checksum><cr><lf>
```

Arguments:

Table 29. $--VTG message field description

Parameter	Format	Description
TalkerID	String, 2 characters	The talker ID (Fixed two characters). GP: If system works in GPS only mode GL: If system works in GLONASS only mode GA: If system works in GALILEO only mode BD: If system works in BEIDOU only mode QZ: If system works in QZSS only mode GN: If system works in multi-constellation mode.
TMGT	ddd.d in degrees	Track in reference to "true" earth poles
T		Indicates "terrestrial"
TMGM	ddd.d in degrees	Track in reference to "magnetic" earth poles
M		Indicates "magnetic"
SoGN	ddd.d in knots	Speed over Ground in knots
N		Indicates "knots"
SoGK	ddd.d in km/h	Speed over Ground in kilometers per hour
K		Indicates "kilometres"

Parameter	Format	Description
D	char	Mode indicator: A = Autonomous mode D = Differential mode E = Estimated mode
checksum	Hexadecimal, 2 digits	Checksum of the message bytes without *<checksum><cr><lf> characters

Example:

```
$GPVTG,73.2,T,,M,0.2,N,0.4,K,D*50
```

6.3.7 $--ZDA

- UTC, day, month and year.
- NMEA message list bitmask (64 bits): 0000 0000 0100 0000

Synopsis:

```
$<TalkerID>ZDA,<Timestamp>,<Day>,<Month>,<Year>,,*<checksum><cr><lf>
```

Arguments:

Table 30. $--ZDA message field description

Parameter	Format	Description
TalkerID	String, 2 characters	The talker ID (Fixed two characters). GP: If system works in GPS only mode GL: If system works in GLONASS only mode GA: If system works in GALILEO only mode BD: If system works in BEIDOU only mode QZ: If system works in QZSS only mode GN: If system works in multi-constellation mode.
Timestamp	hhmmss.sss	UTC Time of GPS Sample: hh: hours (Fixed two digits) mm: minutes (Fixed two digits) ss: seconds (Fixed two digits) .sss: decimal fraction of seconds (Variable length) Note that decimal fraction assumes non zero values when the fix rate is bigger than 1 Hz.
Day	Decimal, 2 digits	Day of month (01 to 31)
Month	Decimal, 2 digits	Month (01 to 12)
Year	Decimal, 4 digits	Year (1994 - 2015)
checksum	Hexadecimal, 2 digits	Checksum of the message bytes without *<checksum><cr><lf> characters.

Example:

```
$GNZDA,204409.000,29,11,2017,,*4C
```

6.3.8 $--GST

- Global Positioning System Pseudorange Noise Statistics.
- NMEA message list bitmask (64 bits): 0000 0000 0000 0008

Synopsis:

```
$<TalkerID>GST,<Timestamp>,<EHPE>,<Semi-major>,<Semi-minor>,<Angle>,<LatErr>,<LonErr>,<Alt
Err Dev>*<checksum><cr><lf>
```

Arguments:

Table 31. $--GST message field description

Parameter	Format	Description
TalkerID	String, 2 characters	The talker ID (Fixed two characters). GP: If system works in GPS only mode GL: If system works in GLONASS only mode GA: If system works in GALILEO only mode BD: If system works in BEIDOU only mode QZ: If system works in QZSS only mode GN: If system works in multi-constellation mode.
Timestamp	hhmmss.sss	UTC Time of GPS Sample: hh: hours (Fixed two digits) mm: minutes (Fixed two digits) ss: seconds (Fixed two digits) .sss: decimal fraction of seconds (Variable length) Note that decimal fraction assumes non zero values when the fix rate is bigger than 1 Hz.
EHPE	dd.d in m	Equivalent Horizontal Position Error
Semi-major	dd.d in m	Standard deviation (meters) of semi-major axis of error ellipse
Semi-minor	dd.d in m	Standard deviation (meters) of semi-minor axis of error ellipse
Angle	dd.d in degree	Orientation of semi-major axis of error ellipse (true north degrees)
LatErr	dd.d in m	Standard deviation (meters) of latitude error
LonErr	dd.d in m	Standard deviation (meters) of longitude error
AltErr	dd.d in m	Standard deviation (meters) of altitude error
checksum	Hexadecimal,2 digits	Checksum of the message bytes without *<checksum><cr><lf> characters.

Example:

```
$GNGST,205512.000,16.5,5.6,4.5,0.8,5.0,5.0,6.7*41
```

or

```
$GAGST,,,,,,,,*46
```

6.4 ST NMEA messages specification

In order to provide further data and information from the ST GNSS receiver, which are not provided by the standard NMEA messages, STMicroelectronics provides "proprietary messages". Any proprietary message on the NMEA port starts with "$PSTM" where "STM" indicates that it is an ST proprietary message ($PSTM).

There are two sorts of "proprietary messages" within an ST-GNSS system. They are either sent repeatedly with a defined or definable reporting rate or they are sent only once as a reaction to a command.

6.4.1 $PSTMINITGPSOK

Message sent in response to command $PSTMINITGPS

Synopsis:

```
$PSTMINITGPSOK*<checksum><cr><lf>
```

Arguments:
None.
Results:
Message sent in case of successful operation.

6.4.2 $PSTMINITGPSERROR

Message sent in response to command $PSTMINITGPS
Synopsis:

```
$PSTMINITGPSERROR*<checksum><cr><lf>
```

Arguments:
None.
Results:
Message sent in case of error.

6.4.3 $PSTMINITTIMEOK

Message sent in response to command $PSTMINITTIME
Synopsis:

```
$PSTMINITTIME OK*<checksum><cr><lf>
```

Arguments:
None.
Results:
Message sent in case of successful operation.

6.4.4 $PSTMINITTIMEERROR

Message sent in response to command $PSTMINITTIME
Synopsis:

```
$PSTMINITTIMEERROR*<checksum><cr><lf>
```

Arguments:
None.
Results:
Message sent in case of error.

6.4.5 $PSTMFORCESTANDBYERROR

Message sent in response to command $PSTMFORCESTANDBY
Synopsis:

```
$PSTMFORCESTANDBYERROR*<checksum><cr><lf>
```

Arguments:
No arguments
Results:
Message is sent in case of error

6.4.6 $PSTMRF

Provides "satellite signal data" for each tracked satellite. Single message contains the relevant fields for max 3 satellites. For all satellites the message is repeated with the data of the other satellites.

Synopsis:

```
$PSTMRF,<MessgAmount>,<MessgIndex>,<used_sats>, [<Sat1ID>,<Sat1PhN>,<Sat1Freq>,<Sat1CN0>],
[<Sat2ID>,<Sat2PhN>,<Sat2Freq>,<Sat2CN0>], [<Sat3ID>,<Sat3PhN>,<Sat3Freq>,<Sat3CN0>],
*<checksum><cr><lf>
```

Arguments:

Table 32. $PSTMRF message field description

Parameter	Format	Description
MessgAmount	Decimal, 1 digit	Number of consecutive $PSTMRF messages
MessgIndex	Decimal, 1 digit	Current number in the sequence of messages
used_sats	Decimal, 2 digits	Number of satellites used in the fix
SatxID	Decimal, 2 digits	Satellite x Number (PRN)
SatxPhN	Decimal, 5 digits	Satellite x Phase Noise
SatxFreq	Decimal, 6 digits	Satellite x Frequency
SatxCN0	Decimal, 2 digits	Satellite x Carrier to Noise Ratio (in dB)

Results:

None

6.4.7 $PSTMTESTRF

Specific message containing information on just one satellite for RF testing purposes.

Synopsis:

```
$PSTMTESTRF,<Sat-ID>,<Sat-Freq>,<Sat-PhN><Sat-CN0>*<checksum><cr><lf>
```

Arguments:

Table 33. $PSTMTESTRF message field description

Parameter	Format	Description
Sat-ID	Decimal, 2 digits	Satellite Number (PRN)
Sat-Freq	Decimal, 5 digits	Satellite Frequency
Sat-PhN	Decimal, 5 digits	Satellite Phase Noise
Sat-CN0	Decimal, 2 digits	Satellite Carrier to Noise Ratio (in dB)

Results:

None

6.4.8 $PSTMEPHEM

Ephemeris Data Dump.

This message is sent as a reply to a $PSTMDUMPEPHEMS command.

Synopsis:

```
$PSTMEPHEM,<sat_id>,<N>,<byte1>,...,<byteN>*<checksum><cr><lf>
```

Arguments:

Table 34. $PSTMEPHEM message field description

Parameter	Format	Description
sat_id	Decimal, 2 digits	Satellite number
N	Decimal, 1 Digit	Number of the ephemeris data bytes
byte1	Hexadecimal, 2 digits	First byte of the ephemeris data
byteN	Hexadecimal, 2 digits	Last byte of the ephemeris data

The N Bytes that are in the message are the dump of a structure that contains all the information of the ephemeris.

Data formats are constellation dependant.

Table 35. $PSTMEPHEM message field description for GPS constellation

Bits	Structure Member	Description
16	week	Week number of the Issue of Data
16	toe	Time of week for ephemeris epoch
16	toc	Time of week for clock epoch
8	iode1	Issue of data 1
8	iode2	Issue of data 2
10	iodc	Issue of data clock
14	i_dot	Rate of inclination angle
8	RESERVED	
24	omega_dot	Rate of right ascension
8	RESERVED	Must be 0
16	crs	Amplitude of the sine harmonic correction to the orbit radius
16	crc	Amplitude of the cosine harmonic correction to the orbit radius
16	cus	Amplitude of the sine harmonic correction to the argument of latitude
16	cuc	Amplitude of the cosine harmonic correction to the argument of latitude
16	cis	Amplitude of the sine harmonic correction to the angle of inclination
16	cic	Amplitude of the cosine harmonic correction to the angle of inclination
16	motion_difference	Mean motion difference from computed value
16	RESERVED	Must be 0
32	inclination	Inclination angle at reference time
32	e	Eccentricity
32	root_A	Square root of major axis
32	mean_anomaly	Mean anomaly at reference time
32	omega_zero	Longitude of ascending node of orbit plane at weekly epoch
32	perigee	Argument of perigee
8	time_group_delay	Estimated group delay differential
8	af2	Second order clock correction
16	af1	First order clock correction

Bits	Structure Member	Description
22	af0	Constant clock correction
1	RESERVED	RESERVED for use by GNSS library – must be 1
1	RESERVED	RESERVED for use by GNSS library – must be 1
1	RESERVED	RESERVED for use by GNSS library – must be 1
1	available	Contains 1 if ephemeris is available, 0 if not
1	health	Contains 1 if the satellite is unhealthy, 0 if healthy
1	RESERVED	Must be 0
4	accuracy	Accuracy

Table 36. $PSTMEPHEM message field description for GLONASS constellation

Bits	Structure Member	Description
16	week	Week number of the Issue of Data
16	toe	Time of week for ephemeris epoch
4	toe_lsb	Time of week for ephemeris epoch (LBS)
11	NA	Calendar day number within the four-year period since the beginning of last leap year (almanac)
7	tb	Time of ephemeris index
2	M	Type of satellite 00=GLONASS 01=GLONASS-M
2	P1	Time interval between two adjacent tb parameters
1	P3	Number of satellites for which almanac is transmitted within this frame 0=4 1=5
1	P2	Flag of oddness ("1") or evenness ("0") of the value of tb
1	P4	Flag to show that ephemeris parameters are present
2	KP	Notification on forthcoming leap second correction of UTC
1	RESERVED	
27	xn	Satellite PZ-90 x coordinate at epoch tb
5	xn_dot_dot	Satellite PZ-90 x velocity at epoch tb
24	xn_dot	Satellite PZ-90 x acceleration component at epoch tb
5	n	Slot number (1\xc9 24)
3	Bn	Healthy flags
27	yn	Satellite PZ-90 y coordinate at epoch tb
5	yn_dot_dot	Satellite PZ-90 y acceleration component at epoch tb
24	yn_dot	Satellite PZ-90 y velocity at epoch tb
8	age_h	Age of predicted ephemeris (hours)
27	zn	Satellite PZ-90 z coordinate at epoch tb
5	zn_dot_dot	Satellite PZ-90 z acceleration component at epoch tb
24	zn_dot	Satellite PZ-90 z velocity at epoch tb
8	RESERVED	Must be 0
11	gamma_n	Satellite clock frequency drift at epoch tb
5	E_n	Age of the ephemeris information
4	freq_id	Frequency ID
12	RESERVED	

Bits	Structure Member	Description
22	tau_n	Satellite clock correction at epoch tb
10	RESERVED	Must be 0
32	tau_c	GLONASS to UTC(SU) time correction
22	tau_GPS	GLONASS to GPS system time correction
10	RESERVED	
11	NT	Calendar day number of ephemeris within the four-year period since the beginning of last leap year
5	N4	Four-year interval number starting from 1996
12	tk	Satellite time referenced to the beginning of the frame
4	FT	Predicted satellite user range accuracy at time tb
32	RESERVED	
5	m_available	Must be 0x1F
1	nvm_reliable	Must be 1
26	spare	
25	RESERVED	
1	available	Contains 1 if ephemeris is available, 0 if not
1	health	Contains 1 if the satellite is unhealthy, 0 if healthy
1	RESERVED	Must be 0
4	RESERVED	

Table 37. $PSTMEPHEM message field description for Galileo constellation

Bits	Structure Member	Description
16	week	Week number of the Issue of Data
14	toe	Time of week for ephemeris epoch
2	RESERVED	
16	toc	Time of week for clock epoch
10	iod_nav	Issue of data
8	SISA	Signal In Space Accuracy
10	RESERVED	Must be 0
10	BGD_E1_E5a	E1-E5a Broadcast Group Delay
10	BGD_E1_E5b	E1-E5b Broadcast Group Delay
2	E1BHS	E1-B Signal Health Status
32	inclination	Inclination angle at reference time
32	eccentricity	Eccentricity
32	root_a	Square root of major axis
32	mean_anomaly	Mean anomaly at reference time
32	omega_zero	Longitude of ascending node of orbit plane at weekly epoch
32	perigee	Argument of perigee
14	i_dot	Rate of inclination angle
1	available	Contains 1 if ephemeris is available, 0 if not

Bits	Structure Member	Description
1	health	Contains 1 if the satellite is unhealthy, 0 if healthy
16	motion_difference	Mean motion difference from computed value
16	crs	Amplitude of the sine harmonic correction to the orbit radius
16	crc	Amplitude of the cosine harmonic correction to the orbit radius
16	cus	Amplitude of the sine harmonic correction to the argument of latitude
16	cuc	Amplitude of the cosine harmonic correction to the argument of latitude
16	cis	Amplitude of the sine harmonic correction to the angle of inclination
16	cic	Amplitude of the cosine harmonic correction to the angle of inclination
24	omega_dot	Rate of right ascension
6	SVID	Satellite Identification
1	E1BDVS	E1-B Data Validity Status
1	RESERVED	Must be 0
8	RESERVED	Must be 0
16	RESERVED	Must be 0
6	af2	Second order clock correction
21	af1	First order clock correction
5	word_available	Must be 0x1F
31	af0	Constant clock correction
1	RESERVED	
6	RESERVED	Must be 0
26	RESERVED	RESERVED for use by GNSS library – must be 1
1	RESERVED	Must be 0

Table 38. $PSTMEPHEM message field description for BEIDOU constellation

Bits	Structure Member	Description
32	inclination	Inclination angle at reference time
32	eccentricity	Eccentricity
32	root_a	Square root of major axis
32	mean_anomaly	Mean anomaly at reference time
32	omega_zero	Longitude of ascending node of orbit plane at weekly epoch
32	perigee	Argument of perigee
17	toe	Time of week for ephemeris epoch
10	time_group_delay	Estimated group delay differential
5	aode	Issue of data, ephemeris
24	omega_dot	Rate of right ascension
8	A0	Ionospheric Delay Model Parameter α_0
24	af0	Constant clock correction
8	A1	Ionospheric Delay Model Parameter α_1
20	sow	Seconds of week
11	af2	Second order clock correction

Bits	Structure Member	Description
1	is_geo	1 for Geostationary satellites, otherwise 0
22	af1	First order clock correction
10	subframe_avail	Must be 0x3FF
16	motion_difference	Mean motion difference from computed value
8	A2	Ionospheric Delay Model Parameter α_2
8	A3	Ionospheric Delay Model Parameter α_3
18	crs	Amplitude of the sine harmonic correction to the orbit radius
8	B2	Ionospheric Delay Model Parameter β_2
4	urai	User range accuracy index
2	RESERVED	Must be 0
18	crc	Amplitude of the cosine harmonic correction to the orbit radius
8	B3	Ionospheric Delay Model Parameter β_3
5	aodc	Issue of data, clock
1	spare	
18	cus	Amplitude of the sine harmonic correction to the argument of latitude
14	i_dot	Rate of inclination angle
18	cuc	Amplitude of the cosine harmonic correction to the argument of latitude
8	B0	Ionospheric Delay Model Parameter β_0
6	spare	
18	cis	Amplitude of the sine harmonic correction to the angle of inclination
8	B1	Ionospheric Delay Model Parameter β_1
6	RESERVED	Must be 0
18	cic	Amplitude of the cosine harmonic correction to the angle of inclination
1	nvm_reliable	Must be 1
11	RESERVED	Must be 0
2	spare	
17	toc	Time of week for clock epoch
13	week	Week number of the Issue of Data
1	available	Contains 1 if ephemeris is available, 0 if not
1	health	Contains 1 if the satellite is unhealthy, 0 if healthy

6.4.9 $PSTMGETUCODEOK

Message sent in response to command $PSTMGETUCODE

Synopsis:

```
$PSTMGETUCODEOK,<unique_code>*<checksum><cr><lf>
```

Arguments:

Table 39. $PSTMGETUCODEOK message field description

Parameter	Format	Description
unique_code	Char, 32 bytes	The Unique ID written in the secondary boots

Results:
Message sent in case of successful operation.

6.4.10 $PSTMGETUCODEERROR

Message sent in response to command $PSTMGETUCODE
Synopsis:

```
$PSTMGETUCODEERROR*<checksum><cr><lf>
```

Arguments:
None.
Results:
Message sent in case of error.

6.4.11 $PSTMEPHEMOK

Message sent in response to command $PSTMEPHEM
Synopsis:

```
$PSTMEPHEMOK*<checksum><cr><lf>
```

Arguments:
None.
Results:
Message sent in case of successful operation.

6.4.12 $PSTMEPHEMERROR

Message sent in response to command $PSTMEPHEM
Synopsis:

```
$PSTMEPHEMERROR*<checksum><cr><lf>
```

Arguments:
None.
Results:
Message sent in case of error.

6.5 ST system configuration messages

6.5.1 $PSTMSETPAROK

Message sent in response to command $PSTMSETPAR
Synopsis:

```
$PSTMSETPAROK ,<ConfigBlock><ID>*<checksum><cr><lf>
```

Arguments:

Table 40. $PSTMSETPAROK message field description

Parameter	Format	Description
ConfigBlock	Decima1,1 digit	Indicates one of the configuration blocks: 1=Current Configuration, 2 = Default Configuration,

Parameter	Format	Description
		3 = NVM Stored configuration.
ID	Decimal, 3 digits	ID - Identifier (see Configuration Data Block as described in FW Configuration document)

Results:
Message sent in case of successful operation.

6.5.2 $PSTMSETPARERROR

Message sent in response to command $PSTMSETPAR
Synopsis:

```
$PSTMSETPARERROR*<checksum><cr><lf>
```

Argument:
No argument
Results:
Message sent in case of error.

6.5.3 $PSTMRESTOREPAROK

Message sent in response to command $PSTMRESTOREPAR
Synopsis:

```
$PSTMRESTOREPAROK*<checksum><cr><lf>
```

Arguments:
None.
Results:
Message sent in case of successful operation.

6.5.4 $PSTMRESTOREPARERROR

Message sent in response to command $PSTMRESTOREPAR
Synopsis:

```
$PSTMRESTOREPARERROR*<checksum><cr><lf>
```

Arguments:
None.
Results:
Message sent in case of error.

6.5.5 $PSTMSAVEPAROK

Message sent in response to command $PSTMSAVEPAR
Synopsis:

```
$PSTMSAVEPAROK*<checksum><cr><lf>
```

Arguments:
None.
Results:
Message sent in case of successful operation.

6.5.6 $PSTMSAVEPARERROR

Message sent in response to command $PSTMSAVEPAR

Synopsis:

```
$PSTMSAVEPARERROR*<checksum><cr><lf>
```

Arguments:

None.

Results:

Message sent in case of error.

6.5.7 $PSTMSETPAR

Message sent in response to command $PSTMGETPAR

Synopsis:

```
$PSTMSETPAR,<ConfigBlock><ID>,<value>*<checksum><cr><lf>
```

Arguments:

Table 41. $PSTMSETPAR message field description

Parameter	Format	Description
ConfigBlock	Decima1, 1 digit	Indicates one of the configuration blocks: 1 = Current Configuration, 2 = Default Configuration, 3 = NVM Stored configuration.
ID	Decimal, 3 digits	ID - Identifier (see Configuration Data Block)
value	Hexadecimal or Decimal	The value of returned parameter. According to the parameter type it could be expressed in hexadecimal format (in case parameter is integer) or decimal format (in case the parameter is floating).

6.5.8 $PSTMGETPARERROR

Message sent in response to command $PSTMGETPAR.

Synopsis:

```
$PSTMGETPARERROR*<checksum><cr><lf>
```

Arguments:

No arguments

Results:

- In case of errors, the error message is returned

6.6 Real Time AGNSS NMEA messages

6.6.1 $PSTMSTAGPS8PASSRTN

Message sent in response to command $PSTMSTAGPS8PASSGEN.

Synopsis:

```
$PSTMSTAGPS8PASSRTN,<DevID>,<Password>*<checksum><cr><lf>
```

Arguments:

Table 42. **$PSTMSTAGPS8PASSRTN message field description**

Parameter	Description
<DevID>	Unique Device ID
<Password>	41-character ASCII password.

Results:

None

7 Firmware Configuration Data Block (CDB)

All configuration parameters are grouped in a data block. Each field is addressed by a unique ID. The IDs are made by three digits: the most significant one represents the parameter type and the others are used to identify different parameters of the same type.

The IDs not reported in the table should be considered as RESERVED and must be left untouched to avoid unexpected system behaviors.

7.1 CDB-ID 102 – NMEA port baudrate setting

Allow setting the baudrate for the NMEA port number. The translation in Table 43.

Table 43. CDB-ID 102 field description

Parameter Value	Baudrate
0x0	300 baud
0x1	600 baud
0x2	1200 baud
0x3	2400 baud
0x4	4800 baud
0x5	9600 baud
0x 6	14400 baud
0x 7	19200 baud
0x 8	38400 baud
0x 9	57600 baud
0xA	115200 baud
0xB	230400 baud
0xC	460800 baud
0xD	921600 baud

System reboot needed to have new setting in use.

7.2 CDB-ID 190 - CDB-ID 201 - CDB-ID 228 - NMEA on UART message list parameters

CDB-ID 201 and CDB-ID 228 allow enabling/disabling each NMEA message in the message list 0. CDB-ID 201 represents the first 32 bits (low bits) of the extended 64 bits NMEA message list. See CDB-ID 228 for the second 32 bits (high bits) of the 64 bits message list.

CDB-ID 190 allows setting the message list output rate for the message list 0. It is a scaling factor referred to the selected fix rate. The default value is 1 and this means that the messages are sent out on every fix. Setting the scaling factor to "N" means that the corresponding message list is sent out every "N" fixes.

Note: The message list 0 is the standard message list. Only the message list 0 should be used if the NMEA multiple rate feature is not required.

For each bit:

- 0 means feature disabled
- 1 means feature enabled

Table 44. **CDB-ID 201 - CDB-ID 228 fields description**

	Bit [1]	Bitmask (32 bits)	Function
Low 32 bits	0	0x1	$GPGNS Message
	1	0x2	$GPGGA Message
	2	0x4	$GPGSA Message
	3	0x8	$GPGST Message
	4	0x10	$GPVTG Message
	5	0x20	RESERVED
	6	0x40	$GPRMC Message
	7	0x80	$PSTMRF Message
	8	0x100	RESERVED
	9	0x200	RESERVED
	10	0x400	RESERVED
	11	0x800	RESERVED
	12	0x1000	RESERVED
	13	0x2000	RESERVED
	14	0x4000	RESERVED
	15	0x8000	RESERVED
	16	0x10000	RESERVED
	17	0x20000	$PSTMSBAS Message
	18	0x40000	$PSTMTESTRF Message
	19	0x80000	$GPGSV Message
	20	0x100000	$GPGLL Message
	21	0x200000	RESERVED
	22	0x400000	RESERVED
	23	0x800000	RESERVED
	24	0x1000000	$GPZDA Message
	25	0x2000000	RESERVED
	26	0x4000000	RESERVED
	27	0x8000000	RESERVED
	28	0x10000000	RESERVED
	29	0x20000000	RESERVED
	30	0x40000000	RESERVED
	31	0x80000000	RESERVED
High 32 bits	32	0x1	RESERVED
	33	0x2	RESERVED
	34	0x4	RESERVED
	35	0x8	RESERVED
	36	0x10	RESERVED
	37	0x20	RESERVED
	38	0x40	RESERVED

	Bit [1]	Bitmask (32 bits)	Function
High 32 bits	39	0x80	RESERVED
	40	0x100	$PSTMEPHEM Message
	41	0x200	RESERVED
	42	0x400	RESERVED
	43	0x800	RESERVED
	44	0x1000	RESERVED
	45	0x2000	RESERVED
	46	0x4000	RESERVED
	47	0x8000	RESERVED
	48	0x10000	RESERVED
	49	0x20000	RESERVED
	50	0x40000	RESERVED
	51	0x80000	RESERVED
	52	0x100000	RESERVED
	53	0x200000	RESERVED
	54	0x400000	RESERVED
	55	0x800000	RESERVED
	56	0x1000000	RESERVED
	57	0x2000000	RESERVED
	58	0x4000000	RESERVED
	59	0x8000000	RESERVED
	60	0x10000000	RESERVED
	61	0x20000000	RESERVED
	62	0x40000000	RESERVED
	63	0x80000000	RESERVED

1. *The Bit-Value indicates the bit position, thus multiple choices are possible.*

Note: The message list 0 is the standard message list. Only the message list 0 should be used if the NMEA multiple rate feature is not required.

7.3 CDB-ID 200 - CDB-ID 227 - Application ON/OFF

Allow enabling/disabling different features in the GNSS library.

All features are mapped in a 64-bit bitmap with one bit for each feature; CDB-ID 200 represents the first 32 bits (low 32 bits) and CDB-227 represents the second 32 bits (high 32 bits).

For each bit:

- 0 means feature disabled
- 1 means feature enabled

Table 45. CDB-ID 200 field description

Bit [1]	Bitmask	Function	Description
0	0x1	RESERVED	

Bit [1]	Bitmask	Function	Description
1	0x2	RESERVED	
2	0x4	SBAS (WAAS / EGNOS) augmentation system	
3	0x8	RESERVED	
4	0x10	RESERVED	
5	0x20	RESERVED	
6	0x40	RESERVED	
7	0x80	QZSS distributed acquisition mode enable	Enable/disable the distributed acquisition operative mode for the QZSS constellation. When distributed acquisition mode for QZSS is enabled, the acquisition stage usage is widespread along the time in order to mitigate the current consumption spikes required by the acquisition engine.
9	0x200	RESERVED	
10	0x400	RESERVED	
11	0x800	RESERVED	
12	0x1000	RESERVED	
14	0x4000	RESERVED	
15	0x8000	RESERVED	
16	0x10000	GPS constellation enable[2]	Enable/disable the GPS constellation. When this bit is enabled GPS satellites are enabled to be tracked and used for positioning. This bit setting affects also the talker ID of GSV and GSA NMEA messages. If only the GPS constellation is enabled the NMEA talker ID for GSV and GSA is "GP". If GLONASS constellation is also enabled "GP" is used for GPS related GSV messages while "GN" is used for the GSA messages.
17	0x20000	GLONASS constellation enable[2]	Enable/disable the GLONASS constellation. When this bit is enabled GLONASS satellites are enabled to be tracked. To be used for positioning also the Bit 21 should be enabled. This bit setting affects also the talker ID of GSV and GSA NMEA messages. If only the GLONASS constellation is enabled the NMEA talker ID for GSV and GSA is "GL". If GPS constellation is also enabled "GL" is used for GLONASS related GSV messages while "GN" is used for the GSA messages
18	0x40000	RESERVED	
19	0x80000	RESERVED	
20	0x100000	RESERVED	
21	0x200000	RESERVED	
22	0x400000	RESERVED	
23	0x800000	RESERVED	
24	0x1000000	RESERVED	
25	0x2000000	RESERVED	
26	0x4000000	RESERVED	
27	0x8000000	RESERVED	
28	0x10000000	RESERVED	
29	0x20000000	RESERVED	
30	0x40000000	RESERVED	
31	0x80000000	RESERVED	

1. *The Bit-Value indicates the bit position (starting from 0 as the least significant bit), thus multiple choices are possible.*

2. *Multi-constellation firmware supports the following constellations: GPS, GALILEO, GLONASS, BEIDOU and QZSS. All constellations cannot be enabled at the same time, allowed combinations to achieve maximum coverage, are: (GPS+GALILEO+QZSS+GLONASS), (GPS+GALILEO+QZSS+BEIDOU) and (GLONASS+BEIDOU). Any constellation can be enabled as standalone satellite navigation system.*

Table 46. CDB-ID 227 field description

Bit[1]	Bitmask	Function	Description
1	0x1	RESERVED	
2	0x2	RESERVED	
3	0x4	RESERVED	
4	0x8	RESERVED	
5	0x10	RESERVED	
6	0x20	RESERVED	
7	0x40	Galileo constellation enable	Enable/disable the Galileo constellation. When this bit is enabled Galileo satellites are enabled to be tracked and used for positioning
8	0x80	Galileo usage for positioning enable	Enable/disable the usage of Galileo satellite for the GNSS position fix. If this bit is disabled and Galileo constellation is enabled, the Galileo satellites are only tracked.
9	0x100	BEIDOU constellation enable[2]	Enable/disable the BEIDOU constellation. When this bit is enabled BEIDOU satellites are enabled to be tracked and used for positioning.
10	0x200	BEIDOU usage for positioning enable	Enable/disable the usage of BEIDOU satellite for the GNSS position fix. If this bit is disabled and BEIDOU constellation is enabled, the BEIDOU satellites are only tracked.
11	0x400	RESERVED	
12	0x800	RESERVED	
13	0x1000	RESERVED	
14	0x2000	RESERVED	
15	0x4000	RESERVED	
16	0x8000	RESERVED	
17	0x10000	RESERVED	
18	0x20000	RESERVED	
19	0x40000	RESERVED	
20	0x80000	RESERVED	
21	0x100000	RESERVED	
22	0x200000	RESERVED	
23	0x400000	RESERVED	
24	0x800000	RESERVED	
25	0x1000000	RESERVED	
26	0x2000000	RESERVED	
27	0x4000000	RESERVED	

1. *The Bit-Value indicates the bit position (starting from 0 as the least significant bit), thus multiple choices are possible.*
2. *Multi-constellation firmware supports the following constellations: GPS, GALILEO, GLONASS, BEIDOU and QZSS. All constellations cannot be enabled at the same time, allowed combinations to achieve maximum coverage, are: (GPS+GALILEO+QZSS+GLONASS), (GPS+GALILEO+QZSS+BEIDOU). Any constellation can be enabled as standalone satellite navigation system.*

Note: If the STAGPS feature is not required (bit 4) and it is disabled, it is strongly suggested to clear all the STAGPS data from the NVM memory. This can be done via NMEA sending the "$PSTMSTAGPSINVALIDATE,7" command. If the NVM was empty (e.g. the STAGPS has been never enabled or the NVM has been completely erased before) the invalidate command is not required.

When GPS and GLONASS constellations are enabled, the GSV messages are sent in two separate sets: one with "GP" as talker ID and one with "GL".

Only "GN" is supported as talker ID for QZSS GSV and GSA messages.

In this case the GSV messages are sent in a single set reporting satellites for all enabled constellations.

7.4 CDB-ID 231 – CDB-ID 232 - NMEA on I2C Port Message List

Allow enabling/disabling each NMEA message in the message list 0 used for sending messages over the I2C port. CDB-ID 231 represents the first 32 bits (low bits) of the extended 64 bits NMEA message list. See CDB-ID 232 for the second 32 bits (high bits) of the 64 bits message list. The message list configuration is done in the same way as for the NMEA message list 0 (see CDB-ID 201 and CDB-ID 228 for details). See CDB-ID 201 also for supported message list table.

7.5 CDB-ID 238 – Default UTC delta time

Allow setting the default value for the GPS time to UTC delta time seconds (leap seconds). This parameter is used by the GNSS software only if the UTC backup data is not available in the backup memory (e.g. first startup after production or in case of backup memory content lost occurrence).

7.6 CDB-ID 260 – WLS algorithm configuration

Allow to configure the WLS algorithm implemented in the positioning stage.

Table 47. CDB-ID 260 field description

Bits	Values	Description
B0	0...1	Enable/Disable the WLS algorithm usage in the positioning stage. 0 = disabled 1= enabled
B1...B7	xxx	Not used
B8...B15	1...100	Parameter1 multiplied by 10. Parameter1 is a coefficient to change the measurements weighting in the position filter. Allowed values are from 0.1 to 10.0 (suggested value is 1.0) means high acceptance of satellites measurements in the position filter. 10.0 means low acceptance of satellites measurements in the position filter.
B16...B23	10...100	Parameter2 multiplied by 10. Parameter2 is a coefficient to change the measurements acceptance threshold. Allowed values are from 1.0 to 10.0 (suggested value is 2.5) means strong satellite exclusions by FDE (high false alarm rate). 10.0 means relaxed satellites exclusions by FDE.

7.7 CDB-ID 500 – Text message

Allow setting a text message which is sent (if enabled – see bit9 of CDB-ID 200 parameter) at startup over the NMEA port. The user is free to use this text as product name or as specific configuration marker.

System reboot needed to have new setting in use.

Appendix A Acronyms and definitions

Table 48 lists the acronyms and definitions used in this document.

Table 48. Acronyms and definitions

Keyword	Definition
Accuracy	Deviation of a GPS-based calculated position from the true position
ADC	Analogue to Digital Converter
Almanac	Contains the information about all available satellites, their orbit data and time of their clocks.
ANF	Adaptive Notch Filter
Azim	Azimuth - Angular distance from a reference
Bank Swap	Exchanging two memory banks for storage of data
BAUD rate	Transmission Rate Measure for the effective transmission of data content. (may differ from Bits/sec).
BEIDOU	China's regional navigation satellite system
Checksum	Calculated from the transmitted characters of a message by "ex-OR"ing the 8 bit character values excluding delimiters $ and *
CN0	Carrier to Noise Ratio - Identifies the quality of a received signal
Cold Start	Start Condition for a GPS system having no position nor time. Almanac and Ephemeris is not available, too.
BeiDou	China's global navigation satellite system (also known as Beidou-2, BD2)
Dead Reckoning	Sensor based process to determine the movement of a mobile unit, utilizing Gyro, Odometer and Wheel Pulses.
Delimiter (within NMEA 0183)	ASCII "$" to indicate Address Field ASCII "," to indicate Data Field ASCII "*" to indicate Checksum Field
DGPS	Differential GPS - GPS Augmentation System providing the accurate location of a Reference Station to reduce system errors.
EGNOS	European Geostationary Navigation Overlay System
Elev	Elevation - Angle between a high level or non-earth bound point and the horizontal plane of the viewer.
Ephemeris	Ephemeris Data is transmitted by each satellite and contains current and predicted satellite position.
FDA	Failure Detection Algorithm - Specific Algorithm to detect failures in position calculation
FDE	False Detection Exclusion
GALILEO	Europe's global navigation satellite system
GDOP	Geometric Dilution Of Position - Quality value representing all geometry based error factors in a system.
GNSS	Global Navigation Satellite System - Satellite based system to calculate the position of the Teseo on the earth surface.
GPS	Global Positioning System - United States Satellite Navigation System
GPS Library	STMicroelectronics C-Library containing all GPS relevant Functions
Gyro	Gyroscope - Sensor to determine rotational movements
HDOP	Horizontal Dilution Of Precision - Quality value representing all 2D plane geometry based error factors in a system.
Hot Start	Start Condition for a GPS System having position, time, Almanac and Ephemeris already available. High time accuracy is required.
IMU	Inertial Measurement Unit

Keyword	Definition
Lat	Latitude - Angular difference of a given position to the Equator. Values include 0° -90° either North or South
Lat-Ref	Latitude Reference - Reference if a Latitude value is North or South
Long	Longitude - Angular difference to a "reference" Longitude indicated as "000". Values include 0° \xc9 180° either West or East.
Long-Ref	Longitude Reference - Reference if a Longitude value is East or West of the "000" Meridian.
NMEA	National Marine Electronics Association - United States Standards Organization For Marine Equipment
NMEA 0183	National Marine Electronics Association - Standard for Interfacing Marine Electronics Devices
NVM	Non Volatile Memory - Any type of memory that conserves data in the absence of regular supply voltage (includes battery buffered memories)
Proprietary Message	Messages within the scope of NMEA0183 which are not standardized. They start with $P and a 3 character identifier.
PRN	Pseudo Random Number - Satellite Specific 1023 Bit Number used for Spread Spectrum Modulation
RAIM	Teseo Autonomous Integrity Monitoring
RF	Radio Frequency - High Frequency for Reception with a RF-Teseo
RS232	IEEE Standard - Physical Layer Standard for Data Transmission
Sat-ID	Satellite Identifier - Satellite specific Number used to generate the corresponding PRN code
SBAS	Satellite Based Augmentation System - GPS enhancement system based on geostationary satellites.
SPS	Standard Positioning Service
Static Position Filtering	Algorithm to detect that the GPS Teseo doesn't move and position output is kept stable.
UTC	Universal Time Coordinated
WAAS	Wide Area Augmentation System - American GPS Augmentation System delivering accurate Ionosphere Data
Warm Start	Start Condition for a GPS system having current Almanac, position and time availability. Ephemeris are not available. Time needs to be available with reasonable accuracy (some seconds).
2D Fix	Fix based on the use of 3 satellites
3D Fix	Fix based on the use of 4 satellites

A.1 Local geodetic datum tables

Table 49. Africa geodetic datum

AFRICA		
REGION	**CODE**	**CDB-ID VALUE**
ADINDAN		
MeanSolution (Ethiopia-Sudan)	ADI-M	0
BurkinaFaso	ADI-E	1
Cameroon	ADI-F	2
Ethiopia	ADI-A	3
Mali	ADI-C	4
Senegal	ADI-D	5
Sudan	ADI-B	6

AFRICA		
REGION	CODE	CDB-ID VALUE
AFGOOYE		
Somalia	AFG	7
ARC_1950		
Mean_Solution	ARF-M	8
Botswana	ARF-A	9
Burundi	ARF-H	10
Lesotho	ARF-B	11
Malawi	ARF-C	12
Swaziland	ARF-D	13
Zaire	ARF-E	14
Zambia	ARF-F	15
Zimbabwe	ARF-G	16
ARC_1960		
Mean_Solution	ARS-M	17
Kenya	ARS-A	18
Tanzania	ARS-B	19
AYABELLE_LIGHTHOUSE		
Djibouti	PHA	20
BISSAU		
Guinea-Bissau	BID	21
CAPE		
South_Africa	CAP	22
CARTHAGE		
Tunisia	CGE	23
DABOLA		
Guinea	DAL	24
EUROPEAN_1950		
Egypt	EUR-F	73
Tunisia	EUR-T	83
LEIGON		
Ghana	LEH	25
LIBERIA_1964		
Liberia	LIB	26
MASSAWA		
Eritrea (Ethiopia)	MAS	27
MERCHICH		
Morocco	MER	28
MINNA		
Cameroon	MIN-A	29
Nigeria	MIN-B	30

AFRICA		
REGION	CODE	CDB-ID VALUE
M'PORALOKO		
Gabon	MPO	31
NORTH_SAHARA_1959		
Algeria	NSD	32
OLD_EGYPTIAN_1907		
Egypt	OEG	33
POINT_58		
Mean_Solution (BurkinaFaso-Niger)	PTB	34
POINTE_NOIRE_1948		
Congo	PTN	35
SCHWARZECK		
Namibia	SCK	36
SIERRA_LEONE_1960		
SierraLeone	SRL	37
VOIROL_1960		
Algeria	VOR	38

Table 50. Asia geodetic datum

ASIA		
REGION	CODE	CDB-ID VALUE
AIN_EL_ABD_1970		
Bahrain_Island	AIN-A	39
Saudi_Arabia	AIN-B	40
DJAKARTA(BATAVIA)		
Sumatra (Indonesia)	BAT	41
EUROPEAN_1950		
Iran	EUR-H	77
HONG_KONG_1963		
Hong_Kong	HKD	42
HU-TZU-SHAN		
Taiwan	HTN	43
INDIAN		
Bangladesh	IND-B	44
India-Nepal	IND-I	45
INDIAN_1954		
Thailand	INF-A	46
INDIAN_1960		
Vietnam (near_16DegNorth)	ING-A	47
ConSonIsland (Vietnam)	ING-B	48

ASIA		
REGION	CODE	CDB-ID VALUE
INDIAN_1975		
Thailand	INH-A	49
Thailand	INH-A1	50
INDONESIAN_1974		
Indonesia	IDN	51
KANDAWALA		
SriLanka	KAN	52
KERTAU_1948		
WestMalaysia-Singapore	KEA	53
KOREAN_1995		
SouthKorea	KGS	54
NAHRWAN		
MasirahIsland (Oman)	NAH-A	55
UnitedArabEmirates	NAH-B	56
SaudiArabia	NAH-C	57
OMAN		
Oman	FAH	58
QATAR_NATIONAL		
Qatar	QAT	59
SOUTH_ASIA		
Singapore	SOA	60
TIMBALAI_1948		
Brunei-East_Malaysia	TIL	61
TOKYO		
MeanSolution	TOY-M	62
Japan	TOY-A	63
Okinawa	TOY-C	64
South Korea	TOY-B	65
South Korea	TOY-B1	66

Table 51. Australia geodetic datum

AUSTRALIA		
REGION	CODE	CDB-ID VALUE
AUSTRALIAN_1966		
Australia-Tasmania	AUA	67
AUSTRALIAN_1984		
Australia-Tasmania	AUG	68

Table 52. **Europe geodetic datum**

EUROPE		
REGION	**CODE**	**CDB-ID VALUE**
CO-ORDINATE SYSTEM 1937 OF ESTONIA		
Estonia	EST	69
EUROPEAN_1950		
MeanSolution	EUR-M	70
WesternEurope	EUR-A	71
Cyprus	EUR-E	72
Egypt	EUR-F	73
England, ChannelIslands, Scotland, ShetlandIslands	EUR-G	74
England, Ireland, Scotland, ShetlandIslands	EUR-K	75
Greece	EUR-B	76
Iran	EUR-H	77
ItalySardinia	EUR-I	78
ItalySicily	EUR-J	79
Malta	EUR-L	80
Norway, Finland	EUR-C	81
Portugal, Spain	EUR-D	82
Tunisia	EUR-T	83
EUROPEAN_1979		
MeanSolution	EUS	84
HJORSEY_1955		
Iceland	HJO	85
IRELAND_1965		
Ireland	IRL	86
ORDNANCE SURVEY OF GREAT BRITAIN 1936		
MeanSolution	OGB-M	87
England	OGB-A	88
England, IsleOfMan, Wales	OGB-B	89
Scotland, ShetlandIslands	OGB-C	90
Wales	OGB-D	91
ROME_1940		
Sardinia	MOD	92
S-42(PULKOVO_1942)		
Hungary	SPK-A	93
Poland	SPK-B	94
Czechoslovakia	SPK-C	95
Latvia	SPK-D	96
Kazakhstan	SPK-E	97
Albania	SPK-F	98
Romania	SPK-G	99

EUROPE		
REGION	CODE	CDB-ID VALUE
S-JTSK		
Czechoslovakia	CCD	100

Table 53. North America geodetic datum

NORTH AMERICA		
REGION	CODE	CDB-ID VALUE
CAPE_CANAVERAL		
MeanSolution (Florida, Bahamas)	CAC	101
NORTH AMERICAN 1927		
MeanSolution	NAS-C	102
WesternUnitedStates	NAS-B	103
EasternUnitedStates	NAS-A	104
Alaska (ExcludingAleutianIslands)	NAS-D	105
AleutianIslands(East180° W)	NAS-V	106
AleutianIslands(West180° W)	NAS-W	107
Bahamas (Excluding San Salvador Island)	NAS-Q	108
SanSalvadorIsland	NAS-R	109
CanadaMeanSolution(Including Newfoundland)	NAS-E	110
Alberta, BritishColumbia	NAS-F	111
EasternCanada	NAS-G	112
Manitoba, Ontario	NAS-H	113
NorthwestTerritories, Saskatchewan	NAS-I	114
Yukon	NAS-J	115
CanalZone	NAS-O	116
Caribbean	NAS-P	117
CentralAmerica	NAS-N	118
Cuba	NAS-T	119
Greenland	NAS-U	120
Mexico	NAS-L	121
NORTH AMERICAN 1983		
Alaska (ExcludingAleutianIslands)	NAR-A	122
Aleutian Islands	NAR-E	123
Canada	NAR-B	124
CONUS	NAR-C	125
Hawaii	NAR-H	126
Mexico,Central America	NAR-D	127

Table 54. South America geodetic datum

SOUTH AMERICA		
REGION	CODE	CDB-ID VALUE
BOGOTA OBSERVATORY		
Colombia	BOO	128
CAMPO NCHAUSPE 1969		
Argentina	CAI	129
CHUA ASTRO		
Paraguay	CHU	130
CORREGO ALEGRE		
Brazil	COA	131
PROVISIONAL SOUTH AMERICAN 1956		
MeanSolution	PRP-M	132
Bolivia	PRP-A	133
Northern Chile (near 19° S)	PRP-B	134
Southern Chile (near 43° S)	PRP-C	135
Colombia	PRP-D	136
Ecuador	PRP-E	137
Guyana	PRP-F	138
Peru	PRP-G	139
Venezuela	PRP-H	140
PROVISIONAL SOUTH CHILEAN		
Southern Chile (near 53° S)	HIT	141
SOUTH AMERICAN 1969		
MeanSolution	SAN-M	142
Argentina	SAN-A	143
Bolivia	SAN-B	144
Brazil	SAN-C	145
Chile	SAN-D	146
Colombia	SAN-E	147
Ecuador (Excluding Galapagos Islands)	SAN-F	148
Baltra, Galapagos Islands	SAN-J	149
Guyana	SAN-G	150
Paraguay	SAN-H	151
Peru	SAN-I	152
Trinidad and Tobago	SAN-K	153
Venezuela	SAN-L	154
SOUTH AMERICAN GEOCENTRIC REFERENCE SYSTEM(SIRGAS)		
South America	SIR	155
ZANDERIJ		
Suriname	ZAN	156

Table 55. **Atlantic Ocean geodetic datum**

ATLANTIC OCEAN		
REGION	**CODE**	**CDB-ID VALUE**
ANTIGUA ISLAND ASTRO 1943		
Antigua, Leeward Islands	AIA	157
ASCENSION ISLAND 1958		
Ascension Island	ASC	158
ASTRO DOS 71/4		
St.Helena Island	SHB	159
BERMUDA 1957		
Bermuda Islands	BER	160
CAPE CANAVERAL		
Mean Solution (Bahamas and Florida)	CAC	101
DECEPTION ISLAND		
Deception Island and Antarctica	DID	161
FORT THOMAS 1955		
Nevis, St.Kitts and Leeward Islands	FOT	162
GRACIOSA BASE SW 1948		
Faial, Graciosa, Pico, SaoJorge and Terceira Islands (Azores)	GRA	163
HJORSEY 1955		
Iceland	HJO	85
ISTS 061 ASTRO 1968		
South Georgia Island	ISG	164
L.C. 5 ASTRO 1961		
Cayman Brac Island	LCF	165
MONTSERRAT ISLAND ASTRO 1958		
Montserrat and Leeward Islands	ASM	166
NAPARIMA,BWI		
Trinidad and Tobago	NAP	167
OBSERVATORIO METEOROLOGICO 1939		
Corvo and Flores Islands (Azores)	FLO	168
PICO DE LAS NIEVES		
Canary Islands	PLN	169
PORTO SANTO 1936		
Porto Santo and Madeira Islands	POS	170
PUERTO RICO		
Puerto Rico and Virgin Islands	PUR	171
QORNOQ		
South Greenland	QUO	172
SAO BRAZ		
Sao Miguel and Santa Maria Islands (Azores)	SAO	173

ATLANTIC OCEAN		
REGION	CODE	CDB-ID VALUE
SAPPER HILL 1943		
East Falkland Island	SAP	174
SELVAGEM GRANDE 1938		
Salvage Islands	SGM	175
TRISTAN ASTRO 1968		
Tristan da Cunha	TDC	176

Table 56. Indian Ocean geodetic datum

INDIAN OCEAN		
REGION	CODE	CDB-ID VALUE
ANNA 1 ASTRO 1965		
Cocos Islands	ANO	177
GAN 1970		
Republic of Maldives	GAA	178
ISTS 073 ASTRO 1969		
Diego Garcia	IST	179
KERGUELEN ISLAND 1949		
Kerguelen Island	KEG	180
MAHE 1971		
Mahe Island	MIK	181
REUNION		
Mascarene Islands	REU	182

Table 57. Pacific Ocean geodetic datum

PACIFIC OCEAN		
REGION	CODE	CDB-ID VALUE
AMERICAN SAMOA 1962		
American Samoa Islands	AMA	183
ASTRO BEACON "E" 1945		
Iwo Jima	ATF	184
ASTRO TERN ISLAND (FRIG) 1961		
Tern Island	TRN	185
ASTRONOMICAL STATION 1952		
Marcus Island	ASQ	186
BELLEVUE (IGN)		
Efate and Erromango Islands	IBE	187
CANTON ASTRO 1966		
Phoenix Islands	CAO	188

PACIFIC OCEAN		
REGION	CODE	CDB-ID VALUE
CHATHAM ISLAND ASTRO 1971		
Chatham Island (New Zealand)	CHI	189
DOS 1968		
Gizo Island (New Georgia Islands)	GIZ	190
EASTER ISLAND 1967		
Easter Island	EAS	191
GEODETIC DATUM 1949		
New Zealand	GEO	192
GUAM 1963		
Guam	GUA	193
GUX I ASTRO		
Guadalcanal Island	DOB	194
INDONESIAN 1974		
Indonesia	IDN	51
JOHNSTON ISLAND 1961		
Johnston Island	JOH	195
KUSAIE ASTRO 1951		
CarolineIslands, Fed.States of Micronesia	KUS	196
LUZON		
Philippines (Excluding Mindanao Island)	LUZ-A	197
Mindanao Island	LUZ-B	198
MIDWAY ASTRO 1961		
Midway Islands	MID_A	199
Midway Islands	MID_B	200
OLD_HAWAIIAN		
Mean Solution	OHA-M	201
Hawaii	OHA-A	202
Kauai	OHA-B	203
Maui	OHA-C	204
Oahu	OHA-D	205
OLD HAWAIIAN		
Mean Solution	OHI-M	206
Hawaii	OHI-A	207
Kauai	OHI-B	208
Maui	OHI-C	209
Oahu	OHI-D	210
PITCAIRN ASTRO 1967		
Pitcairn Island	PIT	211
SANTO (DOS) 1965		
Espirito Santo Island	SAE	212

PACIFIC OCEAN		
REGION	CODE	CDB-ID VALUE
VITI LEVU 1916		
VitiLevuIsland (Fiji Islands)	MVS	213
WAKE-ENIWETOK 1960		
Marshall Islands	ENW	214
WAKE ISLAND ASTRO 1952		
Wake Atoll	WAK	215

Table 58. Non-Satellite Derived Transformation Parameter geodetic datum

Non-Satellite Derived Transformation Parameter		
REGION	CODE	CDB-ID VALUE
BUKIT RIMPAH		
Bangka and Belitung Islands (Indonesia)	BUR	216
CAMP AREA ASTRO		
Camp McMurdo Area, Antarctica	CAZ	217
EUROPEAN1950		
Iraq, Israel, Jordan, Kuwait, Lebanon, Saudi Arabia, Syria	EUR-S	218
GUNUNG SEGARA		
Kalimantam (Indonesia)	GSE	219
HERAT NORTH		
Afghanistan	HEN	220
HERMANNSKOGEL		
Slovenia, Croatia, Bosnia and Herzegovina, Serbia	HER	221
INDIAN		
Pakistan	IND_P	222
PULKOVO 1942		
Russia	PUK	223
TANANARIVE OBSERVATORY 1925		
Madagascar	TAN	224
VOIROL 1874		
Tunisia, Algeria	VOI	225
YACARE		
Uruguay	YAC	226

Table 59. Terrestrial Reference Systems geodetic datum

Terrestrial Reference Systems		
	CODE	CDB-ID VALUE
GLONASS		
PZ90.2	PZ90_2	227
PZ90.11	PZ90_11	254

Appendix B RxNetworks Teseo-LIV4F credential

The table below reports the Teseo-LIV4F credential to access the RxNetworks AGNSS Web Server.

Credential access has to be used as described in the 'AN5160: RxNetworks Assisted GNSS Server Interface Specification'.

Table 60. Teseo-LIV4F credential access on RxNetworks Assisted GNSS Server

String	Value
Server address	stm.api.location.io:80
<cld>	ZYDLLXxEH94dEeX2
<mld>	MYST

Revision history

Table 61. **Document revision history**

Date	Version	Changes
01-Jul-2022	1	Initial release.

Contents

List of figures

List of tables