GSG-5/6 Series GNSS Simulator

User Manual
with SCPI Guide

Orolia Part No.: 4031-600-54001
Revision: 27
Date: 22-Oct-2019

orolia.com

Orolia USA, Inc. dba Spectracom

• 1565 Jefferson Road, Suite 460, Rochester, NY 14623 USA
• 3, Avenue du Canada, 91974 Les Ulis Cedex, France

The industry-leading Spectracom products you depend on are now brought to you by Orolia, the global leader in Resilient Positioning, Navigation and Timing Solutions.

Do you have questions or comments regarding this User Manual?

➔ E-mail: techpubs@orolia.com

Warranty Information

For a copy of Orolia's Limited Warranty policy, see the website: http://www.orolia.com/support/spectracom/warranty-information.

Blank page.

CONTENTS

CHAPTER 6

APPENDIX

INDEX

CHAPTER 1

Introduction

The following topics are included in this Chapter:

1.1 Quick Start

The following procedure is a brief outline on how to get started with your GSG-5/6 unit.

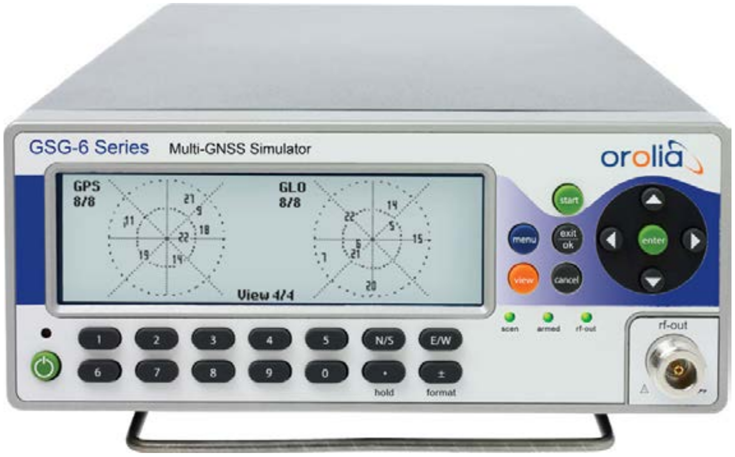
The minimal setup steps are:

1. Unpack the unit (see "Unpacking and Inventory" on page 14), and place it on a desktop or install it in a rack, as described under "Mechanical Installation" on page 15.
2. Connect the receiver antenna cable to the RF Out connector on the front panel. (See also "Electrical Installation" on page 22.)
3. Connect the power cable to a wall socket. Press the ON/OFF key to start the unit.
4. The GSG display will show the **Start** view: Verify that the right-hand side shows an overview of a test scenario (name, date, lat/long/traj, etc.).
5. If no scenario is shown, use the arrow and enter keys to select **Select** from the main menu. This will open up a list of pre-defined scenarios. Select one of the scenarios from this list.
6. Press the start key to begin with the scenario execution.
7. Start the GNSS receiver you want to test.

   > **Note:** It may be necessary to clear the memory of your GNSS receiver, i.e. erase old data. This is typically referred to as a Cold Start, where any ephemeris data and almanac data are removed from the receiver's memory.

8. Your GNSS receiver under test should see and track the generated signals. If the receiver could successfully decode the navigation data included in the signals (this process often takes approximately 40 seconds), the receiver will output the navigation fix as specified in the selected scenario. This navigation solution should correspond to the solution shown on the GSG-5/6 display.

1.2 Welcome



About Orolia's GNSS Simulators

The GSG-5™ and GSG-6™ Series of GNSS Constellation Simulators provide a wide-range of capabilities for in-line production testing and development testing, including navigational fix and position testing, while offering ease-of-operation.

GSG-51 is a single-channel GPS L1 RF generator, capable of emulating a single GNSS signal. One of the main applications for these cost-effective units is fast manufacturing testing of GPS receivers.

GSG-5 Series simulators reproduce the environment of a GNSS receiver. Depending on the configuration, these units simulate up to sixteen GNSS satellites, up to 3 SBAS satellites, together with optional multipath and interference signals. The GSG-5 Series applies models to simulate satellite motions, atmospheric effects, and different antenna types. The movement of the GNSS receiver under test is defined using NMEA data or pre-defined trajectory models.

GSG-6 Series simulators add advanced features and the capability to simulate up to 64 satellites (configuration-dependent) on different frequency bands simultaneously. New signal types include GPS L2P, L2C and L5, GLONASS L2, Galileo E1 and E5a/b, BeiDou B1 and B2, and QZSS L1 C/A, L2C, L5 and L1 SAIF, IRNSS L5.

1.3 Key Features

Since GNSS testing requirements may vary considerably from application to application, GSG Series simulators are available in a multitude of configurations (see "GSG Series Model Variants and Options" on page 203).

Some of the key features are:

- Up to 64 independent satellite channels can be simulated.
- Supported signal types:
  - GPS L1, L2, C/A and P-Code; L2C and L5
  - GLONASS L1, L2, C/A and P-Code
  - Galileo E1/E2 and E5
  - BeiDou compatible
- Support of different types of SBAS simulation: EGNOS, WAAS, MSAS, GAGAN
- Generation of white noise, multipath and interference signals
- Receiver sensitivity testing with accurate, variable output levels ranging from -65 to -160 dBm
- High accuracy time base

GSG Series simulators offer a front panel display with an intuitive software User Interface, allow for remote Web-based operation, and include GSG StudioView™, a PC-based software with Google Maps™ interface to create custom scenarios.

1.4 Typical GSG Applications

GSG-5/6 Series GNSS Simulator are often used for the following testing applications:

Basic Receiver Testing

- **Time-to-First-Fix** (TTFF): How fast can a GNSS receiver obtain a position fix after a cold start.
- **Reacquisition Time**: How fast can a GNSS receiver get a fix after a hot or warm start.
- **Location**: Test position accuracy at different locations in the world.
- **Sensitivity**: Acquisition and Tracking Sensitivity
- **Noise Susceptibility**: SNR limit testing

Advanced Receiver Testing

- **Trajectories**: Test receiver while moving
- **1PPS**: Verify the receiver timing accuracy
- **Leap Second**: Test the leap second handling of the receiver
- **Multipath**: Perform basic receiver tests under multipath conditions

1.5 Intended Use and Operating Principle

DANGER! If the equipment is used in a manner not specified by the manufacturer, the protection provided by the equipment may be impaired.

Orolia GSG-Series Signal Generators and GNSS Simulators are used to test GNSS receivers by generating GNSS signals, as they are transmitted by GNSS satellites. The signals are transmitted via air (using an antenna; see "Signal Power Level Considerations" on page 24), or via an RF cable.

Depending on the model, and the options installed in a GSG unit, generated/simulated signals, as well as user position, time and output power can be manipulated by the user either:

- during the test, i.e. in real-time, via the GSG front panel, or
- before beginning the test, by saving the programmed signal data (as well as trajectory data, if the receiver is to be tested under virtual movement conditions) in scenario files, using the optional StudioView™ software.

In addition to GNSS, other signals such as interference and multi-path can be generated to test the sensitivity to various disruptions.

The number of channels installed in a GSG unit determines how many signals can be generated. If more channels are required than available, two or more GSG units can be synchronized to generate 128, 256, or more signals.

Built-in trajectories (static, configurable circle, and rectangular as defined in 3GPP TS 25.171) or user-designed trajectories (in NMEA standard format) can be run on GSG simulators. Users can upload their own ephemeris data in standard RINEX format or re-use the default data for any time periods. The GSG-6 Series is capable of automatically downloading historical RINEX, WAAS and EGNOS data from official websites, as needed.

The GSG-6 Series can be controlled via an Ethernet network connection, or USB or GPIB. A built-in web interface allows remote operation of the instrument. With the optional GSG StudioView™ PC Software, you can build, edit, and manage the most complex scenarios, including building trajectories via Google Maps, independent of the GSG unit, for later upload.

Besides the variety of built-in navigation/positioning tests, GSG units are also suited for accurate testing of timing GNSS-receivers. The GSG-6 is equipped with an ultra-high-stability OCXO timebase for precision timing of the satellite data, or use external synchronization from a 10 MHz reference from e.g. a Cesium or Rubidium clock. A built-in 1PPS output, synchronized to the generated satellite data, allows comparison with the 1PPS signal from the timing receiver under test.

1.6 Compliance & Legal Notices

Orolia's GSG-Series GNSS Simulator products meet all FCC and CE Mark regulations for operation as electronic test equipment.

> **Note:** For more information about Signal Power Emissions, see "Signal Power Level Considerations" on page 24.

> **Note:** For more information about Software Licensing, see "License Notices" on page 207.

In particular, this instrument has been designed and tested for Measurement Category I, Pollution Degree 2, in accordance with EN/IEC 61010-1:2001 and CAN/CSA-C22.2 No. 61010-1-04 (including approval). It has been supplied in a safe condition.

1.6.1 About this Document

This GSG-5/6 Series User Manual contains directions and reference information for use that applies to the GSG-5/6 Series products.

Study this manual thoroughly to acquire adequate knowledge of the instrument, especially the section on Safety Precautions hereafter and the Installation section.

1.6.2 Declaration of Conformity

A copy of the Declaration of Conformity will be shipped with your unit. The complete text with formal statements concerning product identification, manufacturer and standards used for type testing is available on request.

1.7 Technical Specifications

1.7.1 RF Output Specifications

- **RF** Constellation **Signal** for GPS, GLONASS, Galileo, BeiDou, QZSS, IRNSS
- **Connector**: Type N female
- **Frequency**:
  - L1/E1/B1/SAR: 1539 - 1627 MHz
  - L2/L2C: 1167 - 1255 MHz
  - L5/E5/B2: 1146 - 1234 MHz
  - E6/B3: 1215 - 1303 MHz
- **Number of output channels**: 1 to 64
- **Channel configuration**:
  - Any channel can be GPS, GLONASS, Galileo, BeiDou, QZSS, IRNSS
  - GLONASS freq ch -7 to +6
  - Up to 3 SBAS satellites (instead of 1-3 GNSS satellites)
- **Data format**:
  - 50 bits/s, GPS, Galileo OS, GLONASS frame structure
  - GPS CNAV
  - 250 bits/s, SBAS
- **PRN codes**: 1 to 210, plus GLONASS
- **Spurious transmission**: <-40 dBc
- **Harmonics**: <-40 dBc
- **Output signal level**:
  - -65 to -160 dBm;
  - 0.1 dB resolution down to -150 dBm;
  - 0.3 dB down to -160 dBm.
- **Power accuracy**: ±1.0 dB
- **Pseudorange accuracy** within any one frequency band: 1mm
- **Pseudorange accuracy** across different frequency bands: 30 cm
- **Inter-channel bias**: Zero
- **Inter-channel range**: >54 dB

- **Limits**:
  - **Altitude**: 18240 m (60000 feet)
  - **Acceleration**: 4.0 g
  - **Velocity**: 515 m/s (1000 knots)
  - **Jerk**: 20 m/s^3
- **Extended limits**:
  - **Altitude**: 20200 km
  - **Acceleration**
  - **Velocity**: 20000 m/s (38874 knots)
  - **Jerk**: No limit
- **White noise signal level**:
  - -50 to -160 dBm
  - 0.1 dB resolution down to -150 dBm
  - 0.3 dB down to -160 dBm
  - ±1.0 dB accuracy

1.7.2 Rear Panel Outputs and Inputs

- **External Frequency Reference Input**
  - **Connector**: BNC female
  - **Frequency**: 10 MHz nominal
  - **Input signal level**: 0.1 to $5V_{rms}$
  - **Input impedance**: >1kΩ
- **Frequency Reference Output**
  - **Connector**: BNC female
  - **Frequency**: 10 MHz sine
  - **Output signal level**: $1V_{rms}$ into 50 Ω load
- **External Trigger Input**
  - **Connector**: BNC female
  - **Signal Type**: Single pulse
  - **Level**: TTL level, 1.4 V nominal
  - **Input impedance**: >1kΩ

- **Minimum PW**: 10 ms
- **Active Edge**: Falling

- **1/10/100/1000 PPS Output**
  - **Connector**: BNC female
  - **Output signal level**: approx. 0V to +2.0 V in 50 Ω load
  - **Accuracy**: Calibrated to ±10 nSec of RF timing mark output

1.7.3 Time Base

- **Standard OCXO**
  - **Ageing per 24 h**: <5x10^{-10}
  - **Ageing per year**: <5x10^{-8}
  - **Temp. variation 20 ... 50°C**: <5x10^{-9}
  - **Short term stability** (A_{dev} @1s): <5x10^{-12}

1.7.4 Optional Antenna

- **Frequency**: 1000 MHz to 2600 MHz
- **Impedance**: 50 Ω
- **VSWR**: <2:1 (typ.)
- **Connector**: SMA male
- **Dimensions**: 15 mm diameter x 36 mm length

1.7.5 Environmental Specifications

- **Environmental Data**
  - **Class**: MIL-PRF-28800F, Class 3
  - **Operating Temp.**: 0°C ... +50°C
  - **Storage Temp.**: -40°C ... +70°C, non-condensing, @ <12000 m
  - **Humidity**: 5-95% @ 10...30°C, 5-75% @ 30...40°C, 5-45% @ 40...50°C
  - **Max. Altitude**: 2000 m
  - **Vibration**: Random and sinusoidal according to MIL-PRF-28800F, Class 3
  - **Shock**: Half-sine 30 g per MIL-PRF-28800F, Bench handling

   - **Transit Drop Test**: Heavy-duty transport case and soft carrying case tested according to MIL-PRF-28800F
   - **Reliability**: MTBF 30000 h, calculated
   - **Safety**: Designed and tested for Measurement Category I, Pollution Degree 2, in accordance with EN/IEC 61010-1:2001 and CAN/CSA-C22.2 No. 61010-1-04 (incl. approval)
   - **EMC**: EN 61326 (1997) A1 (1998), increased test levels per EN 50082-2, Group 1, Class B, CE
- **Power Requirements**
   - **Line Voltage**: 100-240 V_{AC}, 50/60/400 Hz
   - **Power Consumption**: 40 W max.
- **Dimensions & Weight**
   - **Width**: ½ x 19" (215 mm)
   - **Height**: 2U (90 mm)
   - **Depth**: 395 mm
   - **Weight**: Net 2.7 kg (5.8 lb)
   - **Shipping**: 3.5 kg (7.5 lb)

CHAPTER 2

Setup

The following topics are included in this Chapter:

2.1 About Your Safety

The following safety symbols are used in Orolia technical documentation, or on Orolia products:

Table 2-1: Orolia safety symbols

Symbol	Signal word	Definition
	DANGER!	Potentially dangerous situation which may lead to personal injury or death! Follow the instructions closely.
	CAUTION!	Potential equipment damage or destruction! Follow the instructions closely.
	NOTE	Tips and other useful or important information.
	ESD	Risk of Electrostatic Discharge! Avoid potential equipment damage by following ESD Best Practices.
	PROTECTIVE GROUND	Shows where the protective ground terminal is connected inside the instrument. Never remove or loosen this screw!
	FUNCTIONAL GROUND	Functional (noiseless, clean) grounding, designed to avoid malfunction of the equipment.
	CHASSIS GROUND	A terminal always connected to the instrument chassis.

2.1.1 Safety Precautions

This product has been designed and built in accordance with state-of-the-art standards and the recognized safety rules. Nevertheless, all equipment that can be connected to line power is a potential danger to life. In particular, its use may constitute a risk to the operator or installation/maintenance personnel, if used under conditions that must be deemed unsafe, or for purposes other than the product's designated use, as it is described under "Intended Use and Operating Principle" on page 5.

DANGER! If the equipment is used in a manner not specified by the manufacturer, the protection provided by the equipment may be impaired.

2.1.2 Basic User Responsibilities

To ensure the correct and safe operation of the instrument, it is essential that you follow generally accepted safety procedures in addition to the safety precautions specified in this manual.

The instrument is designed to be used by trained personnel only. Removing the cover for repair, maintenance, and adjustment of the instrument must be done by qualified personnel who are aware of the hazards involved.

> **Note:** The warranty commitments are rendered void if unauthorized access to the interior of the instrument has taken place during the given warranty period.

Also, follow these general directions:

- The equipment must only be used in technically perfect condition. Check components for damage prior to installation. Also check for loose or scorched cables on other nearby equipment.
- Make sure you possess the professional skills, and have received the training necessary for the type of work you are about to perform (for example: Best Practices in ESD prevention.)
- Do not modify the equipment, and use only spare parts authorized by Orolia.
- Always follow the instructions set out in this guide.
- Observe generally applicable legal and other local mandatory regulations.
- Keep these instructions at hand, near the place of use.

2.1.3 If in Doubt about Safety

Apply technical common sense: If you suspect that it is unsafe to use the product (for example, if it is visibly damaged), do the following:

- Disconnect the line cord.
- Clearly mark the equipment to prevent its further operation.
- Contact your local Orolia representative.

2.2 Unpacking and Inventory

Caution: Electronic equipment is sensitive to Electrostatic Discharge (ESD). Observe all ESD precautions and safeguards when handling the unit.

Unpack the equipment and inspect it for damage. If any equipment has been damaged in transit, or you experience any problems during installation and configuration of your Orolia product, please contact your closest Orolia Customer Service Center (see: "Technical Support" on page 207).

Note: Retain all original packaging for use in return shipments if necessary.

The following items are included with your shipment:

- GSG-5x/6x GNSS Simulator
- Ancillary kit, GSG-5x/6x, containing:
    - AC cord, 5-15P to C13, 18 AWG, 10 A, 125 V
    - Adapter, SMA female–N male, 50 Ω
    - Cable assembly, SMA–SMA, 5ft.
    - USB 2.0 cable, with type A/B connector, 6ft.
- CD with user's manual, Protocol reference document & configuration SW
- Compliance and shipping documentation
- Optional: additional software and license key(s)

2.2.1 Unit Identification

The **type plate** on the rear panel (see "Rear Panel" on page 31) of the unit includes the GSG MODEL, PART No., and SERIAL No.

This information, as well as a list of installed options (if any), can also be found under the menu item **Options** > **Show system information**.

2.3 Mechanical Installation

2.3.1 General Installation Considerations

Orientation

GSG-Series units can be operated in any position, i.e. horizontal, vertical, or at any angle.

Cooling

The air flow through the side ventilation openings must not be obstructed.

Leave 5 cm (2") of space around the unit.

Bench-Top Setup

For bench-top use, a fold-down support is available for use underneath the GNSS Simulator. This support can also be used as a handle to carry the instrument.

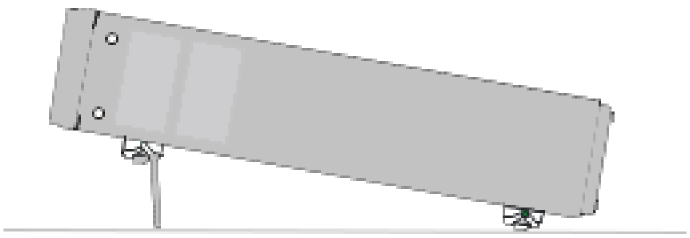


Figure 2-1: Fold-down support

Single-Unit Rack-Mount Installation

With the optional Orolia 22/90 rack-mount kit (P/N **9446-1002-2901**) one GSG unit can be installed in a 19-inch rack (2U). The kit comprises:

- 2 ears, one of which with a pre-assembled face-plate spacer
- 4 screws, M5 x 8
- 4 screws, M6 x 8.

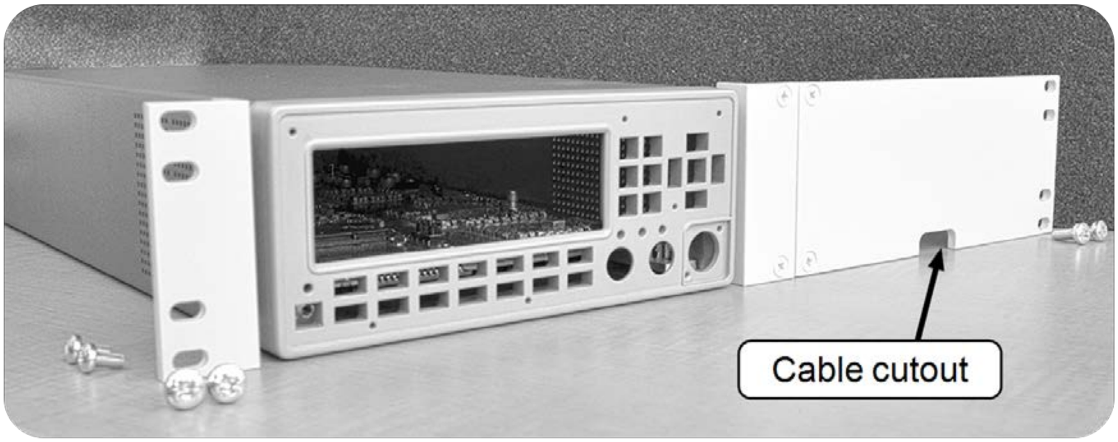


Figure 2-2: Rack-Mount-Kit (the GSG housing shown in the center is not part of the kit)

In order to prepare the GSG unit for rack-mount installation, the housing needs to be opened, in order to remove the bottom feet (otherwise the assembly will not fit in a 2U slot.)

DANGER! Do not perform any work on the internal components of the unit, while the housing is removed, unless you are qualified to do so. Before removing the cover, unplug the power cord and wait for one minute to allow any capacitors to discharge.

1. After making sure that the power cord has been unplugged, carefully turn the unit upside down.
2. Temporarily remove the two **rear feet** by loosening their screws.
3. Remove the four **housing screws** and plugs (if present) at the side panels; discard them.
4. Grip the front panel with one hand, while pushing at the rear with the other hand. Pull the unit out of its housing.
5. Remove the four **bottom feet** from the housing, as shown in the illustration below: Use a screwdriver or a pair of pliers to remove the springs holding each foot, then

push out the foot.

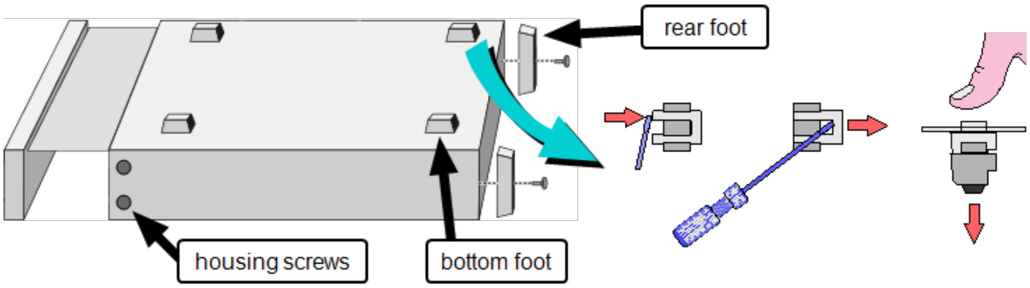


Figure 2-3: Preparing the GSG unit for rack mounting

6. Gently push the unit into its housing again.
7. Re-assemble the two **rear feet**.
8. Install the **ears** that came with the rack-mount kit. Use the rack-mount kit M5 **housing screws**.

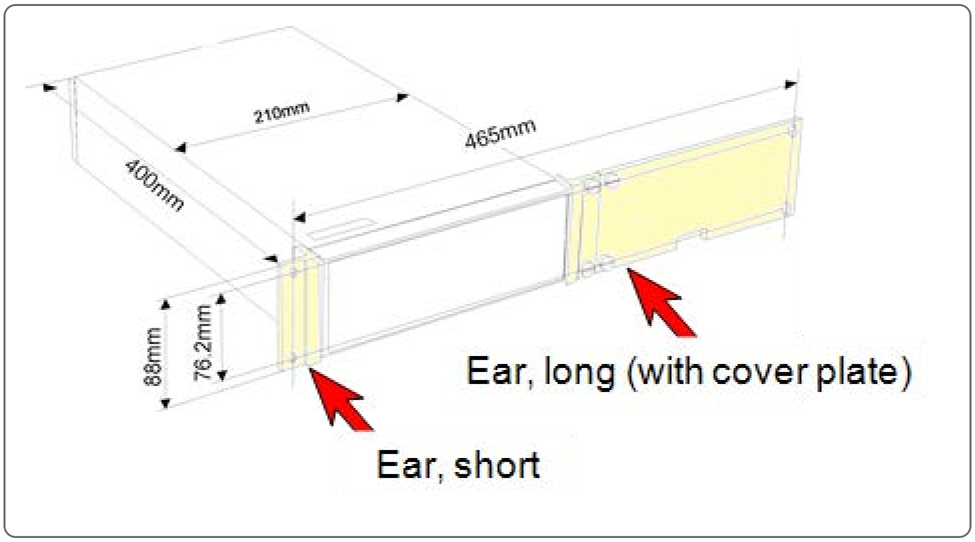


Figure 2-4: Part identification: ears

Note: The unit can also be installed on the right-hand side of the rack by reversing the two ears.

9. Depending on accessibility in your rack, you can connect the cables to the GSG unit now, or after installation of the assembly in the rack. For electrical installation, see "Electrical Installation" on page 22.

10. Install the assembly in your rack, using the M6 screws that came with the rack-mount kit.
11. Complete the electrical installation.

Side-by-Side Rack-Mount Installation

With the optional Orolia 22/05 rack-mount kit (P/N **1211-0000-0701**), two GSG units can be installed side-by-side in one 19-inch rack (2U). The kit comprises:

- 4 x Bracket, rear (1211-1000-0706) [Item 1]
- 2 x Ear, rack (1211-1000-0714) [Item 2]
- 1 x Hinge, right half (1211-1000-0709) [Item 3]
- 1 x Hinge, left half (1211-1000-0709) [Item 4]
- 8 x Screw, oval head phil, M5x10mm (HM25R-D5R8-0010) [Item 5]
- 2 x Screw, pan head phil, M4x8mm (HM10R-04R0-0008) [Item 6]
- 1 x Spacer, Hex, M4x16 (HM50R-04R0-0016) [Item 7]

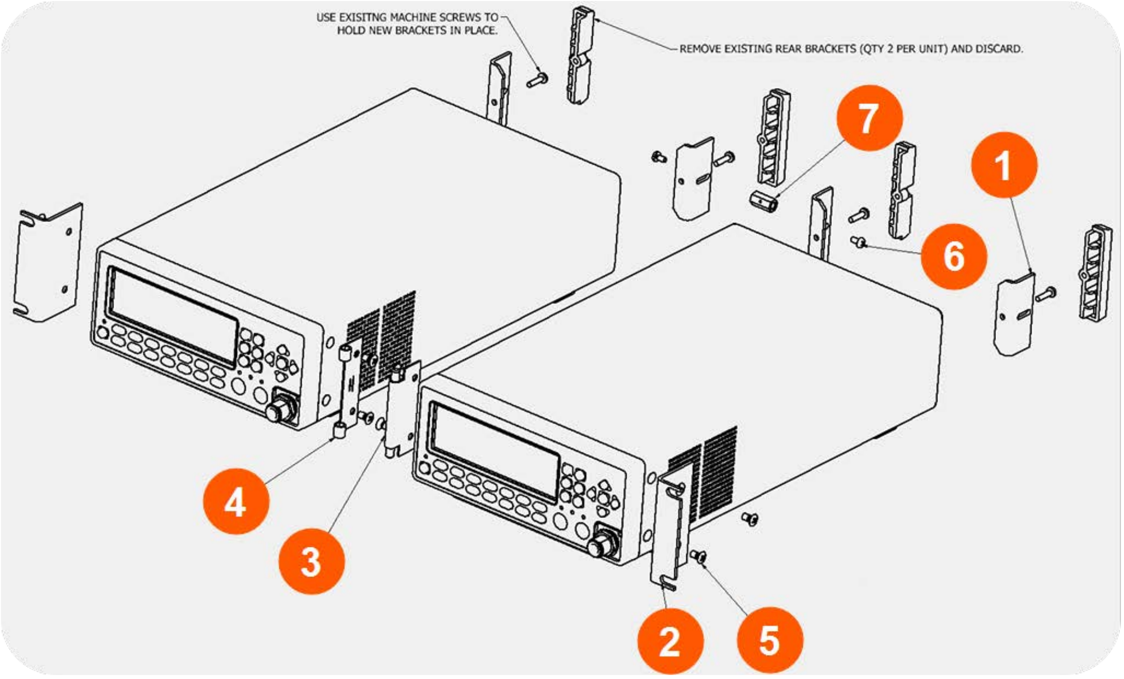


Figure 2-5: Dual rack-mount assembly

In order to prepare the GSG units for rack mount installation, the housings needs to be opened, in order to remove the bottom feet (otherwise the assembly will not fit in a 2U slot.)

DANGER! Do not perform any work on the internal components of a GSG unit, while the housing is removed, unless you are qualified to do so. Before removing the cover, unplug the power cord and wait for one minute to allow any capacitors to discharge.

1. After making sure that the power cord has been unplugged, carefully turn the first GSG unit upside down.
2. Remove the two **rear feet**. Keep the screws, discard the brackets.
3. Remove the four **housing screws** and plugs (if present) at the side panels, and discard them.
4. Grip the front panel with one hand, while pushing at the rear with the other hand. Pull the unit out of its housing.
5. Remove the four **bottom feet** from the housing, as shown in the illustration below: Use a screwdriver or a pair of pliers to remove the springs holding each foot, then push out the foot.

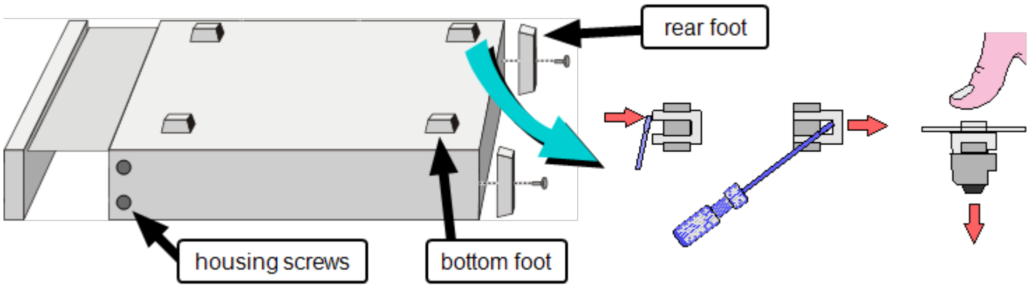


Figure 2-6: Preparing a GSG unit for rack mounting

6. Gently push the unit into its housing again.
7. Install the rear brackets supplied with the mounting kit (item no. 1) where the rear feet were previously attached (see illustration "Dual rack-mount assembly" above). Use the screws saved in step 2.
8. Repeat the procedure described above for the second unit.
9. Using a Philips-head screwdriver, screw the rack ears (item no. 2) into place, using the supplied 10-mm screws (item no. 5).
10. Pinch the hinge pins together, to separate the right and left hinge halves (items no. 3 and 4).
11. Attach hinge halves to the unit with the hinge facing towards the front.

12. Pinch the hinge pins together into the stored position. Align the hinge halves together between the two units, and swing together side by side. The hinge pins should snap into place, securing the front of the two units.
13. In the back of the unit, take the supplied Hex Spacer (item no. 7), and place between middle rear brackets, and secure using the supplied 8-mm screws (item no. 6).
14. Assembly is now ready for installation into standard 19" rack.
15. Depending on accessibility, you can complete the electrical installation before or after installing the assembly in the rack. For electrical installation, see "Electrical Installation" on page 22.

Rack-Mount Installation with an Agilent Power Meter

GSG units are frequently installed adjacent to an **Agilent Power Meter**, using one 19" slot (2U). This can be accomplished with the optional Orolia 22/04 rack-mount kit (P/N **9446-1002-2041**). Also required is the Agilent rack-mount kit.

Note: This kit can also be used to install only one GSG unit in a 19" rack 2U slot, similar to the optional Orolia 22/90 Rack-Mount Kit (P/N 9446-1002-2901).

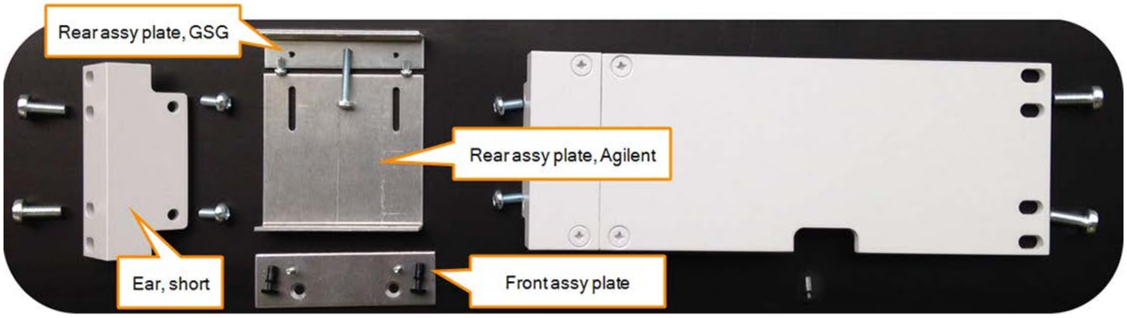


Figure 2-7: 22/04 Rack-mount kit

In order to prepare the GSG unit for rack mount installation, the housing needs to be opened, in order to remove the bottom feet (otherwise the assembly will not fit in a 2U slot.) The same may be necessary for the Agilent unit – follow the manufacturer's instructions.

1. After making sure that the power cord has been unplugged, carefully turn the GSG unit upside down.
2. Temporarily remove the two **rear feet** by loosening their screws.
3. Remove the four **housing screws** and plugs (if present) at the side panels, and discard them.

4. Grip the front panel with one hand, while pushing at the rear with the other hand. Pull the unit out of its housing.
5. Remove the four **bottom feet** from the housing, as shown in the illustration below: Use a screwdriver or a pair of pliers to remove the springs holding each foot, then push out the foot.

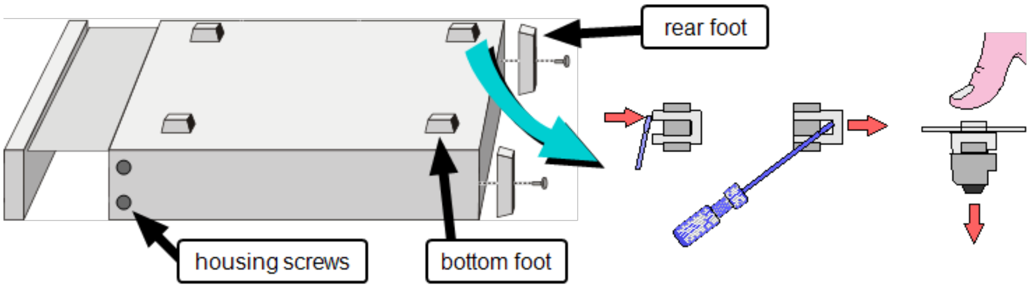


Figure 2-8: Preparing the GSG unit for rack mounting

6. Gently push the unit into its housing again.
7. Re-assemble the two **rear feet**.
8. Decide on which side of the assembly the GSG unit is to be installed: If on the left-hand side, install the **short ear** to the left hand side of the GSG unit, using the rack-mount kit M5 **housing screws**.

> **Note:** The instructions below are based on the assumption that the GSG unit is installed on the left-hand side of the assembly.

9. Install the front assy plate to the Agilent unit, as shown in the illustration below. Use the screws from the Agilent rack-mount kit.
   Take two of the plastic snap caps from the GSG rack-mount kit, remove and discard the caps, and install the sleeves into the housing screw openings.
   Slide the Agilent unit and the GSG unit together, so that the protruding pins of the front assy plate fit into the sleeves.

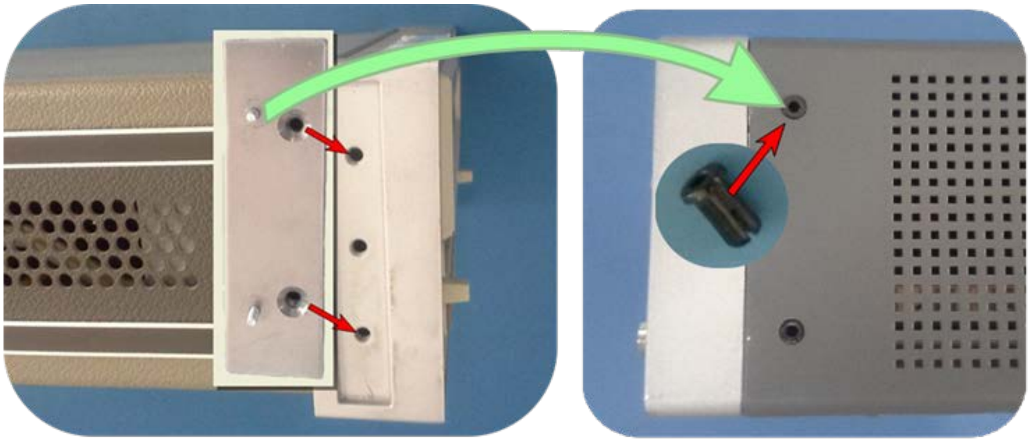


Figure 2-9: Front assembly plate installation Agilent unit (shown left), GSG unit

10. Install the rear assy plate, Agilent, and the rear assy plate, GSG, and assemble them, as shown in the illustrations below.

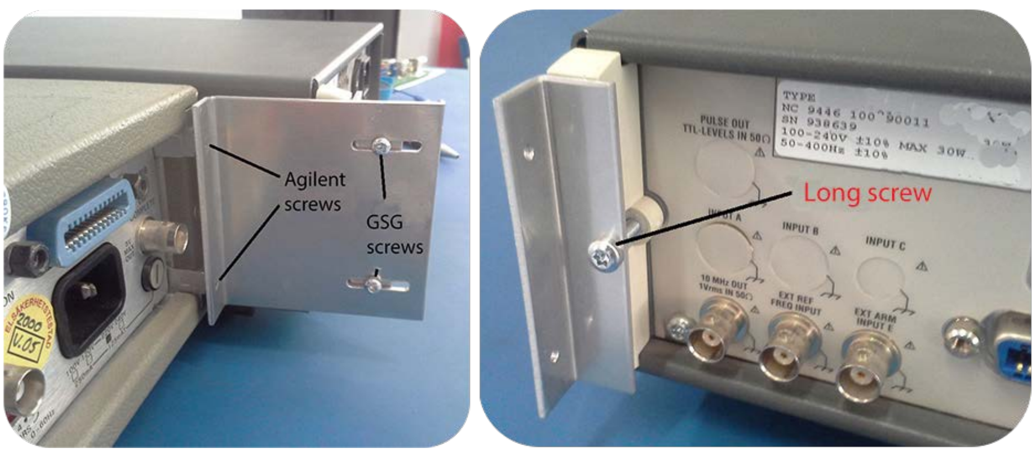


Figure 2-10: Installation of rear assembly plates

11. Equivalent to Step 8., install the front panel ear plate (Agilent rack-mount kit) to the Agilent power meter.
12. The assembly is now complete, and can be installed in the cabinet.

2.4 Electrical Installation

Supply Voltage

GSG Series simulators may be connected to any AC supply with a voltage rating of **100 to 240 V**, at **50/60/400 Hz**. The units automatically adjust themselves to the input line voltage. The maximum power draw is 40 W.

Fuse

The secondary supply voltages are electronically protected against overload or short circuit. The primary line voltage side is protected by a fuse located on the power supply unit. The fuse rating covers the full voltage range. Consequently there is no need for the user to replace the fuse under any operating conditions, nor is it accessible from the outside.

Caution: If this fuse is blown, it is likely that the power supply is badly damaged. Do not replace the fuse. Send the GSG unit to your local Service Center.

DANGER! — Removing the cover for repair, maintenance and adjustment must be done by qualified and trained personnel only, who are fully aware of the hazards involved.

The warranty commitments are rendered void if unauthorized access to the interior of the instrument has taken place during the warranty period.

Grounding

⏚ Grounding faults in the line voltage supply will make any instrument connected to it dangerous. Before connecting any unit to the power line, you must make sure that the protective ground functions correctly. Only then can a unit be connected to the power line and only by using a three-wire line cord. No other method of grounding is permitted. Extension cords must always have a protective ground conductor.

Caution: If a unit is moved from a cold to a warm environment, condensation may cause a shock hazard. Ensure, therefore, that the grounding requirements are strictly met.

DANGER! — Never interrupt the grounding cord. Any interruption of the protective ground connection inside or outside the instrument or disconnection of the protective ground terminal is likely to make the instrument dangerous.

Electrical Connections

For a graphic representation of all electrical connections, see "Rear Panel" on page 31 and "Front Panel" on page 28.

Using any of the **communication interfaces** is not required for GSG to operate in a basic mode. The same applies to the outputs for **1PPS** and **10 MHz**, as well as the inputs **EXT REF FREQ** and **EXT TRIG**: Their usage is not compulsory for basic operation.

The minimum electrical configuration for any test layout requires only the **power cord** and an **RF antenna cable**—or an actual GNSS antenna—to connect the GSG unit to your receiver-under-test (using the front panel RF connector, see "Front Panel" on page 28.)

2.5 Signal Power Level Considerations

2.5.1 Compliance: Using an Antenna

Orolia's GSG GNSS Simulator products meet all required regulations of the FCC and CE Mark for operation as electronic test equipment. However, when using the GSG signal generator with an RF antenna (instead of an RF cable), additional regulations controlling the radiation of GPS-like signals into the air must be taken into account by the user:

In the USA, the GPS spectrum is controlled by the **National Telecommunications and Information Administration (NTIA)**: See Sections 8.3.28 and 8.3.29 of the Manual of Regulations and Procedures for Federal Radio Frequency Management (http://www.ntia.doc.gov/osmhome/redbook/redbook.html).

Depending on your situation, you may need authorization from the FCC to operate at or near the level allowed by the NTIA. A Special Temporary Authorization (STA) or Experimental License may be required. For more information, see the FCC web site: https://fjallfoss.fcc.gov/oetcf/els/.

Countries other than the USA may have their own regulations or restrictions, which you should be aware of and comply with before using the optional antenna.

2.5.2 Transmit Power Level

The U.S. agency **NTIA** (National Telecommunications & Information Administration) restricts the maximum signal level to -140 dBm (24 MHz BW) as received from an isotropic antenna at a distance of 100 feet from the building where the test is being conducted. Therefore, the maximum power level output from the GSG Signal Generator may need to be limited to conform to this regulation. For example, consider the following test setup:

Antenna distance to nearest exterior wall:	100 ft.

Antenna gain:	0 dB (omni antenna)
Cable loss, antenna to GSG:	0 dB (no cable used)

Using the free space loss calculation for radio propagation:

- Loss (dB) = 20 log10 (4л * Distance / λ)
- Where λ = wavelength: @ 1575 MHz= 19 cm = 0.62 ft
  - Distance = 200 ft total => 100 ft from antenna exterior wall + 100 ft to restricted perimeter
- Loss = 72 dB = 20 log10 (4л * 200/0.62)

Using the free space calculation is a worst case scenario as the wall and any other obstructions will likely reduce the signal even more. Therefore, setting the power output of the GSG to:

-140 + 72 = -68 dBm or less will guarantee compliance.

For additional information on path loss, see e.g., this third- party reference [1]: http://en.wikipedia.org/wiki/Path_loss

[1] This link is provided for reference purposes only. It leads to a web page that is not maintained or supported by Orolia.

BLANK PAGE.

CHAPTER 3

Features & Functions

The following topics are included in this Chapter:

3.1 Front Panel

All GSG-5/6 simulators have similar front panels. On the right side are the controls used for managing **scenario execution** and for **display navigation**. At the bottom are the numeric keys used to **input scenario parameters** and other configuration.

Figure 3-1: GSG front panel

There are three **status indicators** on the front panel. When the unit is idle, all three indicators are off.

- scenario will blink when a scenario is running
- armed (or: trig) is lit when the unit is armed, i.e. waiting for a trigger signal to start executing a scenario
- rf-out is lit when there is signal coming out of the RF-connector on the front panel.

Note: The N-type RF-connector is equipped with a DC block to prevent the flow of direct current up to $7V_{DC}$ in order to protect the GSG unit.

3.1.1 Description of Keys

3.1.1.1 Power

The ON/OFF key is a toggling secondary power switch. Part of the instrument is always ON as long as power is applied, this standby condition is indicated by a red LED above the key. This indicator is consequently not lit while the instrument is in operation.

3.1.1.2 Start

- Press start to start the currently selected scenario.
- In the **Signal Generator** menu, press start to start transmitting.

3.1.1.3 Exit

- When editing a field, press exit to end the editing process, and **save your changed field value**. The field label will be highlighted.
- When *not* editing a field, press exit to **return to the previous display**, and **save the changes** you applied to the current display. Confirm your changes.
- When running a scenario, press exit to **stop the scenario** execution (same as cancel).

3.1.1.4 Cancel

- When editing a field, press cancel to abort the editing process, and **discard any field changes**. The field label will be highlighted instead.
- When *not* editing a field, press cancel to **return to the previous display**, and **discard any changes** you applied to the current display. Confirm your cancellation.
- When running a scenario, press cancel to **stop the scenario** execution (same as exit).

3.1.1.5 Menu

- When running a scenario, press menu to display the main scenario configuration (the scenario will continue to run.)
- When reviewing/editing configuration settings, press menu to exit the current sub-menu, and return to the main menu, regardless of the current display. You will be asked to save your changes (same as exit).

3.1.1.6 View

- When running a scenario, press view to toggle between the available views.
- In the **main menu**, pressing view will act as a shortcut to the configuration display of the currently selected scenario.
- In the **Options** menu, press view to make a selection (same as enter).

3.1.1.7 Enter

- Press enter to make a selection.

3.1.1.8 Arrows

- Press any of the arrow keys to navigate in displays.
- When editing an integer value, press the UP/DOWN arrows to incrementally increase or decrease the value.

3.1.1.9 N/S

- When editing **latitude**, press N/S to toggle between north and south latitude.
- During scenario execution, press N/S to open the **transmit power menu**, in order to adjust the scenario's noise settings.

3.1.1.10 E/W

- When editing **longitude**, press E/W to toggle between east and west longitude.
- During scenario execution, press E/W to adjust the units displayed for **Altitude** and **Speed** (m/m/s > ft/kn > ft/mph).

3.1.1.11 Numeric Keys

- Press the numeric keys to input numbers.

3.1.1.12 +/– (format)

- When editing numbers, press +/– (format) to toggle between the **positive and negative** value.
- When configuring or executing a scenario, press +/– (format) to change the **coordinate format** between geodetic coordinates, and ECEF format.
- In scenario execution, **View 2/5** and higher, press +/– (format) to **switch between frequency bands** (L1, L2 and L5).

3.1.1.13 [.] (hold)

- Use the "DOT" [.] (hold) key **together with numeric keys**, where appropriate.
- During scenario execution, press the [.] (hold) key to **hold/resume the simulated movement (trajectory)**.
- While a scenario is loading, press the [.] (hold) key to **initiate a scenario arming** from the front panel.

3.2 Rear Panel

As a means for communication, GSG supports **GPIB**, **USB** and **Ethernet**. Only one connection can be active at a time. The active connection is selected under **Options** > **Interface**. The default setting is **Ethernet**.

The illustration below shows the connections available on the back side of the unit:

Figure 3-2: GSG rear panel

1. **1PPS Output**: TTL level signal with positive slope timed to GPS time of RF out (can be programmed as 10/100/1000PPS).
2. **Reference Output**: 10 MHz derived from the internal or—if present—external reference.
3. **External Reference Input**: Can be selected as a reference via the **Interface and Reference** menu.
4. **External Trigger Input**: Optional signal input for scenario triggering.
5. GPIB **Connector**: The address is set in the **Interface and Reference** menu.
6. **Ethernet Connector**: Data communications port used with TCP/IP networks.
7. **USB Connector**: Data communications port used with Personal Computers.
8. **Line Power Inlet**: AC 90-265 V_{RMS}. 45-440 Hz; automatic input voltage selection.
9. **Protective Ground Terminal**: The protective ground wire is connected at this location inside the instrument. Never tamper with this screw!

10. **Fan**: The fan speed is controlled via a temperature sensor. Normal bench-top use means low speed, whereas rack-mounting and/or installed options may result in higher speed.
11. **Type Plate**: Indicates model number and serial number.

3.3 The GSG Main Menu

The main menu of the GSG user interface is shown on the GSG display when the unit is started. To return to the main menu from any of the sub menus, press the menu key.

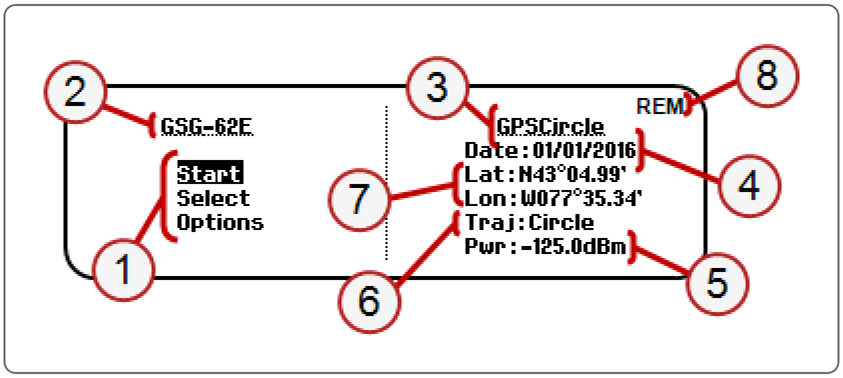


Figure 3-3: GSG's main menu

The main menu displays the following information:

1. Main menu options: **Start**, **Select**, **Options**
2. GSG model number (for more information on models and configurations, see "GSG Series Model Variants and Options" on page 203).

On the **right side** of the menu, the currently selected scenario is shown with some of its key data:

3. Name of the current scenario
4. Scenario start date
5. Transmit RF power (see also: "Setting Transmit Power" on page 113)
6. Trajectory shape
7. Scenario Current Position (latitude/longitude)
8. In the upper right-hand corner, **abbreviations** may be shown:
   - **REM**: remote commanding
   - **EXTREF**: external reference clock is selected in the **Options** menu

- **ARM**: the unit is waiting for a trigger to start the scenario
- **HOLD**: the movement along the trajectory is paused

3.4 "Start" Menu

To start the currently loaded scenario (as previously selected using the ""Select" Menu" on page 38), highlight the main menu option **Start** by pressing the arrow keys. Then press enter.

In its default mode, the GSG simulator will launch the scenario (the delay depends on the size/complexity of the scenario data), and then automatically run the scenario.

To stop the scenario, press exit or cancel, and confirm.

There are, however, interesting alternatives to starting a scenario, mainly to facilitate test automation. The illustration below summarizes the start variations discussed underneath.

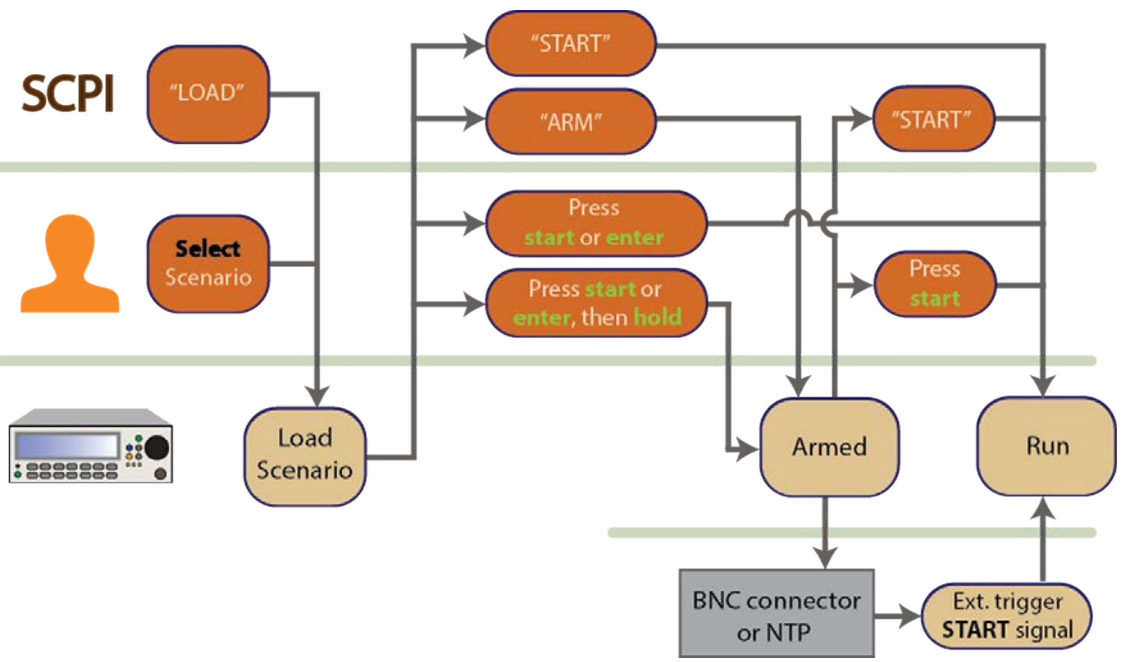


Figure 3-4: Scenario start variations – Flowchart

3.4.1 Scenario Start Variations

Hold before manual start

Once you pressed start, or enter (with the **Start** main menu option highlighted), the GSG unit requires some time to launch the scenario (the delay depends on the size/complexity of the scenario data).

During this wait time, press the [.] (hold) key if you want to prevent the scenario from beginning to run before you are ready. This is called "**arming**" (the **ARM** text icon will display in the upper right corner of the display, and the armed **status indicator** will light up).

Once you are ready, press the start key to run the armed scenario.

SCPI START command

Once you submitted the SCPI command `SOURce:SCENario:LOAD`, submit another command to arm the GSG simulator:

`SOURce:SCENario:CONTrol ARM`.

Then, to start scenario execution, submit the SCPI start command:

`SOURCce:SCNario:CONTrol START`.

Start via external trigger

After arming a loaded scenario (see above), the scenario execution can be started via an external trigger signal, submitted to the GSG unit by means of the BNC input (see "External Trigger Input" under "Rear Panel Outputs and Inputs" on page 8).

3.4.2 Scenario Execution Views

While a scenario is running (also referred to as "scenario execution"), you can display several views, so as to ...

- monitor the current scenario status
- verify the operation of your **receiver-under-test** by comparing its output with the data provided in the scenario execution views
- adjust some of the scenario settings.

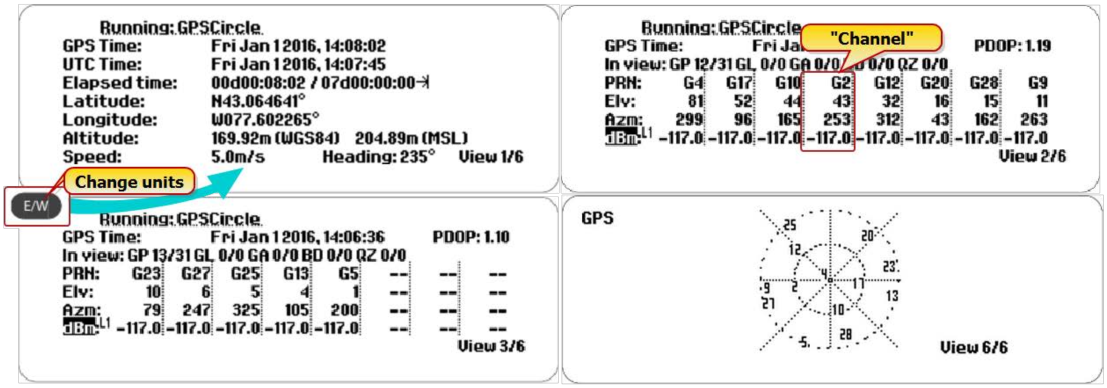


Figure 3-5: Views displayed during scenario execution

To display the views in successive order, press the view key. In the lower right corner e.g., **View 2/6** may be displayed, indicating the current view/total number of views. The total number, and content of views depends on the number of signals used in the scenario.

> **Note:** When you press the exit key to leave a menu, its settings will be taken into use immediately, and all band- or satellite-specific offsets are discarded.

See "Running a Scenario" on page 108 to find out how you can **interact with the system** during scenario execution, and to learn **which scenario settings can be adjusted**.

3.4.2.1 View 1/x

View 1/x displays the **scenario name**, and information about the **simulation GPS date and time**, current **position**, **speed** and **direction**, and **elapsed time**.

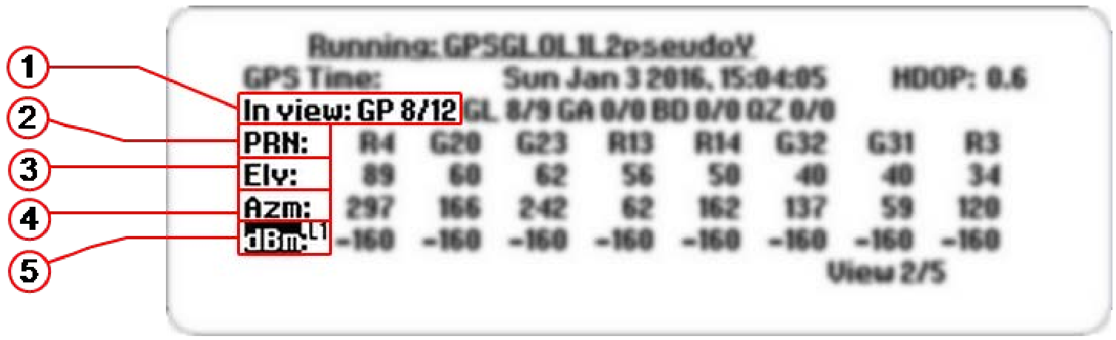
3.4.2.2 View >1/x

Views >1/x display information pertaining to the individual simulated satellites. Up to 8 channels are shown per view (the maximum number of channels is 64, depending on your GSG model, and configuration).

The **first line** repeats the ...

- ... **GPS date and time** (as in **View 1/x**), and displays the ...
- ... **HDoP** (Horizontal Dilution of Precision): A dimensionless number indicating the relative quality of the calculated horizontal position, which is largely a function of the current satellite constellation. [A smaller number is better; the number will never be 0 or 2.]

The remaining lines are:



1. **In view**: Shows the abbreviation of each satellite system, followed by its number of satellites in view/GSG channels reserved. Satellite system abbreviations are:
   - **GP**: GPS
   - **GL**: Glonass
   - **GA**: Galileo
   - **BD**: BeiDou
   - **IR**: IRNSS
   - **QZ**: QZSS
2. **PRN**: Pseudo-Range Number (satellite identifier). The identifiers are:
   - For GPS: **G**xx
   - For Galileo: **E**xx
   - For GLONASS: **R**xx
   - For BeiDou: **C**xx
   - For QZSS: **J**xx
   - For IRNSS: **I**xx
   - For SBAS: **S**xxx.

   Letters are **lower case** if a satellite is unhealthy, or if the ephemeris data is too old to be used.

   For **multipath** replicas, the letter '**D**' will be displayed next to the satellite number.

   Fading satellite signals are indicated by the letter '**F**' (see end of Chapter "Propagation Environment Models" on page 69 for more information).

   Interference signals are recognized by their elevation and azimuth fields since these will be marked as *.

   Furthermore, when the **interference signal is un-modulated** this is identified by a **CG** for GPS interference signals and a leading **C** letter followed by the frequency slot number for GLONASS interference signals.

   Hence, next to the identifiers listed above, the following identifiers may also be displayed:
   - **iUG**, for unmodulated GPS interference signal
   - **iUE**, for unmodulated Galileo interference signal
   - **iUC**, for unmodulated BeiDou interference signal
   - **iUJ**, for unmodulated QZSS interference signal
   - **iUx**, for unmodulated GLONASS interference signal, where 'x' is the frequency slot ranging from -7 to 6

- **iSg**, for sweeping GPS interference
- **iSr**, for sweeping Glonass interference
- **iSe**, for sweeping Galileo interference
- **iSc**, for sweeping BeiDou interference
- **iSj**, for sweeping QZSS interference
- **iNg**, for noise GPS interference
- **iNr**, for noise Glonass interference
- **iNe**, for noise Galileo interference
- **iNc**, for noise BeiDou interference
- **iNj**, for noise QZSS interference

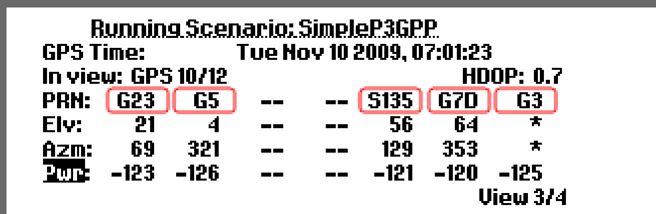
4. ELV: Satellite elevation
   - The angle between the current position's horizontal plane and the satellite position. A low angle is close to 0°, a high angle close to 90° [range = 0 to 90°]
5. AZM: Azimuth
   - The angle around the vertical axis of the current position [north = 0°, east = 90°, south = 180°, west = 270°]
6. dBM: decibel Milliwatt
   - **Transmit Power ratio** in decibels for the frequency band indicated (L1, L2, L5 and ALL). During scenario execution, the Transmit Power (= signal level) can be adjusted for all satellites per frequency band (including ALL bands), or per individual satellite:
     - Press ±/format to toggle through the frequency bands;
       to adjust the power for all satellites on the current band, press ±/power.
     - Press LEFT/RIGHT arrow keys to select a satellite. An **information box** is displayed, showing the satellite ID, elevation, azimuth and frequency bands in use.
       To adjust the **Transmit Power** for this satellite, press the UP/DOWN arrow keys. Press enter to confirm.

   This power adjust functionality is useful for **fine tuning the scenario power level** (see also "Setting Transmit Power" on page 113).

   Adjustments to **dbALL** are saved to the transmit power so that when a scenario is run next time the power is as desired.

Changing the **Transmit Power** setting becomes effective immediately, and also impacts noise generation levels (if in use – available with GSG-5, GSG-55, GSG-56 and GSG-62, 63, and 64).

Example

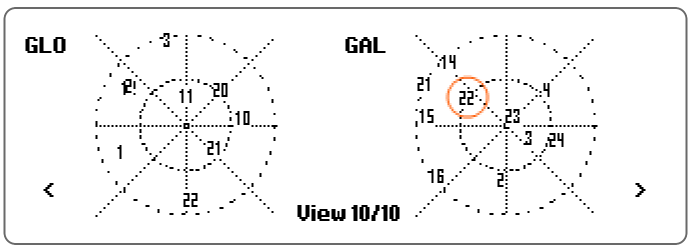


The example above illustrates two GPS signals (**G23** and **G5**), one SBAS signal (**S135**), one multipath signal (**G7D**) and one interference signal (**G3**).

3.4.2.3 Last View

The last view (e.g. **View 4/4**) shows a **skyplot**, illustrating how the simulated satellites are located in the sky.

Press the LEFT/RIGHT arrow keys to scroll through the skyplots, if more than 2 constellations are simulated.



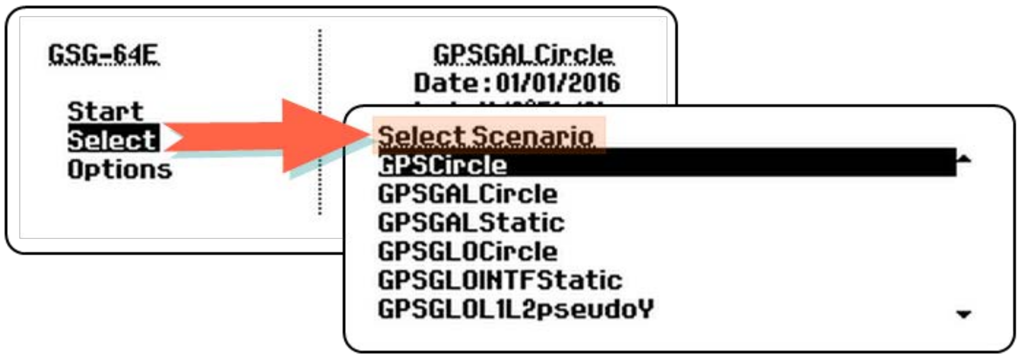
The center of the plot represents the current receiver position, and the outermost circle the horizon, i.e. the **elevation** of a satellite located near this circle is low. The lines represent the **azimuth** (North = 0°). For example, in the **GAL**ileo plot shown above, satellite number 22 would have an elevation of approximately 45°, and an azimuth near 300°.

3.5 "Select" Menu

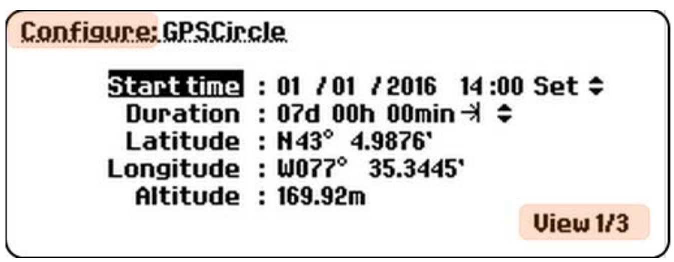
Scenarios are the simulation scripts which you run on the GSG simulator in order to test a GNSS receiver. GSG has pre-installed scenarios which can be executed 'as is', or which you can re-configure to adapt them to your needs. You can also create your own scenarios using the optional GSG StudioView Software.

Prior to running a scenario, you have to select it from the list of scenarios installed on the GSG unit:

1. In the **Main Menu**, highlight **Select** using the arrow keys, then press enter to display the list of scenarios currently loaded:



2. Scroll through the list by using the UP/DOWN arrow keys. Select the highlighted scenario by pressing enter or view: The first **Configuration View** will be displayed:



3. If you want to modify the configuration of the scenario, see "Configuring a Scenario" on page 109.
4. To execute (= run) the selected scenario, press the start key: The scenario will be launched (which will take a moment, depending on the complexity of the scenario chosen), and then started automatically, unless you pressed the [.]/hold key.

Below is a list of all configurable scenario parameters which can be accessed via the **Select Scenario** menu, and which are discussed in the following topics.

> **Note:** Options that are grayed out on your GSG unit are not installed.

» "Start Time" on the next page
» "Duration" on page 42
» "Latitude, Longitude, Altitude" on page 42

3.5.1 Start Time

Start time is the time a scenario uses for simulation purposes, i.e. the simulated time at which the scenario begins every time it is run. The **Start time** can be ...

a. a set time, as configured for the scenario. Whenever you start this particular scenario, the previously set **Start time** will be used, e.g. November 4, 2015 at 19:30.

b. real time, as derived from the NTP server specified in the Network Configuration, and triggered by the user pressing start, or a SCPI start signal being submitted.

> **Note:** If NTP real time is used, the scenario start will be delayed by up to 2 minutes, in order to allow for the simulation data to be loaded.

The **Start time** is aligned to the next full GPS minute. The NTP (UTC) timescale is converted to the GPS timescale by a UTC-GPS offset defined in the NTP Server settings.

GPS time and leap seconds

The **Start time** is based on GPS time, i.e. the displayed time is always GPS time. Unlike UTC time – which is frequently displayed by GNSS receivers – GPS time does *not* include leap seconds.

NTP real time and downloaded Ephemeris

Using NTP as start time in conjunction with **Ephemeris** set to **Download** is subject to licensing options, as it requires the **Simulate Now** option to be present. In this configuration, the GSG unit will simulate the sky as it is in that start position at current time. This functionality is currently only available for the GPS constellation. Please also note that the availability of good ephemeris data cannot be guaranteed, and periods where no data is found and hence no signals can be generated, may occur.

About GPS time and GPS week number

In the GPS data format, there are 10 bits reserved to represent the GPS week number, which leads to a modulo 1024 ambiguity in the week number and hence the GPS date:

The GPS week number count began at midnight of January 5/6, 1980. Since then, the count has been incremented by "1" every week, and broadcast as part of the GPS message. Consequently, at the completion of week 1023, the GPS week number will roll-over to week number 0.

This means that if looking only at the week number (WN) parameter in the GPS data message, it is impossible to determine if WN 1023 corresponds to August 1999, or April 2019, etc. GPS receivers must therefore account for this roll-over problem, and use other means to decide on which 1024 week period they currently are in.

The designers of GPS receivers have a number of ways of ensuring that the WN is interpreted correctly. These techniques range from keeping GPS week numbers in non-volatile memory, keeping a real-time clock, etc.

One popular method involves resolving the year period ambiguities with software revision dates. For example: Since the GPS software knows that it was made on February 11, 2011 (corresponding to GPS week number 1622, and in the data message WN 598), this information can be used to map the WN to a year by concluding that e.g., WN 597 cannot correspond to early February 2011, but rather to mid-September 2030.

This in turn, means that when simulating scenarios using a simulator, going back and forth in time and in GPS week numbers, you may see unexpected behavior in how the WN is interpreted. This could result in a scenario that worked 'correctly' in the past, starts outputting a different date that is 19.7 years forward in time.

The GLONASS system does not have the week roll-over problem that GPS has. When simulating scenarios with historical dates, however, it is likely that a receiver that is trying to compensate for the week roll-over based on the firmware build date mentioned above, will get into a conflict with the GLONASS time stamps and in this case the receiver will not output any solution. This issue, especially with combined GPS+GLONASS scenarios, can be avoided by simulating future dates.

3.5.2 Duration

The duration of the scenario replay can be set to a number of days, hours and minutes.

Any scenario can be run in three different modes:

- Looping: The scenario will be replayed infinite times, re-starting every time after its set duration has expired.
  For this mode, the trajectory should be **loop-shaped**, i.e. have the same start/end point. Otherwise, an error will likely be thrown once the receiver-under-test upon the first replay is moved from the end point to the start point in an unrealistically short time.
- Forever: The scenario will run infinitely (the duration time will be grayed out).
  If your trajectory is **loop-shaped**, i.e. it has the same start/end point, the trajectory will be followed over and over again (just like in the above-mentioned Looping mode), but the simulation time will continue to elapse (contrary to the Looping mode, which will **re-start** the simulation time with every new scenario execution.
  If your trajectory is *not* loop-shaped, in this mode the receiver will travel along the last trajectory vector infinitely.

  > **Note:** The option Endless only works, if the ephemeris option is set to Download. (See also:"Ephemeris" on page 47)

- One-Go: The scenario will be executed once, for the set duration.
  Upon completion of the scenario execution, GSG will return to the Main menu.

3.5.3 Latitude, Longitude, Altitude

The position is specified using WGS84 (for more information on the *World Geodetic System*, see Wikipedia).

Note that the use of the WGS standard also applies to the **altitude** (ellipsoid height), and that this altitude is NOT the same as the MSL often output by receivers.

Select a **different coordinate input format** by pressing the +/– (format) key repeatedly. The choices are:

- decimal degrees
- degrees-minutes

» degrees-minutes-seconds
» ECEF (Earth-Centered, Earth-Fixed) format.

3.5.4 Trajectories

Note: This feature is not available in GSG-51/52/53.

In the context of GNSS testing, a trajectory is the predefined path a receiver is traveling during the execution of a scenario. GSG-5/6 can be used to simulate virtually any user trajectory. You can:

» Use predefined (built-in) trajectories
» Modify predefined trajectories (using the GSG unit, or the **GSG StudioView** software)
» Create trajectory files in **StudioView**, and upload them.

Note: If the RSG Option (OPT-RSG) is installed on your unit, you can also control movement in real-time.

At the start of the scenario the nose of the user is pointing north. The orientation of the vehicle body changes with movement so that its nose is aligned with the vehicle's course. In cases with changing altitude the nose will still point in a horizontal direction, not changing the body attitude. This default behavior can be changed by using SCPI commands which change pitch, roll, and yaw of the simulated vehicle.

3.5.4.1 Predefined Trajectories

GSG units come with several built-in trajectories. The exact list of these predefined trajectories varies from GSG model to model. The following is a selection:

» **Static**: The user is not moving, but the latitude, longitude and altitude defined in the Scenario configuration are used as user position throughout the scenario replay.
» **3GPP**: The user is moving on a rectangluar trajectory as defined in the Technical Specification 3GPP **TS 25 171** V7.1.0, Section 5.5, Table 11 and Figure 1:

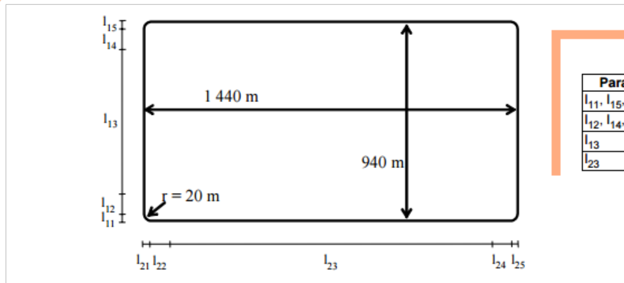


Figure 1: Rectangular trajectory of the moving scenario and periodic update test case

Table 11: Trajectory Parameters

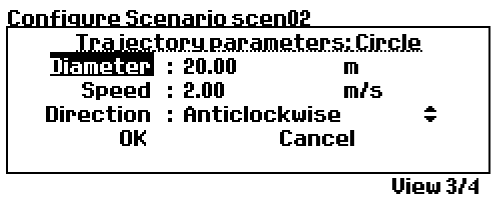
Parameter	Distance (m)	Speed (km/h)
l_{11}, l_{15}, l_{21}, l_{25}	20	25
l_{12}, l_{14}, l_{22}, l_{24}	250	25 to 100 and 100 to 25
l_{13}	400	100
l_{23}	900	100

The specification describes the trajectory as follows:

“The UE [User Equipment, (Orolia)] moves on a rectangular trajectory of 940 m by 1440 m with rounded corner defined in figure 1. The initial reference is first defined followed by acceleration to final speed of 100 km/h in 250 m. The UE then maintains the speed for 400 m. This is followed by deceleration to final speed of 25 km/h in 250 m. The UE then turn 90 degrees with turning radius of 20 m at 25 km/h. This is followed by acceleration to final speed of 100 km/h in 250 m. The sequence is repeated to complete the rectangle.”

The complete specification can be found under http://www.3gpp.org/DynaReport/25-series.htm.

- **Circle**: The user is moving in a circle throughout the scenario replay. When Circle is selected, a dialog is shown asking the parameters describing this trajectory. These parameters include diameter [meter], speed [m/s] and direction [clockwise/anticlockwise].



The **start position** of the trajectories is the position specified in the **Configuration View 1/3**, under Latitude, Longitude and Altitude.

3.5.4.2 User-Created Trajectories

GSG supports the simulation of custom-made trajectories. The trajectories are typically created with the **GSG StudioView** software (see also: "Creating a Trajectory in StudioView"

on page 130).

Two trajectory file types are supported:

RSG Trajectories

Even if the RSG Option (OPT-RSG) is not installed on your GSG, and you can therefore not run scenarios in real time, you can still use the Orolia-proprietary RSG format by up-loading RSG trajectories onto your GSG unit. The RSG format is further described under "RSG Command Reference" on page 332.

NMEA Trajectories

It is also possible to use custom trajectories in the NMEA format (as generated with the help of Orolia's **GSG StudioView** software, or created otherwise) by uploading the NMEA trajectory to your GSG from a Windows PC.

> **Note:** As of firmware version 3.0, Orolia GSG series simulators support 10 Hz NMEA data.

GSG will transform the first **timestamp** in the NMEA trajectory so as to adjust it to fit the scenario start-time and start position. Hence, the start time for the scenario does not need to match NMEA time-stamp. All other timestamps in the NMEA trajectory will be transformed accordingly, thus keeping the relative position/times in the NMEA trajectory intact.

A given NMEA trajectory can be **replayed** in any GPS time frame, utilizing any earth coordinates by specifying the desired start time and start position in the scenario.

Looping trajectories

The NMEA trajectory files can be configured either to be executed once, or to loop repeatedly throughout the scenario execution. For the looping to be allowed, the NMEA trajectory has to be continuous, meaning the first and last specified coordinates of the trajectory must be identical (see also: "Duration" on page 42).

RMC vs. GGA

GSG-5/6 Series GNSS Simulator's accept NMEA streams containing any set of valid NMEA sentences, yet only information from RMC and/or GGA sentences will be used to build the trajectory (for detailed information on GGA and RMC, purchase the NMEA 0183 through nmea.org, or see e.g., here). Either RMC messages, or GGA messages are accepted, or a combination of both. The latter is preferable. All other types of sentences are ignored without user notification.

All NMEA sentences – i.e. all characters between the start marker ('$') and the checksum ('*' plus two hex digits) – are validated for correct syntax. The only accepted error is an incorrect checksum (because incorrect checksums can be useful for manually correcting the contents of an NMEA file).

Date, time, position, speed, heading

Date and time along with longitude, latitude, speed over ground, and heading will be extracted from the RMC message (NMEA's Recommended Minimum), in order to build the trajectory.

Altitude

RMC messages do not include altitude data, hence if no GGA messages are available, the start position altitude specified in scenario parameters will be used instead.

Heading and speed over ground specified in the NMEA file will be applied only to the last epoch of the trajectory, since all other points they will be computed by two adjacent positions. This technique prevents the undesirable behavior of some receivers which generate NMEA data using heading and speed data that does no correspond to position change.

Heading and speed changes

GNSS receivers are generally very sensitive to g-forces, and unrealistic movements will result in the receiver losing track of the simulated signals. Therefore, trajectories should at all times describe smooth, realistic movements, i.e. using gradual transitions in acceleration and heading, rather than abrupt commands (such as random user-set coordinates or speed changes). Any parts of a trajectory describing changes in heading and/or speed must be provided in 10 Hz increments.

Skipped epochs

Note: One GSG epoch equals a 100 ms block of time.

The navigation receiver warning field will always be verified. An epoch will be skipped if...

- ...the field value is 'V', or
- ...there is no date/time data, or
- ...there is no position data.

File size

Note that NMEA trajectory files can become quite large if the sampling rate is high and a large distance is covered. Simulation files uploaded to the GSG unit cannot contain more than 12000 epochs (~19 minutes RMC + GGA at 10 Hz).

If you start a scenario that uses an NMEA file with more than 12000 epochs, GSG will initiate a dialog upon start of the scenario, asking you to either cancel the simulation execution, or to truncate the NMEA trajectory file down to its first 12000 epochs.

Making a One-Line Trajectory

As the GSG unit uses the heading and speed information of the RMC sentences, only one (!) NMEA sentence is actually required to describe a simple, continuous movement.

For example, the following one-line trajectory specifies a continuous north bound trajectory (as the heading field is set to 0.0 degrees) at a speed of 77 knots.

```
$GPRMC,111150,A,6000.0000,N,0100.0000,E,77.000,0.0,010101,0.9,W,A*03
```

One-line trajectories like this can be easily be made by manually creating desired NMEA files. The example above can be taken as a baseline, then edit speed and/or heading fields as required. For the validity of the sentence, the last 2 digits contain a checksum of the data (XOR of all bytes between $ and * symbols) – this checksum must be correct and can be calculated with e.g., this online tool: http://www.hhhh.org/wiml/proj/nmeaxor.html. Note that the NMEA messages, including the checksums, are case sensitive and should be given in UPPERCASE even if the GSG unit (firmware version 3.00 and above) accepts messages in lower case.

3.5.5 Ephemeris

The satellite constellations and the transmitted navigation data of each satellite are dynamically built, once you start the scenario or the signal generation. The constellation and the navigation data is based on RINEX data stored in the unit, or uploaded to the unit. The constellation orbits can be refined by providing precise orbit information in SP3 format (for details, see below).

GPS and QZSS almanac data may optionally be provided in the form of YUMA files (for details, see below).

In addition, SBAS message files are also supported (see "SBAS Satellites" on page 82 and "User-Uploaded Ephemeris" on page 49 below for more details).

Under the menu item **Select** > **Select Scenario** > **Configure** > **Ephemeris**, there are two or three options to choose from (as described below), in order to select a source for your scenario navigation data:

- **Default**
- **Download**
- **User-uploaded files**.

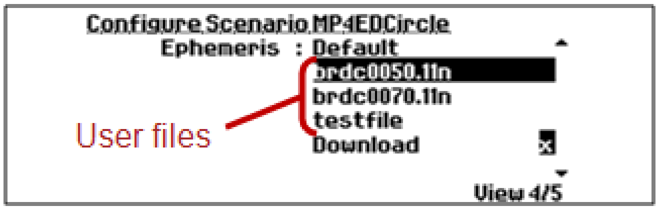


Figure 3-6: Ephemeris selection

3.5.5.1 Default Ephemeris

The default RINEX data for GPS and GLONASS is based on the CDDIS GNSS archive, using the brdc files. The non-redundant **brdc** file merges the individual site navigation files into one, and thus can be used instead of the many individual navigation files.

This data is complemented by GLONASS almanac data downloaded from ftp://www.glonass-iac.ru/MCC/ALMANAC/, covering the same period (file names are prefixed by receiver types, e.g. MCCT_ , MCCJ_, GG-24, or TOPCOM_).

The default navigation data begins Jan 8, 2012 and runs for 33 consecutive days.

For Galileo, BeiDou, and IRNSS, the GSG unit comes shipped with its own ephemeris data set.

When the ephemeris setting is set to Default, the GSG unit builds all scenarios, any start date, using the default data. If there is an exact match for the scenario Start time and pre-loaded navigation files, that navigation data will be used. If an exact date match is not found, then the GSG unit will use the first preloaded navigation data with the same day of the week as the scenario's start time. Further simulation days will use consecutive in date navigation data.

In general, the start time of the scenario always supersedes the time stamps in the navigation data files. If file date and scenario start time do not match, then the loaded data is transformed accordingly to match the scenario's start time. If the scenario defines a GPS almanac files only, the YUMA files will define the almanac and the ephemeris will be derived from the default RINEX data.

3.5.5.2 Download Ephemeris

The user can let the unit automatically download navigation data from official websites. The navigation data, `brdc` files and GLONASS almanac files are retrieved from the same sites as mentioned under "Default Ephemeris" above.

For this feature to work, the following requirements must be met:

1. The GSG unit must have access to the Internet.
2. The correct DNS address must specified, either by setting **Options** > **Interfaces and Reference** > **Network > Obtain IP autom. = Yes**, or—when using a static IP configuration—by manually entering the correct DNS address.
3. The scenario start time must be in the past.

The downloaded navigation data will be locally stored on unit. On subsequent simulations the GSG unit will first look for previously downloaded files before attempting to retrieve them again. Hence once scenarios have run once they can also be replayed at later occasions even if the Internet connection is no longer available.

Note, however, that the unit performs automatic clean-up of downloaded files and that this clean-up will occur when free disk space is less than 20% of the total disk space.

Download cannot be used in conjunction with Galileo, INRSS and/or BeiDou simulation. The download functionality does not support the downloading of GPS almanac files.

Simulate Now

When Download ephemeris is used, it is also possible to simulate the *current time*, provided:

a. the **Simulate Now** license option is installed, and
b. the **Start Time** is set to NTP.

In this case, the navigation data will be based on hourly data retrieved from the official GPS ephemeris site ftp://cddis.gsfc.nasa.gov/pub/gps/data/daily/.

Please note that this functionality is only available for GPS, and that the availability of the data cannot be guaranteed.

3.5.5.3 User-Uploaded Ephemeris

User-specified **RINEX** and **SP3** files can be uploaded to the unit. Multiple files may be selected. The uploaded RINEX files will be used to build both constellation, and navigation data for the satellites. If SP3 data is provided, it will override RINEX data for the definition of satellite orbits in the constellation. If no SP3 data is available, the constellation orbits will be built, using provided or built-in RINEX data.

The number of RINEX files necessary depends on the scenario's start time and duration, and must be equal to the total number of simulation days (including start/end days utilizing less than 24 hours).

In the event that dates for the user-specified data do not match the scenario's start time, then GSG will transform the start time in order to resolve the conflict.

If a satellite system (e.g., GPS, or Galileo) is selected (i.e., number of satellites selected is not 0) and no navigation files are selected for that particular satellite system, then GSG will use default data for that satellite system.

The RINEX format support includes version 2.x and 3.0.

The file extension for SP3 files must be *.sp3 (not case sensitive).

Downloading GPS RINEX files manually:

1. Decide on the start date and time of the scenario, and the duration.
2. Determine the number of files needed to cover the duration. (Each file contains up to 24 hours of information, i.e. midnight to midnight.
3. Go to the website ftp://cddis.gsfc.nasa.gov/pub/gps/data/daily/ and select the required year, and then the day of year.
4. In the directory for that day of year, choose the **XXn** folder, where XX is the 2-digit year.
5. In the **XXn** folder, select and download the file brdcYYY0.XXn.Z, where XX is the 2-digit year and YYY is the 3-digit DOY value.
6. Inside the zipped folder you download is the file to use in the unit.
7. Repeat this procedure for each day you plan on simulating in your scenario.

YUMA

Optionally, GPS and QZSS almanac data may also be provided in the form of YUMA files, which are identified by their **.alm** file extension. GPS and QZSS almanac files are identified by a first-letter file naming convention:

- **g*.alm**: If the first letter of the file name is a ‘g’, GSG assumes the file contains **Galileo** satellite almanac data.
- **b*.alm**: If the first letter of the file name is a ‘b’, GSG assumes the file contains **Beidou** satellite almanac data.
- **q*.alm**: If the first letter is ‘q’, then GSG assumes the file contains **QZSS** satellite almanac data.
- **qg*.alm**: If the first 2 letters are ‘qg’, then GSG assumes the file contains both **GPS, and QZSS** satellite almanac data.
- ***.alm**: If the first letter is anything other than ‘g, b, or q’, GSG assumes the file contains only **GPS** almanac data.

YUMA almanac data can be used with custom RINEX files, or default ephemeris data. If no custom RINEX files are provided, the default data will be used.

This allows testing using GPS and QZSS satellites with the same, or different GPS almanac data. The GSG supports multiple GPS and QZSS almanac files. The YUMA almanac is considered valid for ±3.5 days from the TOA value (Time-of-almanac) listed in the YUMA almanac.

The scenario is restricted to start times within this range. If a scenario runs beyond this range of time, no new satellites will be added. If the user specifies a start time outside this range, a dialog will advise the user that the ephemeris and almanac are dates are mismatched. The SCPI error "Data out of range" will be logged to indicate this issue for remote control users.

CNAV

You can also provide a file with CNAV messages to be used with GPS and QZSS L2C and L5. The file extension is **.cnt** (CNAV train), and the file is satellite-specific. The file name conventions are:

- PRN<satid>_y<4digityear>_d<dayofyear>_h<hourofday>.cnt

e.g., PRNG01_y2013_d105_h14.cnt.

Each row of the file should contain:

- `satSys(A1), satid (I2), 1X, year (I2), 1X, month (I2), 1X, date (I2), 1X, hour (I2), 1X, min(I2), 1X, sec (I2), 1X, msgid (I2), 1X, [optional] hexmsg (A76)`

Example:

G01 13 04 15 14 00 00 11 8B04B4ED919863A6671F473A31412695EFF3C
026C0209FF07D601F775FEFE1FF987800000000

The `hexmsg` part is optional, and if not provided, it will be generated by GSG. This enables for users to specify only the order of messages.

The messages are used in a circular manner, i.e. after the last message is sent, the first message will be sent again. The starting message is selected based on scenario start time, i.e., it can be one of the middle messages in case scenario starts later than the time of the first message.

Since the same file is used for L2C and L5 message trains which have different message duration, only the timestamp of the first message is relevant to decide the starting message. The week number and tow, as well as CRC, are recalculated by GSG.

SBAS

SBAS message files must follow the following file naming conventions so that GSG can recognize them:

- For EGNOS: `PRN*.ems`
- For WAAS: `Geo*`

SBAS message files do not need to be transformed to the scenario date as all timing is relative, i.e. a message file downloaded for a particular date can be used also with any other scenario start date.

ANTEX

You may also specify an **ANTEX file** to be used in simulation. The file extension is `.atx`. It contains satellite antenna phase center offsets and phase center variations. When present, this information is used for improving satellite range calculation.

> **Note:** For GLONASS, matching ephemeris and almanac files must be specified (only the 2-line AGL format is supported, see ftp://ftp.glonass-iac.ru/MCC/FORMAT/Format.agl). In addition, GLONASS almanac files must be named `*YYMMDD.agl` (i.e., a date must be provided at the end of the file name).

> **Note:** The GLONASS data at this publicly available FTP site is known to contain errors. These can cause the GSG to generate signals that are deemed 'bad' by a receiver and may not be used in a fix or for navigation. This data is not maintained by Orolia and is not guaranteed.

> **Note:** The GPS and QZSS almanac files specified must comply with the YUMA file format and match the first 5 characters exactly for field identification. The spacing to the rightmost column of data must be preserved. If the file fails to be processed, verify that the Af0 and the Af1 lines do not contain a space between these prefixes and the (s/s). For example, the line must be `Af0(s/s)`, not `Af0 (s/s)`.

> **Note:** RINEX data files in most cases must be full day files. However, when GPS almanac files are provided, the RINEX records can be of shorter duration. RINEX files of less than a day duration without supporting GPS and QZSS YUMA almanac files are limited to start times only after 1400 hours, and may operate for limited times.

3.5.6 Leap Second

Configuration of leap seconds include two parameters: initial GPS-to-UTC offset (due to accumulated leap second events) and future leap second event.

To set a leap second event, navigate to **Select** > **[Select Scenario]** > **Configure** [selected scenario]: **View 2/3** > **LS:**.

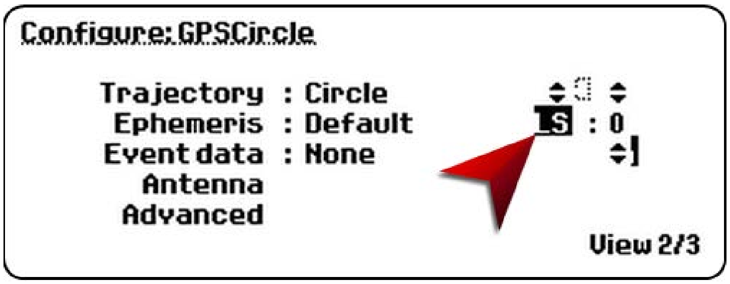


Figure 3-7: Leap second configuration

The leap second event field can be set to -1, 0 or 1, and indicates a future change in GPS-to-UTC offset value.

If the leap second event (LS field) is set to a value other than zero

The following values will be used:

$\Delta t_{LSF} = \Delta t_{LS}$ + value given in the leap second field

WN_{LSF} = The GPS week number (eight bit representation) of the week that includes the 30th of June, or 31st of December, which-ever comes first with respect to the scenario start time.

DN = Day number of the date described above.

If the leap second event (LS field) is set to zero

The GSG will attempt to use leap second event information from navigation data.

When default navigation data is used, OR, the RINEX files attached to the scenario don't contain indication about a future leap second event, then the following values will be used:

$\Delta t_{LSF} = \Delta t_{LS}$

$WN_{LSF} = WN_{LS} - 1$

DN = 1

If the attached RINEX files do contain indication of future leap second event, then WN_{LSF} and DN are set from the RINEX file, and $\Delta t_{LSF} = \Delta t_{LS}$

Initial GPS-to-UTC offset (Δt_{LS}) can be initialized automatically by the scenario start time (using history information about actual leap second events), or by navigation data files.

Alternatively, the user can also specify any desired offset value. This is configured under the scenario menu: **Advanced** > **GPS to UTC offset**, or by using the scenario file parameter **GpsToUtcOffset**.

The allowed values are:

- **Auto**: In this setting, the initial offset is determined by scenario start time
- **Rinex**: determined by information taken from navigation data files attached to a scenario
- **<Fixed Offset in Seconds>**: the user can select an offset from a range of 0-30

An up-to-date leap second events history is embedded in each firmware release, but you can also download the latest list from the Internet using **Options** > **Advanced options** > **Download leap seconds list**.

Considerations

Note that downloaded and default navigation data files do not contain any LSF information (RINEX v2.1). Therefore, it is still necessary to set the LSF when a leap second change will occur, in order to ensure correct behavior.

3.5.7 Event Data

Events can be used to introduce changes into a running scenario. Events can be used to change the power levels of satellites, to control multipath settings, and to control navigation bits, e.g. simulating bit errors in the navigation message. Events are captured in event files.

Each line of an event file describes one event, using one of the following formats:

1. `TIME {scenario | prn SATID | channel NUMBER | system SYSTEM} relpower RELPOWER`
2. `TIME {scenario | prn SATID | channel NUMBER | system SYSTEM} abspower on|off|ABSPOWER`
3. `TIME {prn SATID | channel NUMBER} duplicate RELRANGE RELDOPPLER RELPOWER EFFECTIVETIME [CHTARGET]`
4. `TIME {prn SATID | channel NUMBER} multipath RELRANGE RANGECHANGE RANTEINTERVAL RELDOPPLER DOPPLERCHANGE DOPPLERINTERVAL RELPOWER POWERCHANGE POWERINTERVAL [INSTANCE]`
5. `TIME {prn SATID | channel NUMBER} delete [INSTANCE]`
6. `TIME prn SATID navbits SIGTYPE SFID PAGEID STARTBITPOS ENDBITPOS HEXSTRING REPEAT CRCFLAG PRINTFLAG`

All formats begin with a time tag (TIME), which is the time of application for the event, measured as seconds passed since the scenario Start Time. Events which apply to all satellites use the `scenario` keyword. Events which apply to a specific satellite indicate this by specifying `channel NUMBER` or `prn SATID` values. For relpower and abspower events, it is possible to apply the event to each satellite of a specified constellation: use the `system` keyword followed by the satellite system name (GPS/GLONASS/GALILEO/BEIDOU/QZSS/IRNSS/SBAS).

- The first format, `relpower`, defines a change in the **power level** for the whole scenario, a single satellite identified by SATID or channel number, or all satellites in one constellation.
- The second format, `abspower`, sets the **absolute power** for the whole scenario, a single satellite identified by SATID or channel number, or all satellites in one constellation.
- The third format, `duplicate`, generates a **duplicate signal** from a given satellite, using a specified delay, Doppler and power level. Duplicate channels require 60 seconds to be created, and are introduced at fixed 30-second intervals. Only 4 Duplicate satellites are allowed to be created at a time. Duplicate events closer together than 4 seconds are spread apart automatically to maintain 4 second separation.

  SBAS and Interference satellites cannot be duplicated. The optional `CHTARGET` parameter specifies the channel to be used. If the channel is used by a satellite, this satellite will be disabled, and the multipath satellite replaces it. If the `CHTARGET` parameter is not specified, the multipath satellite will be created in the first unused channel. Multipath, SBAS and interference/jamming channels cannot be duplicated.
- The fourth format, `multipath`, modifies the **multipath parameters** of a satellite. If the satellite is not a duplicate, it becomes a duplicate satellite, which is reflected in its `SATID`. SBAS and interference/jamming channels cannot have their multipath parameters modified.
- The fifth format, `delete`, **deletes** a satellite. If the satellite is not a multipath duplicate, it will typically automatically re-appear after 1 to 2 minutes. SBAS and interference/jamming channels cannot be deleted.
- The sixth format, **navbits**, sets **bits in a navigation message**. The `ENDBITPOS-STARTBITPOS+1 LSB` of the `HEXSTRING` are used to replace the bits between `STARTBITPOS` and `ENDBITPOS`, so that the `ENDBITPOS` is aligned with the LSB of the `HEXSTRING`.

  Should `ENDBITPOS-STARTBITPOS+1` > length (`HEXSTRING`), the `HEXSTRING` will be used as a repeating pattern to replace the bits between `STARTBITPOS` and `ENDBITPOS`.

  Multiple navbits events may be applied to the same message. Note that a navbits event is applied to the first message from the event `TIME` with the `SFID` and `PAGEID` specified in the event. For **GPS** the bit count starts with `MSB`, whereas for **Glonass**, the count starts with `LSB`. Only GPS and GLONASS are currently supported.

The units for the event parameters are:

- TIME in seconds since scenario start time
- SATID is a satellite ID. The format explained in protocol documentation.
- NUMBER is the channel number. Range depends on GSG model.
- RELPOWER relative change in power settings specified in dB
- ABSPOWER absolute value for power settings specified in dBm
- RELRANGE is the relative range delay in meters.
- SYSTEM is the name of the satellite system. Allowed values are: GPS, GLONASS, GLO, GALILEO, GAL, BEIDOU, BDS, QZSS, IRNSS, SBAS
- RELDOPPLER is the relative Doppler offset in meters/sec.
- EFFECTIVETIME numerical number. Reserved for future use.
- CHTARGET is the channel number to where the duplicate is put. Range depends on GSG model.
- RANGECHANGE is the change in range over RANGEINTERVAL. Specified in meters.
- RANGEINTERVAL is the time period in which the RANGECHANGE is updated. Specified in seconds to the tenth of seconds accuracy.
- DOPPLERCHANGE is the change of Doppler in meters/sec.
- DOPPLERINTERVAL is the time period in which the DOPPLERCHANGE is updated. Specified in seconds.
- POWERCHANGE is the change in power over POWERINTERVAL. Specified in dB.
- POWERINTERVAL is the time period in which the POWERCHANGE is updated. Specified in seconds.
- INSTANCE identifies which instance [1..8] of SATID we want to act on. If several (duplicate) satellites exist with the same SATID, INSTANCE can be used to identify a particular duplicate satellite.
- SIGTYPE is one of the signal types supported by the satellite. Allowed values are: L1CA, GPSL1CA, L1P, GPSL1P, L1PY, GPSL1PY, L1CAP, GPSL1CAP, L1CAPY, GPSL1CAPY, L2P, GPSL2P, L2PY, GPSL2PY, L2C, GPSL2C, L5, GPSL5, L1, GLOL1, L2, GLOL2
- SFID is
  - a subframe ID (with GPSL1 and L2P signals)
  - a message type (with L2C and L5 signals)
  - a frame ID (with Glonass)

- `PAGEID` is
  - a page ID (with GPSL1 and L2P signals)
  - 0 (not relevant) when the subframe ID is 1-3
  - 0 (not relevant) with L2C and L5 signals
  - a string idID (with Glonass).
- `STARTBITPOS`, `ENDBITPOS` are positions of bits in a navigation message.
- `HEXSTRING` is a bit pattern to be set in the message.
- `REPEAT`
  - set to 0, if the modification should be applied only once
  - set to 1, if the modification should be repeated on every message.
- `CRCFLAG`
  - set to 0, if `CRC/parity` is not to be corrected after the modification
  - set to 1, if `CRC/parity` needs to be corrected after the bit modification.
- `PRINTFLAG`
  - set to 0, if the modified message does not to be logged (default)
  - set to 1, if the modified message needs to be logged in the execution log. Note that the message is logged only once, even if the modification is repeated on every message (repeat flag is 1).
- `PROPENV`
  - See "Propagation Environment Models" on page 69.

An example event file containing all five formats with explanations is shown below:

```
1.0 channel 7 relpower -3

2.0 prn G32 abspower -110.5

3.0 scenario abspower off

4.0 scenario abspower on

5.0 scenario relpower 2

6.0 system GPS abspower off

7.0 system GAL abspower -120

8.0 system QZSS abspower on
```

```
9.0 system GLO relpower +3

10.0 prn G9 duplicate 30.0 -0.01 -8.3 0

10.0 channel 6 duplicate 30.0 -0.01 -8.3 0

11.0 channel 6 multipath 35.0 0.01 1.0 0.0 0.0 0 -10.0 0.0 0

11.0 prn G9D multipath 25.0 0.01 1.5 0.0 0.0 0 -15.0 0.0 0

12.0 prn G1 navbits L1CA 1 0 77 77 1 0 0

170.0 channel 6 delete

180.0 channel G9D delete
```

- 1.0 seconds into the scenario the power level of the satellite in channel 7 will be attenuated by 3.0 dB.
- At 2.0 seconds, the absolute power for GPS PRN 32 is set to -110.5 dBm.
- At 3.0 seconds, the signal transmissions for all satellites are turned off.
- At 4.0 seconds, the power settings for all signals are restored.
- 5.0 seconds into the scenario, the power level of all satellites is increased by 2.0 dB.
- 6.0 seconds into the scenario, the power level of all satellites is turned off.
- 7.0 seconds into the scenario, the power level of all Galileo satellites is set to -120 dBm.
- 8.0 seconds into the scenario, the power level of all QZSS satellites is turned on (power level will be restored to the level used on this chanel before it was turned off).
- 9.0 seconds into the sceanrio, the power level of all Glonass satellites is increased by 3.0 dB
- At 10.0 seconds, a duplicate of the GPS PRN 9 satellite is created: The range of the duplicate signal is delayed by 30.0 meters, it has a Doppler offset of -0.01 m/s and a power level that is 8.3 dB lower than the original signal.
- At 10.0 seconds, a duplicate of the satellite in channel 6 is created: The range of the duplicate signal is delayed by 30.0 meters, it has a Doppler offset of -0.01 m/s and a power level that is 8.3 dB lower than the original signal.
- At 11.0 seconds, the multipath settings of the newly created duplicate, identified by its channel number 6, is modified: The satellite will have a 35 meter range offset, increasing with 1cm/s. It will have its power attenuated by 10 dB.

- At 11.0 seconds, the multipath settings of the newly created duplicate, identified by its SATID 'G9D', are modified: The satellite will have a 25 meter range offset, increasing with 1.5cm/s. It will have its power attenuated by 15 dB.
- After 12.0 seconds, the MSB is set to 1 in 6-bit health (bits 77-82) in the first GPS L1CA message with subframe ID 1 sent by satellite G1.
- After 170.0 seconds the channel number 6 duplicate is deleted.
- After 180.0 seconds the G9D duplicate is deleted.

Note: Several Events can occur at the same epoch. If so, any PRN/channel event overrules scenario events, see example below.

EXAMPLE:

The output power of channel 1 is set to -142.0 dBm, while all other channels are transmitted with an output power of -147.0 dBm.

```
4.0 scenario abspower -147.0

4.0 channel 1 abspower -142.0
```

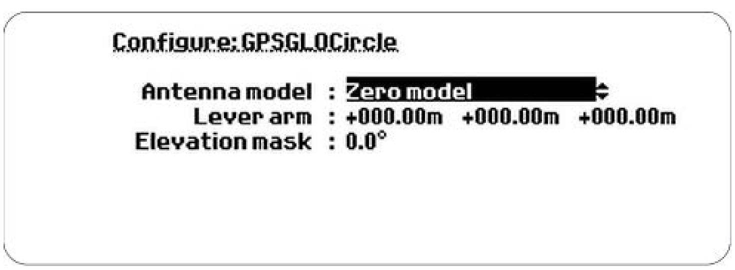
Note also that `abspower` settings of events overrule the **Transmit power** setting specified under **Options** > **Transmit power**, while observing the external attenuation settings.

Duplicating a satellite at **time 00.00** is not permitted.

3.5.8 Antenna Settings

Several antenna-related settings can be configured to allow for optimal scenario simulation: antenna gain pattern, lever arm, and elevation mask.

To configure these settings, navigate to: **Select [Select Scenario]** > **Configure Scenario: View 2/3: Antenna**.



3.5.8.1 Antenna model

The **antenna gain pattern** can be specified for each scenario, using a set of pre-defined **antenna models**, or by utilizing a user-specified file. The built-in antenna models assume an omni-directional gain pattern where the maximum gain is to be found towards the zenith.

The pre-defined antenna models are:

- **Zero model**: Isotropic antenna with a gain of 0 dBic towards all directions. This is the default.
- **Patch**: Gain pattern approximates TOKO DAK Series patch antenna with maximum gain +5 dBic. Size of the patch is 25 x 25 mm and ground plane 70 x 70 mm.
- **Helix**: Gain pattern approximates Sarantel SL1200 (GeoHelix-P2) antenna pattern with maximum gain -2.8 dBic. This is a small helix antenna designed to be embedded in handheld devices e.g. mobile phones. See http://www.sarantel.com for details.
- **Cardioid**: Gain pattern 1+sin (elevation) with maximum gain +3 dBic.
- **GPS-703-GGG**: Gain pattern approximates Novatel's GPS-703-GGG antenna with maximum gain of +5.7 dBic. See www.novatel.com for details.

The format used to describe gain patterns is the FEKO pattern file format version 6.1, Far Field format, File Format 2.0. Gain patterns for various frequencies are to be included in the same file as separate Solution Blocks. The GSG units expect the result type to be either Gain or Directivity, and enforces a maximum value of 50 for the No. of Theta/Phi Samples, with 36 as the recommended choice yielding a 5/10 degree resolution on elevation/azimuth.

The first line of the antenna file is expected to define the File Type. The GSG defines phi 0 degrees, i.e. the x-axis of phi, to point towards the north direction.

3.5.8.2 Lever arm

A **lever arm** can be specified to separate the antenna position from the **body mass center** of the vehicle: All trajectory movements in the simulation will act on the body mass center of the vehicle. By default the antenna is located in this the body mass center position, pointing upward. To specify that the receiver antenna is *not* located in the body mass center position, a lever arm can be configured.

The lever arm settings specify the relative position change in the form of (x, y, z) along the body axis of the vehicle frame, where the coordinate system XYZ is aligned with the body mass center frame. At the start of a scenario, the X-axis corresponds to the east/west axes of the ENU frame and the nose is pointing to the north.

The X-axis has a positive direction towards the right side of the sensor. The Y-axis has a positive direction towards the front of the sensor. The Z-axis has a positive direction towards the top of the sensor.

For more information on vehicle modeling, see "Environment models" on page 68.

3.5.8.3 Elevation mask

The **elevation mask** specifies how low GNSS satellites will be simulated. The elevation mask is set to zero by default.

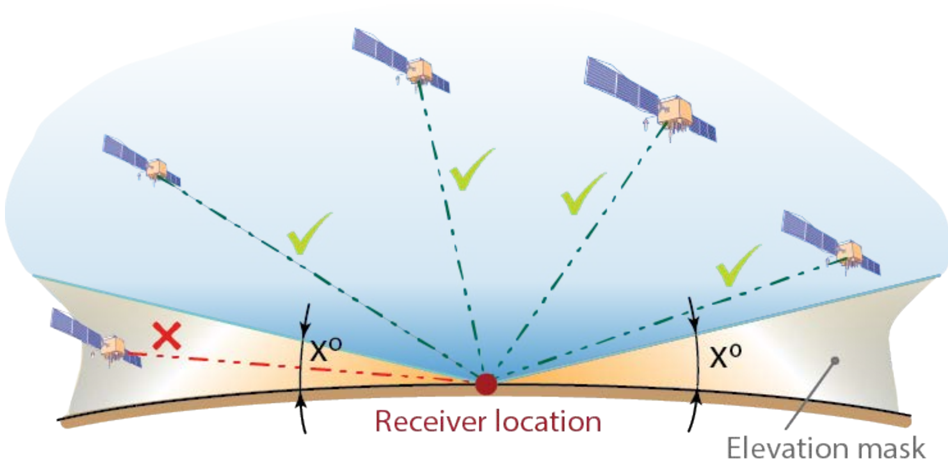


Figure 3-8: Elevation mask

A receiver typically has a higher elevation mask and it will not use any satellite below the elevation angle of its set mask. The recommended setting is to set the elevation mask of GSG to a value equal or less than that of the device under test.

In order to conserve channels by not generating signals the GNSS receiver will not use in its fix, the elevation mask in the GSG can be set to a slightly higher value. This is especially important with, e.g., GSG-52/53 Series units, or GSG-5 models equipped with 4-channels.

3.5.9 Advanced Configuration Options

3.5.9.1 Multipath Signals

A multipath signal is a GNSS signal bouncing off a reflective surface prior to reaching the GNSS receiver antenna. Quite likely, this causes many of the same signals to arrive at the receiver at different times. The receiver then needs to determine which of the signals are received directly.

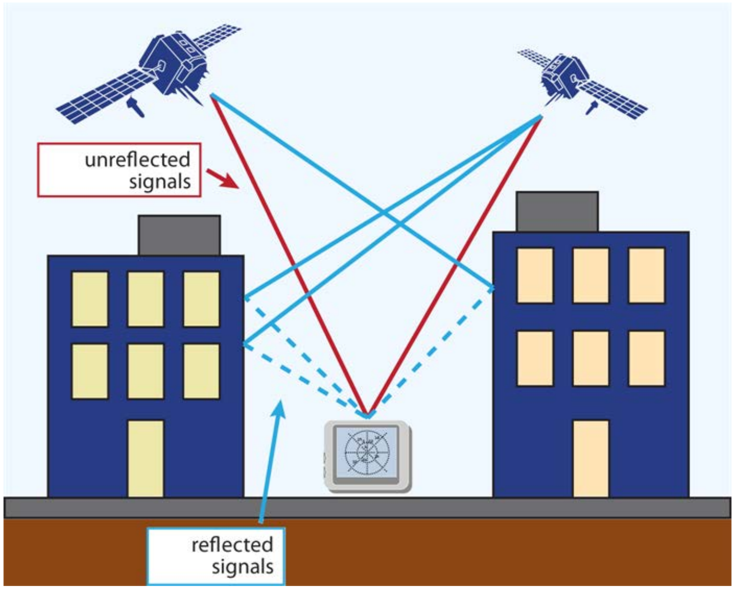


Figure 3-9: Multipath signals in urban environment

To configure a multipath signal, navigate to **Select** > **Select Scenario** > **Configure Scenario, View 2/3: Advanced**, and specify a number greater than zero for **Multipath signals**.

> **Note:** Your GSG unit requires free channel(s) available, in order to allow for the creation and configuration of a (several) new multipath signal(s).

Press enter to display the first configuration view for the first Multipath signal (the number of views equals the number of signals you specified.)

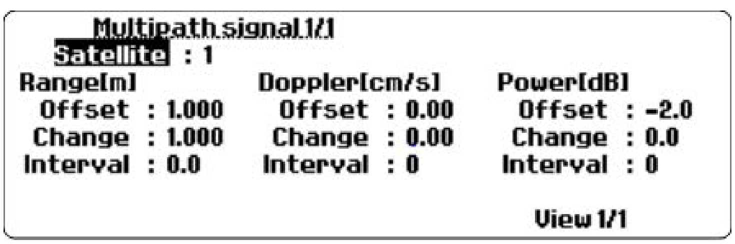


Figure 3-10: Multipath signal configuration view

The following multipath parameters are configurable:

Satellite

» This specifies which satellite is to be duplicated by the multipath signal. The value specified is a running number starting from 1 to the number of satellites defined to be in the scenario. ‘1’ would mean that we will duplicate the satellite in the first position when scenario starts.

Range

- **Offset**: The Range (or: Code) offset in meters. For a multipath signal this value should typically be positive, meaning that the travelled distance of the signal will be longer than that of the original or line-of-sight (LOS) signal.
- **Change**: Change in range offset, given in meters / Interval
- **Interval**: Specify change interval in seconds to the nearest tenth second.

Doppler

- **Offset**: The offset in Doppler in centimeters/seconds
- **Change**: Change in Doppler offset, given in centimeters/seconds/Interval
- **Interval**: Specifying change interval in seconds.

About Range Offset and Doppler

The code (range offset) and Doppler are connected 1-to-1 and cannot be controlled separately in a conflicting manner. For example, a Range Change of 0.019 m/s with Interval ‘1’ has the same effect as specifying Doppler to 1.9 cm/s and leaving all Change/Interval settings at 0.

When both code, and range, and possible change/intervals are specified, the cumulative effect of all things specified will be simulated.

To simulate, e.g., a carrier phase offset that is static relative the LOS signal, please specify the code offset (to, e.g., 0.095 meter) at start and set all Code and Doppler settings to zero.

Random CP

The carrier phase offset can also be randomized on startup by setting the ‘Multipath random CP” to ‘On’ in the GSG menu (or ‘RandomMpCP’ keyword in the configuration file).

Power

- **Offset**: The offset in output Power in dB
- **Change**: Change in Power offset, given in dB / Interval
- **Interval**: Specifying change interval in seconds. If the interval is zero, the offsets will be set at startup and remain static.

Considerations:

- SBAS and interference/jamming channels cannot be duplicated.
- The Change/Interval effect will be interpolated. If the initial interval is zero, the offsets will be set at startup and remain static.
- In a multi-frequency constellation, the multipath configuration will apply to all active bands.
- To match the multipath conditions as specified in the LTE/3GPPS A-GNSS test specification, for GPS the following settings should be used:
  - Range Offset 150 m
  - Doppler Offset 1.9 cm/s
  - Power Offset -6dB
  - Multipath random CP: ON.

Press the view key to configure the next multipath signal, when several multipath signals are configured.

Press the exit key to save your multipath configuration.

3.5.9.2 Interference signals

Note: The Interference feature is only available with GSG-5, GSG-55, GSG-56 and GSG-6 Series products. Some features are only available when OPT-JAM is enabled in the unit (see "List of Available Options" on page 205).

Orolia GSG-Series simulators can generate GNSS interference signals to test GNSS receiver performance. To configure an interference signal, navigate to **Select** > **Select Scenario** > **Configure Scenario, View 2/3: Advanced: Interference Signals**.

After specifying the desired number of interference signals (using the UP/DOWN arrow keys), press enter to display the first interference signal configuration view (the total number of views depends on how many interference signal you specified):

```
Interference Signal 1/5
      Signal type : Unmod GLO L1
   Frequency slot : -5
            Power : -105dBm
 Frequency offset : 0  Hz
         Position : Not set
```

Figure 3-11: Interference configuration view

The following parameters can be configured:

Signal type

You can configure any signal type your GSG unit is licensed for (un-licensed signal types are grayed out).

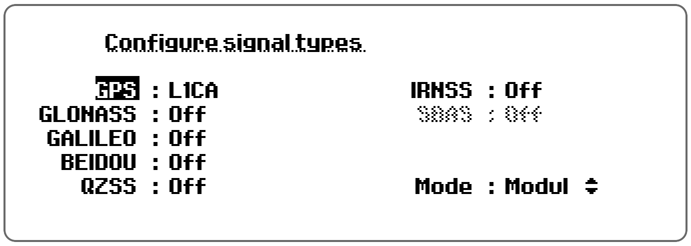


Figure 3-12: Interference signal type configuration view

The interference signal type can be:

- **GPS**: L1CA, L1P, L2P, L1P(Y), L2P(Y), GPS carrier, SBAS
- **GLONASS**: L1, L2 or GLONASS carrier
- **Galileo:** E1, E5a, E5b or a Galileo carrier
- **BeiDou**: B1,B2 or BeiDou B1,B2 carrier signal
- **QZSS**: L1CA or QZSS L1 carrier signal.
- If your GSG unit supports jamming simulations (OPT-JAM), sweep and narrowband noise are available as interference types.

Mode in the lower right-hand corner allows to further manipulate the interference signal by offering the following options:

- **Modulated**: standard signal type (default)
- **PRN**: Pseudo-Random Noise (see e.g., Navipedia: GNSS signal for more information)
- **Unmodulated**: carrier signal (carrier)
- **Sweep** (OPT-JAM only): A dialog is shown asking for startOffset, endOffset, and Sweep-Time.
- **Noise** (OPT-JAM only): A dialog is shown asking for startOffset, endOffset and SweepTime.
  Offsets are used to specify the bandwidth and position of the sweep/noise related to the selected signal frequencies. The range of offsets is ±40 MHz, but can be less when the scenario is executed since signals are not centered in the middle of a frequency band.

Note: Noise interference is not available if wide band noise is set to ON under the Options > Transmit power menu.

Satellite ID/Frequency slot

For GPS, SBAS, Galileo, GLONASS, BeiDou, IRNSS and QZSS signals, the **Satellite ID** must be specified.

For GLONASS carrier signals the **Frequency slot** must be specified.

In some instances, this field is not applicable, and will be grayed out (e.g., GPS carrier).

Frequency offset

The frequency offset refers to nominal frequency of the selected signal/frequency slot.

Power, Position

It is possible to simulate a location-based jamming signal by specifying a position for it. Location-based jamming simulation utilizes the jamming signal **power,** and **position** to calculate the distance from the simulated position, applying the path loss formula given earlier in this document (see "Signal Power Level Considerations" on page 24) to calculate the power of the received jamming signal. As the scenario position moves closer to the location of the jamming transmitter, the jamming power increases, and vice versa.

When configuring a location-based jamming source, the distance to the scenario start position and the jamming coverage are shown, in order to assist you in designing a reasonable jamming test configuration.

Figure 3-13: Configuring the position of a jamming source

Note that the jamming power can be set to +60 dBm, whereas the maximum GSG power level is -65 dBm.

Example

The figure below shows a configuration of a sweeper interference signal for the L1, L2 and L5 bands (OPT-JAM installed).

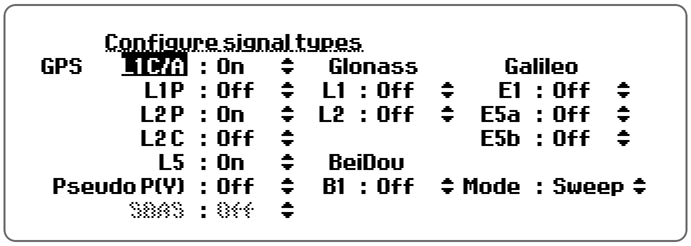


Figure 3-14: Configured sweeper signal

3.5.9.3 Base station

This feature allows you to configure a **Base station**, as it is typically used for high-precision positioning, e.g. in surveying applications: A receiver in a fixed and known position tracks the same satellites the mobile receiver ("rover") does, and in real-time transmits corrective positioning data to the receiver in the rover via a radio transmission stream.

The **Base station** feature can only be enabled with GSG 6-Series units that have the Real-Time Kinematics Option installed (OPT-RTK, see "List of Available Options" on page 205.)

To configure a "virtual" Base station, which supports the output of RTCM differential data to be used as input by a rover receiver, navigate to **Select > Select Scenario > Configure Scenario, View 2/3: Advanced: Base Station**.

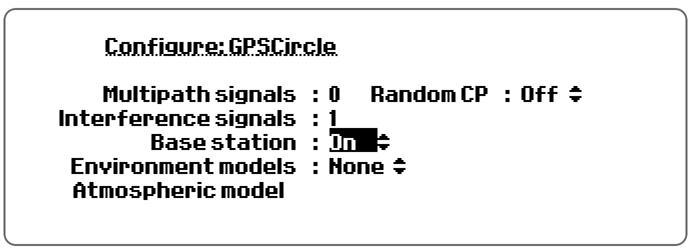


Figure 3-15: Base station configured in Advanced submenu

Once you selected the **On** option for **Base station**, the configuration view will be displayed: Configure the position of the base station and the RTCM messages to be output by it.

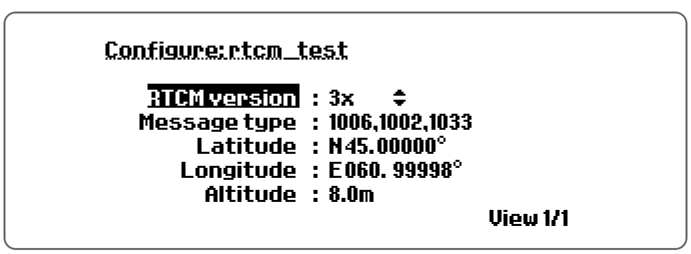


Figure 3-16: Base station configuration dialog

The following **Base station** settings can be reviewed/configured:

RTCM version

The RTCM SC-104 version currently supported is Ver. 3.2. This cannot be changed.

For more information on RTCM standards, see: www.navipedia.net/index.php/RTK_Standards.

Message type

Message types 1002, 1004, 1006, 1010, 1012 and 1033 are supported.

Latitude, Longitude, Altitude

Enter the base station coordinates, using latitude, longitude, and altitude. As with Start position coordinates, the format key can be used to switch between different coordinate formats.

Once a scenario is running, and the base station has been activated, the SCPI command `SOUR:SCEN:RTCM?` can be used to query the GSG for the latest RTCM messages (update rate of 1Hz), as previously configured. The output will be a hexadecimal string.

3.5.9.4 Environment models

Environmental Models allow GSG to simulate **signal obscuration**. (This feature is supported as of software versions 6.1 and higher).

Scenarios utilizing signal obscuration simulate the blocking of GNSS signals by objects placed along the trajectory route. Typical use cases are the simulation of urban "canyons", tunnels, etc.

Environmental models in GSG simulators are supported through compressed keyhole markup language files (kmz), popularized by Google Earth™. A simple way to create these files is by using the 3D drawing tool SketchUp™, available from Trimble Navigation Limited: www.sketchup.com.

Two kinds of models can be configured in a scenario, Vehicle model and Environment model:

Environment model

An environment model is a 3D model of the environment, e.g., buildings, ground, etc. All environment models used must have a geo-location added to them before they can be used for simulation purposes.

Vehicle model

A vehicle model represents a 3D model of the vehicle. The vehicle model will move with the simulated trajectory. The body center of a simulated vehicle will be in the origin position of the model, and all trajectory movements defined in the simulation will act on the body center. The vehicle model should be placed so that its nose points to the north.

The vehicle model will also follow any pitch/roll/yaw movements simulated, i.e. if the vehicle model rolls by 90 degrees, half of the sky is likely to be blocked by the vehicle itself (depending on vehicle model used).

The antenna position oftentimes is not in the same location as the vehicle body center position. In the simulation, this can be adjusted by configuring the lever arm values (see "Lever arm" on page 60).

The antenna position can also be specified in the vehicle model file by adding a component named RecAnt. In the event that both lever arm, and RecAnt are set, the receiver antenna position as set in the Vehicle model takes precedence. The vehicle model does not need a geo-location.

If a satellite is blocked by an object from either environment or vehicle model, i.e. it is not visible by the receiver antenna, its power will be set to OFF.

GSG can successfully handle vehicle models with up to 130 triangles. Models should be optimized for a low polygon count. The triangle count is limited to a total of 300 for the combined environment and vehicle models.

For additional information, see the Orolia Technical Note Vehicle Modeling.

Propagation Environment Models

Built-in signal propagation models can be used to simulate multipath propagation in rural, sub-urban and urban areas. Used propagation models are specified in ITU-R Recommendation M.1225, "Guidelines for evaluation of radio transmission technologies for IMT-2000" (see Section 2.1.4 Parameters of the wideband models). The document is available on the ITU website (http://www.itu.int/rec/R-REC-M.1225/en).

The ITU model corresponds to a tapped-delay line structure with a fixed number of taps: 3 taps in rural and sub-urban environments and 5 taps in an urban environment.

The first tap (i.e. the direct path) may be either Rice or Rayleigh fading, corresponding to LOS and NLOS situations, respectively. The other taps are always Rayleigh fading.

The ITU model describes multipath propagation for a single satellite either in a LOS or NLOS situation. Propagation environment model generates multipath taps for the entire satellite constellation. Based on the satellite elevation angle, the satellites are divided into three zones, as illustrated below:

- Open Sky, Multipath Zone, Obstruction Zone

Figure 3-17: ITU multipath propagation model

Satellites above the **Open Sky** limit are not affected by multipath propagation.

Satellites in the **Multipath Zone** (elevation angle between Obstruction Limit and Open Sky Limit) are considered LOS signals, but affected by multipath propagation. The ITU model for LOS situation is used for these satellites.

For satellites in the **Obstruction Zone** (elevation angle below Obstruction Limit), the direct signal path may be obstructed, e.g., by a building. This is modelled by giving a probability for an NLOS situation. With the given probability, the simulator classifies satellites as NLOS and takes the ITU model for the NLOS situation into use. The NLOS situation changes only when a satellite leaves the Obstruction Zone.

Note that, in addition to the two elevation limits mentioned above, the **Elevation mask** setting applies to the simulation as normally.

The Propagation environment is defined by the **environment type** (open/rural/sub-urban/ urban) and three parameters:

» Open sky limit, Obstruction limit and NLOS probability.

Default values for the parameters in each environment type are given in the table below. The Open environment type is the default, meaning that all satellites assume free-space propagation.

Table 3-1: Propagation environment type parameters

Environment	Open sky limit	Obstruction limit	NLOS probability
Rural	20°	15°	0.1

Environment	Open sky limit	Obstruction limit	NLOS probability
Suburban	40°	30°	0.2
Urban	60°	40°	0.3

The Propagation environment model is taken into use by setting an event `scenario propenv`. If stated without parameters, the default parameter values given above will be used. In this case the format of the even line is:

```
TIME scenario propenv {open|rural|suburban|urban}
```

Note: For more information on Event simulation, see "Event Data" on page 54.

Alternatively, parameter values can be provided in the format:

```
TIME scenario propenv {rural|suburban|urban} OPENSKYLIMIT
OBSTRUCTIONLIMIT NLOSPROBABILITY
```

Example

```
0.0 scenario propenv suburban
300.0 scenario propenv urban
600.0 scenario propenv urban 90.0 60.0 0.75
```

The example event file above will create a simulation starting from sub-urban environment (default parameters). After five minutes the simulation changes to an urban environment (default parameters) and after ten minutes to a highly obstructed urban environment where open sky satellites do not exist (open sky limit at 90 degrees), and satellites below 60 degrees elevation are likely to be NLOS (NLOS probability 0.75).

The Propagation environment model can be defined in the scenario configuration by using the Scenario editor in **StudioView**.

The Propagation environment model can also be set by using the corresponding SCPI commands (see "SOURce:SCENario:PROPenv" on page 265).

When using the Propagation environment model, note that:

» It takes 1 minute to create multipath taps during simulation. Therefore the time interval between switching the environment model should be more than one minute.

» The Event `scenario propenv` must be stated without parameters, or alternatively with all three parameters specified.

- Valid ranges for the parameter values are:
  - `OPENSKYLIMIT`: 0.0 to 90.0 (degrees)
  - `OBSTRUCTIONLIMIT`: 0.0 to OPENSKYLIMIT (degrees)
  - `NLOSPROBABILITY`: 0.0 to 1.0
- It is possible that all multipath taps cannot be created because of limited number of channels available. The Tap number defines the precedence of tap creation (direct path first, and then second tap etc.)
- The maximum number of satellites to be simulated should be set to a fixed value. If any satellite system is set to 'Auto', no new duplicate channel can be created while the scenario is running.
- The number of multipath signals should be set to zero. When using the Propagation environment model, the simulator automatically assigns the multipath channels.
- Fading satellite signals (i.e. all satellites below the Open sky limit) are indicated by the letter 'F' next to the satellite number in the satellite information display when the scenario is running. Created multipath taps (taps 2 to 5) are indicated by letter 'D'.

3.5.9.5 Atmospheric model

Atmospheric conditions have an effect on the propagation of GNSS signals, and as such can be an error source. GSG allows for these effects to be simulated, by applying tropospheric and ionospheric models to a scenario.

To configure these models, navigate to:

Select > **[Select Scenario]** > **Configure scenario, View 2/3** > **Advanced** > **Atmospheric model**.

Iono model

The GSG unit comes with built-in support for a model of the ionosphere. By default the used model is a reverse model of the model described in IS-GPS-200D, Section 20.3.3.5.2.5, called **Klobuchar**.

The a0-3 and b0-3 parameters set in the default model are set by the used navigation data files. When set to **Off**, no delays caused by the ionosphere are used in the simulation.

Under normal testing conditions, the **Klobuchar** ionosphere model should be used.

> **Note:** The GSG also supports simulation of ionosphere delays using files in the IONEX format.

Tropo model

A number of tropospheric models are supported by the device. These are:

- **Saastamoinen** model. The model is based on Saastamoinen, J., 'Atmospheric Correction for the Troposphere and Stratosphere in Radio Ranging of Satellites,' The Use of Artificial Satellites for Geodesy, Geophysics Monograph Series, Vol. 15., American Geophysical Union, 1972
- **Black** model. The model is based on Black H., 'An Easily Implemented Algorithm for the Tropospheric Range Correction', JOURNAL OF GEOPHYSICAL RESEARCH, 1978
- **Goad&Goodman**, a tropospheric model based on Goad and Goodman(1974), "A Modified Hopfield Tropospheric Refraction Correction Model", 1974
- **STANAG** model. The model is based on NATO Standardization Agreement (STANAG) Doc. 4294, Appendix 6.

The tropospheric model can also be set to **Off**, and no tropospheric delays are used in simulation. Under normal testing conditions, one of the tropospheric model *should* be used.

The tropospheric model also allows for the temperature, pressure and humidity to be configured:

- **Temperature**: to be specified in degrees Celsius
- Atmospheric **pressure**: in millibars
- **Humidity**: relative humidity in percent.

The graph below illustrates the delays for the different models available, using default values for environmental conditions.

Note that the tropospheric delay added to satellites with low elevation angles are 'capped' at a maximum value. The capping delay value and the elevation angle are a function of the model used.

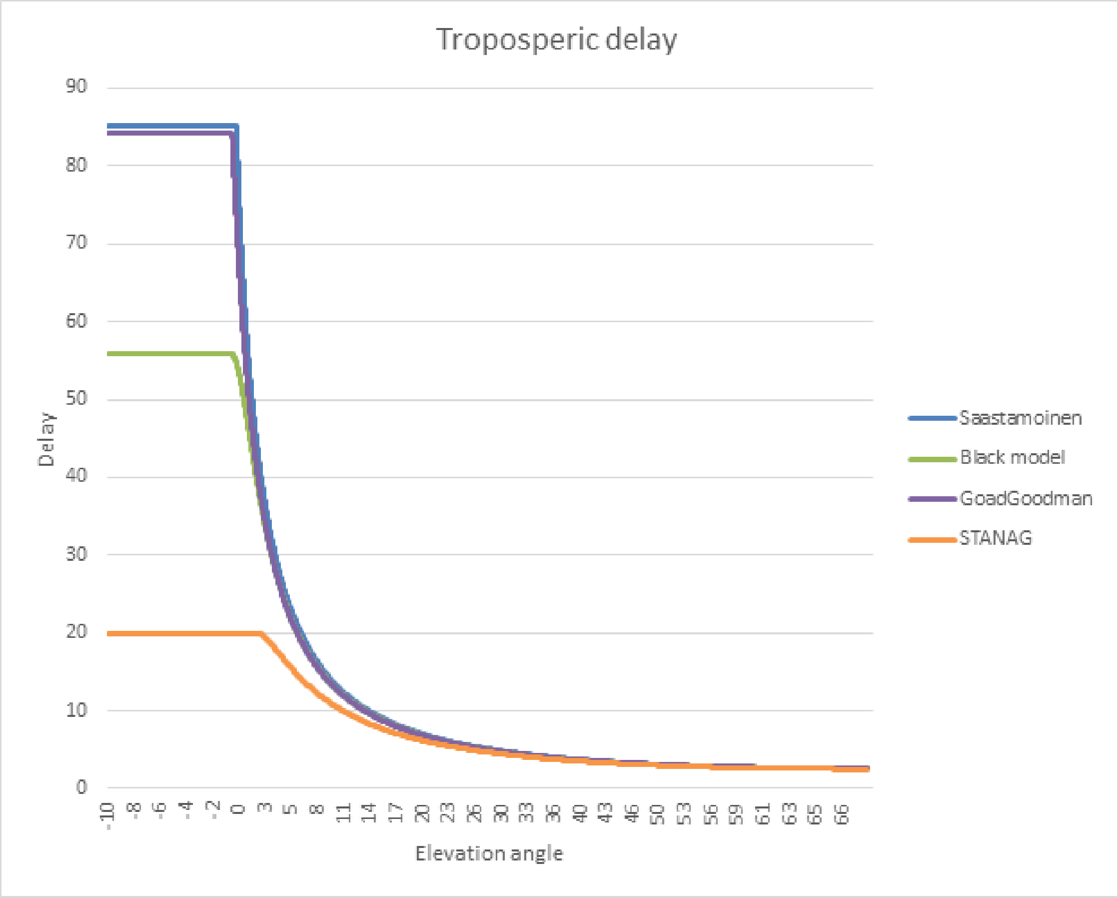


Figure 3-18: Tropospheric delay vs. elevation angle

3.5.10 Satellite Configuration

Depending on the model and configuration of your GSG unit, and the scenario chosen, several satellite systems can be simulated in a scenario, each of which you may want to configure in accordance with the requirements for your receiver-under-test.

The illustration below shows the configuration of GPS-based satellites as an example:

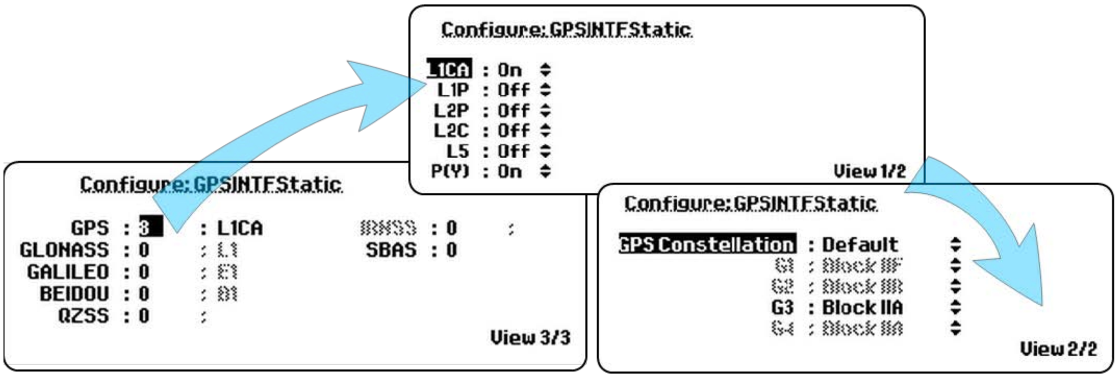


Figure 3-19: GPS satellite configuration

To access the first satellite configuration view, navigate to **Select** > **[Select scenario]** > **Configure scenario: View 3/3**.

The following satellite-relevant settings can be configured:

- **Satellite System**, e.g., GPS, Glonass (see "Satellite Systems" below)
- **Number of satellites** simulated for a given satellite system (see "Number of Satellites" on the next page)
- **Signal Type**, e.g., L1, L2 (see "Frequency Bands and Signal De-/Activation" on the next page)
- **Satellite Constellation** [GPS: "block"] (see "Satellite Constellations" on page 79)
- **Encryption** (see "Encryption" on page 81)
- **SBAS/Augmentation** (see "SBAS Satellites" on page 82)

3.5.10.1 Satellite Systems

The following navigation satellite systems can be simulated by GSG-series constellation simulators, depending on unit configuration, see also "GSG Series Model Variants and Options" on page 203:

- GPS
  - USA; globally operating system, very accurate, regular modernization and upgrading
- GLONASS
  - Russia; globally operating system, works independently from US military controlled system; combination of Glonsass + GPS solves "urban canyon" problem

- GALILEO
  - Europe; globally operating system; yet, not fully operational as of summer 2015; high-quality signals, multiple uses
- BEIDOU
  - China; regional system (Asia); planned global expansion; open system
- QZSS
  - Japan; regional system
- IRNSS
  - India; regional system

3.5.10.2 Number of Satellites

The maximum number of satellites to be simulated by GSG in a given scenario is specified separately for each available GNSS system. (For SBAS, see "SBAS Satellites" on page 82).

To edit the number of satellites for a GNSS system, navigate to: **Select** > [**Select Scenario**] > **Configure Scenario**: **View 3/3** > [**Satellite System**]: Enter a number"Number of Satellites" above

The theoretical maximum number of satellites that can be simulated is 64, but this number also depends on:

- The **license** and **GSG model** used (number of available **channels**)
- How many **frequency bands** are used, e.g., if 64 channels are available, 64 GNSS L1 satellite signals can be simulated, or, e.g., 32 L1/L2 satellite signals. (Note that GPS L2 and L2C are using separate channels, as are the Galileo bands E5a and E5b.)

The default setting is **Auto**, i.e. GSG will determine the number of satellites simulated at any given time during scenario execution.

> **Note:** If GSG runs out of free channels when in Auto mode, not all satellites will be simulated.

3.5.10.3 Frequency Bands and Signal De-/Activation

When testing GNSS receivers, it is oftentimes required to test for multi-frequency, multi-constellation performance. All of the four major GNSS systems, i.e. GPS, Glonass, Galileo, and BeiDou, transmit numerous signals across several frequencies, but through international cooperation, these frequency bands have been coordinated:

The RF signals transmitted from satellites of different constellation systems...

- ... are transmitted on frequencies close to each other, yet they do not interfere with each other
- ... can be decoded by one receiver (if supported by the receiver manufacturer)
- ... can be grouped into four main bands.

These four frequency bands are:

Constellation	Frequency Bands			
	1	2	3	4
GPS	L1	L2/L2C	L5	
Glonass	L1	L2		
Galileo	E1		E5	E6
BeiDou	B1		B2	B3

For multi-frequency, multi-constellation testing it is suggested to test any of the constellations, frequency bands, or any combination together.

The following frequency bands can be generated (GSG-configuration dependent):

For GPS:

- L1CA
- L1P
- L2P
- L2C
- L5
- P(Y): Pseudo encryption

For Glonass:

- L1
- L2

For Galileo:

- E1
- E5A
- E5B

For BeiDou:

- B1
- B2

For QZSS:

- C/A
- SAIF
- L2C
- L5

Active Signals

Frequency bands can be turned ON/OFF separately, so as to configure which types of RF signals specific to each supported satellite system shall be active/inactive when a scenario is running.

Depending on the configuration of your GSG unit, all of the frequency bands listed above can be turned ON/OFF.

To turn ON/OFF a signal band, navigate to: **Select** > [**Select Scenario**] > **Configure Scenario**: **View 3/3** > [**Satellite System**]: Enter a number of satellites > 1 (see "Number of Satellites" on page 76).

The satellite constellation (see "Satellite Constellations" on the facing page) must be configured accordingly, in order to allow for, e.g., the L2C band to be simulated. In other words, if you chose to disable satellites that can generate this signal, it will not be generated, even if you activate the signal. Hence, it is recommended to leave all signal types ON (default), thereby letting the configured satellite type determine which RF signals are active.

Use cases for turning OFF the transmission of individual frequency bands are:

- simulating a one-band antenna
- reserving the maximum number of channels for other requirements (e.g., L1-only transmission)

Considerations:

- Altering active RF signals will not alter the navigation message. Hence from a receiver point of view, choosing to de-active L2 and L5 will mimic the situation of using a single band (L1) antenna.
- Settings are GNSS-specific, not satellite-specific.
- For GLONASS, C/A code is always used.

3.5.10.4 Satellite Constellations

Once existing GNSS satellites of a satellite system in orbit are being replaced by new, more modern satellite types, the satellites are often categorized by their **generation**, or historic **constellation**. In the case of the GPS system, these constellations are named by their **block** numbers, e.g., "IIA".

> **Note:** The functionality described below only applies to GPS and Glonass. Other installed satellite systems, such as Galileo, still have their first generation of satellites in orbit.

GSG offers three options to configure satellite constellations:

I. The **Default** setting refers to the constellation state for April 22, 2015.

II. Constellation-wide setting of the satellite generation, e.g., by setting all GPS satellites to **Block IIR-M**:

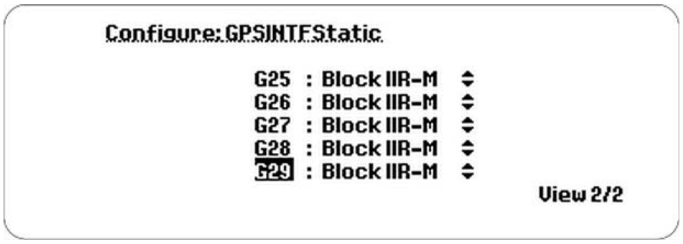


Figure 3-20: Assigning one constellation block to all satellites

To access this configuration view:

1. Navigate to **Select** > [**Select Scenario**] > **Configure Scenario**: **View 3/3**
2. Next to the desired **Satellite System** enter a **number of satellites** greater than "0", or **Auto** (see "Number of Satellites" on page 76), and press Enter to open the first configuration view, then the RIGHT arrow key to open View 2/2.

> **Note:** The G## numbers refer to the individual GPS satellites (Glonass satellites are named R##).

III. Explicitly **specify** the constellation for each individual satellite, using GSG StudioView:

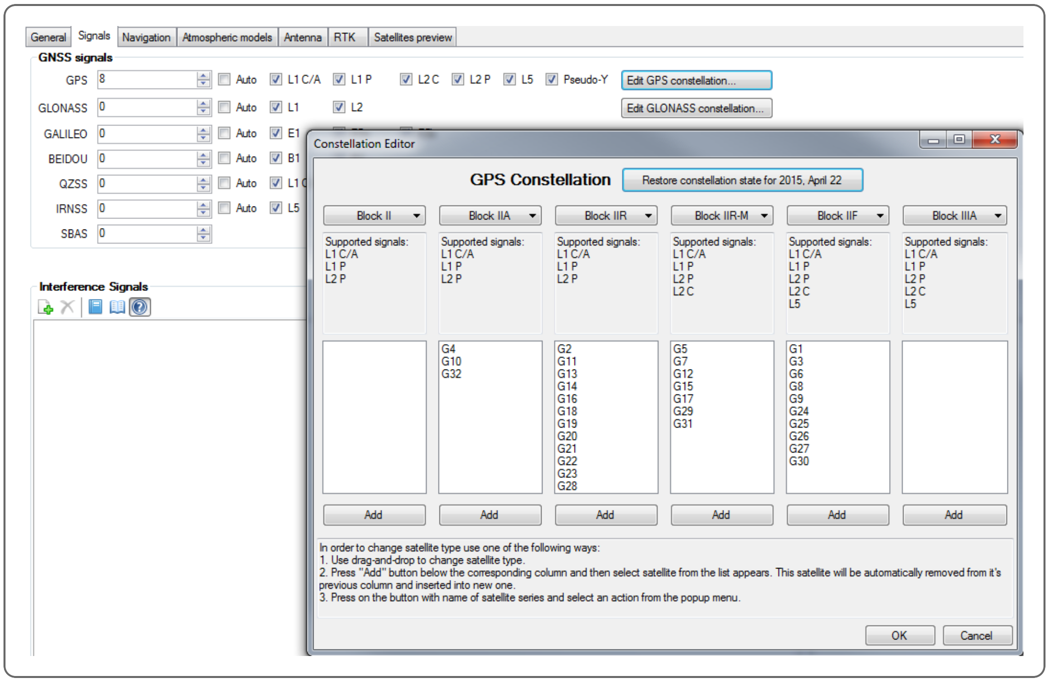


Figure 3-21: GPS Constellation configuration (StudioView)

This functionality may be required for the configuration of scenarios taking place in the past, or 'What-if' scenarios.

Consider the following when configuring satellite constellations:

- The selected satellite constellation will impact the **navigation message** to mimic the type of simulated satellite.
- The satellite type will also impact the types of RF signals generated (see "Frequency Bands and Signal De-/Activation" on page 76), i.e. for the signal type **L2C** to be transmitted, the satellite type must be Block IIR-M (or higher), for **L5** to be transmitted, the satellite type must be of type Block IIF (or higher), etc.

Possible settings are:

For GPS:

- II
- IIA
- IIR
- Block IIR-M
- IIF

- (default)

For Glonass:

- Glonass-K1
- Glonass-M
- (default)

3.5.10.5 Encryption

Next to the unencrypted L1 band Coarse/Acquisition Pseudo-Random Noise code (C/A PRN code), the Precise (P), but encrypted Pseudo Random Noise code is used to modulate both the L1, and the L2 carriers.

While GSG cannot replicate the encryption, it can emulate, and thus represent the P(Y) code, so as to allow for commercial GPS surveying receivers to be tested for their ability to derive the carrier in a codeless fashion.

Note that this technology does NOT use controlled encryption. Instead, it mimics the encryption so as to provide an RF signal in the L1/L2 P(Y) location.

Note: GPS receivers that use genuine encryption methods will NOT be able to use the L1/L2 P with Pseudo P(Y) code enabled because the encryption used is not as expected and they cannot decode it.

To turn P(Y) ON/OFF:

1. Navigate to: **Select [Select a Scenario] > Configure Scenario: View 3/3**.
2. Next to the **GPS**, enter a **number of satellites** greater than "0", or **Auto**, then press Enter to open this **Configuration** view:

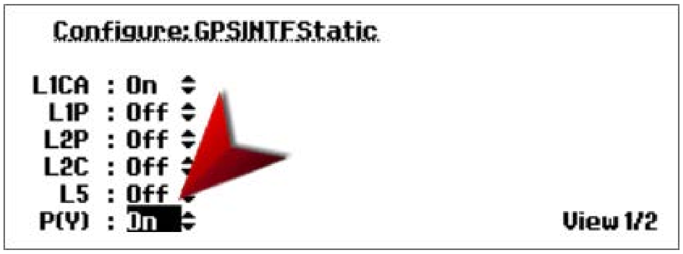


Figure 3-22: Turning pseudo encryption ON/OFF

3. Navigate to the **P(Y)** entry at the bottom of the view, and select **On**, or **Off**.

Considerations:

For most L1/L2 GPS receivers, there are two valid configuration modes:

1. Enable L1 C/A, L1P, and L2P only:
   - The L1P and L2P will be transmitted without encryption.
2. Enable L1 C/A, L1P, and L2P, and Pseudo P(Y):
   - The **P code** will be scrambled to mimic a realistic P(Y) signal for use in receivers that can make use of L1/L2 P(Y) signals for codeless applications, or to provide a signal in the band to better emulate the real world.
   - In the **GSG-6** series, the NAV message transmitted by the GPS satellites is updated to reflect if (pseudo-) encryption is active or not. This is specified by bit 19 in the second word of subframe one. This bit represents the anti-spoof (A-S) flag, where “1” indicates that the A-S mode is on in that satellite. It is recommended to enable Pseudo P(Y) when the GSG-unit supports it. This will set the A-S flag to ON which is required in some receivers. GPS receivers may reject L1CA code if the A-S flag is off.
   - In **GSG-5x** units, where it is not possible to transmit Pseudo P(Y), the A-S bit is always set to ON to indicate that encryption is on (although the actual RF signal is not transmitted on such units).
   - The **NAV message** also holds information on the type of L2 signal being transmitted (bits 11 and 12 of word three in subframe one). These bits are always set to indicate that the P code is active on L2.

3.5.10.6 SBAS Satellites

Several GNSS augmentation systems, e.g., differential GPS, exist to further improve positioning, navigation, and timing functionality (see also: www.gps.gov). Space Based Augmentation Systems (SBAS) incorporate system components such as additional SBAS geo satellites, ground reference stations, and user equipment which together aid the GPS system, thereby allowing greater precision and integrity, among other things.

SBAS systems support specific GNSS systems, are available for civil use, and have been/are being developed for all of the GNSS systems worldwide:

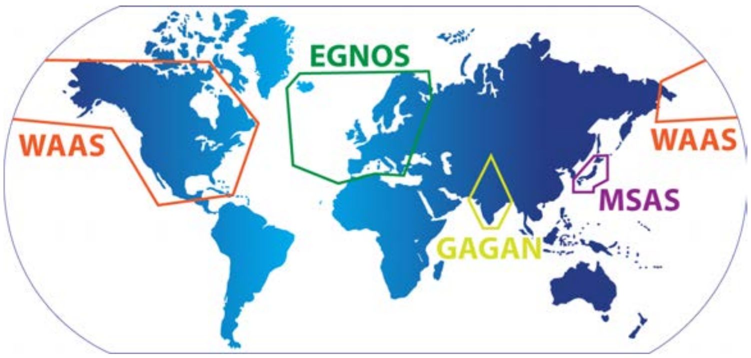
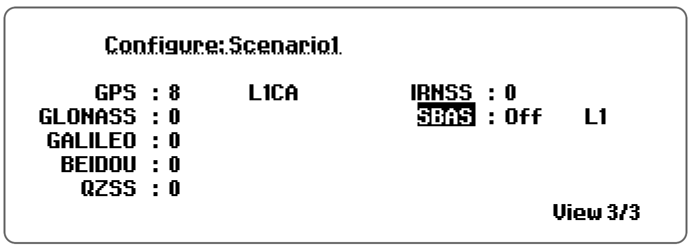
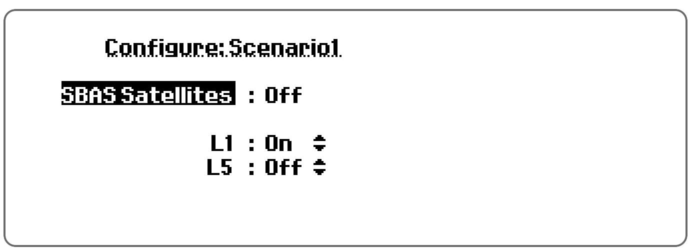


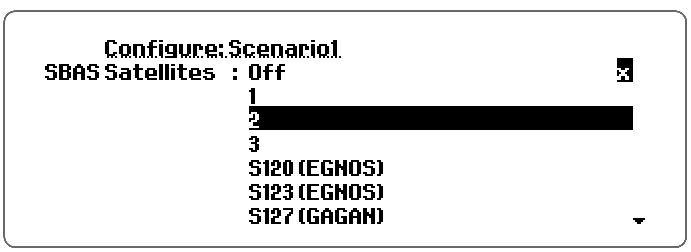
Figure 3-23: GNSS SBAS systems

GSG can simulate SBAS satellites at frequency bands L1 and L5. Each scenario defines the number of SBAS satellites that should be simulated. There can be 0, 1, 2, or 3 SBAS satellites per scenario. You can also specify individual SBAS satellite IDs to simulate.

To review/edit the number of SBAS satellites for the scenario chosen, navigate to: **Select** > [**Select Scenario**] > **Configure Scenario**: **View 3/3**"Number of Satellites" on page 76







If an integer number of SBAS satellites is specified, the GSG unit will select SBAS space vehicles based on their elevation relative to the user position. When the scenario is running, the SBAS satellite positions and speed will be updated with the information found in the SBAS messages. These messages comprise different **Message Types**, one of which—MT9—is used to update the satellite's position and speed.

You can also select up to 3 specific satellites to simulate.

The SBAS satellites transmit their signals utilizing Coarse/Acquisition Pseudo-Random Noise (see also "Encryption" on page 81). **PRN numbers**, which have been internationally coordinated, have been allocated to each of the SBAS constellations. Although PRN120 ... PRN158 are all reserved for SBAS systems, only a few of them are actually used by satellites.

When determining the elevation angle of SBAS satellites, the GSG unit looks for the SBAS satellites listed below. This is in contrast to the signal generator mode (see "Signal Generator" on page 94) where the user can specify any SBAS PRNs to be simulated.

The currently supported SBAS satellites are:

» **EGNOS**: 120, 123, and 136

» **WAAS**: 131, 133, 135, and 138

» **MSAS**: 129, 137

» **GAGAN**: 127, 128, and 132

The simulator uses two approaches for SBAS messages:

» Default SBAS messages (MT63)

» EGNOS/WAAS message files

The default SBAS messages are always available. These messages should be recognized by SBAS-compatible receivers. However, they carry no information and will therefore not enable the receiver to correct GPS signals.

SBAS message files for both EGNOS, and WAAS are supported. EGNOS files (.ems) are ASCII and hourly, while WAAS files are typically in binary format and cover a whole day. Both systems share the same format of the messages. For details, see www.navipedia.net.

When the scenario has the Ephemeris set to **Download**, the GSG unit will download the SBAS messages from official sites and match these messages to the time of the scenario. The SBAS messages broadcast by these satellites are downloaded automatically from the following public FTP sites:

» **EGNOS**: ftp://131.176.49.48

» **WAAS**: ftp://ftp.nstb.tc.faa.gov

» **MSAS**: www.enri.go.jp

» **GAGAN**: `default MT63`

GSG logs into these sites anonymously. However, note that both FTP sites are likely to track and record all FTP access, including access by GSG simulators.

Considerations

If a scenario needs SBAS messages that cannot be downloaded from these FTP sites, the scenario continues, but the GSG unit transmits null-messages (SBAS message type: MT63). An SBAS-compatible receiver should still be capable of seeing the SBAS signals, but it will not find any useful information (range corrections, time offsets, etc.) in these messages.

Because of these reasons, SBAS scenarios run best with a live Internet connection. Furthermore, since the aforementioned FTP sites store only a limited amount of SBAS records, the start time of SBAS scenarios has to be chosen carefully:

Usually, SBAS records that are less than a year (EGNOS)/6 months (WAAS) old, can be found on the FTP sites mentioned above. Therefore, it is advisable to select a start time that is not older than one year for EGNOS scenarios, and not older than 6 months for WAAS scenarios.

Moreover, the start time shall not be too close to the current time. For EGNOS, there can be a one-day delay before the SBAS messages are published on the FTP site. For WAAS the delay can possibly be longer (up to 3 or 4 days).

An Internet connection is not *always* needed, however: All downloaded ephemeris data and SBAS data will be locally stored on the unit, once they have been downloaded. Hence, the next time the same scenario runs, the ephemeris data and SBAS messages are read from the local storage, not from the online ftp sites.

GSG will performs automatic clean-up of downloaded files, once the remaining free disc space falls below 20% of the total disc space.

Note: The SBAS corrections are 'applied backwards' to the output GPS signals by adjusting the signal ranges.

It is also possible to download the EGNOS and WAAS files from the ftp servers, and select them for use in the scenario: The file name holds the information on the applicable time & date, which is NOT available in the content of the file (all time is relative), and must follow these naming conventions:

- For **EGNOS**: PRN<prn>_y<YYYY>_d<doy>_h<hour>.ems
- For **WAAS**: Geo<prn>_<GPSWeek>_<dayOfWeek>

Note: WAAS files do not have a file extension.

Should the files downloaded from the ftp server do not meet these format requirements, it will be necessary to rename the files accordingly.

QZSS L1 SAIF

The QZSS satellites transmit also a SBAS signal, called L1 SAIF. The GSG unit can emulate this signal. The signal is enabled by setting the value of "QZSSL1SAIF" to "1" in a scenario file.

If the user does not specify a file containing the messages for transmission, the unit will transmit only the default (MT63) messages. The naming convention for the transmitted files is the same as for the WAAS satellites above. The PRN numbers reserved for QZSS L1 SAIF transmission start from 183, so the name of the message file for J01 should start with "Geo183_", for J02 with "Geo184_", etc.

For the best results, the user should specify the Rinex navigation file(s) used in the scenario, together with the SAIF message files. This way the user can ensure that the simulated satellite position based on Rinex NAV files is in line with the position information transmitted in the L1 SAIF messages.

3.6 "Options" Menu

Features and functions that are not directly related to the scenarios are typically found under the **Options** Menu.

3.6.1 Transmit Power

The term **Transmit Power** refers to the satellite signal power (signal level) transmitted by GSG during the execution of the currently chosen scenario.

The Transmit Power can be adjusted as described under "Adjusting Transmit Power" on page 88, or during scenario execution (see "Setting Transmit Power" on page 113).

Caution: To learn more about signal level compliance in the United States, see "Signal Power Level Considerations" on page 24. If you live in other countries, check your local emission standards.

- The transmit power is specified in **dBm**.
- The supported range is: Max. **-65 dBm** ... min. **-160 dBm**.
- The resolution is: **0.1 dBm**.
- Default setting: **-125.0 dBm**.

Note: The External Attenuation setting decreases the set Transmit Power level.

> **Note:** When the power settings of individual channels are changed during scenario execution (via the > Events menu, or protocol) the power range will be further limited so that the maximum difference between the strongest and the weakest signal is never more than 72 dB.

To access the **Transmit Power** view, navigate to **Options** > **Transmit Power**. This view also allows you to adjust the **external attenuation** (see "Adjusting External Attenuation" on page 90), and **noise** (see "Adjusting Noise Generation" on page 91).

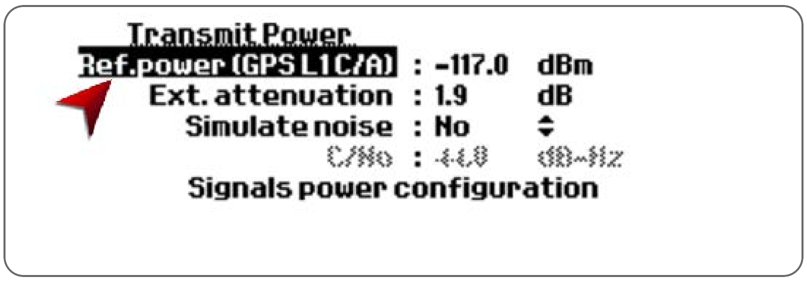


Figure 3-24: Configuring transmit power

Antenna cable length

The **recommended Transmit power setting**, assuming relatively short cables and that no external attenuators are used, is **-125.0 dBm**. If long cables are used, it is recommended that these are simulated by adjusting the external attenuation (see also "Adjusting External Attenuation" on page 90).

The Transmit Power set in the **Options** menu is assigned to the signal type with the highest power level, and all others are set relative to that.

Considerations

A common problem is that signals too strong or too weak are used. A signal too strong will typically 'jam' the receiver, causing it to erroneously find many shadow signals. It is recommended that you familiarize yourself with the typical signal/noise values for real satellites, and try to obtain similar values when using this unit. When the signal strength is correctly set, the receiver will respond directly and logically to changes in signal power.

The following table shows the offsets when referencing **GPS L1 C/A** as **zero dB** offset:

Table 3-2: Transmit power offsets

Constellation	Signal	Power offset, dB
GPS	L1 C/A	0.0
	L1 P	-3.0
	L2 P	-3.0
	L2C	0.0
	L5	+1.5
GLONASS	L1	-2.5
	L2	-8.5
Galileo	E1	+1.5
	E5a	+3.5
	E5b	+3.5
Beidou	B1	-4.5
	B2	-4.5
QZSS	L1 C/A	0.0
	L1 SAIF	-2.5

3.6.1.1 Adjusting Transmit Power

The term **Transmit Power** refers to the satellite signal power (signal level) transmitted by GSG during the execution of a scenario. **Transmit Power** can be controlled individually by signal type:

As of software version 7.1.1 (January 2018) the **Reference Power** setting is used to control the absolute power level of the GPS L1 C/A signal. Then, the default relative power offset can be adjusted for each individual signal type other than GPS L1 C/A.

Furthermore, for BeiDou only, you can assign different relative power offsets, depending on the orbit type (MEO/GEO/IGSO).

This power configuration defines by channel which power level will be used by satellites once they appear in view.

The power configuration can be changed only before starting a simulation. During the simulation, you can see the power configuration in read-only mode (press the N/S key to open the **Transmit power** menu). Note that the effect of changing the **Reference power** *during* the simulation is different than *before* the simulation: During the simulation, the specified **Absolute power** will be set for all active channels.

There are two ways to adjust the power configuration: Via the front panel (see below), or by using SCPI commands (see "SOURce: Subsystem Commands" on page 236). Once you modified the power configuration, it will be saved with all other settings when the unit is turned off.

To configure signals power:

1. Navigate to **Options > Transmit Power**.
2. Adjust the GPS L1C/A band **Ref. power**. The power is specified in **dBm**. The supported range is: Max. -65 dBm ... Min. -160 dBm.
3. Select **Signals power configuration** to open the corresponding menu.
4. Select the desired constellation, and adjust the power for each signal type supported by this constellation.

> **Note:** When changing the power setting for a signal type, the Reference Power (absolute power level of GPS L1 C/A) and Relative Power offsets for all the remaining signal types will remain unchanged.

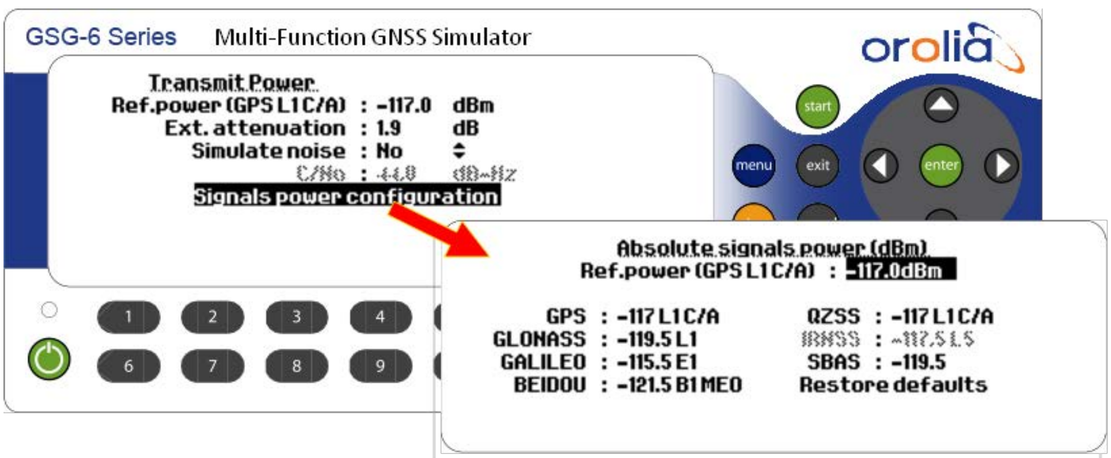


Figure 3-25: Signals power configuration menu

Use the format key to switch between the absolute or relative mode of displaying/editing power. When the **Absolute Power** mode is active and **Reference Power** is changed, the **Relative Power** offsets will stay unchanged, so that absolute powers will be "shifted" together with the **Reference Power**.

Default Power Configurations

Signal name	Orbit type name	Absolute power (dBm)	Relative power (dBm)
GPSL1CA (reference)		-128.5	0

Signal name	Orbit type name	Absolute power (dBm)	Relative power (dBm)
GPSL1P		-131.5	-3
GPSL2P		-131.5	-3
GPSL2C		-128.5	0
GPSL5		-127	+1.5
SBASL1		-131	-2.5
SBASL5		-127.9	+0.6
GLOL1		-131	-2.5
GLOL2		-137	-8.5
GALE1		-127	+1.5
GALE5A		-125	+3.5
GALE5B		-125	+3.5
GALE6		-125	+3.5
BDSB1	MEO*	-133	-4.5
BDSB1	IGSO*	-133	-4.5
BDSB1	GEO*	-133	-4.5
BDSB2	MEO*	-133	-4.5
BDSB2	IGSO*	-133	-4.5
BDSB2	GEO*	-133	-4.5
QZSSL1CA		-128.5	0
QZSSL2C		-130	-1.5
QZSSSAIF		-127.9	+0.6
IRNSSL5		-129	-0.5

*Orbit type for BeiDou satellites determined by PRN number (for more information, see https://www.glonass-iac.ru/en/BEIDOU/): GEO: 1<=PRN<=5, IGSO: 6<=PRN<=10 and PRN=13, MEO: others

3.6.1.2 Adjusting External Attenuation

External attenuation allows you to specify attenuation between the GSG power output, and the receiving device. This allows the unit to compensate e.g., for antenna cable lengths. Any of the power settings (Transmit power, Event settings) will observe the specified external attenuation.

- The range is: 0 ... 30.0 dB
- Resolution: 0.1 dB.

To adjust External Attenuation, navigate to **Options** > **Transmit Power**.

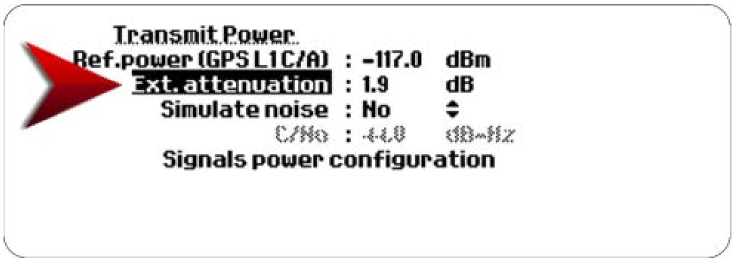


Figure 3-26: Adjusting external attenuation

3.6.1.3 Adjusting Noise Generation

GSG-5/6 has the capability to simulate noise on the GPS L1 band. Noise simulation can be a powerful tool for receiver testing, since it allows for a strong signal to be submitted, without jamming the receiver.

To access the **Transmit Power** view, which—among other things—allows to adjust the noise settings, navigate to **Options** > **Transmit Power**:

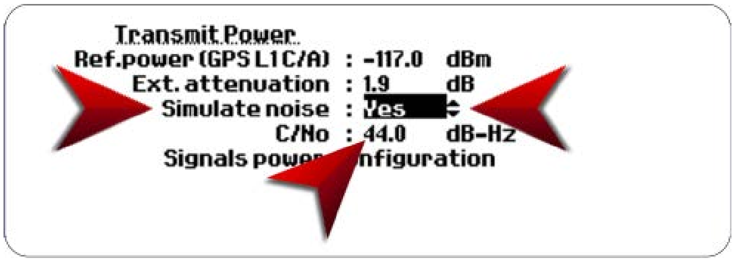


Figure 3-27: Adjusting noise settings in the Transmit Power view

Noise generation (GSG-5, GSG-56 and GSG-6)

The noise generated by GSG-5, GSG-56 and GSG-6 is similar to the noise of GSG-55, but differs in that the noise bandwidth is constant and set to cover all simulated bands. The noise central frequency is not configurable.

Noise-related adjustable parameters

- Simulate Noise: Yes/No (Default: Yes)
- C/N_O: Carrier-to-noise density. Range: 0 ... 56 dB-Hz

General considerations

International regulations keep the L1 band practically clean from disturbing signals, so the only noise source is the natural background noise, as expressed in the following equation:

» $P_N = kTB_N$

Where k is the Bolzmann's constant, T is the ambient temperature (in Kelvin), and BN is the bandwidth (in Hertz).

For example, an ideal GPS L1 C/A code filter would have a passband of 2MHz, and noise power passed by the filter at a temperature of 290 K would be equal to -141 dBW.

The ambient noise power spectral density is given by the equation:

» $N_T = kT = 4.00 \times 10^{-21}$ W/Hz = -204 dBW/Hz = -174 dBm/Hz

By definition, carrier-to-noise density is the carrier power divided by the noise power spectral density. The GPS ICD specifies that the received signal level at the surface level is -130 dBm or better. Carrier-to-noise density is then:

» C/N_0 = -130 dBm/(-174 dBm/Hz) = 44 dB Hz

C/N_0 (not SNR) is the figure that the receivers typically display as an indication of quality for the received, digitally modulated signal. If the receiver has bandwidth of 6 MHz, SNR would be:

» SNR = 44 dB Hz/(6×10^6 Hz) = 44 – ($10 \times \log_{10}(6 \times 10^6)$) dB = -23.8 dB.

If a stronger input signal for the receiver is required, while maintaining the same C/N_0, additional noise needs to be introduced into the transmitted signal. One may think of this as having an active antenna at the receiver input. The signal level is higher, but so is the noise level.

Interaction of Transmit Power and External Attenuation

When you change the values in the Transmit Power dialog, you may notice that other settings may change as a consequence of the changes made. For example, if you have Transmit Power set to -70 dBm, and External Attenuation set to 5.0 dB, the unit actually transmits signals at -65 dBm to compensate for the external losses.

Note, however, that manually adjusting the attenuation to 10 dB in such a situation will cause the Transmit Power to drop to -75 dBm as a consequence. This is a result of the hardware configuration, as the unit cannot deliver more than a total of -65 dBm. The Transmit Power setting gives the power level at the end of your antenna cable.

Adjusting Transmit Power: Best practices

In general, when changing the Transmit Power setting, it is recommended to follow this order:

1. Set the External Attenuation
2. Set the Transmit Power
3. Set the Noise Bandwidth
4. Set the Carrier-to-Noise Density
5. Set the Noise Offset (this can be done at any time without affecting the other settings)

Adjusting Power/Noise via SCPI command

If you use the SCPI protocol to change the power/noise settings, use the order above to do modifications, and check the SCPI error after each command. If there is a Parameter Conflict error, it would indicate that the unit accepted your command, but due to a conflict with a different parameter, your parameter value was modified.

The conditions under which a Parameter Conflict may occur include the following:

1. **A Transmit Power value has been requested that is too high**. The requested Transmit Power is within the specified limits, but the External Attenuation setting limits the maximum power to below the requested setting. Transmit Power is set to the maximum available, rather than the value requested by the user. Increasing the Transmit Power may lead to an increase of C/N_0, as described under bullet #3 below. To prevent this from happening, especially when using the SCPI protocol for making adjustments, always use the command order described above, and check the SCPI errors after each command.
2. **An Unachievable Carrier-to-Noise ratio has been requested**. The requested value is within specifications, but the Transmit Power setting is too low to achieve the required setting. In this case, the ambient noise power spectral density limits the achievable carrier-to-noise ratio. The Carrier-to-Noise density will be set to its maximum value, not to the value requested by the user. The noise generator does not generate any additive noise in this situation. Increase the Transmit Power, then set C/N_0 again.
3. **A Carrier-to-Noise ratio has been requested that is too low.** The requested value is within specifications, but the Transmit Power setting is too high to achieve the required setting. The signal/noise generator does not have the capability to generate a noise signal this strong (remember that noise power is more than the signal power – SNR is negative). The Carrier-to-Noise density will be set to its minimum value, not to the value requested by the user. Decrease the Transmit Power to decrease the required noise power.
4. **A Noise Bandwidth value has been requested that is too wide.** (SCPI command only) In effect, this leads to the same situation described under bullet # 3 above. GSG accepts the noise bandwidth setting, but increases the C/N_0 to its minimum value. The noise bandwidth required depends on the filters of the receiver. You have to search for the value that is wide enough for your receiver. Set up a relatively

strong signal (for example: -100 dBm, C/N_0 44 dB-Hz), and narrow noise bandwidth. Then increase the noise bandwidth until the C/N_0 value shown by your receiver stabilizes. It is a good idea to use the narrowest bandwidth needed.

> **Note:** The receivers use different methods to calculate C/N_0 (or SNR), so the value given by the receiver may be different from the C/N_0 setting of the GSG unit.

3.6.2 Signal Generator

Every GSG model can be operated as a signal generator, i.e. to generate one, or—if so equipped—several satellite signals (with no Doppler), or one carrier frequency.

In **Signal Generator** mode, advanced GSG units can support: GPS, GLONASS, Galileo, BeiDou, SBAS. If equipped with the L2 and/or L5 options, GSG allows the selected satellite(s) to transmit all signals enabled on that satellite.

> **Note:** For more information on available GSG models and options, see "GSG Series Model Variants and Options" on page 203.

To configure the **Signal Generator** mode, navigate to **Options** > **Signal Generator**:

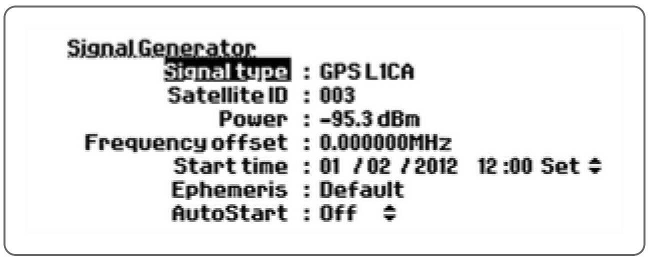


Figure 3-28: Signal Generator configuration view (depends on licensing options installed)

The following **Signal Generator** options can be configured:

3.6.2.1 Signal type

The Signal type selection will open a new view, as shown below. Note that the view depends on the licensing options installed on your unit.

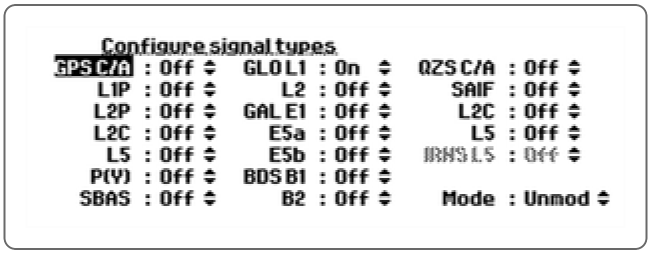


Figure 3-29: Signal types configuration view

Combining signals from different GNSS systems

If your GSG unit is licensed for multiple channel operation, in **Signal Generator** mode it is not only possible to choose between multiple frequency bands and codes, but also to simulate several GNSS signals, e.g., both GPS and GLONASS, at the same time. This can be achieved by enabling several GNSS systems from the **Configure signal types** menu.

In Signal Generator mode, GSG offers the following **Signal type** configuration options:

- GNSS systems currently supported are: GPS, Glonass, Galileo, BeiDou, and QZSS, and their corresponding signal types. For information on signal types, see also "Frequency Bands and Signal De-/Activation" on page 76.
- Pseudo-encryption (P(Y)): For more information, see "Encryption" on page 81.
- SBAS Signals: It is possible to generate a signal for any of the SBAS PRNs. However, GSG can generate a real SBAS message stream only if the chosen PRN corresponds to a live SBAS satellite (see "SBAS Satellites" on page 82 for further details).

  > **Note:** The SBAS signal type is only available with GSG-55, GSG-56, and GSG-6 Series units.

  Note that in signal generator mode (unlike in constellation simulation mode), GSG will always attempt to download SBAS data. If such data is not available, then MT63 (i.e., “null messages”) will be transmitted.
- Modulation options:
  - **Modulated**: This is the default mode, transmitting standard, modulated signals.
  - **Sweep or Noise**: In addition to the modulated signals, sweeping interference or narrowband noise interference will be transmitted. Currently it is not possible to use sweep/noise with unmodulated signals.
  - **Pure Carrier Signal**: GSG will transmit an un-modulated signal (pure carrier), using the user-specified signal strength, and frequency offset from the

nominal frequency. With GSG-6 series simulators, it is possible to generate carrier signals for L1 and L2 at the same time.

- **Prn**: Pseudo-random noise

3.6.2.2 Satellite ID

The **Satellite ID** field is used to specify the GPS PRN, Galileo PRN, and the GLONASS satellite ID, therefore it is limited to **24** (the highest GLONASS satellite ID). If this field is set to a value higher than 24, then GLONASS will not be selectable under **Configure signal types**.

3.6.2.3 Transmit Power

The term **Transmit Power** refers to the signal power transmitted by GSG during the execution of a scenario.

- The transmit power is specified in **dBm**.
- The supported range is: Max. -65 dBm ... Min. -160 dBm.

> **Note:** External Attenuation setting decreases the Max value. For more information, see "Adjusting Transmit Power" on page 88"Adjusting External Attenuation" on page 90.

> **Note:** When the power settings of individual channels change during scenario execution (via the > Events menu, or SCPI commands) the power range will be further limited so that the maximum difference between the strongest and the weakest signal is never more than 72 dB.

- The resolution is: 0.1 dBm.
- Default setting: -125.0 dBm

For more information on Transmit Power, see "Adjusting Transmit Power" on page 88.

> **Caution:** If you are using an antenna (rather than an RF cable), see "Signal Power Level Considerations" on page 24 regarding signal level compliance in the U.S. If you live in other countries, check your local emission standards.

3.6.2.4 Frequency offset

The **Frequency offset** applies to ALL of the simulated signals in the signal generator mode, i.e. once you set an offset, the code phases of the simulated signals start to shift compared to each other.

3.6.2.5 Start time

The **Start time** can be a set time, or the current time derived from an NTP server, as specified in your Network Configuration. If the current time is used, provided by an NTP server, the scenario start will be delayed, in order to allow the simulation to load required data, and start aligned to the nearest GPS minute.

The NTP (UTC) timescale is converted to the GPS timescale by a UTC-GPS offset defined in the firmware.

For more information, see "Start Time" on page 40.

> **Note:** If this field is grayed out, it is not applicable for the chosen configuration.

3.6.2.6 Ephemeris

If NTP start time is used, the **Ephemeris** cannot be downloaded, as this data is not available in real time. The simulated range equals to (25.0E-3*speed_of_light), so the 1PPS Out from the back panel would trail the time mark determined from the RF Out signal by 25 ms.

> **Note:** If this field is grayed out, it is not applicable for the chosen configuration.

3.6.2.7 AutoStart

If set to **ON**, **AutoStart** will start the Signal Generator mode automatically, once you powered up the GSG unit.

If set to **OFF**, the signal generation is started by pressing the START key, and stopped by pressing CANCEL. When a signal is generated, the simulated GPS time and the text **Transmission ON** are displayed (see illustration below).

Note that **Power** and **Frequency offset** can be edited while the transmission is ON.

Figure 3-30: Signal Generator running

All signal parameters are stored to non-volatile memory and are set when the unit is started.

3.6.3 Interface and Reference

GSG interface options can be configured via the **Options** > **Interface and Reference** view:

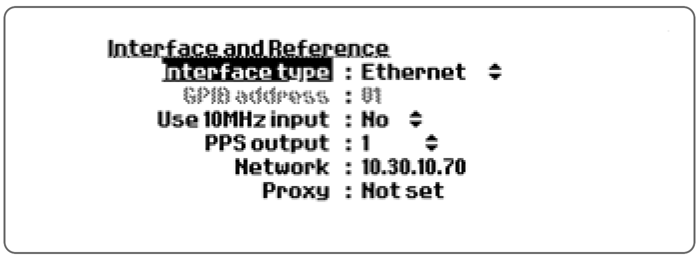


Figure 3-31: Interface/Reference Configuration

Depending on the type of interface chosen, only relevant fields are editable.

The remote interface type can be:

- **USB**
- **Ethernet**
- **GPIB**: Set the address here.
- **SCPI-Raw** network clients can use a socket connection to port 5025 and send/receive SCPI commands terminated by a newline.

The **10 MHz** input can also be selected via this view. When it is selected, a small symbol containing the text **EXTREF** is displayed in the upper right corner of the GSG display.

In all models except GSG-52 and GSG-53, the **PPS output** on the rear panel can be configured to send 1, 10, 100 or 1,000 pulses per second. The pulse ratio is always 1/10 (1/10 high, 9/10 low). **PPS Out** is active on the rising edge of the signal.

3.6.3.1 Network Configuration

To access the **Network configuration** view, navigate to **Options** > **Interface and Reference** > Select **Interface Type**: **Ethernet**. Highlight the menu item **Network**, and press enter; the **Network configuration** screen will be displayed:

IP Configuration

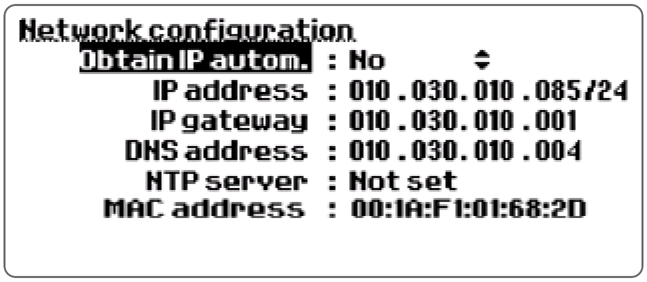


Figure 3-32: Static IP address configuration

In the **Network configuration** screen, you can configure GSG either to obtain an IP address automatically from a DHCP server, or you can specify a static IP address.

To specify a static IP address manually, you must provide:

- the **IP address**
- the **network mask**
- and the **gateway**.

> **Note:** In order for the ephemeris download to work, the correct DNS address must specified, either by setting Options > Interfaces and Reference > Network > Obtain IP autom. = Yes, or—when using a static IP configuration—by manually entering the correct DNS address.

If in doubt, consult your network administrator about the IP address configuration.

NTP Configuration

Under **Network configuration**, you can also—among other things—enable the current time, as delivered by an NTP server, to be used as the Start Time, by setting an NTP Server address.

1. To access the **Network configuration** view, navigate to **Options** > **Interface and Reference** > Select **Interface Type**: **Ethernet**. Then highlight the menu item **Network**, and press enter; the **Network configuration** screen will be displayed. Highlight the menu item **NTP server**, and press enter.
2. Enter the IP address of the NTP server on your network.

NTP client unable to set time

In the event that GSG cannot resolve the NTP server address, upon start-up, the error message NTP client unable to set time will be displayed:

1. Confirm the message by pressing enter, and navigate to:
2. **Network configuration** view, **Options** > **Interface and Reference** > Select **Interface Type**: **Ethernet**.
3. Then highlight the menu item **Network**, and press enter; the **Network configuration** screen will be displayed. Highlight the menu item **NTP server**, and press enter.
4. Enter a valid NTP address, or—if the IP address is correct—navigate up to **Network configuration**, and verify that the appropriate static IP address and gateway are selected so that GSG can resolve the path to the NTP server.

Download Server

The download server for the GPS ephemeris and almanac data can be configured under:

Network configuration > **Options** > **Interface and Reference** > [**Interface Type**: set to **Ethernet**.] > **Network** > **Network configuration**: **Download server**.

The choices are **Default**, and [**user-entered custom address**].

For more information on automatic download of ephemeris and almanac data see "Ephemeris" on page 47.

> **Note:** In order for the ephemeris download to work, the correct DNS address must specified, either by setting Options > Interfaces and Reference > Network > Obtain IP autom. = Yes, or—when using a static IP configuration—by manually entering the correct DNS address.

3.6.3.2 Proxy Configuration

If your GSG unit is used in a network behind an HTTP proxy, access to the proxy can be configured as described below:

1. To access the **Proxy configuration** view, navigate to **Options** > **Interface and Reference** > **Proxy**.

2. The Proxy address must be the address of the proxy including the http:// -prefix, and a port number after the address separated by ':'. Optionally, a username and password for the proxy can be given. If in doubt, consult your network administrator about the Proxy server settings.

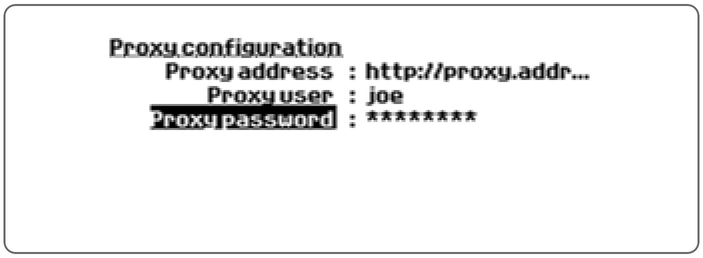


Figure 3-33: Proxy Configuration view

3.6.4 Manage Files

The **Manage Files** view display allows management of the navigation files, scenario files, trajectory files and event files. To access this view, navigate to **Options** > **Manage Files**:

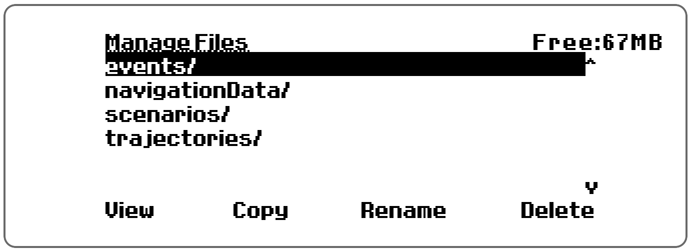


Figure 3-34: Manage Files top level view

Navigating

- The top level view shows the directories. To **select a directory**, use the UP/DOWN arrow keys and press ENTER.
- **Select files** within a directory by using the UP/DOWN arrow keys.
- To **go up** one level, select "**../**".
- To **perform an action** on a file, first select it, and then use LEFT/RIGHT arrow keys to select the desired action (View, Copy, Rename or Delete).
- To **return** to the previous level, press the CANCEL key. (i.e., this is the same as selecting "**../**". EXIT returns to the main menu).

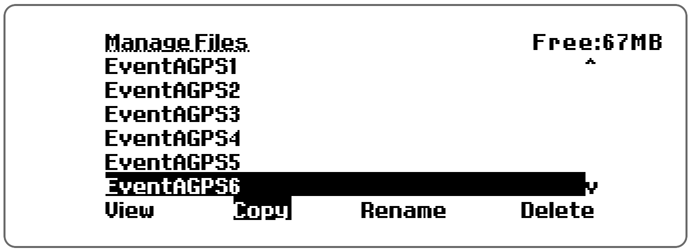


Figure 3-35: Choosing a file and an action

Copying and renaming files

- When **copying** or **renaming** a file, a keyboard is displayed for entering a new file name. Use the arrow keys and the ENTER key to select letters for the file name.
- The LEFT/RIGHT arrow keys move the cursor.
- The DEL key removes a letter **left** to the cursor.
- When your new file name is complete, press the EXIT key, or the DONE option.

> **Note:** If the file already exists or is in use, a confirmation for the action is requested.

- Use the CANCEL key to cancel the operation.

Figure 3-36: Keyboard

> **Note:** Directories cannot be created or deleted, and files cannot be copied between directories.

Viewing file contents

When viewing file contents, the screen can be scrolled up and down, and left and right using the arrow keys.

To exit the viewer, use the EXIT or CANCEL keys.

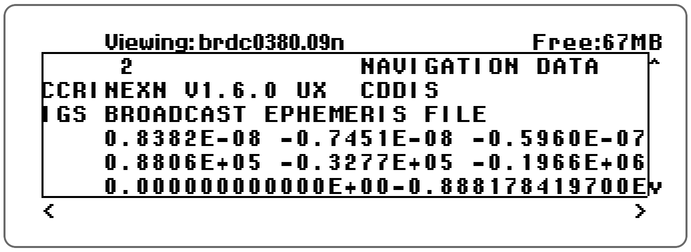


Figure 3-37: Viewing file content

3.6.5 Show System Information

The **System Information** view displays information about the GSG model, serial number, firmware version, oscillator type, and installed options (if any). In addition, the amount of free storage space available for scenarios and other user files is shown.

To access this view, navigate to **Options** > **Show system information**:

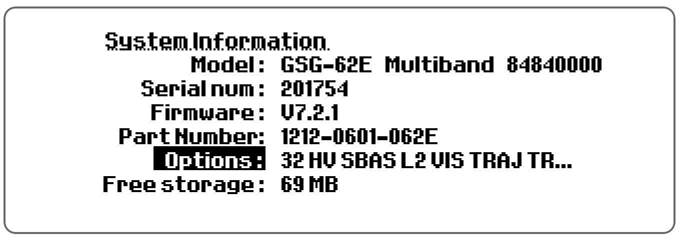


Figure 3-38: System information view

By selecting **Options** and pressing the ENTER key, you can also view the available and installed license options:

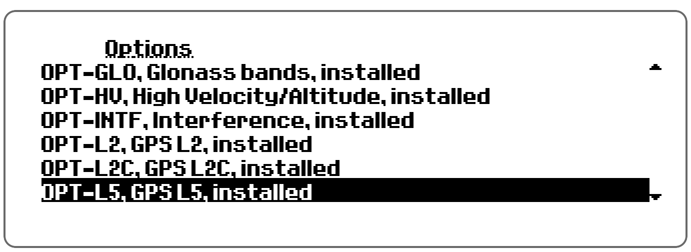


Figure 3-39: System information – Options

For more information on GSG Options, see "GSG Series Model Variants and Options" on page 203.

3.6.6 Restore Factory Defaults

This option restores the GSG unit to its factory default configuration.

To access this view, navigate to **Options** > **Reset to factory defaults**:

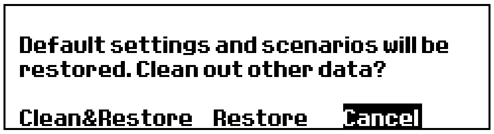


Figure 3-40: Restore factory defaults

- **Clean&Restore** will restore the original pre-defined **scenario/trajectory/event/navigation data files**, and **delete all user created/uploaded files and execution log**. Please wait for this operation to complete. When prompted that the operation is complete, press “OK”. Wait until this operation is complete before power cycling the GSG unit.
- **Restore** will only restore **data files** to their defaults – all user data on the unit will remain stored (unless they have same file names as the factory data, which is not recommended).
- **Cancel** will do nothing, and return to the **Options** menu.

3.6.7 Calibration

> **Note:** This chapter describes the Calibration menu items. Calibration itself should only be attempted by qualified technicians. Alternatively, you can send your GSG unit to Orolia to be calibrated.

Via the **Calibration** view, you can:

- calibrate the unit’s maximum output power, and OXCO frequency
- view the results of a previous user calibration.

To access the **Calibration** view, navigate to **Options** > **Calibration**:

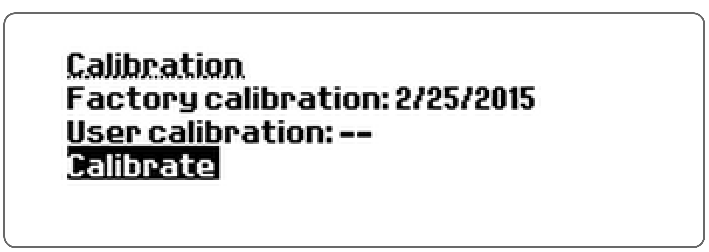


Figure 3-41: Calibration view

The **Calibration** view displays when the **Factory calibration** was done, and if and when the last **User calibration** was done.

Calibration Recommendations

Is recommended that you adhere to the following **calibration guidelines**:

1. Orolia recommends calibration **every 2 years** to ensure the frequency is within specification and the power levels are correct.
2. If your unit is equipped with an **Ultra-High-Stability OCXO** (an option that is no longer available), Orolia recommends calibration **every year** for this reference to ensure operation to specifications.
3. Regardless of which oscillator option is installed in your GSG unit: If you are testing GPS timing receivers and are testing the precision of the 1 PPS output, comparing it to the 1 PPS output from your device under test, Orolia recommends calibration every year.

To carry out a user calibration, highlight **Calibrate** and press enter. Confirm your choice, and enter the password, in order to make sure the calibration settings are not changed unintentionally. The password is **62951413** (first 8 digits of π backwards).

Figure 3-42: Entering the calibration password

Note: It is strongly advised to write down the current values before making any changes. Once new values are saved, the old values cannot be recalled.

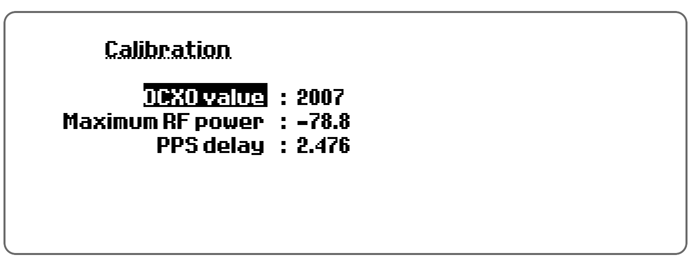


Figure 3-43: User Calibration view

During the calibration, the unit generates an unmodulated signal at full power. Maximum RF power is measured by a spectrum analyzer connected to the RF output of your GSG unit.

The **OCXO DAC** value is adjusted according to the frequency measured by the GSG unit from the 10 MHz output at the back panel. Using a frequency counter, adjust the OCXO value until the GSG shows 10 MHz.

The **PPS delay** is essentially an “equipment delay” of the generated signal. To measure it properly, you need to measure the difference between the GSG's **PPS out** (Trigger out) and the **PPS out** of a trusted GPS timing receiver. The value is always positive, and is set in microseconds. Three digits can be given, enabling nanosecond resolution. The allowed range is [0.000-4.000] microseconds.

Note that if you try to measure this delay, remember to take into account the GPS time to UTC time offset set in the scenario you use. Timing receivers typically output the UTC synchronized PPS signal.

After the calibration is complete, the new values can be saved by pressing exit or menu. Press cancel to discard the values and keep the previous calibration settings.

CHAPTER 4

Frequent Tasks

This Chapter includes several tasks that GSG users frequently perform.

This list is constantly being updated. Should you miss a task that is currently not included in this list, please let us know: techpubs@orolia.com. Thank you.

The following topics are included in this Chapter:

4.1 Working with Scenarios

The tasks described here are frequently performed in the context of scenario execution and configuration.

4.1.1 Scenario Start/Stop/Hold/Arm

See under: ""Start" Menu" on page 33.

4.1.2 Running a Scenario

During scenario execution, you can ...

- Press view to display up to 6 different **views** to monitor the execution of your test scenario (see "Scenario Execution Views" on page 34).
- Press menu to display the scenario configuration (grayed out, because editing is not permitted during scenario execution).
- Press the [.] / hold key to pause/resume moving along the trajectory. When the trajectory is paused, the HOLD symbol is displayed in the corner of the screen, the speed is 0.0 m/s, but the simulation clock continues to run.
- Press the arrow keys to change **all** power levels (for more power adjustment options, see "Setting Transmit Power" on page 113.)
- Press ± / format...
  - ...in **View 1**, to change the **coordinate format** between three geodetic and one geocentric formats, i.e., Lat, Lon, Alt will be shown either in format `DD MM.mmmm`, `DD MM SS.ss`, `DD.dddd` or `X, Y, Z`.
  - ...when **dBm** is highlighted, to toggle between frequencies (L1 ...ALL) and their power levels
- Press N/S to show the **Transmit Power** menu, and enable/disable/adjust noise settings.
- Press E/W to adjust the units displayed for **Altitude** and **Speed** (m/m/s > ft/kn > ft/mph).

> **Note:** The scenario will continue to run in the background, even if you view a display other than the "Scenario Execution Views" on page 34.

> **Note:** When you press exit to leave a menu, its settings will be taken into use immediately, and all band- or satellite-specific offsets are discarded.

4.1.3 Holding a Scenario

Holding a scenario means to temporarily prevent your GNSS receiver from continuing to move along its scenario trajectory (i.e., halting the trajectory), while the simulation continues to run otherwise (time continues to elapse). This can be done manually, by pressing the [.] / hold key, or by using the SCPI command **SCENario:CONTrol** , see "SOURce:SCENario:CONTrol" on page 263.

The display will show the **HOLD** icon in the upper right corner.

> **Note:** Holding a scenario is not the same as arming a scenario (see "Scenario Start Variations" on page 33).

A typical use case for holding a scenario would be to simulate a red traffic light.

4.1.4 Configuring a Scenario

Prior to configuring a scenario, you have to select it: In the **Main Menu**, highlight **Select**, scroll through the list of scenarios, and press enter (for more information, see ""Select" Menu" on page 38).

For a list of scenarios pre-installed on a GSG 6, see "Pre-Installed Scenarios" on page 188 (the scenarios pre-installed on your unit may be different, depending on your GSG model.)

Once a scenario has been selected, a number of Views will guide you through the list of parameters configurable with the chosen scenario.

> **Note:** For a list of all configurable scenario parameters, see ""Select" Menu" on page 38.

First Configuration View

In the first Scenario Configuration View, basic information like **position**, **date** and simulation **duration** are provided:

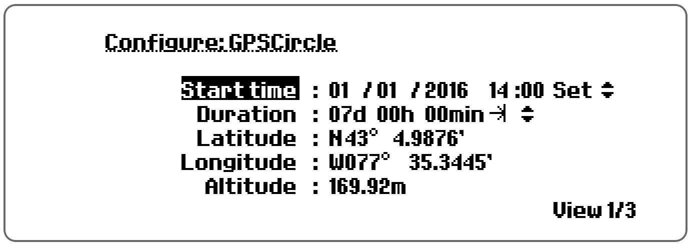


Figure 4-1: Scenario Configuration View 1/3

To **navigate between fields**, first press the UP/DOWN arrow keys to select the field label. Then press enter or the RIGHT arrow key, to begin editing the values. To move to the next field on the same line, press enter, or the RIGHT arrow key. To move to the previous field on the same line, press the LEFT arrow key.

To proceed to the **next View**, press view.

To **finish editing**, press start, exit, or cancel. (With start and exit, you will be given the choice to save the scenario under a different name.)

Second Configuration View

The second Scenario Configuration View allows you to configure the **Trajectory**, **Ephemeris** data, **Event data** and **Leap Second** simulation (**LS**).

Also accessible from this View are:

- The **Antenna** submenu, which allows the configuration of the **Antenna model**, **Lever arm**, and **Elevation mask**.
- The **Advanced** submenu, which provides access to **Multipath** and **Interference signals**. Also, the **Base station** (RTK option) can be turned on, and **output messages** defined. **Environment models** can be changed to 'set' which allows the selection of **Environment** and **Vehicle** models (created with the third-party tool SketchUp (a Technical Note about Vehicle Modeling using Sketchup is available upon request).
- The **Atmospheric** model submenu.

Note: Some of the functionality shown is optional. For more information on GSG Options, see "GSG Series Model Variants and Options" on page 203.

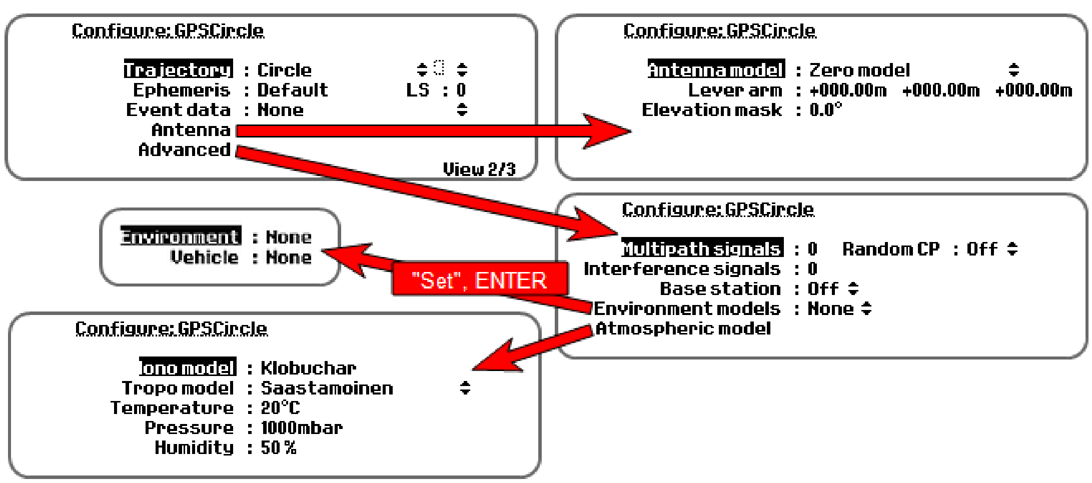


Figure 4-2: Scenario Configuration View 2/3

Third Configuration View

The third Scenario Configuration View allows you to configure the satellites to be simulated.

> **Note:** Some of the functionality shown is optional and may be grayed out. For more information on GSG Options, see "GSG Series Model Variants and Options" on page 203.

For each satellite constellation your GSG unit can simulate (e.g., GPS), you can:

- In the Satellites View (see illustration below), set the maximum number of satellites to be simulated (using the UP/DOWN arrow keys). Or, use the **Auto** setting, which lets the GSG simulator automatically select the highest number of satellites available for the number of channels supported by your GSG unit.
  In the Satellites View, you can also configure the number of **SBAS** satellites (see "SBAS Satellites" on page 82).
- In the Signal Type View, select the signal types to be simulated for the highlighted constellation (e.g., "L1CA"), and enable (pseudo-P(Y)) encryption (if available).

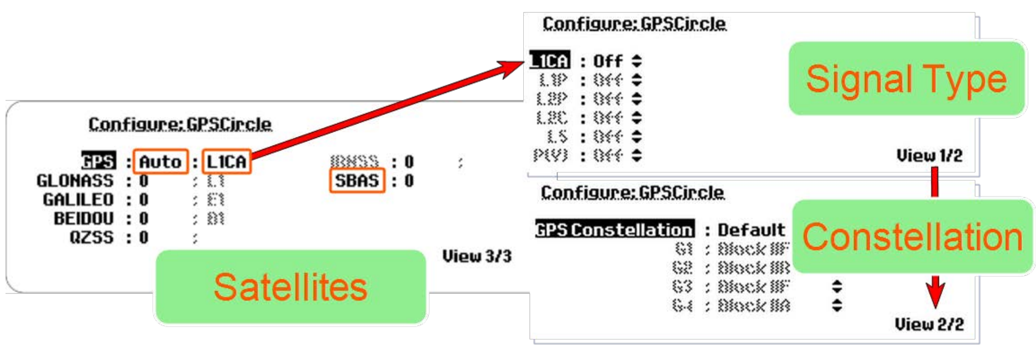


Figure 4-3: Scenario Configuration View 3/3

- In the Signal Type View, press the view key to access the Constellation View, which allows you to specify the Blocks (or the "vintage") of satellites simulated. For more information about this subject, see "Default Scenario Satellites" on page 190.

  The **Default** setting in the first row of Constellation View, **GPS Constellation** (or **GLONASS Constellation**, respectively) will simulate the satellite constellations as they existed on April 22nd, 2015.

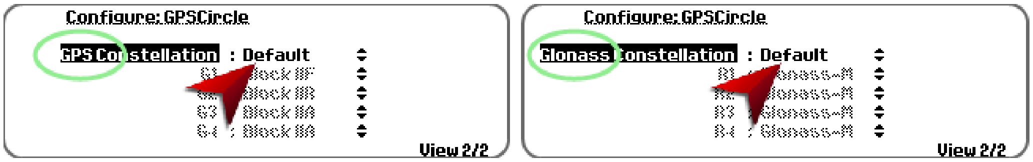


Figure 4-4: Types of GPS (Glonass) satellites simulated

Note: For a list of all configurable scenario parameters, see ""Select" Menu" on page 38.

Note: For the different options on how to start a scenario, see "Scenario Start Variations" on page 33.

4.2 Locking/Unlocking the Keyboard

The keyboard locking functionality prevents any unwanted modifications from being made.

When the keyboard lock is engaged, it is not possible to change parameters, or edit scenario execution via the front panel. It is, however, possible to view scenario configuration and observe scenario execution, using the view key, and toggle between the position coordinates, using the format key.

To **engage** the keyboard lock:

» In any of the GSG-5/6's menus or execution views, using the numeric keys on the front panel, key in the keyboard lock code, and confirm. The default keyboard lock code is "`1122`".

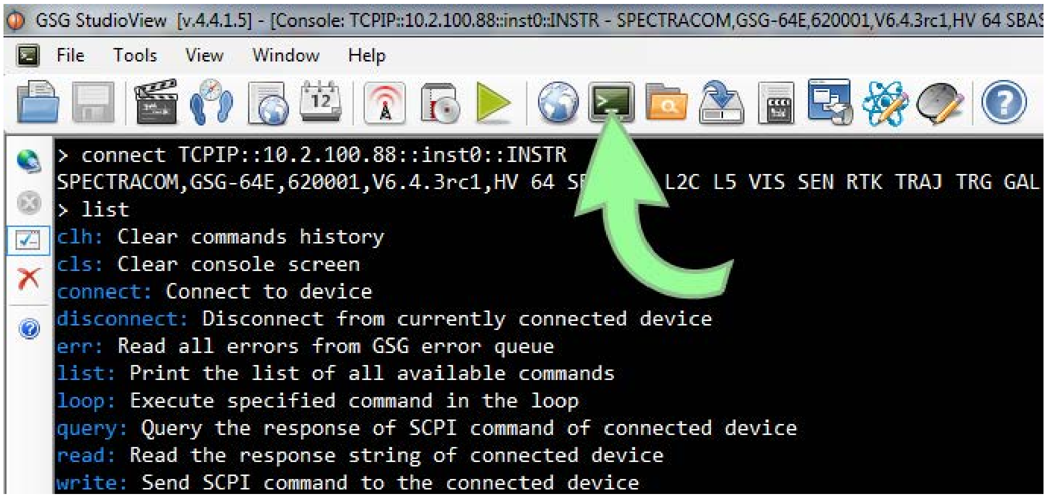
To **disengage** the lock:

» On the front panel, press any key other than view or format, and enter the current lock code. When all digits have been entered, navigate the DONE in the lower right-hand corner, and press enter.

Note: The keyboard lock can also be engaged/disengaged via a SCPI command, see "SOURce:KEYLOCK:STATus" on page 317 for details.

To **change** the keyboard lock *code*, use the `KEYLOCK:PASSWord` SCPI command:

» Open the , and then the command line interpreter (CLI) by clicking the MONITOR icon:



» Ensure that the CLI is connected to your GSG unit (see "Using the CLI" on page 116).

» Enter the following command: `write SOURce:KEYLOCK:PASSWord [wxyz]`

» The keyboard lock code `[wxyz]` can be changed at any time. The user-issued lock code must be 4-8 digits in length, and contain only numerical characters. The default keyboard lock code is "`1122`".

4.3 Setting Transmit Power

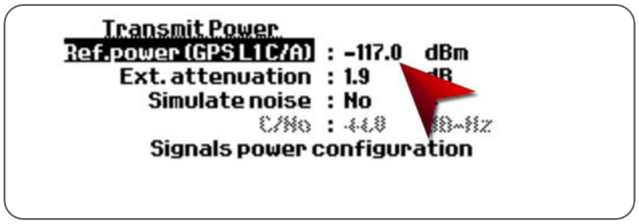
There are three different ways to alter the **Transmit Power** (also referred to as signal level):

1. While configuring a scenario:

   a. in Studioview e.g., by using the **Events** editor, or

   b. via the GSG menu **Options** > **Transmit Power**:

      » Navigate to the **Transmit Power** menu, then press the RIGHT arrow key to highlight the current value, then the UP/DOWN arrows or the numeric keys to adjust the value, and then enter to confirm.



This sets the **Reference Power** which is used to control the absolute power level of the GPS L1 C/A signal. You can also adjust the default relative power **offset** for each individual signal type other than GPS L1 C/A by selecting **Signals power configuration**. (For more information, see "Adjusting Transmit Power" on page 88.)

2. While the scenario is running — by pressing the N/S key:

   » In Scenario Execution **Views 2 to 5/x** (see "Scenario Execution Views" on page 34) highlight **dBm** for all satellites, or press the LEFT/RIGHT arrow keys to highlight individual satellites. Then press the N/S key to adjust the scenario power.

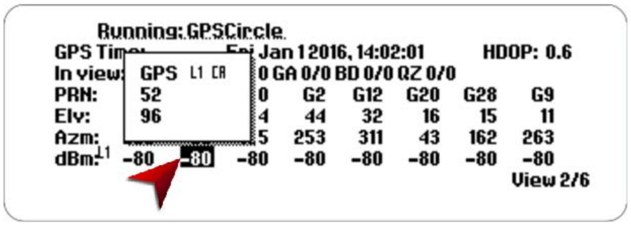
   > **Note:** Pressing the ± (format) key while dBm is highlighted changes the frequency band to be adjusted: L1 > L2 > L5 > ALL.

   While this option will also open the **Transmit Power** menu (as under **1.** above), it is used to adjust only the Transmit Power level for the current scenario (running now, or in the future). All other scenarios will continue to use the default value, or the value you set under **1**.

   Note that the adjusted power level will also apply to any new satellites coming into view later during any execution of this scenario.

3. While the scenario is running — by pressing the press the UP/DOWN arrow keys:

   » In the Scenario Execution **Views 2 to 5/x** (see "Scenario Execution Views" on page 34) highlight **dBm** for all satellites, or press the LEFT/RIGHT arrow keys to highlight individual satellites. Then press the UP/DOWN arrow keys to adjust the **Transmit Power** level, or enter a new value by using the numeric keys.



> **Note:** Pressing the ± (format) key while dBm is highlighted changes the frequency band to be adjusted: L1 > L2 > L5 > ALL.

Contrary to option **2.**, this will only adjust power for the selected satellites in view, not for new satellites coming into view later.

> **Note:** If a value is not accepted, it is likely out of spec, see "Transmit Power" on page 86.

4.4 Accessing the GSG Web Interface

To connect to the "The GSG Web UI" on page 176, follow these steps:

1. Determine the IP address of the GSG unit you want to connect to, by navigating to **Options** > **Interface and Reference**. The IP address will be listed under the **Network** menu item.
2. Open a Web browser (such as Mozilla Firefox or Internet Explorer), and enter the IP address into the address bar.
3. Once connected, the browser will display a graphical representation of the front panel of your GSG unit. You can click the buttons to perform operations as you would if you were physically doing so from the front panel of the unit. The functionality of the buttons and options is detailed in the Section "Front Panel" on page 28. The only exception is the **Power** button, which restarts the unit (instead of powering it OFF).
   The Primary Navigation Menu on top of the Web UI provides access to the following menu items:
   - **GSG FILES**: Provides access to all scenario configuration files and log files of your GSG unit. The files can be viewed and downloaded with a Web browser.
     - To **upload** a file, click **Choose Files** to browse directories, e.g., on a connected PC. Multiple files can be uploaded at one time, provided that

the combined size of the files does not exceed 10 MB.

- Then click **Upload**. If the upload is successful, the directory will be refreshed, otherwise a status page will list the files that could not be uploaded, and the reason why the upload failed.
- Please note the following file type requirements:
antennaModels: *.ant
calibration: *.cal
events: *.even
scenarios: *.scen
trajectories: *.traj or *.nmea.
File types of uploaded observations and navigationData will not be verified.

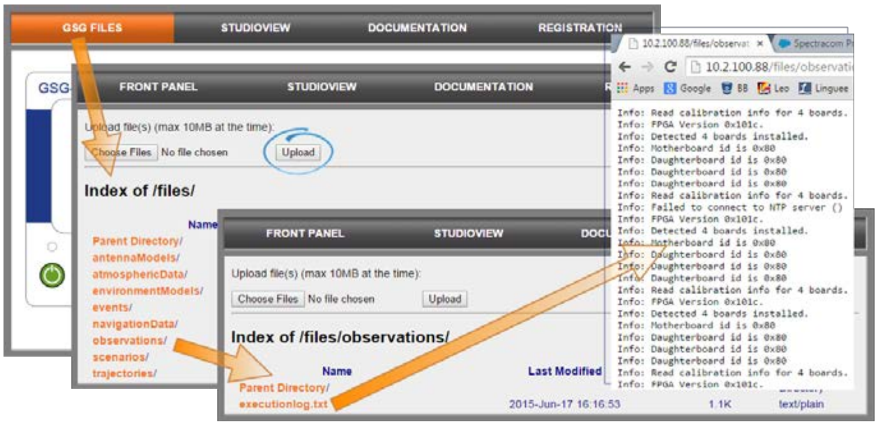
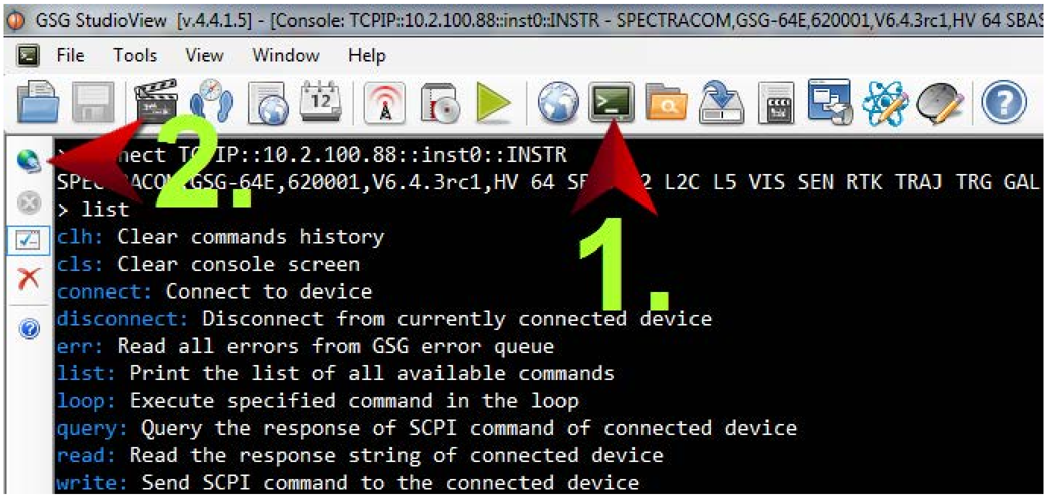


Figure 4-5: Example GSG Web UI, showing a logged GPS almanac file

- **STUDIOVIEW**: Opens the Orolia website/StudioView web page: https://www.orolia.com/products-services/gnss-simulation/gpsgnss-simulators
- **DOCUMENTATION**: Opens the Orolia website/StudioView web page: https://www.orolia.com/support/spectracom/gsg-series-gps-simulators-support
- **REGISTRATION**: http://register.spectracom.com/.

4.5 Using the CLI

1. Open the , and then the Command-Line Interpreter (CLI) by clicking the MONITOR icon:



2. Click the Globe icon. The **Connections** window will open.
3. Click the green PLUS icon in the top-left corner, and enter the name of the new connection, and its IP address (which can be found under the GSG menu **Options** > **Interface and Reference**. The IP address will be listed under the **Network** menu item.
4. Test the connection, and click OK.

The connection between the CLI and your GSG unit is now established, and you can start communicating by sending SCPI commands.

4.6 Performing a Receiver Cold Start

A **Warm Start** is performed by most GNSS receivers after a power reset. The data (ephemeris, almanac) is remembered to aid in obtaining the satellites during next power-up.

To perform a **Cold Start**, initiate a cold start command to the receiver, or clear its memory by using other means intended for this purpose. Resetting the power does not perform a cold start by design.

> **Note:** ALWAYS force a Cold Start, or a full reset of a receiver after it had been used with generated signals!

Without a Cold Start:

- The receiver will reject the generated signals as invalid.
- The receiver may not find the generated satellites.
- The receiver may fail to navigate or behave poorly.

4.7 Creating a One-Line Trajectory

As the GSG unit uses the heading and speed information of the RMC sentences, only one **NMEA** sentence is actually required to describe a simple, continuous movement.

For example, the following one-line trajectory specifies a continuous north-bound trajectory (as the heading field is set to 0.0 degrees) at a speed of 77 knots:

```
$GPRMC,111150,A,6000.000,N,0100.000,E,77.000,0.0,140715,0.9,W,A*03
```

RMC	Recommended Minimum Sentence C
111150	Fix taken at 11:11:50 UTC
A	Status [A = Active, V = Void]
6000.0000,N	Latitude: 60 deg 00.000' N
0100.0000,E	Longitude: 01 deg 00.000' E
77.000	Speed over ground: 77 knots
0.0	Heading: 0.0 degrees true
140715	Date: 14-July-2015
0.9,W	Magnetic Variation: 0.9 deg West
A	Positioning system mode indicator: [A = Autonomous]
*03	Checksum data, always begins with *

One-line trajectories like this can be easily be made by manually creating the desired **NMEA** files: The example above can be taken as a baseline, then edit speed and/or heading fields as required.

To allow for testing the sentence's validity, the last 2 digits contain a checksum of the data (XOR of all bytes between $ and * symbols) – this checksum must be correct and can be calculated with e.g., this online tool: www.hhhh.org/wiml/proj/nmeaxor.html.

Note that the **NMEA** messages, including the checksums, are case sensitive and should be given in UPPERCASE even if the GSG unit (firmware version 3.00 and above) accepts messages in lower case.

To find out more about **NMEA**, and purchase a copy of the **NMEA 0183** Standard, visit www.nmea.org.

To learn more about trajectories, see: "Trajectories" on page 43.

4.8 Leap Second Configuration

A leap second can occur on two dates, December 31 or June 30. It is announced approx. 6 months in advance. The GPS almanac changes to 'announce' the leap second, and GPS receivers must correctly implement the leap second at the proper time.

To configure a leap second for a scenario, **Select** the scenario, then navigate to the configuration **View 2/3**:

Figure 4-6: Leap Second field

Alternatively, the leap second can be configured in StudioView.

The leap second field can be set to -1, 0 or 1, and indicates a future change in the leap second value. While Δt_{LS} is set automatically based on information in the used ephemeris data, the value given in the leap second field will impact values related to L_{SF} (Leap Seconds Future).

When the leap second is set to a value other than zero

The following values will be used:

$\Delta t_{LSF} = \Delta t_{LS}$ + value given in the leap second field

WN_{LSF} = The GPS week number (8-bit representation) of the week holding the 30th of June, or 31st of December, whichever comes first with respect to the scenario start time.

DN = Day number of the date described above.

When the leap second is set to zero

The following values will be used:

$\Delta t_{LSF} = \Delta t_{LS}$,

$WN_{LSF} = WN_{LS} - 1$, and

DN = 1

Note that downloaded and default navigation data files do not contain any LSF information (RINEX v2.1). Therefore it is still necessary to set the L_{SF} when a leap second change will occur, in order to ensure correct behavior. The default UTC/GPS offset—as of 2016—is set to 17 seconds (it will be 18 seconds in 2017).

The four leap second almanac variables are:

WN_{LSF}: Week number when the leap second becomes effective

DN: Day number when the leap second becomes effective

$\Delta \mathbf{t}_{LS}$: Current or past leap second value

$\Delta \mathbf{t}_{LSF}$: Current or future leap second value.

4.9 Studioview Tasks

4.9.1 What Is StudioView?

GSG StudioView™ software for Windows® enables you to create and edit scenarios, and perform file management tasks with Orolia's GSG series GPS/GNSS simulators.

While GSG simulators are capable of configuring and running scenarios without the need for an external computer, StudioView Windows® software offers several additional benefits:

- Create, edit and organize all scenario parameters including dynamic events
- Create, edit and visualize trajectories with mapping tools
- Convert trajectories from CSV, KML, KMZ and GPX files to the required NMEA format
- Create scenario files (including events and trajectories) without the need to be connected to a simulator.

4.9.1.1 StudioView Tasks

This Chapter describes some of the tasks you will likely perform with GSG StudioView:

- "Installing StudioView" on page 122
- "Connecting StudioView to GSG" on page 122
- "Updating the GSG Firmware via StudioView" on page 124
- "Accessing GSG Remotely via StudioView" on page 128
- "Creating a Trajectory in StudioView" on page 130
- "Creating an RSG Trajectory with StudioView" on page 140
- "Configuring a Scenario" on page 148
- "Playing RSG Scenarios in StudioView" on page 148
- "Record and Playback" on page 163

4.9.1.2 StudioView Functionality Overview

File Management and Control

StudioView's File Manager allows you to upload or download scenario files to or from your GSG unit: Connect your Windows® PC running StudioView to the GSG unit via its network, USB, or GPIB interface. Then drag & drop the files, or use the copy/delete/rename hotkeys on either the PC or the GSG unit.

The **Uploader** is designed to batch-upload scenarios, or firmware files – if needed, to several GSG units simultaneously.

Also included is the **Console** function which is used to send SCPI commands to a connected GSG unit, and to view the response.

> **Note:** GSG StudioView requires a license to activate all features after the 30-day trial period. After the trial period, all features are locked out except for the Uploader. The Uploader is used to perform firmware updates or upload scenario files to the GSG.

Trajectory Building and Dynamic Event Management

A key feature of GSG StudioView is the ability to create and modify the simulation of a moving receiver: So-called trajectories can be created with Google Maps® or imported from other devices and applications such as Google Earth®.

StudioView converts a list of waypoints from a CSV file, or waypoints, routes and tracks from a GPX file (GPS exchange format) into the NMEA format, as required by the unit. Google KML and KMZ files can be opened, edited, and saved. Waypoints can be edited individually or in batches.

When converting a trajectory file to the NMEA format, you can specify altitude and speed at each waypoint or use a constant value for the trajectory. Trajectory waypoints can be interpolated at a set interval from 100 ms to 1 hour.

Pre-defined events occur along the trajectory: Whether a vehicle stops at a red traffic light, or a building temporarily blocks the line of sight to a satellite – in StudioView you can create and edit different types of events that will make any test scenario more realistic and test the capabilities of a GNSS receiver even under difficult conditions.

Ephemeris and Almanac Data

Based on the scenario start time and duration, GSG StudioView software identifies and pre-downloads the relevant RINEX files from the official websites. Once a scenario is uploaded to the GSG simulator, no further downloads are required.

4.9.2 Installing StudioView

A free 30-day demo version of StudioView can be downloaded from the Orolia website, see the link below.

After the 30-day trial period you need to purchase a StudioView license to activate all features, with the exception of the **Uploader**, which does not require a license. The Uploader is used to perform firmware updates or upload scenario files to a GSG unit.

To install StudioView:

1. Download the software: https://files.spectracom.com/public-downloads/gsg-56-update-files-and-documentation
2. Locate the downloaded `.exe` file, and launch it. Follow the on-screen installation instructions.
3. Towards the end of the installation process, among other things you will be asked if you want to install the National Instruments VISA driver: This driver is required for Studioview to communicate with your GSG unit, so please check the box (unless you do not plan to connect your PC to a GSG unit).
4. Save the VISA runtime `.exe` file, and launch it to install the driver.

The topic "Connecting StudioView to GSG" below describes how to establish communication between StudioView and a GSG unit (or any other device on the network).

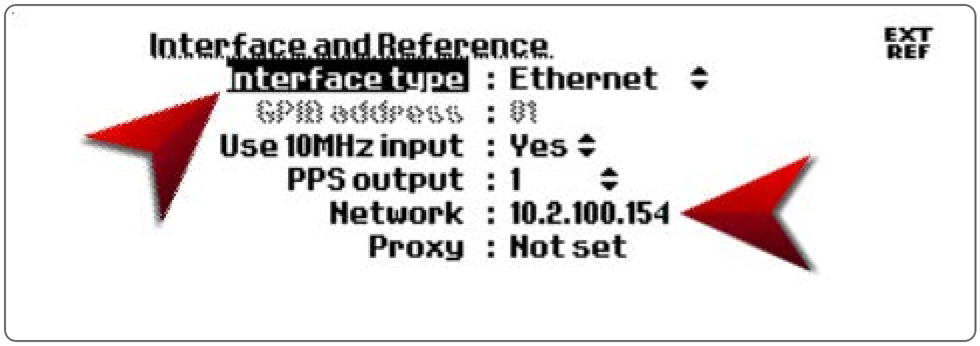
4.9.3 Connecting StudioView to GSG

StudioView needs to be connected to your GSG unit so that you can up-/download files, record data, or use the GSG web interface.

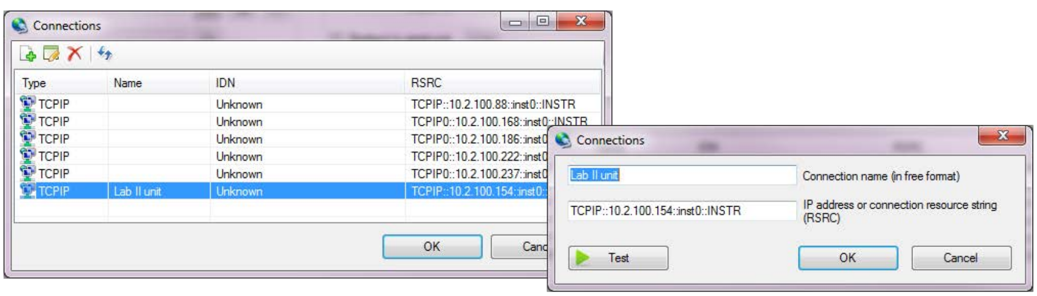
On the GSG side, you can use the Ethernet port, USB port, or GPIB interface to establish the hardware connection. On the StudioView side, the **Connections Manager** tool is used to detect and configure the connection.

Once you have connected a cable to your GSG unit, follow the procedure below to establish communication between StudioView and the GSG unit (Ethernet is used as an example).

1. On the GSG unit, navigate to **Options > Interface and Reference**:



2. Select the **Interface type** that matches your hardware configuration, i.e. the connection between the GSG unit and the StudioView computer: TCP/IP, USB, or GPIB (note that SCPI-Raw does not work with StudioView).
3. In the same menu dialog, take note of the **Network address** displayed.
4. Launch StudioView on a PC. From the main menu, or the TOOLS dropdown menu, select the tool you would like to use e.g., the GSG web interface, the Uploader, or the Data recorder.
5. In the tool window, next to the **Address** field, click the Connections button. The **Connections Manager** window will be displayed, showing a list of previously created connections:



If you can see the connection pointing to your GSG unit, click it and then click OK. To refresh the list and search for more GPIB or USB devices, press the **Refresh** button. Otherwise click the Add icon to add a new connection:

6. Enter the IP address that was displayed under Step 3. above, and click to validate the connection. Click OK to close validation dialog. If the connection was successful, click OK to add this connection.
7. Highlight your connection (blue background) and click OK. The StudioView tool window should now show the connection to your GSG unit. (Note: The actual

connection (e.g., for the Data Recorder) will not be established until you press the **Start** button.)

4.9.4 Updating the GSG Firmware via StudioView

GSG StudioView is used to perform a firmware update. It is recommended that you always use the latest available GSG **firmware**, and that you use the latest available **StudioView** version to update the GSG firmware. For the latest StudioView version, and GSG firmware version, see https://www.orolia.com/support/spectracom/gsg- series- gps-simulators-support

1. To determine which firmware version is currently installed on your GSG unit, navigate to **Options** > **Show system information**.
2. To determine the latest GSG firmware version, and download it to your Personal Computer, see the link above. This link also points to firmware update instructions.

   Please note the following:

   - Prior to updating the firmware, please read the corresponding **release notes** containing further update instructions. If the firmware version on your GSG unit is V4.05 or lower, several updates must be performed in the correct sequence. It is possible, albeit not recommended, to back out a firmware update by installing the previous version on top. To obtain the firmware updates 2.03, 2.04, and 4.07 please contact Orolia Support at https://www.orolia.com/support/spectracom/request-service-product-support.
   - In order for the instrument to communicate with the PC Software the NI VISA runtime software is required. The StudioView installation wizard will notify you, but if necessary, the NI VISA runtime software can also be downloaded directly from the National Instruments website:

     http://joule.ni.com/nidu/cds/view/p/id/3342/lang/en.
   - Once a firmware upload is initiated from StudioView, any running scenario will be stopped, and the upgrade process status will be displayed instead. First, the progress of the file transfer is displayed. After the file transfer is complete, the actual upgrade operation is made. Finally, the GSG-5/6 unit will reboot with the new firmware installed.
   - The Windows Personal Computer onto which you will download the new firmware must have **GSG StudioView** installed on it, and must be connected to your GSG unit e.g., via Ethernet.
3. Once you have prepared the firmware upgrade as outlined above, proceed to the topic "Uploading StudioView Files" on the facing page.

4.9.5 Uploading StudioView Files

GSG StudioView's **Uploader** tool is used to upload scenario files and firmware updates from your StudioView PC to a connected GSG unit, and vice versa. (For more information on firmware upgrades, see "Updating the GSG Firmware via StudioView" on the previous page.)

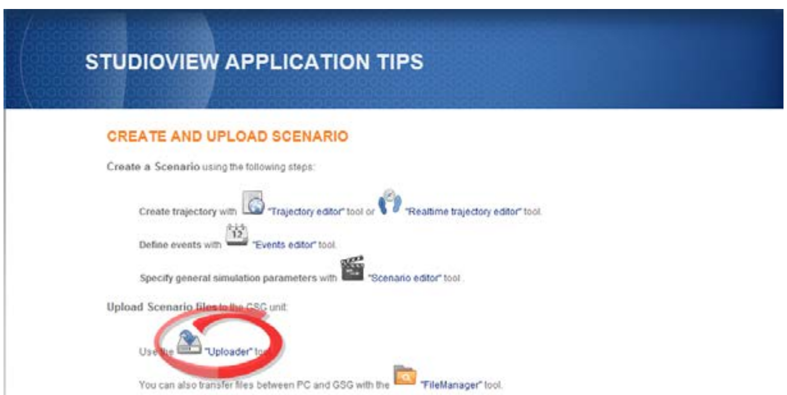
Note that for uploading scenarios the **Uploader** is the preferred tool over the **File Manager** (see "Transferring Files With StudioView" on page 127), since the **Uploader** automatically uploads all files belonging to a scenario. The **File Manager** is meant to transfer individual files e.g., a standalone trajectory file.

4.9.5.1 Using the StudioView Uploader for the First Time

In order for GSG to communicate with StudioView, the **NI VISA Run-Time** engine is required, which can be downloaded directly from the National Instruments website.

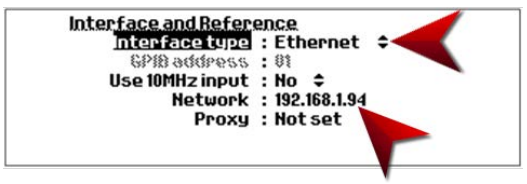
1. Once **VISA Run-Time** and **GSG StudioView** are installed, start **GSG StudioView.**

   From the **Application Tips** screen, or from the toolbar under **Tools**, select **Uploader**.



2. On the **Uploader** screen, click **Select devices for uploading**.
3. The **Connections** window will open, displaying all TCP/IP, GPIB and USB connections. Click the **PLUS** icon to add an ETH device, or the **Refresh** icon to search for more devices and update the list.

To obtain your GSG's IP address or change the interface type, select **Interface and Reference** from the GSG **Options** menu:



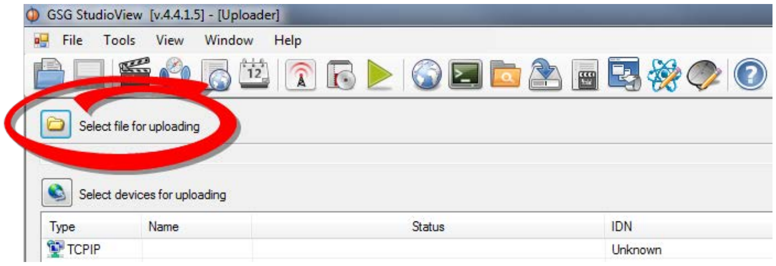
Note: This screen may vary, based on your installed firmware version.

4. Click the **Test** button to verify the connection:

Uploading Firmware

1. In order to update the firmware onto your GSG unit, in the StudioView **Uploader**, click the **Open Folder** button (next to **Select file for uploading**), and navigate to the downloaded firmware file on your PC.

2. Click the **Start** button to start the transfer. The unit will first transfer the file, and then update the firmware.



Uploading a Scenario

1. In order to upload a scenario, first ensure that the scenario file (`.scen` file) and any trajectory, event, or navigation file associated with the scenario are stored in the StudioView repository. By default this location is:

   `C:\Users\username\Documents\Spectracom\GSG StudioView\Repository`

   > **Note:** The username location may depend on your version of the Windows operating system you are using.

2. In the StudioView **Uploader** window, click **Select file for uploading**, and navigate to the scenario file in the repository.
3. Click the **Start** button to start the upload. The software will automatically upload the scenario file as well as any trajectory, event, or navigation file associated with that scenario, and place the files in the proper locations in the GSG.

4.9.6 Transferring Files With StudioView

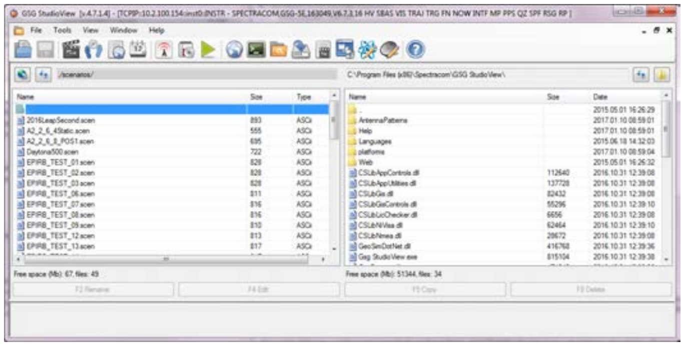
If there is a need to transfer individual files from your StudioView PC to a connected GSG or vice versa, StudioView's **File Manager** is the tool of choice.

> **Note:** Note that for uploading scenarios the Uploader (see "Uploading StudioView Files" on page 125) is the preferred tool over the File Manager, since the Uploader automatically uploads all files belonging to a scenario.

Transferring a File using the StudioView File Manager

1. Open the **File Manager** tool by navigating to **Tools > File Manager**, or click .

2. Establish a connection to the GSG unit by clicking . (For details, see "Connecting StudioView to GSG" on page 122.)
3. Once a connection has been established, you should see GSG's file/folder tree in the left window, and StudioView's file list in the right window:



4. Double-click on any folder to open it. Click on the top folder .. to navigate to the next higher folder. Highlight any file in order to **F5** Copy or **F8** Delete it.

Downloading a File from a GSG Unit via the Web Interface

In order to download a file from a connected GSG unit to a destination of your choice on the StudioView PC, instead of using the **File Manager**, you can also use the **GSG Web Interface**:

1. In StudioView, open the **GSG Web Interface** by clicking the Globe button (or – if not using Studioview – see "Accessing the GSG Web Interface" on page 115).
2. Select GSG FILES in the top-left corner. Then select the observations/ directory. Click on any file to display its contents. Click the green arrow in the top right corner to navigate back to the file listing.
3. Download a file by right-clicking on it and selecting **Save target as ...**

4.9.7 Accessing GSG Remotely via StudioView

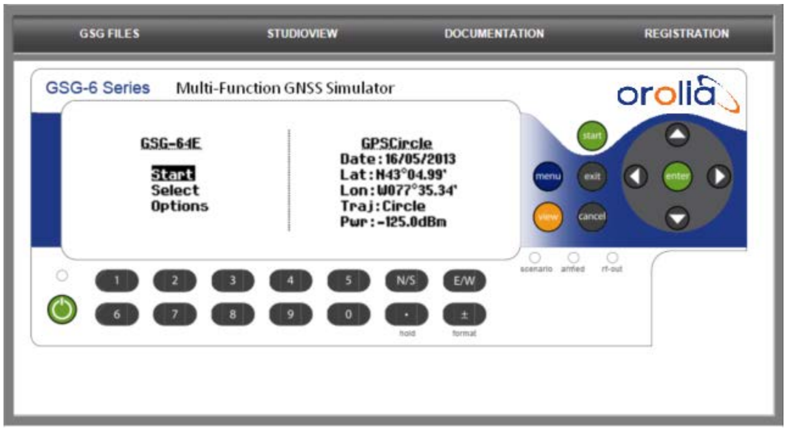
If a PC has been connected to the GSG unit via one of the communication interfaces (Ethernet, USB, or GPIB, see "Rear Panel" on page 31), the GSG unit can be controlled remotely, either by using its web interface, or by submitting commands via StudioView's console tool:

Access via the Web User Interface

The StudioView Web User Interface ("Web UI") tool provides web access to the front panel of your GSG unit.

To open the Web UI in StudioView:

1. Navigate to **Tools > Web interface**, or click the icon.
2. Click to open the Connections Manager tool (for details, see "Connecting StudioView to GSG" on page 122.)
3. After setting up the connection, a visual representation of the front panel will appear:



You can control the unit as if physically pressing the buttons on the unit.

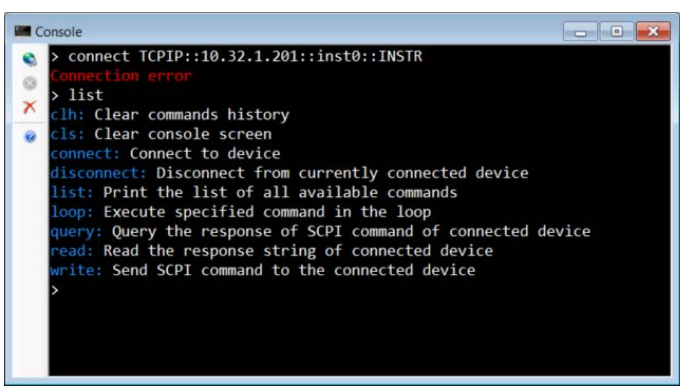
To access the files stored on the GSG unit, click the **GSG FILES** button in the top left corner.

Access via the Console Tool

The StudioView Console allows you to communicate with the GSG unit via SCPI protocol.

To open the StudioView Console tool:

1. Click or navigate to **Tools > Console**. The Console window will display:



2. Click to open the Connections Manager tool (for details, see "Connecting StudioView to GSG" on page 122.)
3. The following generic commands are supported (to display this list in the console window, type `list`.):
   - `clh` – clear commands history
   - `cls` – clear console screen
   - `connect` <resource[1]> – connect to a specific GSG unit
   - `disconnect` – disconnect from currently connected device
   - `list` – print the list of available commands
   - `loop` <interval> <SCPI command> – execute specific command in a loop with specific interval in milliseconds between repetition
   - `query` <SCPI command> – send SCPI command to connected device and read response string of connected device
   - `read` – read the response string of connected device
   - `write` <SCPI command> – send SCPI command to the connected device

GSG has en error queue that can be checked automatically after every execution of a command. To enable or disable this function, click .

To clear the console screen, click .

4.9.8 Creating a Trajectory in StudioView

What is a Trajectory?

In the context of GNSS testing, a trajectory is the predefined path a receiver is traveling during the execution of a scenario. It is the input to the scenario that defines how the virtual vehicle will move in up to 6 degrees of freedom during the test (X, Y, Z; pitch, roll, yaw).

While the definition of the vehicle path itself is an important constituent of any scenario, the GNSS signal reception at any given time during scenario execution is of equal importance, since the reception will be affected by these trajectory-dependent factors:

- The **environment** along the trajectory changes: Infrastructure blockages such as tall buildings ("urban canyons") or other topographic characteristics will cause the signal reception to vary.

[1] A resource string (e.g., "TCPIP::10.32.1.203::inst0::INSTR") can point to a USB or GPIB connection. It is generated automatically when you add a new connection and its IP address becomes validated.

- The **attitude** of the vehicle in motion can affect signal reception due to on-vehicle blockages and antenna affects.
- The **acceleration/jerk** in the context of position and attitude can have a significant effect on signal reception.

Ultimately, the objective of any test is to determine how the receiver/the system under test responds when subjected to the above-mentioned control factors and noises.

Note: For additional information on trajectories, see "Trajectories" on page 43.

How to Obtain a Trajectory?

There are different ways to generate a trajectory:

a. Authentic NMEA data logged during a real-world drive along the trajectory route under live sky conditions (see "Record and Playback" on page 163)

b. StudioView's **Trajectory Editor**, which can be used to manually create a trajectory as a sequence of points on Google Maps.

c. **StudioView RSG Trajectory Editor**, which can be used to describe a motion-based trajectory (proprietary format) (Option-RSG is **NOT** required to use this editor)

d. **Real-time motion simulator** input; (Option-RSG **IS** required to input the data in realtime)

e. Several GSG models are shipped with pre-installed circle and 3GPP trajectories.

Realistic Trajectories

Trajectories generated by means other than (a.) should always realistically reflect the dynamic capabilities of the type of vehicle in motion e.g., car, aircraft, ship. To this end, Orolia recommends using 'smooth' methods to describe the movements, i.e. changes in acceleration, heading or altitude should be gradual, not sudden or 'hard'.

When using coordinates to describe a trajectory, the data must be provided in 10 Hz format and must not contain sudden changes in speed, direction or elevation; GNSS receivers generally are very sensitive to G-force and unrealistic movements will result in the receiver losing track of the signals.

Supported Trajectory Formats

GSG accepts trajectory data in the `.nmea` and `.traj` formats, while StudioView also accepts other formats, which can then be converted to NMEA for use in the GSG.

GSG supports the following trajectory-related workflows:

1. NMEA trajectory files, generated with the StudioView Trajectory Editor
2. NMEA trajectory files, as recorded by a receiver under the live sky, then imported into and converted by Studioview
3. `.traj` format, as generated with the RSG trajectory editor in StudioView (supports 6 DoF)
4. Real-time motion (6 DoF), generated while the simulation is running (Hardware-in-the-Loop testing) (requires RSG-option)
5. `.tle` (two-line element), as used for space vehicle simulation (see "Trajectory Two-Line Element Format (TLE)" on page 363).

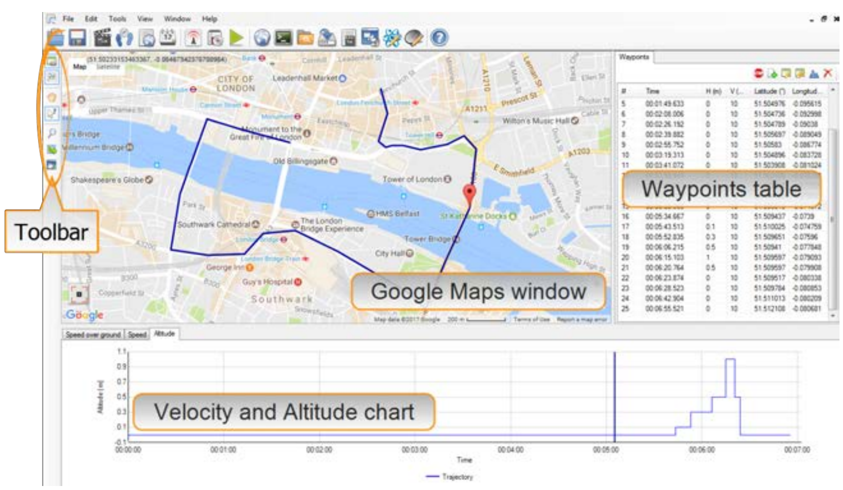
Workflow no. 1. is described below – generating a trajectory using StudioView's Trajectory Editor:

Using the Trajectory Editor for the First Time

To create or edit a trajectory, open the StudioView Trajectory Editor:

1. On a Windows PC, start StudioView. The StudioView Application Tips screen will be displayed.
2. Navigate either to the menu **Tools > Trajectory Editor** or, in the main toolbar click .

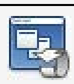
In its default view mode, the Trajectory Editor displays a **Google Maps** window, a **Way-points** table, and a Trajectory **Velocity and Altitude** chart:



The vertical toolbar on the left side of the screen provides access to the Trajectory Editor's functionality:

Note: A little blue box indicates that a feature is active (except "Export ..."). Some features cannot be active at the same time.

Table 4-1: The Trajectory Editor Toolbar

No.	Icon	Hover Tooltip	Usage
1.		Parameters panel	Display/hide the **Waypoints** table on the right side of the screen.
2.		Charts panel	Display/hide the **Speed and Altitude** chart.
3.		"Drag only" map mode	**Drag** (pan) in Google Maps™, while avoiding inadvertent insertion of way-points.
4.		Cursor coordinates	Display the **coordinates** of the cursor.
5.		"Search" panel	Access the Google Maps **Search** functionality (enter the name of a loc-ation, or WGS-84 coordinates).
6.		"Build route" panel	Opens a panel in Google Maps that is needed to build a route. (Consider this to be the Trajectory Editor's main tool.)
7.		Export ...	Open the StudioView **Trajectory Converter** to export a trajectory. For more information, see "Converting a Trajectory in StudioView" on page 136.

Building a Route by Creating Google Maps™ Waypoints

The main step towards creating a trajectory in StudioView is to build a **route** by creating waypoints in StudioView's Google Maps window.

While it is possible to create waypoints in the Google Maps window by left-clicking on dif-ferent points – thereby building a route – this route will *not* follow streets or make allow-ances for any topographic characteristics. Instead, use Google Maps to plot a realistic route by entering start point and end point, as you would with any GNSS navigation system (and as described below).

For airborne vehicle trajectories the RSG trajectory editor is more suitable, since its tra-jectories natively support 6 DoF, and since its trajectories are not tied to a specific geo-graphic location (so they can be played in different locations, if so required. For more information, see "Creating an RSG Trajectory with StudioView" on page 140).

To build a route:

1. In the StudioView Trajectory Editor, click the **Route Builder** icon . (See also "Using the Trajectory Editor for the First Time" on page 132.) The **Build route** panel will open:

2. Enter an address for the first point of the new route.
3. Click the + button to create a new waypoint or the x to delete an existing one.
4. Enter additional waypoints as needed.

> **Note:** To search for particular place on the map, but not set a waypoint (yet), click the Search icon. Note that any previously set waypoints will NOT be lost.

5. Once all the waypoints for the new route are set (you can add waypoints later), click the Route builder button in the lower right corner of the panel. The route will be built. The **Waypoints** table and the **Velocity and Altitude** chart will be populated.

Editing a Route

While a route built with the help of Google Maps would suffice to be used in a scenario, it is advisable to add or change some additional altitude and speed data, thus developing the trajectory further into a realistic trajectory. Also, you may want to edit individual waypoints, or add a stop.

> **Note:** You can also edit the route of an existing trajectory. To import an existing trajectory, see "Transferring Files With StudioView" on page 127. Open the trajectory by clicking File > Open. Make sure you have selected the correct file format (the default is "All supported files"), then locate the file you want to open, and click Open. The Trajectory Editor window will open automatically.

To edit an existing route:

In the StudioView Trajectory Editor **Waypoints** table you can:

» Add elevation to all waypoints by retrieving Google Maps altitude data (by default, all elevation is set to 0). Click the **Update Elevations** button.

» Double-click on any waypoint to **edit** its elevations, its coordinates, or the speed setting.

You may enter **geographic coordinates** in any of Degrees, Deg Min, Deg Min Sec format or ECEF coordinates. StudioView will recalculate the other formats automatically.

Note that altitude values are not MSL, but above the surface of the ellipsoid.

Table 4-2: Speed conversion table (Note: mph and knots are rounded down.)

m/s	km/h	mph	knots
2	17.2	4.5	3.9
5	18	11.2	9.7
10	36	22.4	19.4
15	54	33.6	29.2
20	72	44.7	38.9
25	90	55.9	48.6
30	108	67.1	58.3
35	126	78.3	68.0
40	144	89.5	77.8
50	180	111.8	97.2
60	216	134.2	116.6
70	252	156.6	136.1
80	288	179.0	155.5
90	324	201.3	175.0
100	360	223.7	194.4
200	720	447.4	388.8

Note: When changing speed settings, the time values in the Waypoints table will be updated automatically.

Note: As noted before, changes to speed, altitude and heading should be gradual and realistic for the type of vehicle simulated.

SHIFT-right click or CTRL-right click any group of waypoints to **batch-edit** their settings by clicking .

Add or delete waypoints by clicking the + icon, or the x icon, respectively. By default, StudioView will insert the waypoint at the end of list. However, you may manually set a **Point #**, which will add the waypoint to that place in the trajectory.

Select a waypoint in the Waypoints table, and then **drag** its Google Maps pin to relocate it.

Add a brief **stop** (e.g., to simulate a red stop light) by highlighting any waypoint and clicking the STOP button. Enter the stop duration, and the speed at which to continue after the stop.

Saving the Trajectory

To save the trajectory, click [save icon], or select **File > Save**. The default file format is `.nmea`.

4.9.9 Converting a Trajectory in StudioView

Trajectories can be captured in different formats, depending on how they have been generated and/or what their intended use is. GSG can read `.nmea` , `.traj`, or `.tle` files.

To learn more about ...

- `.nmea` trajectories, see "Creating a Trajectory in StudioView" on page 130.
- `.traj` files, see "Creating an RSG Trajectory with StudioView" on page 140.
- `.tle` files, see "Trajectory Two-Line Element Format (TLE)" on page 363.

The conversion of trajectories can become necessary e.g., if a trajectory had been created by ...

- ... recording it using a GNSS receiver
- ... using third-party software e.g., Google Earth
- ... manually entering it into an Excel spreadsheet
- ... using other means to generate trajectories.

However, it may also be required to change or add certain settings to an .nmea trajectory e.g., to smoothen, interpolate or equalize its data. This also can be accomplished with the Trajectory Editor, see under "Converting a Trajectory in StudioView" above.

StudioView's Trajectory Converter can convert the following input and output file formats:

- `.nmea`
- `.csv`
- `.gpx`
- `.kml`
- `.kmz`

Note: Note that the .csv format represents StudioView waypoints table from the Trajectory Editor and therefore has the same fields. Working with .csv files, StudioView assumes that for each waypoint the .csv file will have four values: Latitude, Longitude, Speed and Altitude.

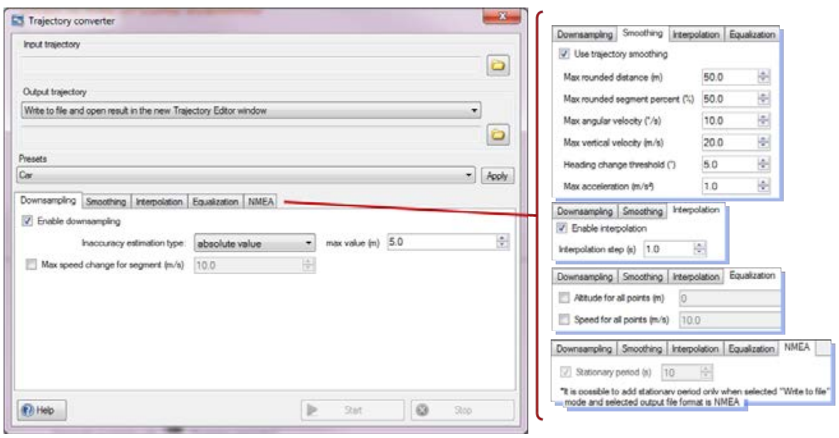
Realistic Trajectories

Trajectories should always realistically reflect the dynamic capabilities of the type of vehicle in motion e.g., car, aircraft, ship. To this end, Orolia recommends using 'smooth' methods to describe the movements, i.e. changes in acceleration, heading or altitude should be gradual, not sudden or 'hard'.

When using coordinates to describe a trajectory, the data must be provided in 10 Hz format and must not contain sudden changes in speed, direction or elevation; GNSS receivers generally are very sensitive to G-force and unrealistic movements will result in the receiver losing track of the signals.

Using the Trajectory Converter for the first Time

In StudioView, open the **Trajectory Converter** tool by clicking the icon, or navigate to **Tools > Trajectory converter**:



1. Select the **Input trajectory** you want to convert by clicking the file folder icon.
2. Decide what to do with the new trajectory and then select one of the 3 available options:

- » **Open result in the new Trajectory Editor window**
- » **Write to file and open result in the new Trajectory Editor window**
- » **Write to file.**

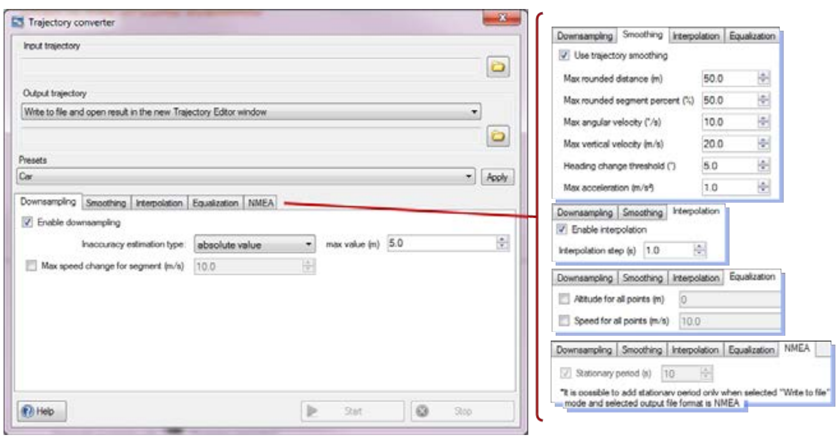
With the latter two options, click the lower file folder icon, and select a file location and file format .

3. Select a **Preset** (Car, Aircraft, Ship) to pre-populate the fields below. Click **Apply**.
4. Adjust the parameters as described under "Converting a Trajectory in StudioView" on page 136 and click Start to begin the conversion. Look out for possible error messages and follow the screen instructions to resolve any found issues.

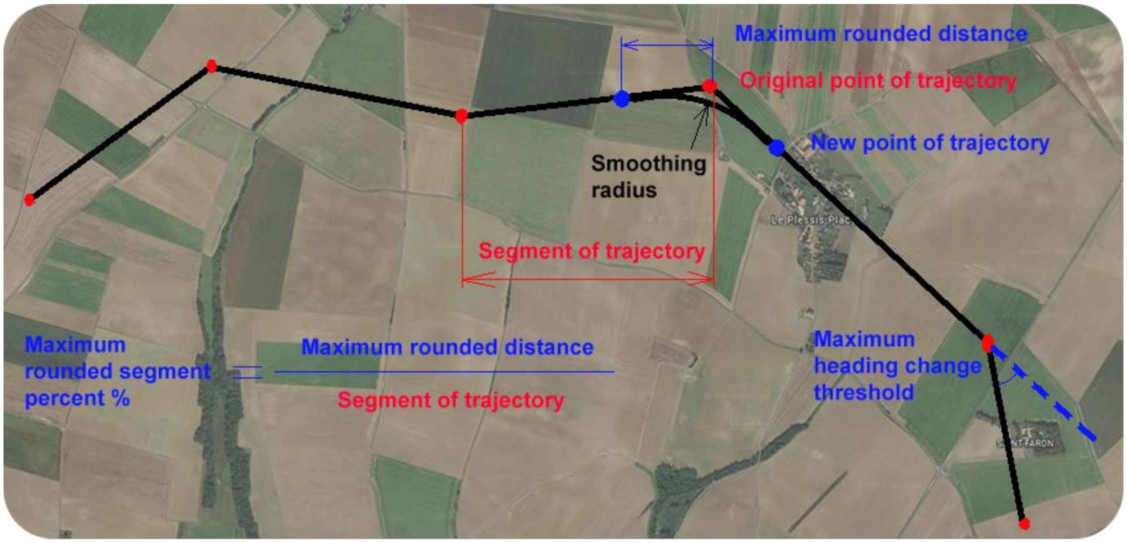
4.9.10 Improving a Trajectory

StudioView Trajectory Converter (see "Converting a Trajectory in StudioView" on page 136) has several adjustable parameters that can be used to enhance a trajectory, thus making it more suitable for simulation.

In StudioView, open the **Trajectory Converter** tool by clicking the icon, or navigate to **Tools > Trajectory converter**:



Apply changes to the original trajectory as needed, following the tabs from left to right. The diagram below illustrates some of the parameters.



Downsampling

Downsampling means decreasing the number of points in a trajectory, so as to make those parts of a trajectory with constant movement parameters as long as possible. The downsampling algorithm excludes points which are within a specified deviation from the general movement direction. If your trajectory uses a substantial number of points, it is strongly recommended to apply downsampling.

Enable it by checking the box on **Downsampling** tab. To define a deviation, select an **Inaccuracy estimation type**, and enter a max. value for it. Also enter a maximum speed change for the trajectory segment.

Smoothing

Smoothing is used to adjust movements parameters that are critical for the receiver performance. By changing the smoothing parameters, you can achieve more realistic speed changes and turn abrupt heading changes into more realistic gradual turns. **Enable** it by checking the box on the **Smoothing** tab. The smoothing algorithm will add points to the trajectory as determined by the algorithmn.

» The **Max rounded distance** is measured from a particular rounded point.

» The **Max rounded segment percent** represents the value of the maximum rounded distance in % from the rounded point relative to the length of entire trajectory segment. The segment of the trajectory refers to any part of a trajectory between two consecutive points.

» The **Max angular velocity** defines the speed of a turn. While rounding angels of trajectory, resulted angular speed will be interpolated within specified limit.

- The **Max vertical velocity** defines the maximum permitted vertical movement between consecutive points. If the altitude has changed between two consecutive points, it will be interpolated, applying this not-to-exceed vertical velocity.
- The **Heading change threshold** defines the maximum change of movement direction at any particular point. If the actual value is greater than the heading change threshold, the trajectory will be rounded at this point.
- The **Max acceleration** in m/s^2. ($3.8 m/s^2$ is a typical value for a performance car [0-60 mph in 7s]).

Interpolation

Interpolation allows to add points to the original trajectory. **Enable** it by checking the box on the **Interpolation** tab.

An **Interpolation step** value of 1 per second generates 1Hz data.

> **Note:** The maximum number of points for an NMEA trajectory is 12000!

Equalization

Use equalization if a steady speed or constant altitude is needed. The value entered will be applied to all waypoints of the trajectory.

NMEA Stationary Period

The NMEA tab allows to set up a **Stationary period** in seconds. It will be added to the beginning of trajectory, allowing the receiver to obtain a fix before any movement starts (this will help to avoid that the receiver under test possibly does not capture the initial part of the trajectory).

> **Note:** This feature only works if the file type of both the Input, and Output trajectory is NMEA, and if the Output destination is set to Write to file.

4.9.11 Creating an RSG Trajectory with StudioView

What is an RSG trajectory?

RSG trajectories[1] are used primarily to simulate airborne applications, such as flight, missile or orbital trajectories. Contrary to a *standard* trajectories (as described under "Creating a

[1]RSG = Real-time Scenario Generation

Trajectory in StudioView" on page 130), an RSG trajectory is not defined by its geographic position, but by **relative changes of movement**.

These motion changes are captured in user-defined parameters which are assigned to SCPI commands (to learn more about SCPI commands, see "SCPI Guide: Introduction" on page 224). RSG trajectories are created and edited with StudioView's RSG Trajectory Editor.

About Real-time Scenario Generation

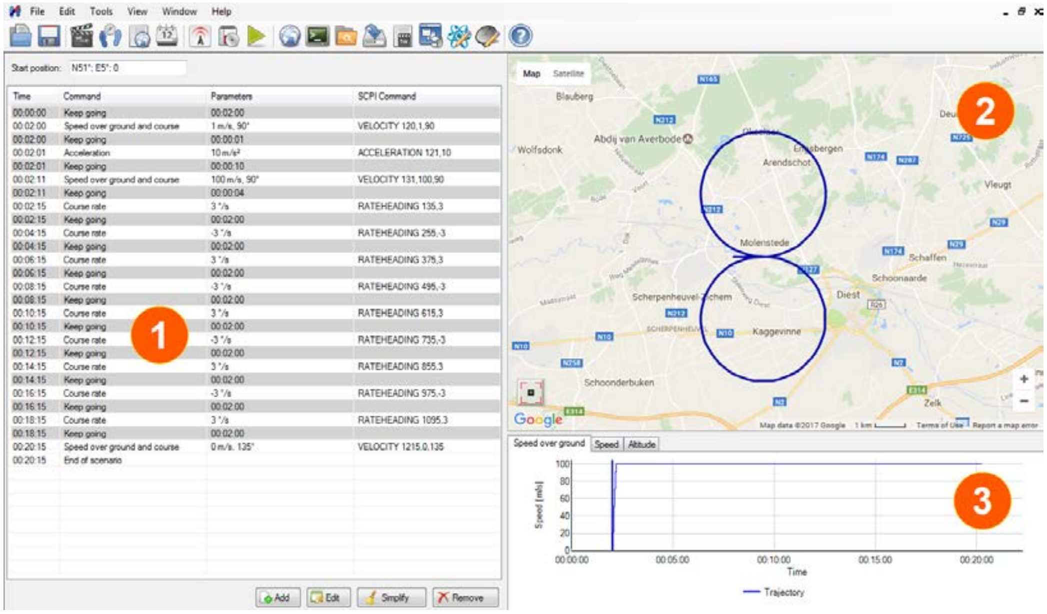
Note that RSG trajectories must be built *prior* to running them. However, there is also the concept of feeding trajectory data into the GSG unit in real-time, i.e. while the trajectory is being generated: This functionality requires the option kit OPT-RSG, which allows the GSG unit to receive trajectory information in real-time from e.g., a motion simulator or a computer running simulation software.

4.9.11.1 Using the RSG Trajectory Editor for the First Time

To open the StudioView RSG Trajectory Editor:

» In StudioView, click in the main toolbar, the **Tools** menu, or in the StudioView Application Tips startup screen. The editor screen will show (the image below shows a loaded scenario for illustration purposes.)
The editor has three panels:

1. The left panel shows a list of all RSG commands for the trajectory currently open (if any).
2. The corresponding trajectory is visualized on the Google Map on the right.
3. The charts below the m ap show speeds and altitude over time.



To open an existing RSG trajectory:

» Navigate to **File > Open...**, or click .

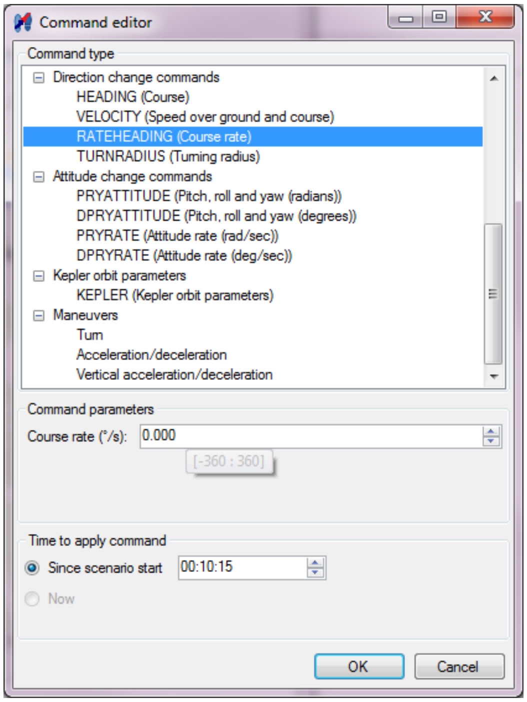
To create a new RSG trajectory:

1. Enter a **Start position**, and an altitude in meters. Use a semicolon and a space as separators:

   Start position: N51°30'10"; E5°; 100

   Note that this data actually is not part of the trajectory, it is used only to assign a relative location to the trajectory. You can later change this location, thereby moving the entire trajectory to a different place (unless the trajectory includes geographic or ECEF position change commands).

2. To add a new command to the list, highlight the command after which you would like to insert the new command by clicking it. Then click the Add button. The RSG **Command Editor** window will appear:



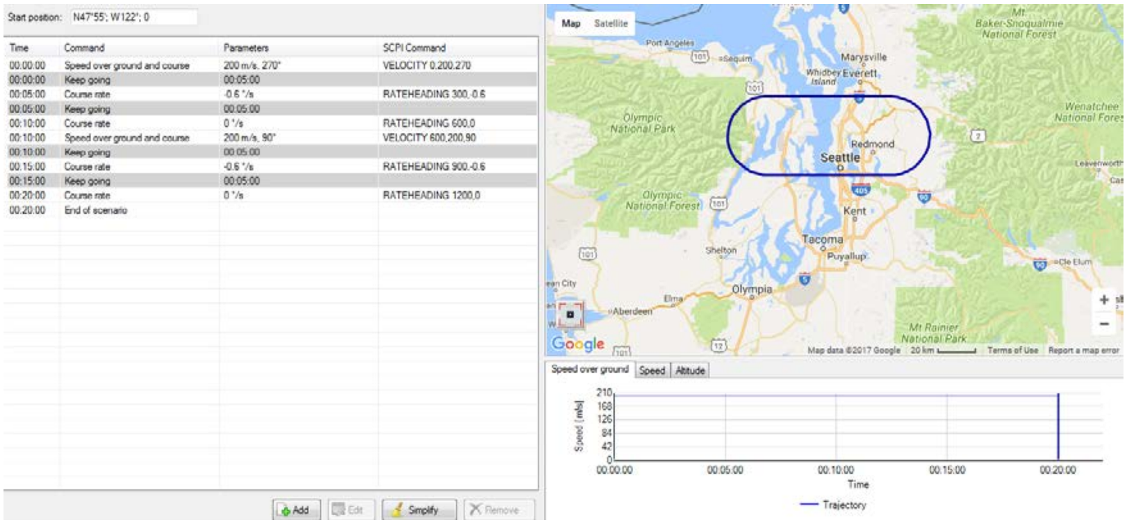
Select the command you want to use, and enter the required parameters. Detailed command descriptions can be found in the "SCPI Guide" on page 223.

3. To edit an existing command, double-click it, or click the Edit button. To delete a command, use the Remove button.
4. To copy or move a command, you can drag & drop it using your mouse in combination with/without the CTRL key.
5. To undo a command, press CTRL + Z. To redo a command, press CTRL + Y.

The map and the chart on the right side of the screen will reflect any visible changes.

4.9.11.2 RSG Example: Racetrack Pattern

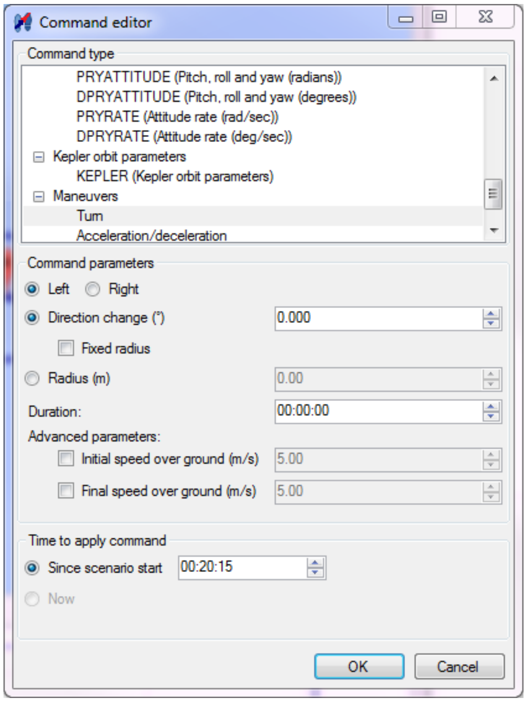
To create a racetrack-shaped pattern:



1. Open the RSG Trajectory editor.
2. Enter a **Start position** and altitude (m), or leave the default.
3. Add **VELOCITY** with initial speed and heading.
4. Next, add a **KEEP GOING** instruction in order to assign a duration to the previous command. This command will be shown with a gray background because it serves as a filler command that is not written into the trajectory file. It is used only to display the trajectory on the map, and to properly time out an action. Select a duration for the **KEEP GOING** filler command e.g., 5 minutes.

> **Note:** With some of the other commands, the KEEP GOING command is created by StudioView (you may still need to assign a duration manually).

5. Next, create a turn e.g., by 180° within 5 minutes: While this can be accomplished with the command sequence **RATEHEADING**, **KEEP GOING**, **STOP**, this would require some calculation to determine that the course rate change for this turn is -0.6°/s. Instead, use the **Maneuver** command **Turn**, and define the Direction change, or the radius of the turn:

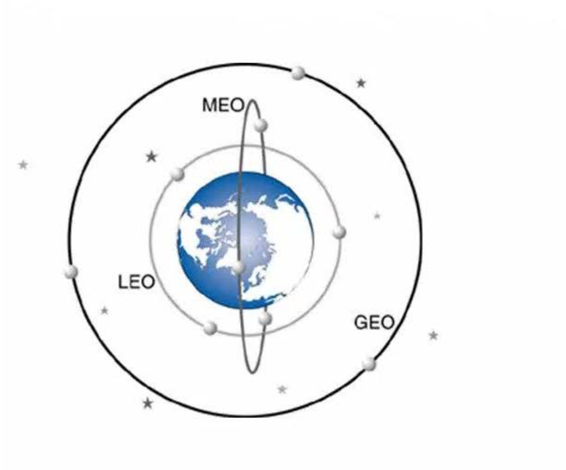


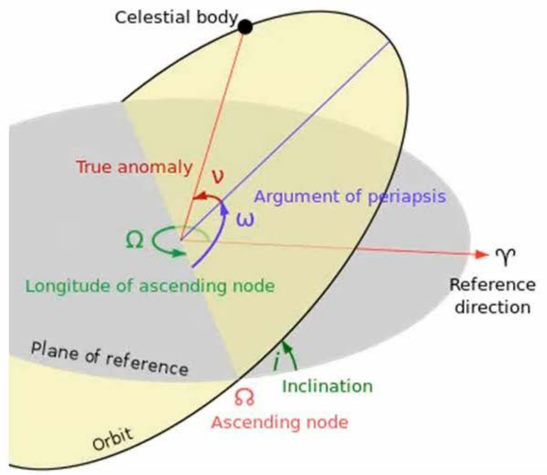
For the turn, select **Left**, **Direction change**: 180°, and **Duration**: 5 minutes.

6. Lastly, repeat all of the steps above to complete the racetrack pattern (replace the heading with the opposite heading).

4.9.11.3 Kepler Orbit

KEPLER orbits are used to build a trajectory for space vehicles. The RSG trajectory editor's Command Editor offers a KEPLER trajectory that is – as all Keplerian orbits are – described by six parameters. These standard parameters make speed and heading change calculations unnecessary, but their specifications are beyond the scope of this documentation, and hence are not further described herein.

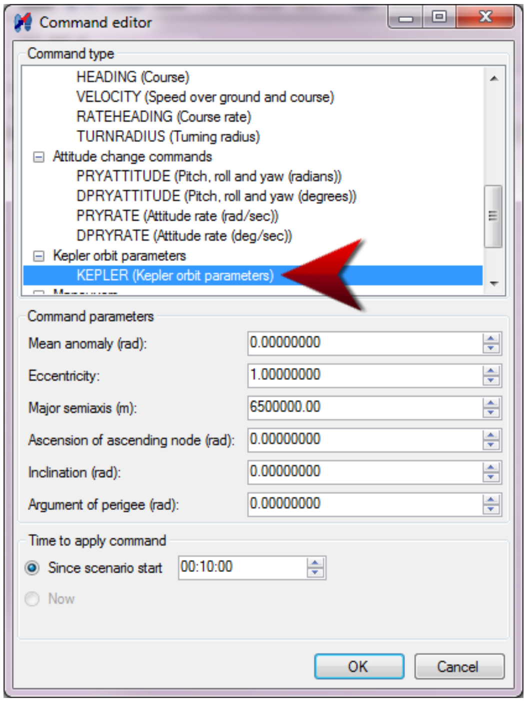




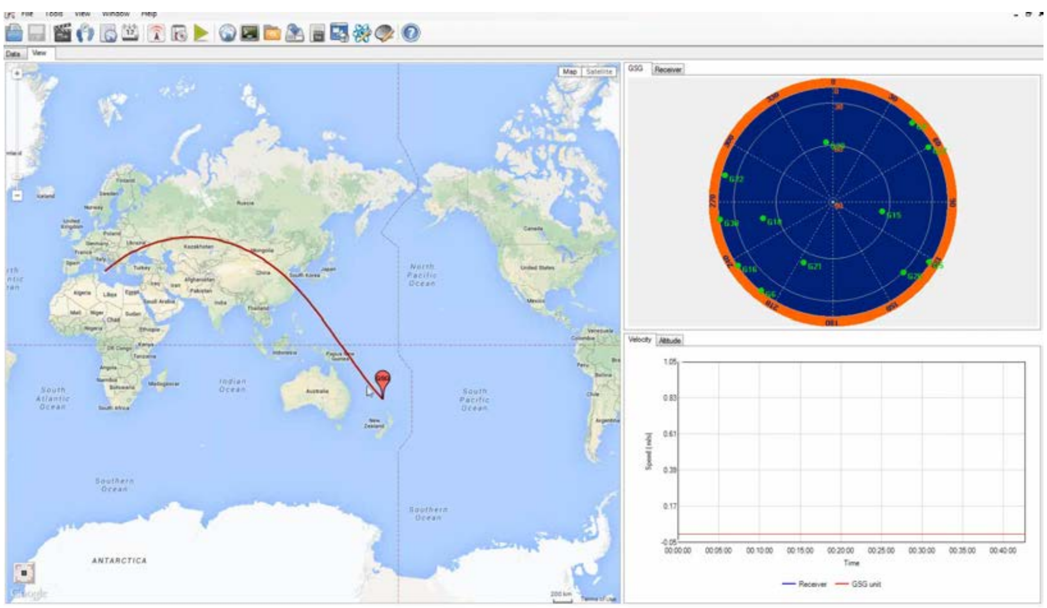
Note: The preferred way to describe space vehicle trajectories are TLE-formatted trajectories, see "Trajectory Two-Line Element Format (TLE)" on page 363.

To access the Kepler trajectory dialog window:

1. In StudioView, navigate to the RSG Trajectory Editor.
2. Click **Add** to open the Command Editor.
3. Scroll down to **Kepler orbit parameters** and click OK. The parameter dialog will show:



4. Populate the fields.
5. Attach the Kepler trajectory to a scenario (see "Configuring a Scenario" on the next page). The result will look similar to the illustration below:

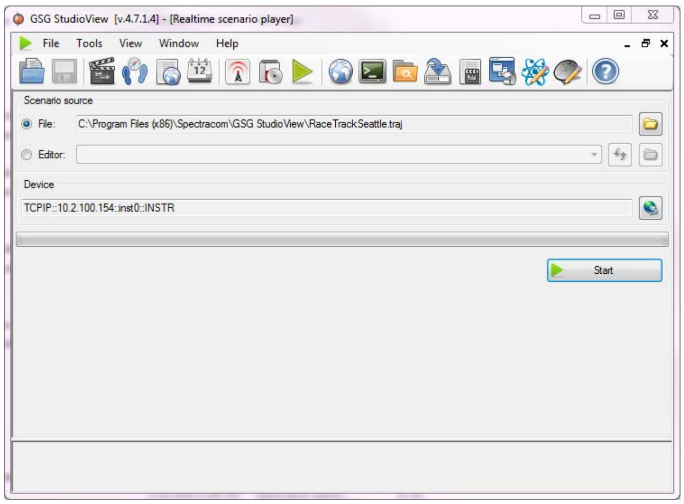


> **Note:** Higher-end GSG models have a sample Kepler trajectory for the ISS pre-installed.

4.9.12 Playing RSG Scenarios in StudioView

StudioView's Realtime Scenario Player allows to play RSG scenarios in real-time.

1. To open the Realtime Player, navigate to **Tools > Realtime scenario player**, or click . The player dialog will open:



2. Use the File or Editor radio-buttons to select the source RSG trajectory (.traj file extension) for realtime playing.
3. Click to open the **Connections Manager** tool (for details, see "Connecting StudioView to GSG" on page 122.)
4. Click Start to start playing.

A list of RSG commands will appear in the RSG Trajectory Editor window.

After a particular RSG command has been fulfilled, it will become crossed out e.g., ~~VELOCITY 0,5,0~~.

4.9.13 Configuring a Scenario

A scenario is the dataset describing a simulation in terms of starting position, duration, trajectory, events and other parameters which you may want to include in your simulation. GSG units come with several predefined scenarios (depending on the GSG model). You may also use StudioView to create your own scenarios, save them to a file and upload them to the GSG unit. The GSG unit will execute the simulation in accordance with the parameters specified in the scenario file.

Scenario data is stored in a text file. To show/hide text of scenario file, click . To configure the scenario, fill in the appropriate fields under the tabs described below.

Once you have completed your scenario configuration, save it and upload the scenario file to the GSG unit by clicking .

Note: StudioView stores all files in a directory chosen during the installation process. By default, the repository is located at C:\Users\UserName\Documents\Spectracom\GSG StudioView\Repository. You may save your scenario in any other folder, but please note you must also save any trajectory, event, antenna pattern, or navigation files you may want to include to your scenario in the same folder.

The **Scenario Editor** provides access to all essential scenario parameters. To access the Scenario Editor, click , or navigate to **Tools > Scenario Editor**:

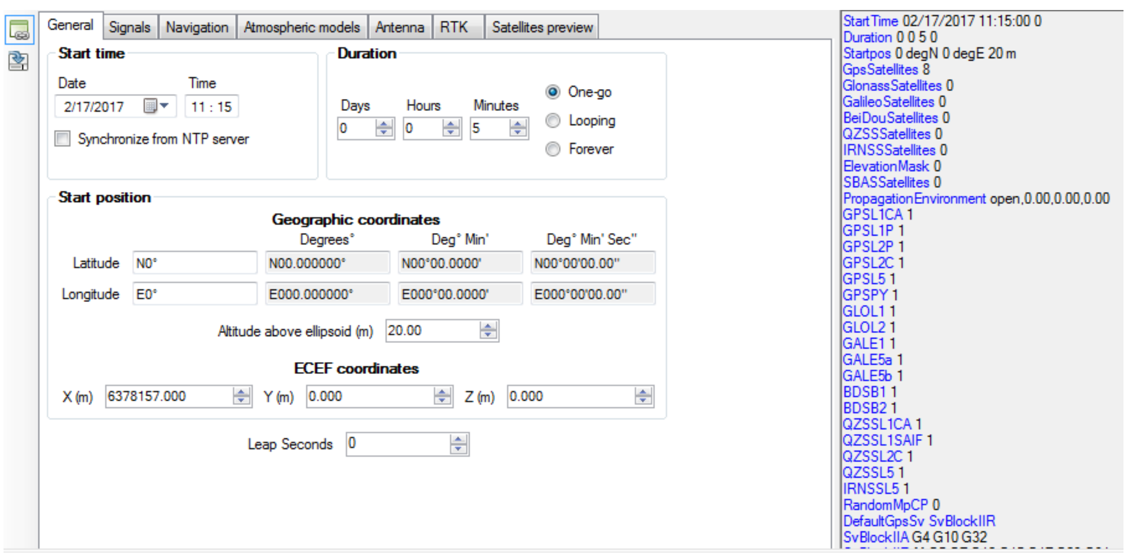


Figure 4-7: Scenario Editor

General tab

Under the **General** tab, you can edit basic parameters like Start time, Duration and Start position:

- The **Start time** is specified using GPS Time. The GPS Time is always used when displaying time. This is not equal to the UTC time frequently displayed by the receivers. Contrary to the GPS time, UTC contains leap seconds.
  The Start Time can be a set time, or the current time derived from an NTP server specified in the Network Configuration settings of GSG device. To use this feature, check **Synchronize** from the NTP server checkbox.
- If the current time from the NTP server is used, next the startup will be delayed up to 2 minutes to allow the simulation to load required data. The start time is aligned to the next full GPS minute. The NTP (UTC) timescale is converted to the GPS timescale by a UTC-GPS offset defined in the NTP server settings.
- Using NTP as start time in conjunction with Ephemeris set to Download is subject to licensing options, as it requires the **Simulate Now** option to be present. In this configuration, the GSG will simulate the sky as it is in that start position at current time. This functionality is currently only available for the GPS constellation. Please also note that the availability of good ephemeris data cannot be guaranteed, but periods where no data is found and hence no signals can be generated, may occur.
- The Duration of the scenario replay can be set to a number of days, hours and minutes. The scenario can be set to:
  - **Looping**, means that scenario will restart again right after execution is finished
  - **Forever**, GSG will download needed navigation data from Internet and run scenario until user will stop it
  - **One-go**, in which case it executes only ones and then returns to main menu of GSG device. Note that the option "forever" only works when the Ephemeris option is set to 'Download' (Start) Position.
- The Start position is specified using WGS84. Note that this also concerns the altitude (ellipsoid height) and that this is not the same as the MSL often output by receivers. StudioView provides automatic conversion between different coordinate input formats; decimal degree, degrees-minutes, degrees-minutes-seconds and ECEF format.

Signals tab

Under the **Signals** tab, you can determine which satellite signals you want to use, the type of environment, and possible Interference signals:

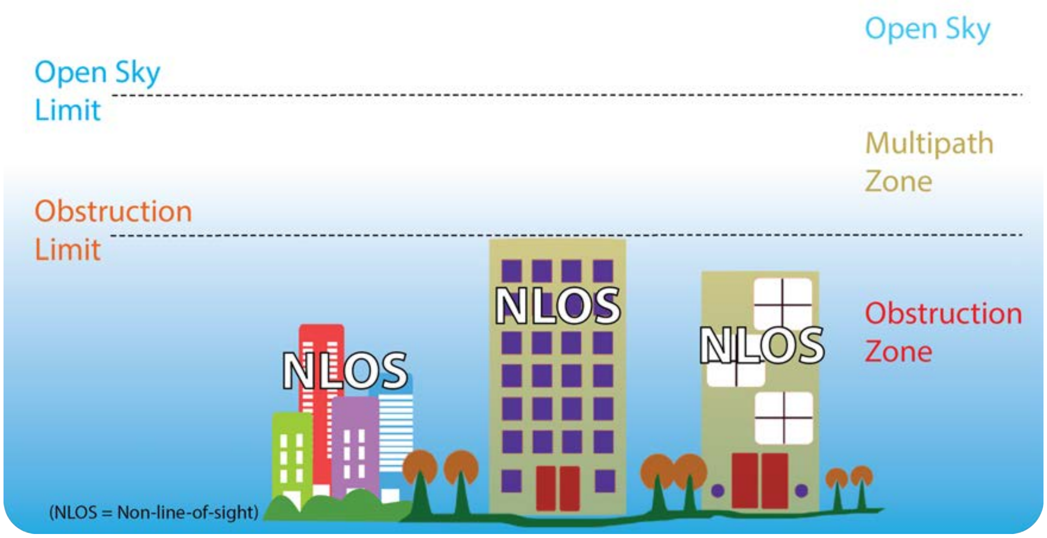
- Under this tab, you can explicitly set the maximum number of satellites to be simulated, with separate settings for GPS, GLONASS, Galileo and Beidou. When the **Auto** keyword is used, the GSG unit will automatically select the highest satellites available and generate the maximum number of satellites that your GSG model allows. You can also configure the number of SBAS satellites to be simulated.
- It is possible to configure the **frequency bands** and possible **(pseudo-P(Y))**

encryption by clicking on the checkbox for the corresponding constellation and band. The availability of all elements for simulation (e.g., GPS L2C, L5 and Galileo) is dependent on the installed licensing options and your GSG model .

- For each constellation, you may specify the **maximum number of signals** in view for a given time, clicking the up or down arrows. Or, just type the number. For maximum number, also see the field tooltips.

> **Note:** The maximum number of signals depends on your GSG model. The unit will decrease the number of signals specified in the scenario to fit your license options. If Auto is selected, the GSG unit will use maximum number of channels.

- Use checkboxes to include or exclude a particular frequency band (e.g. L1, L2, E1, L2 P, etc.) from your simulation scenario. If a checkbox is grayed, it means that it is not supported or the only displayed choice is available.
- There can be 0, 1, 2, or 3 SBAS satellites per scenario. The GSG unit will select SBAS SV based on their elevation with respect to the user position. When the scenario is running the SBAS satellite positions and speed will be updated with the information found in the SBAS messages. You can also select specific SBAS satellites by their IDs (up to 3 total).
- Under **Propagation environment** you can select an environment model which will impact signal propagation. There are four models available:
    - Urban
    - Suburban
    - Rura
    - Open (full clear view of the sky, i.e. no obstructions).



The simulation is carried out based on probability, applying different building densities (sparse <> dense). The feature offers some adjustability. For more information, see "Propagation Environment Models" on page 69.

» By specifying the **Elevation mask**, you define the satellite-in-view cut-off range. All satellites which are below this range will be dropped off and replaced with better/higher satellite (if available). For more information, see "Elevation mask" on page 61.

» You may also add **Interference signals** and **Multipath signals** to the scenario. The maximum number of Interference/Multipath signals is 8.

Interference signals are used to degrade the reception of GNSS receivers. To add an Interference signal, click . To add a signal, use default values or specify Interference signal parameters by expanding the list. You may also collapse or expand all items by clicking on the closed book or open book icon, respectively. To delete an Interference or Multipath signal, click .

Navigation tab

Under the **Navigation** tab, you link files that describe the trajectory, events, environment, vehicle model and navigation data to your scenario:

» Any user **trajectory** can be simulated using the GSG. You can choose to use one of the built-in trajectories or upload a trajectory file created in the Trajectory Editor or RSG Trajectory Editor of StudioView. To select one of **built-in** trajectories, click on Circle, Static, 3GPP and set up parameters if needed.

» To attach a pre-installed trajectory or your own trajectory to the scenario, click **File** and pick a trajectory file from the dropdown menu. To add your own trajectory file to this dropdown list, you have to create a new trajectory first and then save it in the repository.

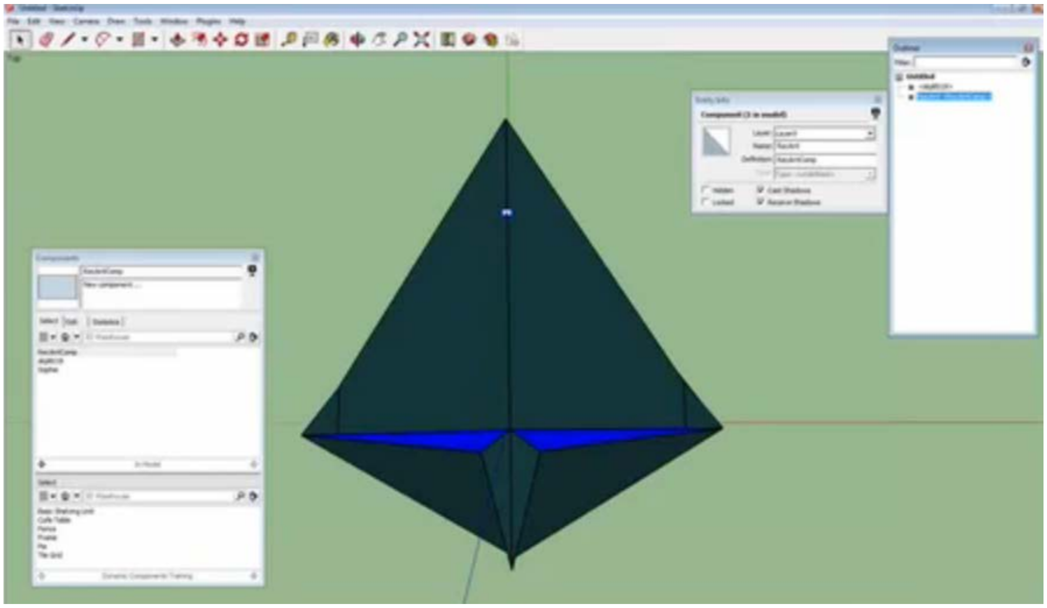
- An **Events** file describes some specified events during scenario execution. To create events file, see "Defining Events in StudioView" on page 155.

  Support for environmental or vehicle models in GSG simulators is via compressed keyhole markup language files (kmz) popularized by Google Earth. A simple way to create these files is with the tool SketchUp available from Trimble® Navigation, see https://www.sketchup.com/.

- An **Environment** model is a 3D model of the environment, describing terrain, buildings, etc. All environment models used must have a 'geo-location' added to them before they can be used by in simulation. Environmental Modeling is used first and foremost to simulate urban canyons, or tunnels. You can create blocks, representing buildings/obstructions, and place them on the map along the trajectory. The power level of the satellites will be blocked or reduced in the vicinity of the buildings due to the obstruction of the line of sight near these virtual buildings.

- A **vehicle model** represents a 3D model of the vehicle. The vehicle model will move with the simulated trajectory. The vehicle model will also follow any pitch/roll/yaw movements simulated, i.e. if the vehicle rolls by 90 degrees, half of the sky is likely to be blocked by the vehicle itself, depending on vehicle model used. The body center of the simulated vehicle will be in the origin position of the model. The antenna position can differ from the body center position by configuring lever arm values in the scenario configuration. The antenna position can also be specified in the vehicle model file by adding a component named "RecAnt". If both lever arm and RecAnt are set, the receiver antenna position as set in the vehicle model takes the precedence. The vehicle model does not need a geo-location.

  Vehicle models can also be created with the software tool "Sketchup", see above.



If a satellite is blocked by an object from either the environment or vehicle model, i.e. it is not visible by the receiver antenna, its power level is set to OFF.

For more information, see https://www.orolia.com/documents/environmental- modeling- gpsgnss-simulators.

GSG can successfully handle vehicle models with up to 130 triangles and models should be optimized for low polygon count. The triangle count is limited to a total of 300 for the combined environment and vehicle models.

» **Navigation data** allows you to specify Almanac and Ephemeris files to be used during simulation. Next to the Default option, you can download navigation data from the official web sites, or use your own Almanac or RINEX files. For more information on RINEX files, see "Editing RINEX Files in StudioView" on page 170.

To download navigation data from the official web sites, click , and then click . The navigation data for the scenario start time and number of satellites you specified under the **Signals** tab will be downloaded. To add the navigation data files to your scenario, click and select the files needed.

To edit navigation data in the RINEX editor, select the file from the list and click .

Atmospheric models tab

Under the **Atmospheric** models tab, you can model the Ionosphere and Troposphere.

» The GSG unit comes with built-in support for an Ionospheric model. When set to **On**, by default the used model is a reverse model of the model described in IS-GPS-

200D, section 20.3.3.5.2.5. When set to **Off**, no delays caused by the Ionosphere are used in the simulation. GSG also supports simulation of Ionospheric delays using files in IONEX format. To specify a particular file, select **Files** and choose it from Repository.

Antenna tab

Under the **Antenna** tab, you determine which type of antenna you would like to simulate, as well as the lever arm, which specifies the antenna position relative to the vehicle center of movement.

RTK tab

Under the **RTK** tab, it is possible to simulate a virtual Base Station: Specify its geographic coordinates and altitude, as well as an RTCM protocol version and type of RTCM messages to be simulated.

Satellites preview tab

Under the **Satellites preview** tab, you can visualize the satellites in view.

4.9.13.1 Defining Events in StudioView

To make a simulation more realistic, you can introduce events that can change power levels of certain signals, add or modify multipath signals, change the propagation environment, and modify navigation messages.

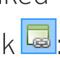
In order to describe an event, you need to specify the **event time** in seconds counted from the beginning of scenario, and set the **event parameters**.

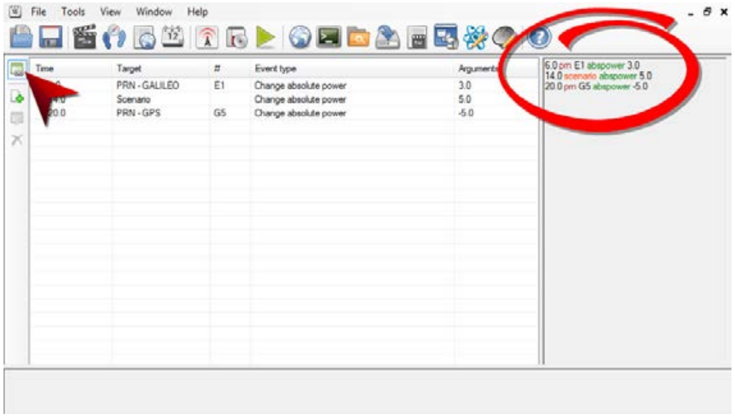
Several events can occur in the same epoch. Note that PRN/channel events overrule scenario events.

> **Note:** One GSG epoch equals a 100 ms block of time.

Using the Events Editor

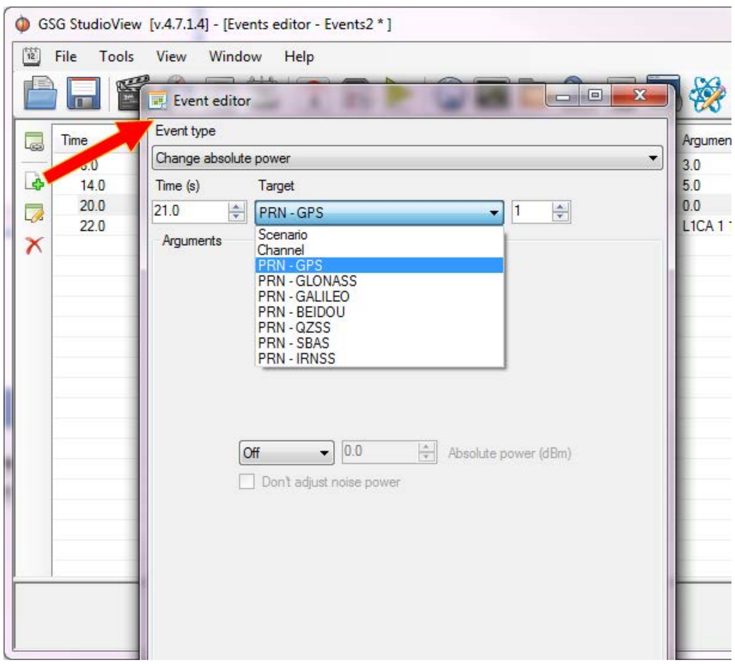
To open StudioView's Events Editor, navigate to **Tools > Events editor**, or click .

Events are listed in the table on the left and stored in a text file that is linked to the scenario. To **show/hide** the event file text on the right side of your screen, click :



Adding an Event

To add a new event, click :



To set the type of event, choose an option from the **Event type** drop-down menu:

- Change absolute power: Defines a power level for a given channel or PRN code.
- Change relative power: Defines a change in the power level for a given channel or PRN code.
- Create new multipath signal
- Delete multipath signal

- Change multipath signal parameters
- Change navigation message bits
- Change signal propagation model.

To set the **Time** at which the event is to occur, specify the number of seconds from the beginning of scenario.

Define a **Target**:

- **Scenario** will apply the specified event to all satellites simulated in scenario.
- **Channel** will apply the specified event to one of GSG's channels.
- **PRN**[1]**-x** will apply the specified event to a particular GNSS satellite or SBAS; select a PRN code number.

There are two Parameters types available for each event type - Absolute power or Relative power - in next drop-down menu.

For Channel event, type Duplicate is also available.

Editing or Deleting an Event

To edit an event, highlight it, then double-click it, or click .

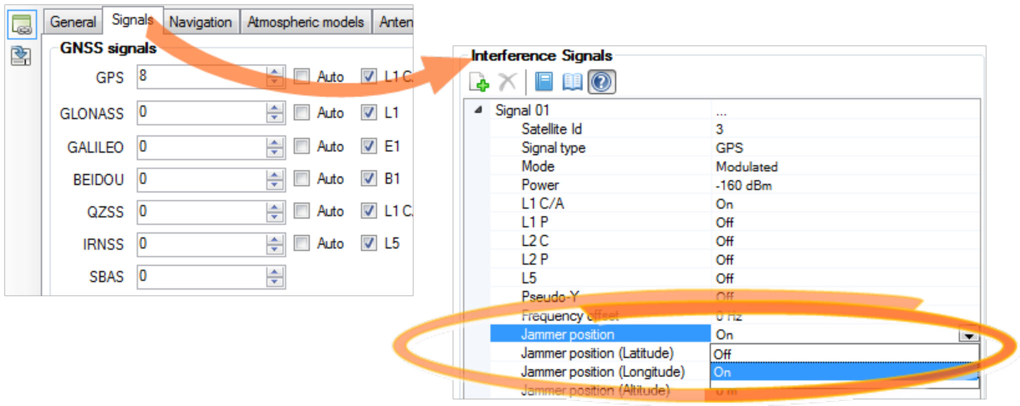
To delete an event highlight it, then click .

4.9.13.2 Adding a Jammer Signal in StudioView

It is possible to add a localized jamming signal to a scenario (or several of them), so as to determine the response of a receiver-under-test to a jamming/interference condition.

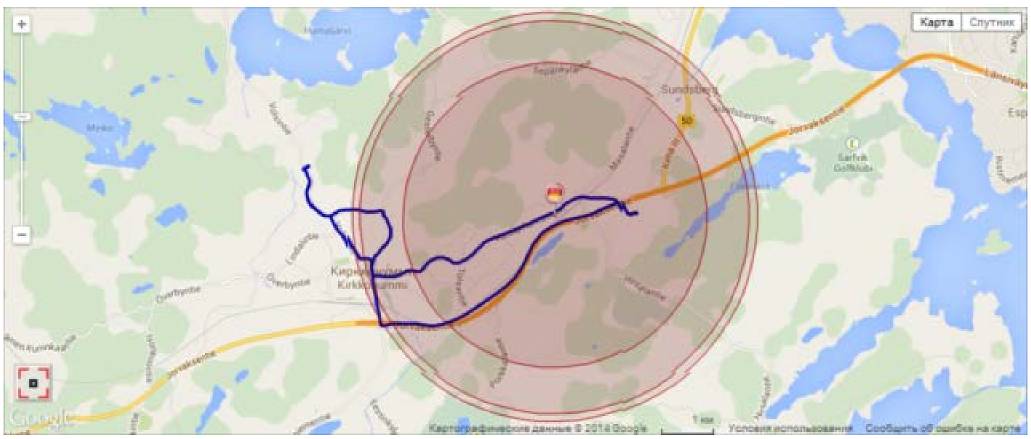
1. In StudioView, open the Scenario editor by navigating to **Tools > Scenario** editor, or by clicking .
2. Open the scenario or trajectory of your choice, or start a new one. For more information, see "Configuring a Scenario" on page 148.
3. To add a jamming source to your scenario, go to **Signals** tab, and under **Interference Signals** add a new signal by clicking , or edit an existing signal.

[1]Pseudo-Random Noise



4. Locate the line item **Jammer position**, and turn it ON.
5. Enter the **geographic position** for the new interference signal.
6. Specify the **frequency bands** you wan to jam and other parameters for the new interference signal.

Please note that now your jammer will be displayed on the map in the **Trajectory Editor** and the **RSG Trajectory Editor**. However, it will only show the area along your trajectory impacted by the jammer **in 2D**.

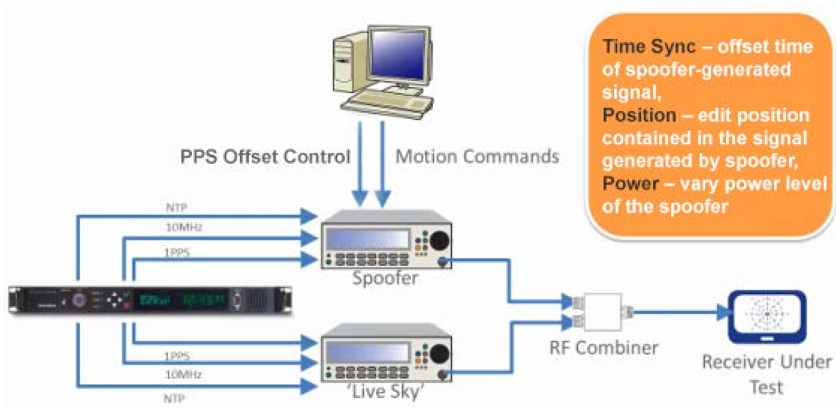


4.9.13.3 Spoofing a Signal in StudioView

> **Note:** This functionality is used with the VTS System (Vulnerability Test System), which includes a GSG spoofing license.

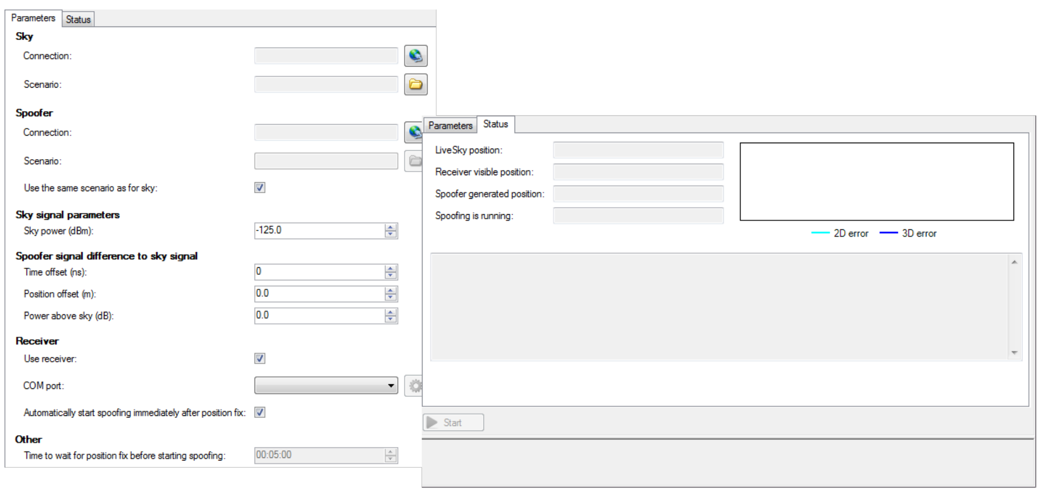
A spoofing test in StudioView exposes the device-under-test not only to the authentic **Sky** signal, but also to a second signal generated by the **Spoofer**. This second signal can be used to challenge the capability of the receiver to discern between the genuine and the fake signal. Several parameters can be adjusted to tweak the test scenario, if needed.

The testing environment illustrated below is used to test a system against spoofing vulnerability.



To configure a spoofing scenario:

1. Navigate to **Tools >** Spoofing test. The **Parameters** panel will open (the **Status** tab is used during the scenario execution, see "Running a Spoofing Simulation" on the next page):



2. Populate the menu fields:

   » **Sky**

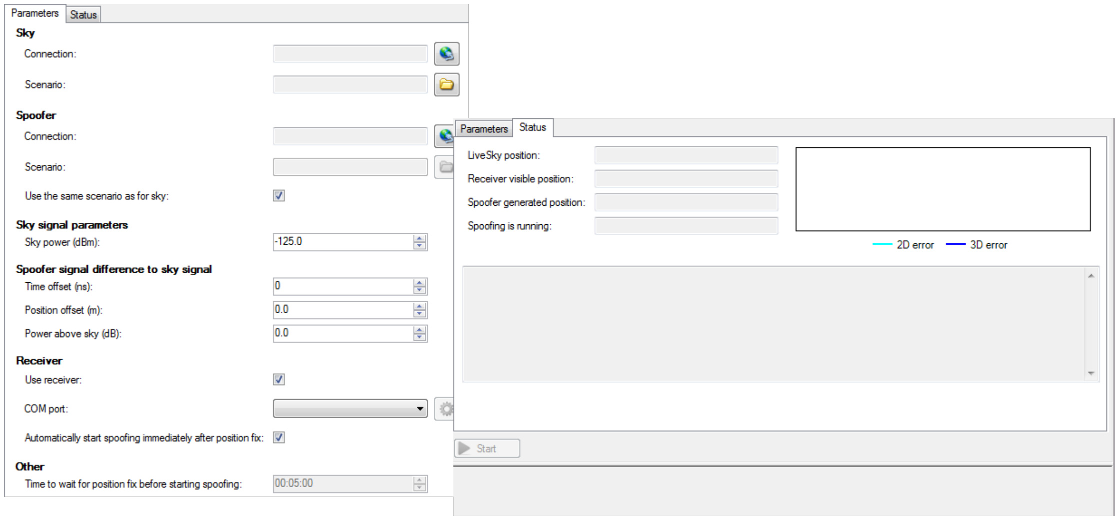
      » Connection: Open the **Connections Manager** to establish a connection to the device that generates the authentic satellite data (simulated, recorded, or live sky)

      » Scenario: Select a scenario for the "authentic" simulation.

- **Spoofer**
  - Connection: Open the **Connections Manager** to establish a connection to the device that generates the spoofing data (simulated, recorded, or live sky)
  - Scenario: Select a scenario for the "spoofed" simulation.
  - Use the same scenario as for sky: [Yes/No] Check if you want to use the same scenario for both the sky data, and the spoofing data.
- **Sky signal parameters**
  - Sky power (dBm): Select a signal strength for the authentic signal.
- **Spoofer signal difference to sky signal**
  - Time offset (ns): Determine by how much the spoofed signal's time shall be offset from the sky signal's time. The time is not directly related to UTC, it only states the time difference between two signals.
  - Position offset (m): Determine by how much the spoofed signal shall be offset from the sky signal.
  - Power above sky (dB): Determine how much stronger the spoofed signal shall be in comparison to the sky signal.
- **Receiver**
  - Use receiver: [Yes/No] Determine if you want to feed the GNSS receiver data into StudioView during the simulation.
  - COM port: Determine to which port the receiver is connected
  - Automatically start spoofing immediately after position fix: [Yes/No] Determine when to start the spoofing. When checked, StudioView starts spoofing immediately after the receiver can determine its position. If unchecked, the field **Time to wait for position fix before starting spoofing** (see below) is used to determine the spoofing start delay.
- **Other**
  - Time to wait for position fix before starting spoofing: [min:sec] Determine when to start the spoofing.

Running a Spoofing Simulation

Once you have configured the spoofing parameters under **Tools >** Spoofing test > **Parameters** tab, open the **Status** tab in order to run the scenario:



- Click the **Start** button in the bottom-left corner. The following parameters will be updated in real time:
  - LiveSky position: Provides the actual position, as determined using the live sky satellite data.
  - Receiver visible position: Provides the actual position, as measured by the receiver.
  - Spoofer generated position: Provides the position, as calculated using the spoofed signals.
  - Spoofing is running: Tells if the spoofing signal is active at this moment.

  The chart on the right will visualize the 2D error, as well as the 3D error between the LiveSky position and the receiver-visible position.

4.9.13.4 Using SBAS in a Simulation

GSG will select SBAS SV based on their elevation with respect to the user position. When the scenario is running the SBAS satellite positions and speed will be updated with the information found in the SBAS messages. In particular, for each MT9 message, the satellite's position and speed are updated.

Although PRN120 - PRN158 are all reserved for SBAS systems, only a few of them are actually used by satellites. When determining the elevation angle of SBAS satellites, GSG unit looks for the SBAS satellites listed below. This is in contrast to the signal generator mode where you can specify any SBAS PRNs to be simulated.

The currently supported SBAS satellites are:

- EGNOS: 120, 124, and 126
- WAAS: 133, 135, and 138
- MSAS: 129, 137
- GAGAN: 127, 128

The simulator uses two approaches for SBAS messages:

1. Default SBAS messages (MT63)
2. EGNOS/WAAS/MSAS message files

The default SBAS messages are always available. These messages should be recognized by SBAS-compatible receivers. However, they carry no information and will therefore not enable the receiver to correct GPS signals.

SBAS message files for both EGNOS, and WAAS are supported. EGNOS files (`.ems`) are ASCII and hourly, while WAAS files are typically in binary format and cover a whole day. Both systems share the same format of the messages and details can be found in http://www.navipedia.net/index.php/The_ EGNOS_ SBAS_ Message_ Format_ Explained.

When the scenario has Ephemeris set to "Download", the GSG unit will download the SBAS messages from official sites and match these messages to the time of the scenario. The SBAS messages broadcast by these satellites are downloaded automatically from these public FTP sites:

- EGNOS: ftp://131.176.49.48
- WAAS: ftp://ftp.nstb.tc.faa.gov
- MSAS: www.enri.go.jp
- GAGAN: `default MT63`

GSG uses an anonymous login. However, note that both FTP sites are likely to track and record all FTP access, including access by the GSG-55.

The SBAS download starts when the constellation simulation of the scenario has started; not during initialization of the scenario.

If a scenario needs SBAS messages that cannot be downloaded from these FTP sites, the scenario continues, but the GSG unit transmits null-messages (SBAS message type: `MT63`). An SBAS-compatible receiver should still be able to see the SBAS signals, but it will not find any useful information (range corrections, time offsets, etc.) in the SBAS messages.

It follows that SBAS scenarios run best with a live Internet connection. Furthermore, since the aforementioned FTP sites store only a limited amount of SBAS records, the start time of SBAS scenarios has to be chosen carefully. Usually, SBAS records that are less than a year (EGNOS)/6 months (WAAS) old can be found on the aforementioned FTP sites. Select a start time that is not older than one year for EGNOS scenarios, and not older than

6 months for WAAS scenarios. Moreover, the start time shall not be too close to the current time. For EGNOS, there can be a one day delay before the SBAS messages are published on the FTP site. For WAAS the delay can possibly be longer (up to 3 or 4 days).

The Internet connection is not always needed. All downloaded ephemeris data and SBAS data will be locally stored on the unit once they are downloaded. So, the next time the same scenario runs, the ephemeris data and SBAS messages are read from the local storage and no Internet connection is needed. The unit performs automatic clean-up of downloaded files. Such clean-up will occur when free disc space is less than 20% of the total disc space.

Note: Currently SBAS corrections are not 'applied backwards' to the outputted GPS signals, even though the corrections will be transmitted in the SBAS signal.

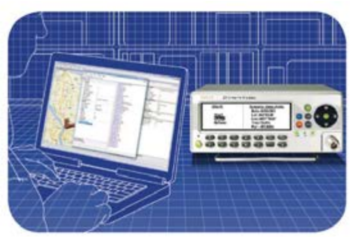
4.9.14 Record and Playback

Note: This is an optional feature for which the Record and Playback option (OPT-RP) is required.

The Record and Playback software converts recorded NMEA messages into scenario, trajectory, and event files. You can then upload these files to your GSG unit where they can be played back in order to recreate the original scenario.

4.9.14.1 Standard Workflow

1. Connect the recording receiver to a laptop running the software and record



2. Connect the laptop to the simulator, edit the scenario as desired, and run the playback generator

3. Connect the simulator the receiver under test and run the scenario

1. While traveling in a car along the planned test route, **record NMEA data** by using a GNSS receiver and antenna (included in the **OPT-RP** kit), connected to a laptop computer with StudioView installed on it. StudioView will record the data generated.

2. Run the recorded trajectory through the **Scenario Generator** in StudioView. Note that your StudioView computer must be connected to the GSG unit, and that the OPT-RP option must be enabled.

   StudioView will automatically generate the scenario, event and trajectory files for you.
3. **Upload** the generated scenario files to your GSG unit, and playback the scenario.

4.9.14.2 Installation of the OPT-RP Software

From the GSG Software CD, copy the file `setup_vx.x.exe` to your PC. Run the executable to install the Record and Playback software.

4.9.14.3 Usage Notes

i. The Record and Playback software is intended for use with a properly licensed GSG. If your GSG needs a Record and Playback license, please contact Orolia (see "Technical Support" on page 207).

ii. The Record and Playback program uses the GSG VISA address to check for a valid license file. This address is also used to upload the results to the GSG if the **Auto upload** feature is enabled. GSG supports communication via TCP/IP, USB, and GPIB. For TCP/IP connections, this program will accept IPv4 addresses as well as VISA resource strings. The following lines describe the resource name syntax.

iii. VISA resource string format:

   » `TCPIP[board]::host address[::LAN device name][::INSTR]`

   » `USB[board]::manufacturer ID::model code::serial number[::USB interface number][::INSTR]`

   » `GPIB[board]::primary address[::GPIB secondary address][::INSTR]`

   For more information about the VISA address, please see:

   http://zone.ni.com/reference/en- XX/help/371361N- 01/lvinstio/visa_ map_address/

iv. The signal to noise ratio (SNR) is used to compare the level of a desired signal to the level of background noise. The SNR is expressed in decibels (dB) and is used to describe the GNSS signal strength in NMEA 0183 GSV sentences.

v. On the GSG unit, signal strength is specified in dBm, i.e. the power ratio in decibels of the measured power referenced to one milliwatt (mW). Since dB cannot be directly converted into dBm, the Record and Playback program relies on a decibel offset value. This offset value maps the NMEA signal strength (in dB) to the GSG signal strength (in dBm). An offset value of [-160] is recommended to begin with. This value can be adjusted up or down until the offset is satisfactory.

vi. You must also specify any scenario options that cannot be extracted from NMEA sentences. These options include signal type, antenna model, troposphere model, temperature, pressure, humidity, and elevation mask. For more information, see under ""Select" Menu" on page 38.

vii. The input file name specifies the NMEA -0183 file to parse. This file should only contain NMEA -0183 GGA, RMC, and GSV sentences.

viii. The output file name specifies the name of the scenario as it will appear on the GSG unit. Do not include a file extension. The Record and Playback program outputs scenario, trajectory, and event files. It will automatically append the correct file extension for each output file. By default, the output files will be generated and saved in the same directory as the input file.

4.9.14.4 Recording Data with StudioView

StudioView's **Data Recorder** allows to obtain all required data automatically without the need to execute SCPI commands. The Data Recorder generates NMEA data from the trajectory currently being played on the GSG unit, collecting RINEX navigation and observation data, recording RSG parameters, satellites information and navigation messages.

How can the recorded data be retrieved?

The data can be retrieved either by using the GSG Web Interface, or via SCPI commands, or by using Studioview.

Using the StudioView Data Recorder for the First Time

To open the Data Recorder, navigate to **Tools > Data Recorder**, or click :

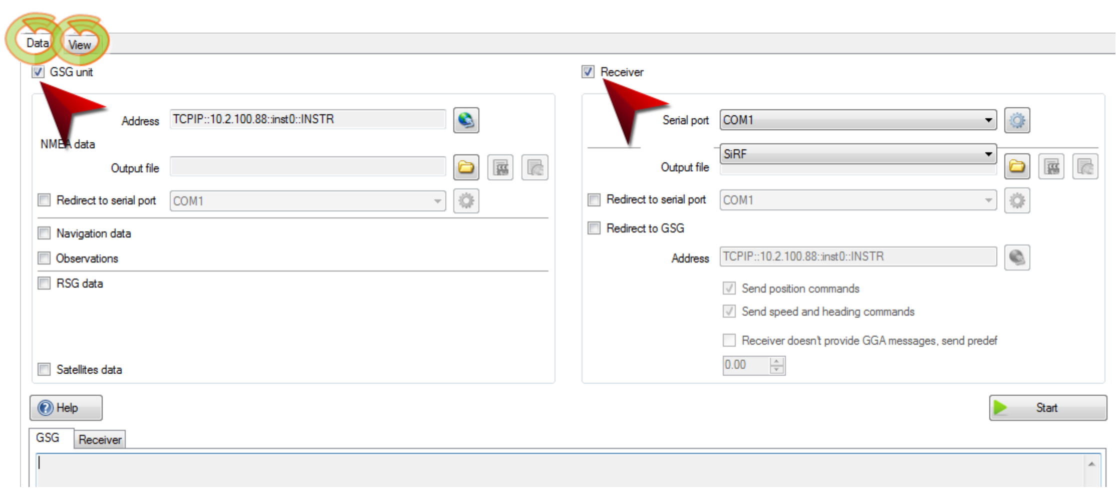


Figure 4-8: StudioView's Data Recorder

The Data tab

Under the **Data** tab you decide where the data to be recorded comes from (a GSG unit, or a GNSS receiver), and which data to log:

- RINEX navigation files
- RINEX observation files
- NMEA from GSG unit
- NMEA from connected receiver (This can be used to record NMEA data for the GSG Record and Playback option.)

You can also redirect NMEA data via a serial port, or – in case of Receiver data – to a GSG unit.

The Receiver tab

Under the **Receiver** tab you can track the position in real time. On the right side of the **View** panel, the satellites in view are displayed, as well as speed, altitude and error data.

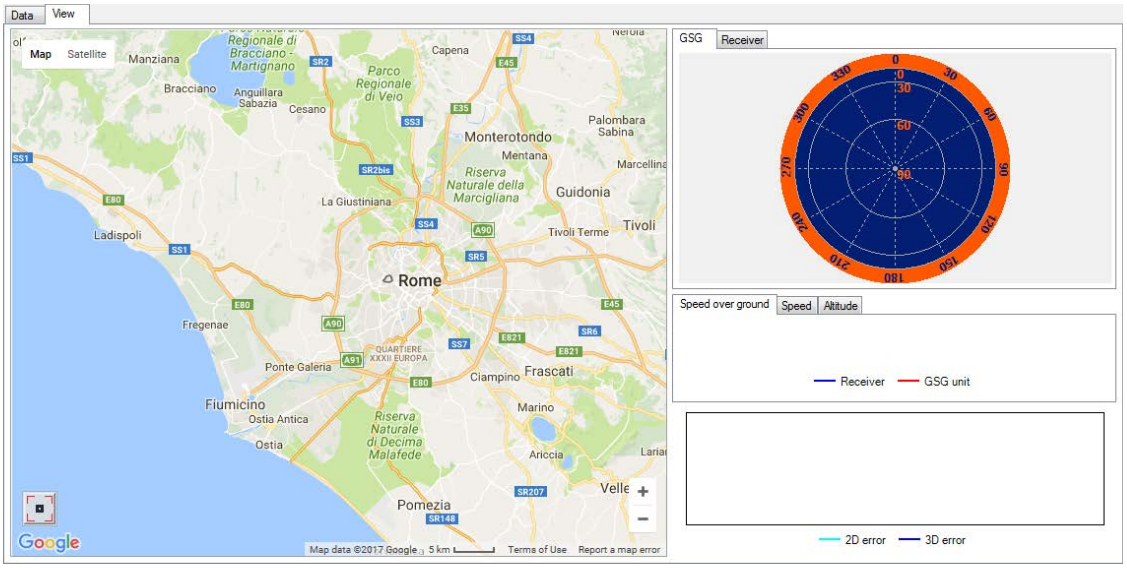


Figure 4-9: Data recorder View window

Preparing the Recording of Data Generated by a GSG Unit

On the **left side** of the screen:

1. Check the box **GSG unit**, then click to configure the connection (for details, see "Connecting StudioView to GSG" on page 122.)
2. Choose which data to record: Navigation data, Observations, RSG data, Satellites data (e.g., satellite position, Doppler shift, etc), navigation messages.

   Select **Output files** for each recorded data category.
3. As an option, the NMEA data can also be **redirected** to a serial port e.g., to use it with a device that utilizes real-time NMEA data (such as a marine plotter). If so desired, select and configure the port to be used to redirect the data.

Preparing the Recording of Data Generated by a GNSS Receiver

On the **right side** of the screen:

1. Select **Receiver** if you have a GNSS receiver connected to your StudioView computer, and you want to record the life sky data it reads while e.g., driving a car
2. Select and configure the **Serial port** (if in doubt, use the default settings)
3. Select the type of **Receiver chip**: If you have a SiRF chip, StudioView will configure this receiver automatically.
4. Select an **Output file** for the recorded NMEA data that the receiver sees.
5. If so desired, you can not only record the generated data, but also redirect it to a different **serial port** of your computer e.g., to consume the data with a third-party application. If applicable, select and configure that port.
6. As an option, the NMEA data can also be **redirected** to a GSG unit, in order to use it with a device that utilizes real-time NMEA data (e.g., a marine plotter or receiver demo software).
   Note that GSG does not accept real-time NMEA data, only planned NMEA trajectory data. When redirecting NMEA data to a GSG unit, StudioView actually converts the NMEA data to RSG commands prior to sending the data.

   To configure this feature, click (for details, see "Connecting StudioView to GSG" on page 122.)
7. Select which data to send to the GSG unit. If the receiver does not send GGA data, i.e. the data stream does not include any altitude information (as is the case with RMC data), you may set a predefined altitude.

Recording the Data

Now, that the configuration is complete, click Start and begin with your test drive.

Or, load the desired scenario on your GSG unit, and click Start.

DO NOT start the scenario via the GSG unit, since the RINEX navigation data will not be captured! (Unless you manually submitted the SCPI navigation data logging command.)

Once data starts to be generated, the Data Recorder will display the incoming raw data in the bottom section of the Data tab. Under the **View** tab (top-left corner of the screen) you can also visually display the progress in real-time on the Google Map and see a skyplot with all visible satellites, as well as velocity and altitude charts.

4.9.14.5 Processing Recorded Data for Playback

Note: This is an optional feature for which the Record and Playback option (OPT-RP) is required.

Part of the StudioView **Record & Playback** workflow (see "Record and Playback" on page 163) is to augment and convert the data previously recorded (see "Recording Data with StudioView" on page 165) so that it can be played back as a scenario on a GSG unit. This conversion is done with the StudioView **Scenario Generator** tool.

The Scenario Generator translates the GGA, RMC and GSV sentences[1] contained in the NMEA into a syntax that is used by GSG scenario, trajectory and event files. These files are required to playback the recorded data on a GSG unit.

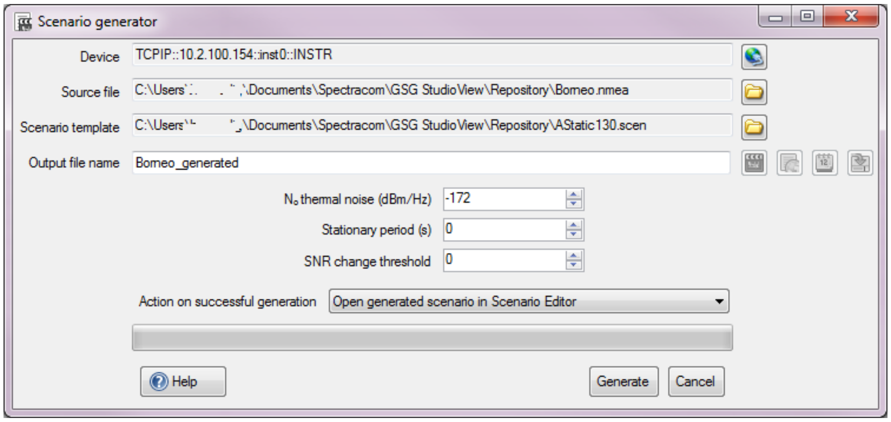
Note: The GSG unit requires an internet connection to replay the recorded data.

Generating the Scenario Playback Data

In order to generate the playback data, it is necessary to have a GSG with the Record and Playback Option (OPT-RP) enabled connected to the PC running StudioView.

1. In StudioView, navigate to **Tools > Scenario Generator**, or click . The Scenario generator dialog window opens:

[1]For example, the GGA and RMC sentences contain position, speed, heading and altitude information, while the GSV sentences record which satellites had been in view and what had been their power levels at any given time during the trajectory.



2. Click to open the **Connections Manager** tool (for details, see "Connecting StudioView to GSG" on page 122.)

3. Select your recorded NMEA file as the source file by opening the file dialog and navigating to file.

   The **Scenario Generator** uses the GSG default scenario parameters for the Playback function. To review these default parameters, open the **Scenario Editor**.

4. Alternatively, you can select a different scenario as a template, in order to use non-default scenario parameters: Click next to **Scenario template**, and locate the scenario you want to use as a template on your PC.

   If you want to use the GSG default scenario parameters, you can leave the **Scenario template** blank.

5. Populate the following settings:

   » **No thermal noise (dBm/Hz)**: On the GSG, signal strength is specified in dBm. The Record and Playback generation relies on a decibel offset value. This offset value maps the NMEA signal strength (in dB) to the GSG signal strength (in dBm). An offset value of [-160] is recommended to begin with. This value can be adjusted up or down until the offset is satisfactory.

   » **Stationary period (s)**: Choose to add a stationary period to your trajectory if the movement starts immediately. If the recording already contains a stationary period, then adding an additional one is not necessary.

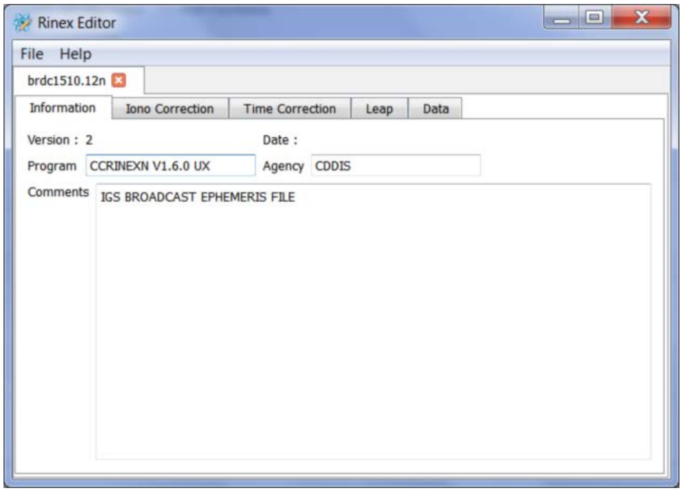
   » **SNR change threshold**: The Signal-to-noise ratio (SNR) is used to compare the level of a desired signal to the level of the background noise. SNR is expressed in decibels (dB) and is used to describe the GNSS signal strength in NMEA 0183 GSV sentences.

6. You may also choose **Actions** after generation is complete. Use the drop-down menu to choose to open the files in the editors or open the Uploader to load them onto the unit.
7. Click **Generate** to create the files.

4.9.15 Editing RINEX Files in StudioView

RINEX files contain Ephemeris data that can be edited with StudioView's RINEX editor:

» Open the RINEX Editor by navigating to **Tools > Rinex Editor**, or by clicking .
The Editor window will open:



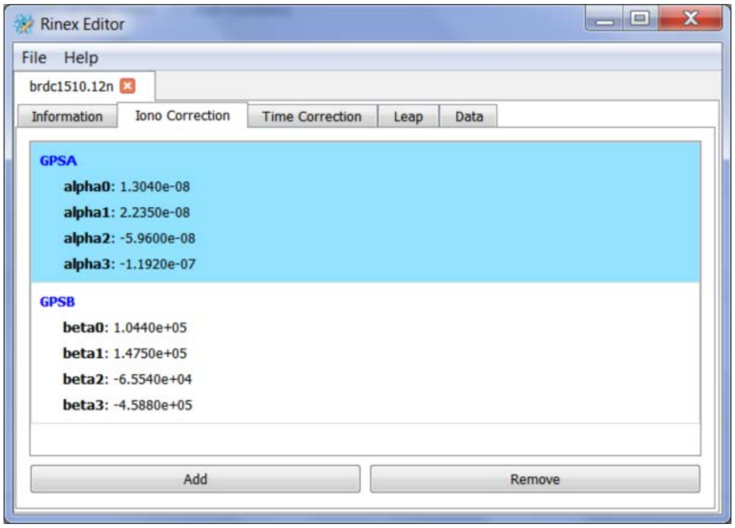
» Open an existing RINEX file to edit it.
The Editor functions are grouped under five tabs:

The Information tab

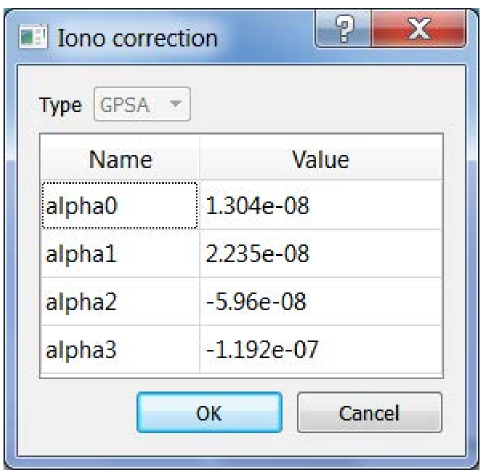
Under the **Information** tab (see illustration above) you can edit the **Program**, **Agency** and **Comments** fields of a RINEX file.

The Iono Correction tab

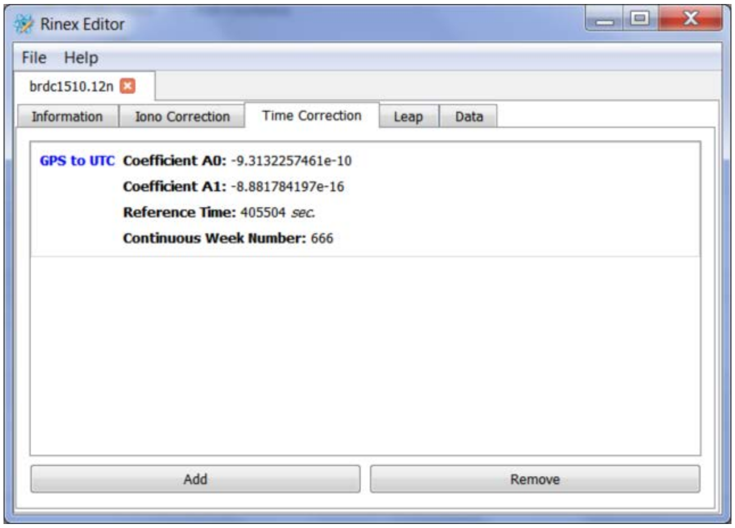
Under the **Iono Correction** tab, you can change the values of correction coefficients.



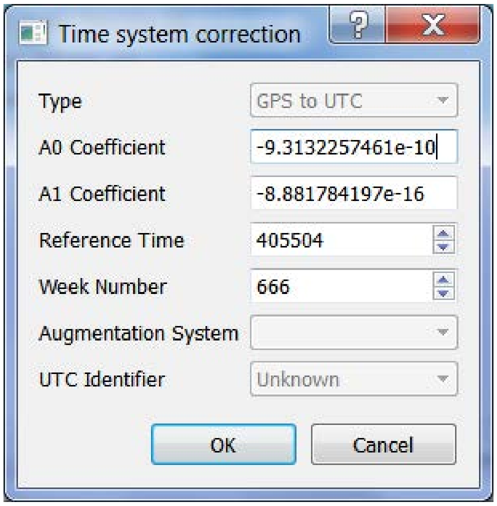
The following dialog box will appear if you double-click on a highlighted **GPSA** or **GPSB** row:



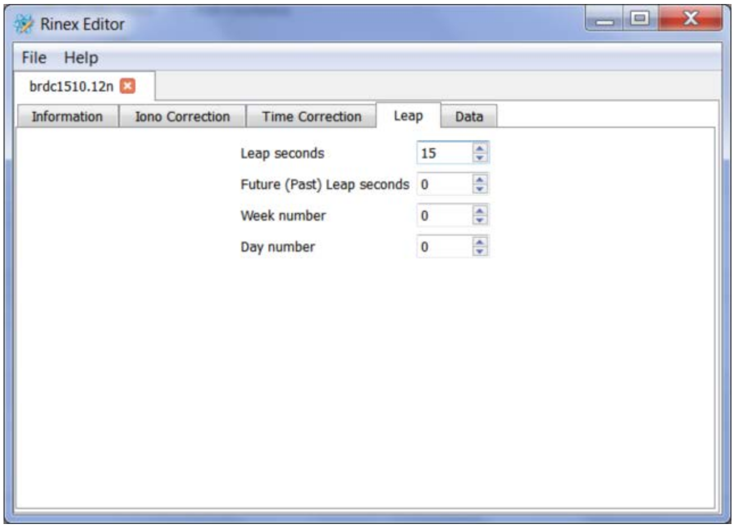
The Time Correction tab



The **Time Correction** toolbar allows you to change the **A0** and **A1** coefficients, Reference Time and the Continuous Week Number. Double-click a highlighted row to open the following dialog box:

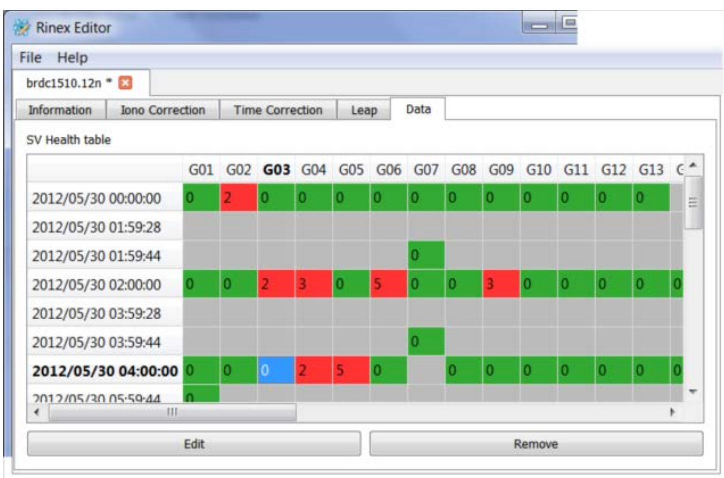


The Leap tab



Under the **Leap** tab you can change the **Leap seconds**, **Week number**, an **Day number**.

The Data tab



Under the **Data** tab you can change the health of satellites. In each cell, you can enter a number directly, or click the **Edit** button and select a value from the dropdown list:

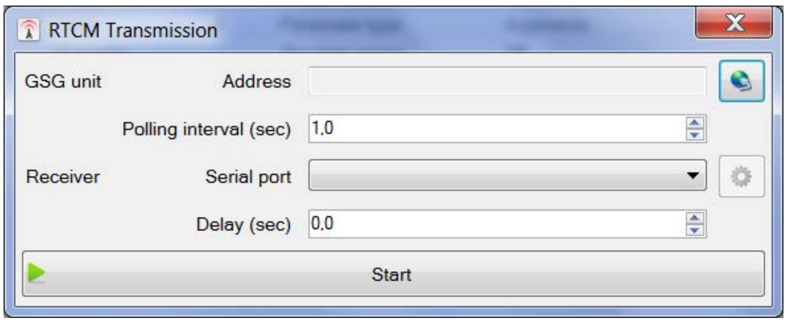
ALL DATA OK
PARITY FAILURE
TLM/HOW FORMAT PROBLEM
Z-COUNT IN HOW BAD
SUBFRAMES 1, 2, 3
SUBFRAMES 4, 5
ALL UPLOADED DATA BAD
ALL DATA BAD

To edit multiple cells at once, select the cells by holding down the Shift or Ctrl key while clicking, and then click the **Edit** button.

4.9.16 Transmitting RTCM Messages With StudioView

The **RTCM Transmission**[1] function allows to re-transmit RTCM message from the GSG unit to the external receiver. In order to use this function, a virtual **Base Station** has to be configured (see also: "Base station" on page 67) and used in your scenario, and the external receiver has to be connected to the serial port.

1. To open the RTCM Transmission window in StudioView, navigate to **Tools > RTCM Transmission** or click .



2. Click to open the Connections Manager tool (for details, see "Connecting StudioView to GSG" on page 122.)
3. Click to configure the serial port, or use the dropdown menu.
4. If required, specify the **Delay** in seconds.
5. To start transmitting RTCM messages to the selected unit, click Start.

   The Stop button will appear, with a counter of all the RTCM messages transmitted.
6. To stop the transmission, click Stop .

See also these SCPI commands: "SOURce:SCENario:RTCM?" on page 298 , "SOURce:SCENario:RTCMCFG?" on page 299 and "SOURce:SCENario:RTCMCFG" on page 299.

[1]Radio Technical Commission for Maritime Services

CHAPTER 5

Reference

This Chapter includes reference information, such as listings of default settings, logs, protocols, file formats and error messages.

The following topics are included in this Chapter:

5.1 The GSG Web UI

Orolia GSG Series simulators feature a Web-based user interface (throughout this documentation referred to as "**Web UI**"), accessible via a standard Web browser (e.g., Mozilla Firefox or Internet Explorer) installed on a computer with access to the same network to which your GSG unit is connected.

From the GSG Web UI you can perform operations remotely via HTTP as you would directly from the unit, and access product technical support documents and materials.

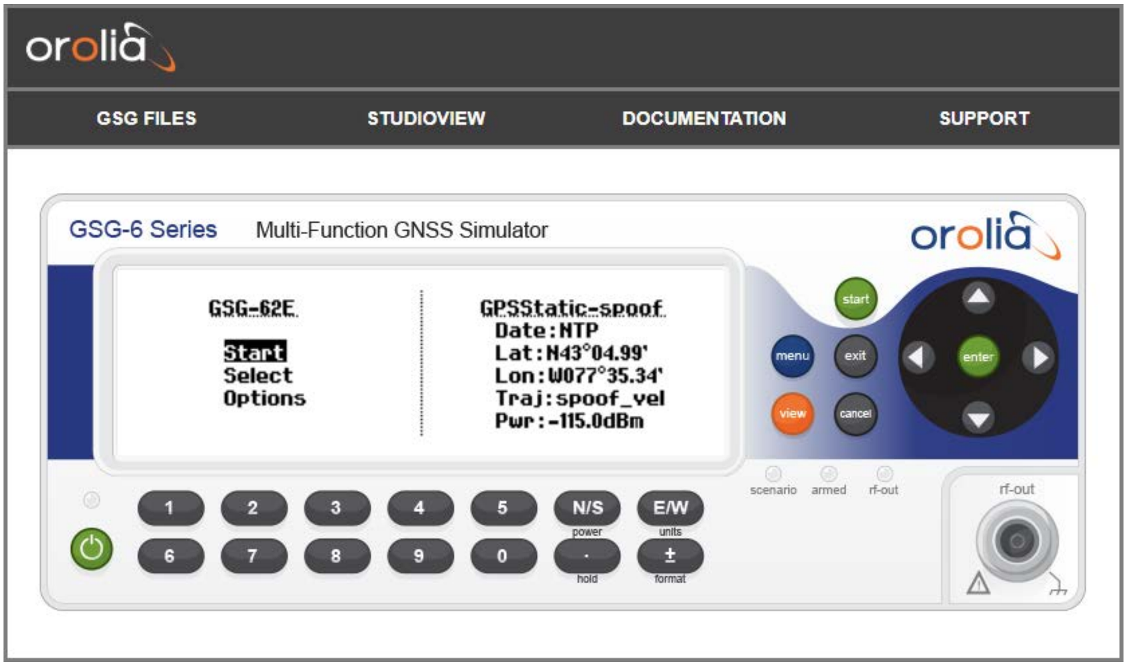


Figure 5-1: GSG-6 Web UI

Note: The information and text displayed on your computer screen will vary depending upon the configuration of your GSG unit.

For more information, and instructions on how to access the Web UI, see "Accessing the GSG Web Interface" on page 115.

5.2 Messages

Below is a listing of messages, as they may appear in message dialog boxes. The text below each message explains its meaning, context and – if applicable – suggested remedial action.

Could not initialize the keyboard.

A possible hardware issue exists. Please contact service.

Could not initialize web interface.

A possible firmware issue exists. Re-install firmware. If problem persists, then contact service.

Scenario is modified. Do you want to save changes using a different name?

Scenario configuration has been modified. “Yes” allows you to specify a new scenario name to save it under. If “No” is selected, the current scenario name is used to save the configuration. “Cancel” does not start scenario execution and returns to previous menu.

Problem in scenario configuration. Please edit coordinates.

Invalid coordinates given for scenario.

Problem in scenario configuration. Please edit dates.

Invalid date or duration given for scenario.

Problem in scenario configuration.

General problem in scenario configuration when it is read into memory.

Error in scenario: if duration is 'forever', ephemeris shall be 'Download’.

When setting duration to be “forever”, then navigation data/ephemeris setting must be on “Download”.

Speed or attitude above regulation limits, aborting scenario execution.

Scenario execution has encountered values that are too high. Simulated speed and altitude are limited in non-export GSG versions. Speed limit is 520 m/s and altitude is 18470 m. Change the scenario configuration.

No valid navigation data available.

Please review scenario settings. Restart GSG-5/6.

Can't generate Subframes.

Please review scenario settings. Restart GSG-5/6.

Are you sure you want to restore factory defaults?

Confirmation request before restoring factory defaults.

IP address configuration failed.

Configuring the Internet Protocol address failed. Please check your network settings and try again.

Cannot save unit parameters.

Saving of settings failed. Try restarting the device and saving parameters again. If it still fails, then please contact service.

Cannot load unit parameters.

The device was not able to use the configured parameters. Defaults are used in this case. Please go to the Menu and set the parameters again and store them with EXIT button.

Cannot load unit calibration data.

Unit is not calibrated, or calibration data is corrupted. The device should be re-calibrated. Please contact service. Device can be used, but the observed power levels may differ from the ones shown in display.

Cannot save unit calibration data.

Manual calibration data could not be saved. Restart unit and try again. If saving is still not possible, contact service.

Problem with the license file.

The License file is corrupted. The device will work, but only as GSG-51. Contact service.

Are you sure you want to calibrate the unit?

Confirmation for user calibration.

Password is invalid.

Calibration password entered is invalid. Try again.

Save calibration?

After manual calibration you can choose to save or not save the values.

Could not start signal.

Unexpected problem occurred. Please restart the GSG-5/6.

Please check start date!

GPS Start date is invalid in scenario configuration or signal generator. (Note: The earliest allowed start date is 6.1.1980.) Please check the date and correct it.

Invalid Rinex file selected.

Check that selected navigation data file is a valid RINEX file (only version 2.1 and upwards supported).

Please check navigation data.

Navigation data is not valid.

No ephemeris for this PRN/date.

In the signal generator, navigation data is not found for the selected PRN and/or date. Try again with different values.

File missing.

The navigation data file was missing when starting the signal generator. Select another option in the ephemeris list and try again.

No reference clock detected.

External 10 MHz input is enabled, but no signal is detected. Connect the reference signal. If 10 MHz input is disabled and you still receive this message, contact service.

Too old firmware!

Mismatch detected between firmware components. Try to update the firmware. If you still receive the message contact service.

No text entered.

When giving name for a file, the name is empty. Please enter a file name.

Cannot copy directory.

Copying a directory is forbidden.

Cannot delete directory.

Removing a directory is forbidden.

Cannot rename directory.

Renaming directory is forbidden.

No manageable files available.

If this happens the device is faulty. Please contact service.

Do you want to delete the file?

Confirmation request when removing file.

Cannot delete file.

Removing of file fails.

Cannot rename file.

Renaming a file failed. This condition may occur when there is no free storage space. Try removing any unnecessary files, and rename the file again.

Cannot copy file.

Copying a file failed. This condition may occur when there is no free storage space. Try removing any unnecessary files, and copy the file again.

File already exists.

File is copied or renamed over an existing file. You can choose if you want to overwrite it or not.

File is in use.

The file for a given file manager operation (Copy, Rename or Delete) is performed is in use. You can choose whether to continue the operation or not. If the scenario in use is deleted, the current scenario becomes “None”.

No scenario selected.

This happens when the current scenario is “None” and scenario execution is started. Select a scenario to be executed.

Scenario failed to start.

Please review scenario settings. Restart GSG-5/6.

Could not start scenario.

Restart GSG-5/6.

No scenarios available.

This message appears when there are no scenarios in the scenarios directory. Reset factory defaults to restore the default scenarios, or transfer your own scenarios from a PC using the StudioView software.

Could not start data loading.

An unexpected problem has occurred while loading navigation data. Try scenario again. If problem persists, then contact service.

Could not download SBAS data.

An unexpected problem has occurred while downloading and/or loading SBAS data. Try scenario again. If problem persists, then contact service.

Problem opening trajectory file.

Please review scenario settings. Restart GSG-5/6.

Cannot save scenario configuration.

Saving of scenario settings failed. Try restarting the device and saving scenario configuration again. If it still fails, then please contact service.

Available disk space too low xx% free.

Navigation/SBAS data download detected that the free storage space was too low. Please use **Options** > **Manage files** to free some space.

Invalid scenario field *field* - > modified.

After reading in scenario file it was detected that field is invalid, therefore its value has been “capped” or set to a default valid value (for details, see "Scenario File Format" on page 191.)

5.3 Timing Calibration

GSG units with the **Timing Calibration Option** (OPT-TIM) have an additional timing calibration file installed, named `pps.cal`.

Provided the option is enabled, you can access and modify this file via the StudioView **File Manager** (> **Tools** drop-down menu), or the Web UI under the menu item **GSG FILES**, or by using **SCPI file management** commands (see "MMEMory:COPY" on page 321).

> **Note:** Restoring factory defaults on the unit will also reset this file to the factory default for the unit.

File Format "`pps.cal`"

[boardid]

FREQ_BAND offset

Boardid-board0, board1, board2, board3

FREQ_BAND - GPS_L1, GPS_L2, GPS_L5, GLO_L1, GLO_L2, GLO_L3

Example file for a GSG-64 with GPS L1 and L2:

[board0]
GPS_L1 -40.0E-9

GPS_L2 -40.0E-9

[board1]

GPS_L1 -60.0E-9

GPS_L2 -60.0E-9

[board2]

GPS_L1 -60.0E-9

GPS_L2 -60.0E-9

[board3]

GPS_L1 -60.0E-9

GPS_L2 -60.0E-9

5.4 NMEA Logging

GSG offers the possibility to log a scenario's execution in NMEA data. Every second a "snapshot" of the user and satellites' status is taken and recorded in the form of 3 standard NMEA sentence types.

- **RMC** sentence describes essential GPS position, velocity and time
- **GGA** sentence describes essential fix data, providing 3D location and accuracy data. Height of geoid above WGS84 ellipsoid is being approximated according to EGM96 geoid model and fix quality defaults to GPS fix (SPS).
- **GSV** sentences (1 to 4 depending upon the number of SV's) describe the actual satellites in view.
  NOTE: In GSV sentences, an SV's SNR estimate is given based on the following:

  When noise is ON:

  SNR = CNo - NF

  When noise is OFF:

  SNR = min (56, Channel Power + BN - NF - Lc)

  Where NF is ("Noise Figure" of receiver) = 1,

  Lc (cable loss) = 1,

  and BN (background noise level) = 174 dB

You can access this "snapshot" by using the SCPI command `SOURce:SCENario:LOG?`, which can be queried at a maximum rate of 1Hz. See the GSG SCPI Guide ("Command Reference" on page 227) for more details.

It is also possible to use StudioView to log NMEA data: **Tools** > **Data Recorder**.

5.5 Execution Log

During everyday operation, your GSG unit will maintain a log, which is kept in the file `observations/executionlog.txt`. In this log file, the unit stores information about the scenarios run, and possible errors that may have occurred, which can be helpful with troubleshooting.

To view the Execution log:

- On the GSG front panel display, navigate to **Options** > **Manage files** > **observations**: **executionlog.txt**. Press the Arrow right key to highlight the **View** option, and press enter to display the log.
- In GSG StudioView, use the **File Manager**.

» Using the GSG Web UI on a connected PC, click on **GSG FILES**, then navigate to **observations**, and click on **executionlog.txt**.

The maximum size of the execution log file is 20,000 lines. Once the limit is reached, the oldest entries will be overwritten by new data.

Please note that the log is not updated in real time, but is updated when a scenario stops, for example.
Also note that the log will be deleted when a **Clean & Restore** operation is performed, i.e. when restoring the unit to factory default configuration (**Options** > **Reset to factory defaults**).

5.6 Saving RINEX Data

[For more information on theRINEX data format, refer to e.g., Wikipedia: RINEX.]

Saving RINEX Observation Data

It is possible to store RINEX observations of a running scenario. This feature can be enabled using the SCPI command:

`SOURce:SCENario:OBS <start>,<duration>,<interval>`

`<start>` Specifies the number of seconds since scenario start to expire before starting to log observations. Using the parameter '-1' will start the logging of a running simulation immediately.

`<duration>` The observation period in seconds. Using the parameter '-1' will log until the end of the scenario.

`<interval>` Is the sample rate.

After the duration/hour is passed or the scenario is stopped, the observations are saved to:

`/observations/scenarioNameYYYYMMDDHHMMSS.obs`

Each file can contain at maximum 1 hour of data. The generated files can be retrieved via **GSG StudioView**, the Web UI, or by using SCPI commands (see "MMEMory:COPY" on page 321). Additionally, a link named `/observations/latest.obs` points to the latest generated file.

Observation files supported include GPS, GLONASS, Galileo, BeiDou and mixed. The RINEX file can be used in post-processing with the navigation data obtained from receiver.

Saving RINEX Navigation Data

It is also possible to log RINEX navigation data for a running scenario. This feature can be enabled using the SCPI command:

`SOURce:SCENario:NAV <ON|OFF>`

After each six hours or when the scenario is stopped, the navigation data is saved to:

`/observations/scenarioNameYYYYMMDDHHMMSS.nav`

The generated files can be retrieved via **GSG StudioView**, the Web UI, or by using SCPI commands (see "MMEMory:COPY" on page 321).

Additionally, a link named `/observations/latest.nav` points to the latest generated file.

The files are in RINEX 3.0.2 mixed format. As the unit generates new **navigation data sets** quite rarely, it is recommended that navigation data logging is enabled before starting a scenario. Navigation data logging is turned off when scenario stops.

5.7 YUMA Almanac File

During scenario start-up, the GSG simulator generates a GPS constellation almanac in the Yuma format. The generated file is named `observations/alm_gps.txt` and can be retrieved from the **observations** folder, using:

- the StudioView **File Manager**
- the GSG **Web UI**
- the **SCPI command** set.

> **Note:** This file will be overwritten every time a new scenario is started, so only the YUMA file for the last run scenario will be available.

The almanac file will be empty, if GPS satellites were not included in the latest scenario executed.

The following is an example of one entry in the almanac file:

```
******** Week 755 almanac for PRN-01 ********
ID: 01
Health: 000
Eccentricity: 5.8191653807E-04
Time of Applicability(s): 233472.0000
Orbital Inclination(rad): 0.9597571420
Rate of Right Ascen(r/s): -8.1828408482E-09
SQRT(A) (m 1/2): 5153.650309
Right Ascen at Week(rad): 1.2710842193E+00
```

```
Argument of Perigee(rad): 1.117132574
Mean Anom(rad): -3.0896651067E+00
Af0(s): 1.7600243882E-04
Af1(s/s): 1.3415046851E-11
week: 755
```

5.8 RLS (Return Link Service)

As part of the Cospas-Sarsat System, Galileo satellites are capable of picking up emergency signals emitted on 406 MHz by distress beacons, and transmitting a signal back to the beacon via the E1 frequency to confirm receipt of the distress signal. This technology has been developed under the international Cospas-Sarsat program. It comprises the Galileo-enabled distress beacons, the SAR payload on the Galileo satellites, as well as ground-based receiving stations (LUTs) and Mission Control Centers.

If the option "Galileo" (Opt-GAL) is installed, GSG is capable of simulating the Return Link Message (RLM) transmitted by the Galileo signal to the distress beacon.

As a user you interact with the GSG by inputting the RLM parameters as a SCPI command into the computer; GSG will then take this message and simulate its transmission on the E1 frequency from the satellite to the distress beacon.

For more information on SCPI command syntax, and examples, see "SOURce:SCENario:RLM" on page 300.

5.8.1 SAR Data

Each Return Link Message encapsulated in an SAR data page contains the following data:

- **Beacon ID** (60 bits): This field is used by the beacon to discern whether the RLM received is, indeed, addressed to itself or to some other beacon.
- **Message code** (4 bits): Defines the message type.
- **Parameters field** (16 bits for the short RLM, 96 bits for the long RLM). This field provides the information that SAR operators wish to send to the Galileo-equipped beacon.

Short RLMs are used to provide the activated Galileo-equipped beacon with a short acknowledgement of various kinds of commands (e.g., to reduce its transmission rate).

Long RLMs are intended for more complex commands in which several parameters may be required (e.g., to contain operational information or the coordinates of a location).

For more information on SCPI command syntax and examples, see "SOURce:SCENario:RLM" on page 300.

5.8.2 Requirements

In order for the RLM simulation to work with GSG, the following pre-requisites must be met:

- The Galileo-Option OPT-GAL must be present
- GSG-5 or -6, with 8 channels (or 16 channels if GPS + GAL is required)
- Firmware version 6.6.1 or greater
- StudioView software Version 4.6.1.3 or greater

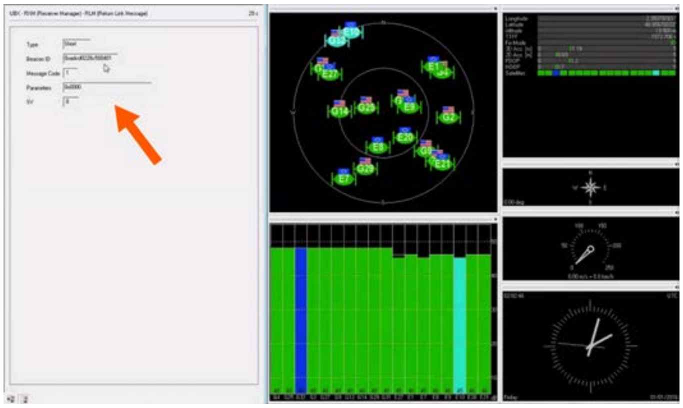
5.8.3 Simulating RLMs

The following is a brief outline how to setup a basic test system:

1. Using a GSG unit, and a GNSS receiver, start a scenario that is configured with GAL signals e.g., **GPSGALStatic**.
2. Open StudioView on a Personal Computer (PC).
3. Open the Console tool (Tools > Console).
4. Connect the PC to the GSG unit by clicking the small GLOBE with PLUG icon on the left (or click Refresh if a connection has been established before).
5. Issue the RLM command(s), using the Console e.g.:

```
write SOUR:SCEN:RLM 0,8,711888,141509,1025,65536
```

6. The submission should be confirmed: `OK [no errors]`.
7. View the received message on the receiver, see the following example:



For automated testing e.g., when integrating GSG into a larger test system, a SCPI program can be written. In this case, StudioView would not be needed.

5.9 Galileo E6-B/C Signal

GSG's optional Galileo E6-B/C signals are pseudo-signals for the Galileo Commercial Service (CS). Both signals are sent unencoded: A secondary code (pilot tone) is sent on the E6-C frequency band, with no data. On the E6-B band dummy data is sent at 448 bps.

To utilize this optional functionality a GSG-6 unit with an available frequency band is required, as well as the **OPT-GAL** and **OPT-L6** option packages.

5.10 Default Settings

The factory settings are considered default settings. You can restore these settings at any time by navigating to **Options** > **Reset to factory defaults**.

The default settings are:

- **Transmit power**: -125 dBm
- **External attenuation**: 0 dB
- **Interface type**: Ethernet
- **Obtain IP automatically**: Yes
- **GPIB address**: 01
- **Use 10 MHz input**: No
- **Simulate Noise**: No
- **RequiredCN0**: 44
- **NoiseBW**: 20.46
- **NoiseOffset**: 0
- **Scenarios**: See "Pre-Installed Scenarios" below.
- **Trajectories**: See "Trajectories" on page 43.
- **Events**: See "Event Data" on page 54.

5.11 Pre-Installed Scenarios

GSG-5/6 units are shipped with a set of predefined scenarios and supporting files. Which files are installed on your unit depends on the model and options purchased.

Listed below is a selection of scenarios, as they are installed on many GSG units.

The output power of these scenarios has to be set by the user (**Options** > **Transmit power**).

More advanced scenarios, events, and trajectories can be found in the StudioView repository.

Table 5-1: Scenarios

Model	Scenario	Position	Start Time
All	GPSStatic	N043° 04' 59.257",W077°35'20.674"	01/07/2022 14:00
GSG-5/54/55/56/62/63/64 OPT-TRAJ	GPSCircle	N043° 04' 59.257",W077°35'20.674"	01/07/2022 14:00
GSG-5/55/56/62/63/64 OPT-SBAS	GPSStaticSBAS	N043° 04' 59.257",W077°35'20.674"	01/07/2022 14:00
GSG-53/56/62/63/64 OPT-GLO	GPSGLOStatic	N60°27'24.41", E42°7'24.42"	01/07/2022 9:00
GSG-56/62/63/64 OPT-GLO OPT-TRAJ	GPSGLOCircle	N60°27'24.41", E42°7'24.42"	01/07/2022 9:00
GSG-5/62/63/64 OPT-GAL	GPSGALStatic	N48°51'24.12", E002°21'2.88"	01/07/2022 02:00
GSG-5/62/63/64 OPT-GAL OPT-TRAJ	GPSGALCircle	N48°51'24.12", E002°21'2.88"	01/07/2022 02:00
GSG-5/62/63/64 OPT-BDS	GPSBDSStatic	N22°16'41.88", E114°9'32.04"	01/07/2022 02:00
GSG-5/62/63/64 OPT-BDS OPT-TRAJ	GPSBDSCircle	N22°16'41.88", E114°9'32.04"	01/07/2022 02:00
GSG-62/63/64 OPT-L2	GPSL1L2Pcode	N60°27'24.41", E42°7'24.42"	01/07/2022 15:00
GSG-62/63/64 OPT-L2	GPSL1L2pseudoY	N60°27'24.41", E42°7'24.42"	01/07/2022 15:00
GSG-62/63/64 OPT-L2 OPT-GLO	GPSGLOL1L2pseudoY	N60°27'24.41", E42°7'24.42"	01/07/2022 15:00
GSG-5/55/56/62/63/64 OPT-MP	GPSMP2Static	N10°39'0.00", W061°27'0.00"	01/07/2022 10:02
GSG-5/55/56/62/63/64 OPT-MP	GPSMP4Static	N39°54'0.00", E116°24'0.00"	01/07/2022 10:02
GSG-5/55/56/62/63/64 OPT-INTF	GPSINTFStatic	N43°4'59.25", W077°35'20.67"	01/07/2022 09:20
GSG-5/55/56/62/63/64 OPT-INTF OPT-GLO	GPSGLOINTFStatic	N60°27'24.41", E042°7'24.42"	01/07/2022 09:20

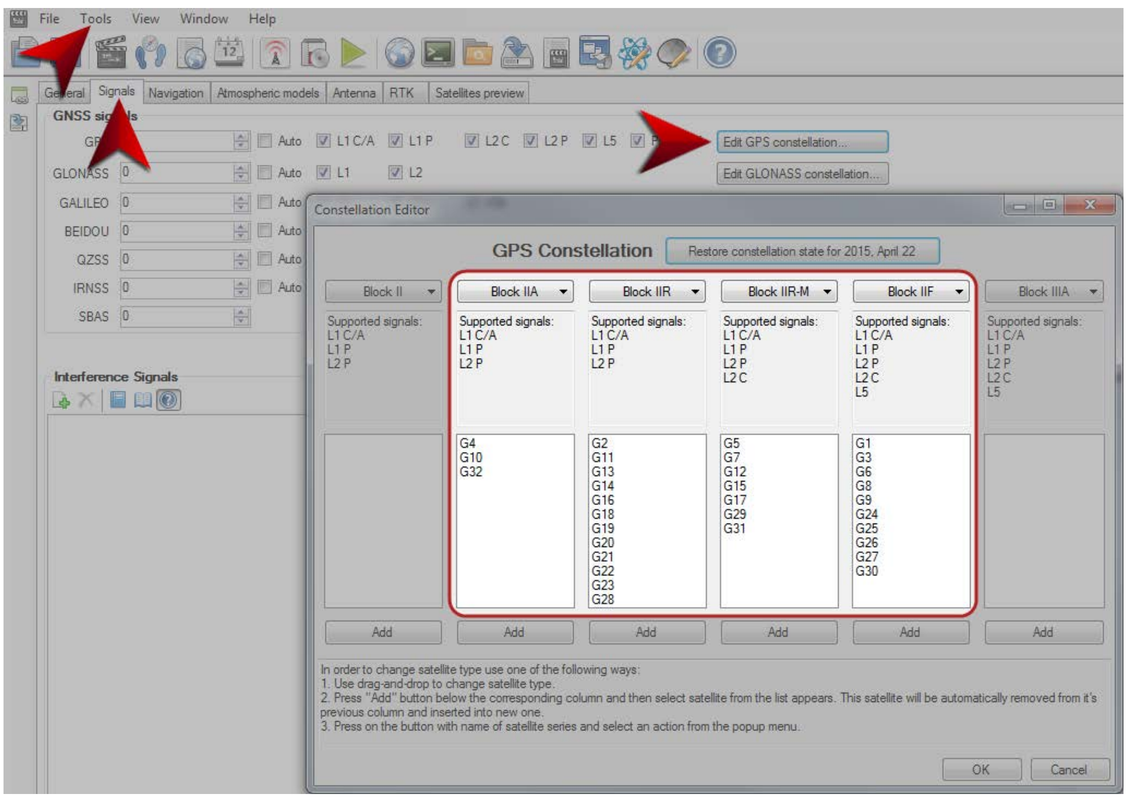
5.12 Default Scenario Satellites

As of spring 2015, the default **GPS constellation** consists of the following active satellites:

- 3 x Block IIA satellites
- 12 x Block IIR satellites
- 7 x Block IIR-M satellites
- 10 x Block IIF satellites

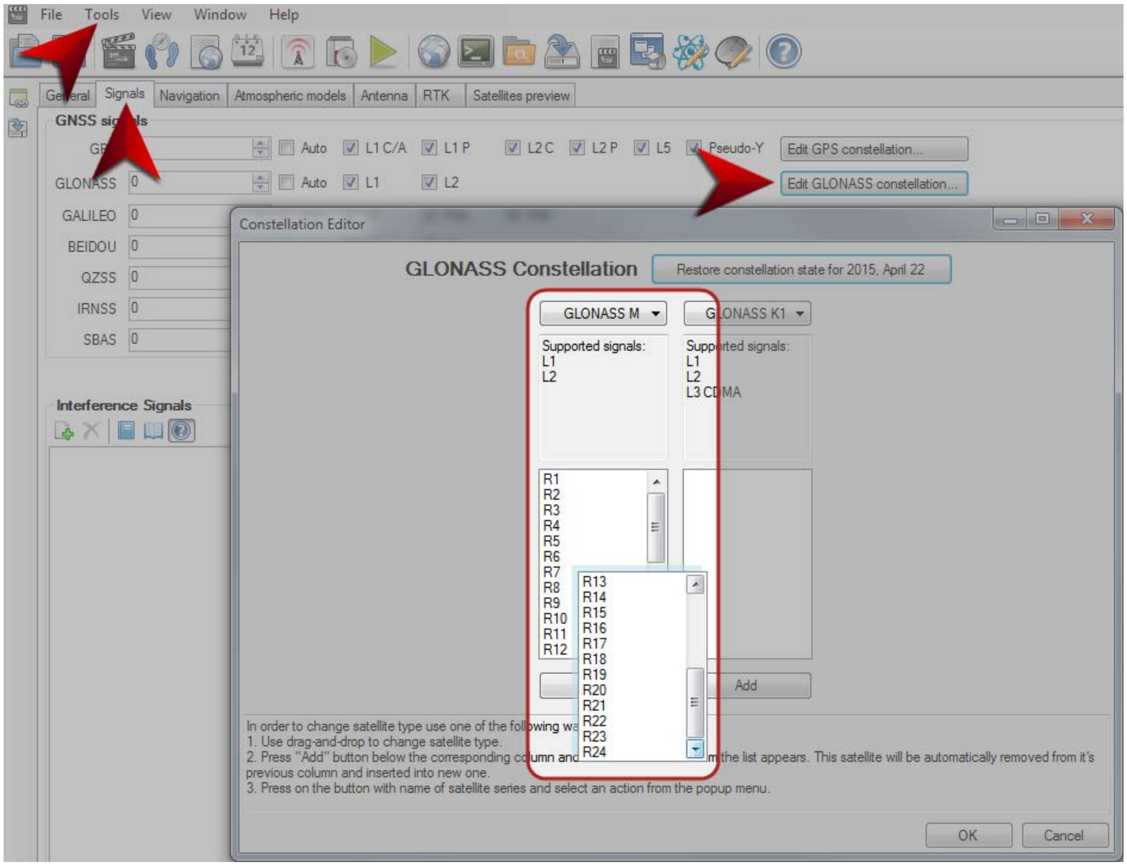
GSG uses this constellation as the default for its scenarios. You can, however, change this when developing or manipulating scenarios, e.g., to simulate an event in the past.

The following illustration shows the **Constellation Editor** window with the default satellite types.
The window can be accessed in **Studioview**, via the **Tools** menu > **Scenario Editor** > **Signals** tab > **Edit GPS constellation**... button:



5.12.1 GLONASS Default Satellite Types

All satellites default to the GLONASS-M type. Alternatively, you can select GLONASS K1.



5.13 Scenario File Format

Scenarios are defined by means of text files which contain a set of keywords and values, as described below.

Scenario files used with GSG-5/6 units must follow the described format. All fields are optional and will assume default values if not provided. Any specified invalid values will be modified such that any field will be within its own valid range.

Scenario file keywords:

StartTime `MM/DD/YYYY HH:MM:SS 0|1`

Duration `DAYS HOURS MINUTES 0|1|2`

Ephemeris `FILENAME[,FILENAME*] | Default | Download`

EventData `FILENAME | None`

Environment `FILENAME | None`

Vehicle FILENAME | None

GpsSatellites INTEGER

GlonassSatellites INTEGER

GalileoSatellites INTEGER

BeiDouSatellites INTEGER

QZSSSatellites INTEGER

IRNSSSatellites INTEGER

Startpos DECIMAL degN DECIMAL degE DECIMAL m

BaseStationPos DECIMAL degN DECIMAL degE DECIMAL m

UserTrajectory FILENAME [0|1] | Static | 3GPP | Circle

LeverArm DECIMAL DECIMAL DECIMAL

DeltaLSF -1|0|1

TrajectoryParameters DECIMAL DECIMAL 1|-1

AntennaModel FILENAME | Zero model | Patch | Helix | Cardioid | GPS-703-GGG

IonoModel FILENAME[,FILENAME*] | On | Off

TropoModel Saastamoinen | Black model | Goad&Goodman | STANAG | Off

Temperature DECIMAL

Pressure DECIMAL

Humidity DECIMAL

ElevationMask DECIMAL

SBASSatellites INTEGER | <comma-separated list of SBAB satellite IDs>

SBASL1 [0|1]

SBASL5 [0|1]

DefaultGpsSV Default| SvBlockII | SvBlockIIA | SvBlockIIR | SvBlockIIR-M | SvBlockIIF | SvBlockIIIA

DefaultGlonassSV Default | SvGlonassM | SvGlonassK1

SvBlockII Space delimited list of SVID's

SvBlockIIA Space delimited list of SVID's

SvBlockIIR Space delimited list of SVID's

SvBlockIIR-M Space delimited list of SVID's

SvBlockIIF Space delimited list of SVID's

SvBlockIIIA `Space delimited list of SVID's`

SvGlonassM `Space delimited list of SVID's`

SvGlonassK1 `Space delimited list of SVID's`

GPSL1CA `1 | 0`

GPSL1P `1 | 0`

GPSL2P `1 | 0`

GPSL2C `1 | 0`

GPSL5 `1 | 0`

GPSPY `1 | 0`

GLOL1 `1 | 0`

GLOL2 `1 | 0`

GALE1 `1 | 0`

GALE5a `1 | 0`

GALE5b `1 | 0`

BDSB1 `1 | 0`

BDSB2 `1 | 0`

QZSSL1CA `1 | 0`

QZSSL2C `1 | 0`

QZSSL5 `1 | 0`

QZSSL1SAIF `1 | 0`

IRNSSL5 `1 | 0`

RandomMpCP `1 | 0`

MultipathSignals `INTEGER`

[**MultiPathSignal** `INTEGER`]

mpChannel `INTEGER`

rangeOffset `DECIMAL`

rangeChange `DECIMAL`

rangeInterval `DECIMAL`

dopplerOffset `DECIMAL`

dopplerChange `DECIMAL`

dopplerInterval `DECIMAL`

powerOffset `DECIMAL`

powerChange DECIMAL

powerInterval DECIMAL

InterferenceSignals INTEGER

[**InterferenceSignal** INTEGER]

GPSL1CA 1 | 0

GPSL1P 1 | 0

GPSL2P 1 | 0

GPSL2C 1 | 0

GPSL5 1 | 0

GPSPY 1 | 0

GLOL1 1 | 0

GLOL2 1 | 0

GALE1 1 | 0

GALE5a 1 | 0

GALE5b 1 | 0

BDSB1 1 | 0

BDSB2 1 | 0

QZSSL1CA 1 | 0mode 0 | 1 | 2 | 3

SatId INTEGER

Power INTEGER

FreqOffset INTEGER

JammerPosition DECIMAL degN DECIMAL degE DECIMAL m | Not set

StartOffset DECIMAL

EndOffset DECIMAL

SweepTime INTEGER

RtcmConfig 3x,INTEGER [,INTEGER*]

GpsToUtcOffset Auto | Rinex | <INTEGER>

Keyword Parameters:

Scenario File Keyword	Parameter Value	Comment
StartTime	Valid date in the format: MM/DD/YYYY HH:MM:00 Source Valid range limited to: MIN GPS: 00:00 on 6th of January 1980 MIN GLONASS: 00:00 on 1st of January 1996MAX: 23:59 on 31st of December 2100 Source: [0, 1]	Start time in the GPS Time time frame. Note that seconds must be set to zero. When scenario ephemeris data is set to Download, then StartTime must be in the past (typically with a 1-day added margin) to allow the data to be available for downloading. The optional Source value defines where the simulation gets its Start Time. The default value is set. 0 – Set (fixed value) 1 – NTP (current time)
Ephemeris	Keyword or filename. Available keywords: {'Default', 'Download'}	Default indicates that the unit will re-use internally available files to build a navigation data. 'Download' means that the unit attempt to download the data from ftp site. Default: 'Default' Filenames are used to specify RINEX and GPS/QZSS YUMA Almanac navigation files. Several files are separated by comma.YUMA almanac files are identified by the .alm case- insensitive file extension. The old keyword *NavigationData* has the same meaning as keyword *Ephemeris* and is accepted for backward compatibility.
Duration	DAYS HOURS MINUTES REPEAT where the values are INTEGER values with the following ranges; DAYS: [0, 31] HOURS: [0, 23] MINUTES: [0, 59] REPEAT: [0,2]	The REPEAT value indicates what the scenario will do once scenario has reached its end. 0 – stops 1 – re-starts 2 – forever. The 'forever' option (2) requires *Ephemeris* to be set to *Download*.
GpsSatellites	[-1, 5], for GSG-52/53 [-1, 8], for GSG-54 [-1, 16], for GSG-55 [-,16], for GSG-56	Maximum number of signals in view at any given time. Keyword '-1' implies 'Auto' – maximum number of Satellites in view to be simulated Default: 5/8/16 depending on GSG model The old keyword *NumSignals* is accepted and interpreted as *GpsSatellites*.
GlonassSatellites	[-1, 5], for GSG-53 [-1,16], for GSG-56	Default: 0 Keyword -1 implies 'Auto' – maximum number of Satellites in view will be simulated.
GalileoSatellites	[-1, 36]	Default: 0 Keyword -1 implies 'Auto' – maximum number of Satellites in view will be simulated.

Scenario File Keyword	Parameter Value	Comment
BeiDouSatellites	[-1, 37]	Default: 0 Keyword -1 implies 'Auto' – maximum number of Satellites in view will be simulated.
QZSSSatellites	[-1, 4]	Default: 0 Keyword -1 implies 'Auto' – maximum number of Satellites in view will be simulated.
IRNSSSatellites	[-1, 7]	Default: 0 Keyword -1 implies 'Auto' – maximum number of Satellites in view will be simulated.
Startpos	latitude: [-89.999999, 89.999999] longitude: [0,360] altitude: [0.0, 18,240.0]	The Startpos position is specified in latitude north, meaning that a latitude south is reached by using a minus sign. Latitude values span from -90 to 90. The longitude is always specified as an east coordinate, ranging from 0 to 360. The coordinates must be written as DD.dddddddd in this file. Altitude is specified in meters, and can be specified up to decimeter level. Maximum altitude can be increased with 'extended limits' option, raising it to 20200km.
BaseStationPos	(as for 'Startpos')	(as for 'Startpos')
UserTrajectory	Keyword or filename (in NMEA or RSG format). Available keywords: {'Static', 'Circle', '3GPP' } When a file is selected, additional looping parameter specifies trajectory execution method. Valid values: {0, 1}	Default: 'Static' With NMEA trajectories, the optional Looping parameter will define the how the trajectory will be executed. Default looping status is 0; execute once. 0 – Execute once 1 – Loop continuously
TrajectoryParameters	DECIMAL DECIMAL 1|-1 The first DECIMAL value specifies circle diameter in meters. Valid range: [0.0, 1 000 000.0] The second DECIMAL value specifies speed in meters /second. Valid range: [0.0, 1000.0] The third value specifies direction. Valid values: {-1, 1}	The parameter is (only) required if UserTrajectory is 'Circle'. For the direction values '-1' is interpreted as anti-clockwise and '1' as clockwise.

Scenario File Key-word	Parameter Value	Comment
IonoModel	Keyword of comma-separated filenames Available keywords:[Off, On] 'Off' = 'Off' (in Graphical user interfaces) 'On' ='Klobuchar' (in Graphical user inter-faces)	Default: On
Tropo Model	{'Saastamoinen' | 'Black model' | 'Goad&Good-man' | 'STANAG' | 'Off' }	Selected tropospheric model. Default: 'Saastamoinen'
Temperature	[-99,99]	Tropospheric model's temperature. Default: 20. [In degrees Celsius.]
Pressure	[800,1200]	Tropospheric model's pressure. Default: 1000. [In mBar]
Humidity	[1,100]	Tropospheric model's relative humidity. Default: 50. [In percentage.]
SBASSatellites	[0,3] <list of SBAS satellite IDs>	Maximum number of SBAS channels in view at any given time. Default: 0 Satellite IDs are determined automatically depend-ing on start position. Instead of an integer, the user can specify a comma-separated list of desired satel-lite IDs, e.g. "S131, S133"
AntennaModel	Keyword or filename Available keywords: {'Zero model', 'Helix', 'Patch', 'Cardioid', 'GPS-703-GGG'}	Selected antenna model. Default: 'Zero model'
EventData	Filename or 'None'	Selected Event file. Default: 'None'
ElevationMask	[-10.0, 89.0]	Minimum allowed elevation of an SV in degrees. Default: 0.0
DeltaLSF	[-1,1]	An integer signaling future leap second change of ±1. Default: 0
GpsToUtcOffset	Auto Rinex [0; 30]	Auto - initial GPS-to-UTC offset is determined by scewnario start time. Rinex - initial GPS-to-UTC offset is determined by RINEX files attached to scenario. Any integer [0; 30] - GPS-to-UTC offset is this spe-cific value.

Scenario File Keyword	Parameter Value	Comment
DefaultGpsSV	{'Default', 'SvBlockII', 'SvBlockIIA', 'SvBlockIIR', 'SvBlockIIR-M', 'SvBlockIIF', 'SvBlockIIIA'}	Defines the default GPS satellite series simulated, would the ID not explicitly be specified to be of a different type. Default: 'Default'
DefaultGlonassSV	{'Default', 'SvGlonassM', 'SvGlonassK1' }	Defines the default GLONASS satellite series simulated, would the ID not explicitly be specified to be of a different type. Default: 'Default'
{'SvBlockII', 'SvBlockIIA', 'SvBlockIIR', 'SvBlockIIR-M', 'SvBlockIIF', 'SvBlockIIIA' }	Example: DefaultGpsSv SvBlockIIR SvBlockIIA G10 G14 SvBlockIIR-M G17	A space delimited list of Satellite ID specifying what satellites are mapped to a non-default satellite series. For GPS the Satellite ID is built up by the letter 'G' followed by PRN number, e.g. G3. Default: not specified
{SvGlonassM', 'SvGlonassK1' }	Example: DefaultGlonassSV SvGlonassK1 SvGlonassKM R10 R11 R14	A space delimited list of Satellite ID specifying what satellites are mapped to a non-default satellite series. For GLONASS the Satellite ID is built up by the letter 'R' followed by PRN number, e.g. G3. Default: not specified
GPSL1CA	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Default: 1 ('On')
GPSL1P	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Default: 1 ('On')
GPSL2P	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Default: 1 ('On')
GPSL2C	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Default: 1 ('On')
GPSL5	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Default: 1 ('On')
GPSPY	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Default: 1 ('On')
GLOL1	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Default: 1 ('On')
GLOL2	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Default: 1 ('On')

Scenario File Keyword	Parameter Value	Comment
GALE1	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Default: 1 ('On')
GALE5a	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Default: 1 ('On')
GALE5b	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Default: 1 ('On')
BDSB1	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Default: 1 ('On')
BDSB2	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Default: 1 ('On')
QZSSL1CA	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Default: 1 ('On')
QZSSL1SAIF	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Default: 1 ('On')
QZSSL2C	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Default: 1 ('On')
QZSSL5	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Default: 1 ('On')
IRNSSL5	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Default: 1 ('On')
SBASL1	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Default: 0 ('Off') if SBAS parameter is zero, otherwise 1 ('On')
SBASL5	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Default: 0 ('Off')
RandomMpCP	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Enables randomization of initial multipath Carrier Phase Offset value. Default: 0
MultipathSignals	[0, min(8, 16 – NumSignals)]	The number of multipath channels. Default: 0

Scenario File Keyword	Parameter Value	Comment
[MultipathSignal INTEGER]	INTEGER range: [1, MultipathSignals]	'Header' for the parameters for each multipath signal. When MultipathSignals is 1 or greater, then a set of parameters must be specified for each multipath signal.
mpChannel	[1, GpsSatellites + GlonassSatellites + GalileoSatellites]	Specifying which channel that is to be duplicated.Only GPS or GLONASS channels may be duplicated. Default: 1
rangeOffset	[-999.999, 999.999]	Specifying range offset in meters. Default: 0
rangeChange	[-99.99, 99.99]	Specifying range offset change in meters / rangeInterval. Default: 0
rangeInterval	[0, 600]	Specifying range change interval in seconds with one decimal accuracy. Default: 0
dopplerOffset	[-99.99, 99.99]	Specifying Doppler offset in meters/seconds. Default: 0
dopplerChange	[-99.99, 99.99]	Specifying Doppler offset change in meters/seconds / dopplerInterval. Default: 0
dopplerInterval	[0, 600]	Specifying Doppler change interval in seconds. Default: 0
powerOffset	[-30.0, 0.0]	Specifying power offset in dB. Default: 0
powerChange	[-30.0, 0.0]	Specifying power offset change in dB / powerInterval. Default: 0
powerInterval	[0, 600]	Specifying power change interval in seconds. Default: 0
InterferenceSignals		The number of interference channels. Default: 0
[InterferenceSignal INTEGER]	INTEGER range: [1, InterferenceSignals]	'˜Header'™ for the parameters for each interference signal. When InterferenceSignals is 1 or greater, then a set of parameters below must be specified for each interference signal.
GPSL1CA	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Specifies the type of signal interference
GPSL1P	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Specifies the type of signal interference
GPSL2P	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Specifies the type of signal interference

Scenario File Keyword	Parameter Value	Comment
GPSL2C	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Specifies the type of signal interference
GPSL5	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Specifies the type of signal interference
GPSPY	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Specifies the type of signal interference
GLOL1	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Specifies the type of signal interference
GLOL2	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Specifies the type of signal interference
GALE1	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Specifies the type of signal interference
GALE5a	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Specifies the type of signal interference
GALE5b	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Specifies the type of signal interference
BDSB1	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Specifies the type of signal interference
BDSB2	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Specifies the type of signal interference
QZSSL1CA	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Specifies the type of signal interference
QZSSL2C	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Specifies the type of signal interference
QZSSL5	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Specifies the type of signal interference
QZSSL1SAIF	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Specifies the type of signal interference

Scenario File Keyword	Parameter Value	Comment
IRNSSL5	[0, 1] , where 0 corresponds to 'Off' , and 1 to 'On'	Specifies the type of signal interference
mode	[0. 1, 2, 3] , where 0 is modulated,1 is unmodulated, 2 is sweep and 3 is noise	Specifies the type of signal interference
SatId	GPS: [1,32] GPS carrier: 0 Glonass: [1, 24] Glonass carrier: [-7,6] Galileo: [1, 36] SBAS: [120, 158] BeiDou: [1,37] QZSS: [1,4] IRNSS: [1:7]	Specifies the satellite ID (PRN / frequency slot) for interference signal Default: 3 For 'Glonass carrier', the term 'SatId' refers to the frequency slot.
Power	[-65, -160]	Signal strength in dBm. Default: unit's transmit power
FreqOffset	[-999999, 999999]	Frequency offset in Hz. Default: 0
JammerPosition	latitude: [-89.999999, 89.999999] longitude: [0,360] altitude: [0.0, 18,240.0] or 'Not set'	Position of the jammer. See StartPosition above.
StartOffset	[-40.000000,40.000000]	Start offset for sweeper or noise in MHz
EndOffset	[-40.000000,40.000000]	End offset for sweeper or noise in MHz
SweepTime	[4,20]	Sweep time in microseconds
RtcmConfig	3x, corresponding to RTCM 3.2 [1002 | 1004 | 1006 | 1010 | 1012 | 1033], any combination of one or more of these	Comma separated fields. First, field is RTCM version. Next, is a list of 1 or more message types, involving any combination of the supported types.
Environment	Filename or 'None'	Selected file with Environment model. Default: 'None'
Vehicle	Filename or 'None'	Selected file with Vehicle model. Default: 'None'
LeverArm	[-500.0,+500.0] [-500.0,+500.0] [-500.0,+500.0]	The XYZ offset for the antenna position versus the body mass center. Default: 0 0 0

5.14 GSG Series Model Variants and Options

Orolia GSG Series GNSS constellation simulators and signal generators are available in several different Model configurations, and with numerous Option packages, in order to allow for application-specific customization:

Hdwe Models	Channels	Channel Option	# of Simultaneous Frequencies	Upgrade	Constellations	Frequency Bands
GSG-51	1	–	1	–	GPS Included Optional constellations which can be ordered: • GLONASS (OPT-GLO) • Galileo (OPT-GAL) • BeiDou (OPT-BDS) • QZSS (OPT-QZ) - IRNSS (OPT-IRN)	GPS L1 C/A Included Included if constellation is ordered: • GLONASS L1 C/A • QZSS L1 • Galileo E1 • BeiDou B1
GSG-5	4	OPT-4	1	Software		
	8	OPT-8				
	16	OPT-16				
GSG-62	32	–	2	Factory OPT-32/2		Above included for ordered constellations. Options if constellation is ordered: • GPS L1P, GPS L2P, GLONASS L2 C/A (OPT-L2) • GPS L2C (OPT-L2C) • GPS L5, Galileo E5a/b, BeiDou B2 (OPT-L5)IRNSS
GSG-63	48	–	3	Factory OPT-48/3		
GSG-64	64	–	4	Factory OPT-64/4		

Figure 5-2: GSG options overview

5.14.1 Which GSG Model & Options Do I Have?

The **model** will be displayed in the top-left corner of the Main screen (startup screen).

To find out which **options** are installed on your GSG unit, navigate to **Options** > **Show System Information** > **Options**. Press enter to access this list:

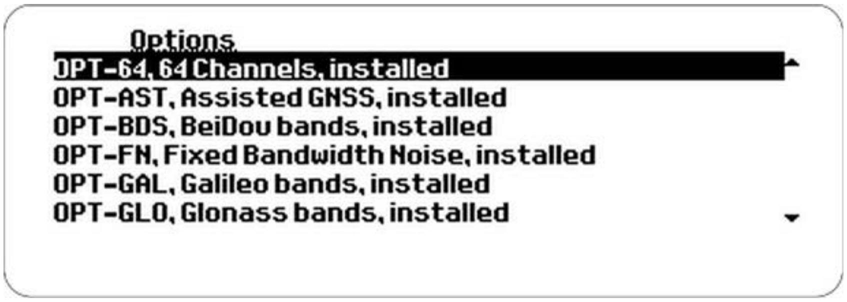


Figure 5-3: List of installed options

5.14.2 GSG Models & Variants

5.14.2.1 GSG-51 Series

- Single-Channel GPS Factory Tester

Options for GSG-5:

- GLONASS
- GALILEO
- BEIDOU
- QZSS
- Upgrade to 4-channel unit

5.14.2.2 GSG-5 Series

- **Base unit**: 4 channels, GPS L1 C/A

Options for GSG-5:

- Upgrade to 8, or 16-channels
- Upgradable to GSG-6 Series

Advanced Feature Set included:

- SBAS
- Trajectories

- Multipath
- White Noise
- Programmable PPS (1, 10, 100, 1000)
- StudioView software
- Antenna modeling
- Front Panel Lockout
- NTP Synchronization
- Arm and trigger
- Leap Second Simulation

5.14.2.3 GSG-6 Series

Multi-frequency, Multi-system GNSS constellation simulators:

- Up to 64 Channels and 4 simultaneous frequencies
- GPS L1 C/A
- Includes Advanced Feature Set of GSG-5

GSG-6 Model variants:

- **-62**: 32 channels and 2 freq bands
- **-63**: 48 channels and 3 freq bands
- **-64**: 64 channels and 4 freq bands

Options for GSG-6:

- Upgradeable to 48 channels and 3 simultaneous frequencies, and 64 channels and 4 simultaneous frequencies
- GLONASS, GALILEO, BEIDOU, QZSS, IRNSS
- Add New GPS Signals L2C and L5
- Add GPS and GLONASS L2 (includes P Code on both L1 and L2)
- Add Galileo E5a/b and BeiDou B2

5.14.3 List of Available Options

OPT-04,-08,-16,-32,-48.-64: enables all channels that the GSG hardware can support.

OPT-AST: enables features for Assisted-GNSS testing.

OPT-BDS: enables all BeiDou signals supported.

OPT-ECL: enabling this option loads and enables the predefined scenarios for eCall.

OPT-FN: fixed bandwidth noise

OPT-GLO: enables Glonass signals supported.

OPT-GAL: enables all Galileo signals supported.

OPT-HPWR: High Power Option

OPT-HV: high velocity/altitude enables the simulation of high velocity vehicles.

OPT-INTF: interference option to simulate interference signals. See also "Interference signals" on page 64.

OPT-IRN: enables all IRNSS signals supported.

OPT-JAM: jamming enables the ability to define a point source of interference signals. See "Interference signals" on page 64.

OPT-L2: enables GPS L2P, L1P and GLONASS L2 signals.

OPT-L2C: enables GPS L2C signals.

OPT-L5: enables GPS L5, Galileo 5a/b, BeiDou B2, IRNSS, SBAS L5 signals.

OPT-L6: reserved for Galileo E6 and BeiDou B3.

OPT-MP: allows for the simulation of multipath signals. See also "Multipath Signals" on page 61.

OPT-NOW: uses downloaded Ephemeris and NTP time set to align GSG generator signals with live sky.

OPT-PPS: PPS Output allows for configuring 1/10/100/1000 pulse-per-second output aligned to the GPS on-time point

OPT-QZ: enables the simulation of QZSS signals.

OPT-RP: offers the ability to convert and record GPS data to a GSG scenario to replay the actual route and satellites in view during that route, including power levels.

OPT-RSG: allows GSG to receive trajectory information in real time via SCPI commands.

OPT-RTK: RTK/DGNSS RTK Real time kinematics enables the generation and use of base station information for RTK receivers.

OPT-SBAS: enables the simulation of satellite base augmentation systems (see "SBAS Satellites" on page 82).

OPT-SEN: sensor simulator package generates various sensor type data in response to a query via SCPI command (included with RSG option)

OPT-SPF: Spoofing range compensation

OPT-TIM: timing calibration option offers 10x improvement in the timing accuracy. See also "Timing Calibration" on page 182.

OPT-TLM: sets the TLM word in all messages to all 1's

OPT-TRAJ: supports receiver trajectories to be used by GSG.

OPT-TRG: external trigger enables the use of an external trigger signal to start scenarios.

OPT-VIS: visibility package allows for using environmental models to test satellite shadowing by vehicles or surrounding.

5.15 Problems?

5.15.1 Technical Support

To request technical support, please go to the "Spectracom Support" page of the Orolia Corporate website, where you can not only submit a support request, but also find additional technical documentation. Or contact our office:

Phone support is available during regular office hours under the telephone numbers listed below.

5.15.1.1 Regional Contact

Orolia operates globally and has offices in several locations around the world. Our main offices are listed below:

Table 5-2: Orolia contact information

Country	Location	Phone
France	Les Ulis, Cedex	+33 (0)1 6453 3980
USA	Rochester, NY	+1.585.321.5800

Additional regional contact information can be found on the Contact page of the Orolia website.

5.16 License Notices

Some parts of this product use free software released under the terms of the GNU Lesser General Public License, version 3, as published by the Free Software Foundation. Upon request Orolia will give out source code in accordance with applicable licenses. For contact information, see: orolia.com.

GNU LESSER GENERAL PUBLIC LICENSE, Version 3, 29 June 2007

Copyright© 2007 Free Software Foundation, Inc. <http://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License. "The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.

3. Object Code Incorporating Material from Library Header Files.

The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates(ten or fewer lines in length), you do both of the following:

a. Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b. Accompany the object code with a copy of the GNU GPL and this license document.

4. Combined Works.

You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a. Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b. Accompany the Combined Work with a copy of the GNU GPL and this license document.

c. For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d. Do one of the following:

    0. Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

    1. Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e. Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)

5. Combined Libraries.

You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a. Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b. Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

GNU GENERAL PUBLIC LICENSE Version 3, 29 June 2007

Copyright© 2007 Free Software Foundation, Inc. <http://fsf.org/> Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that

(1) displays an appropriate copyright notice, and

(2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work)run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law. You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force.

You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a. The work must carry prominent notices stating that you modified it, and giving a relevant date.

b. The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c. You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d. If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a. Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b. Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c. Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d. Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e. Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a. Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
b. Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
c. Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
d. Limiting the use for publicity purposes of names of licensors or authors of the material; or
e. Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
f. Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a)provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License. An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or

(b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation.

If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation. If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

--

The GPSTk is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or any later version. Copyright 2004, The University of Texas at Austin.

--

Some parts of this product use free software released under the terms of MIT/X11 License. Upon request Orolia will give out source code according to applicable licenses. Contact information can be found at web address: orolia.com.

This license applies to GeographicLib, versions 1.12 and later.

CHAPTER 6

SCPI Guide

The following topics are included in this Chapter:

6.1 SCPI Guide: Introduction

The SCPI Guide describes the data exchange between a GSG unit and a PC. SCPI (pronounced: "skippy") stands for **Standard Commands for Programmable Instruments**. The SCPI standard describes the syntax of commands widely used to communicate with programmable instruments. Some of the commands are generic in nature, others are GSG-specific and may use arguments and parameters.

While the communication channels are not standardized under the SCPI specification, in case of GSG all of the SCPI commands can be used via any of the available communication ports, i.e. Ethernet, USB, and GPIB.

EXAMPLE – Using the ETH port:

GSG can be setup to receive raw SCPI commands through the Ethernet port over TCP port 5025.

In the Option Menu, change the connection type from USB or Ethernet to SCPI-RAW. Then send the SCPI commands to the GSG.

6.2 Protocol

6.2.1 General Format of Commands

The general format of protocol commands follows the SCPI syntax. For example:

```
SOURce:SCENario:CONTrol start
SOURce:SCENario:CONTrol?
```

Commands ending with ?-mark are queries. Keywords can be shortened by typing only the capital letters. Case does not matter. For example:

```
sour:scen:cont?
```

If using SCPI-Raw all commands should be terminated with newline "\n". All responses from GSG are also terminated with newline. Exceptions are commands and responses where the length of data is inside the command/response. These are `SOURce:FILe:data` command and response to `MMEMory:DATA?` query. Some commands, such as `sour:scen:log?` and `sour:scen:advlog?` imply multiline responses. When a multiline response is not empty, each line of data is terminated with a newline symbol. Additionally, multiline responses (even empty ones) are terminated with an empty line (a line with no symbols except a newline symbol in it).

When using, for example, a telnet client to control the unit, it is enough just type commands and press enter to send the command.

6.2.2 Protocol Errors

Below is a list of possible errors and their explanations. They can be retrieved using `tSYSTem:ERRor[:NEXT]?`

- **-100,"Command error"**. This is the generic syntax error for devices that cannot detect more specific errors. This code indicates only that a Command Error defined in IEEE-488.2, 11.5.1.1.4 has occurred.
- **-102,"Syntax error"**. An unrecognized command or data type was encountered.
- **-105,"GET not allowed"**. A Group Execute Trigger was received within a program message (see IEEE-488.2, 7.7).
- **-108,"Parameter not allowed"**. Parameter given for command which does not have any parameters.
- **-109,"Missing parameter"**. Command requiring parameter(s) is issued without them.
- **-112,"Program mnemonic too long"**. Protocol keyword too long. All keywords are less than 12 characters long.
- **-113,"Undefined header"**. The header is syntactically correct, but it is undefined for this specific device; for example, BXYZ is not defined for any device.
- **-120,"Numeric data error"**. Number format is not recognized.
- **-129,"Numeric data out of range"**. Numeric parameter value is invalid.
- **-140,"Character data error"**. This error, as well as errors 141 through –149, is generated when parsing a character data element. This particular error message is used when the device cannot detect a more specific error.
- **-141,"Invalid character data"**. Either the character data element contains an invalid character or the particular element received is not valid for the header.
- **-148,"Character data not allowed"**. Character data detected when number is expected.
- **-150,"String data error"**. String. This error as well as errors –151 through –159 is generated when parsing a string data element. This particular error message is used when the device cannot detect a more specific error.
- **-151,"Invalid string data"**. A string data element was expected, but was invalid for some reason (see IEEE-488.2, 7.7.5.2); for example, an END message was received before the terminal quote character.

- **-158,"String data not allowed"**. String data detected when number is expected.
- **-160,"Block data error"**. This error, as well as errors –161 through –169, is generated when parsing a block data element. This particular error message is used when the instrument cannot detect a more specific error.
- **-161,"Invalid block data"**. A block data element was expected, but was invalid for some reason (see IEEE-488.2, 7.7.6.2); for example, an END message was received before the length was satisfied.
- **-190,"Execution in progress"**. Command not allowed in current state.
- **-191,"Execution not in progress"**. Command requiring scenario/signal generator executing issued when device is idle.
- **-192,"Unused channel"**. Query or command for a channel that is currently not allocated to any signal.
- **-193,"RSG command overflow occurred"**. Too many RSG commands were received to process within a GSG 10 Hz epoch.
- **-194,"RSG command underflow occurred"**. Underflow detection was enabled, and no GSG RSG commands were received within a GSG 100 ms epoch.
- **-200,"Execution error"**. Scenario execution failed to start.
- **-220,"Parameter error"**. Scenario/signal generator started without a scenario.
- **-221,"Settings conflict"**. Indicates that a legal program data element was parsed but could not be executed due to the current counter state (see IEEE-488.2, 6.4.5.3 and 11.5.1.1.5.)
- **-221,"Settings conflict; invalid combination of channel and function"**. See above.
- **-222,"Data out of range"**. Indicates data values are out of range or input data such as Navigation data files have incompatible ranges of validity.
- **-224,"Illegal parameter value"**. Scenario configuration has illegal coordinates or date.
- **-225,"Out of memory"**. Command processing was interrupted because of the lack of memory.
- **-241,"Hardware missing"**. This error is given only when the reference clock signal is missing when scenario/signal generator is started or when the Ethernet MAC address cannot be found. Check that the external reference clock is connected. Verify Network Ethernet Port detection and activity lights.
- **-250,"Mass storage error"**. Copying/moving files is not allowed between directories.
- **-256,"File name not found"**. File operation attempted on a non-existing file or directory.

- **-257,"File name error"**. File name is empty.
- **-410,"Query INTERRUPTED".** Indicates that a condition causing an INTERRUPTED Query error occurred (see IEEE-488.2, 6.3.2.3).
- **1401,"Wrong program data checksum found"**. Checksum of file transferred is invalid.
- **1403,"File length error"**. In file transfer length is invalid. This can be happen if there is not enough memory or storage space on device to retrieve file.
- **1404,"File type error"**. Invalid file type given.

6.3 Command Reference

6.3.1 Common Commands

6.3.1.1 *CLS

Clear Status Command

The `*CLS` common command clears the status data structures by clearing all event registers and the error queue. Also possible executing of scenario or signal generator is stopped. It does not clear enable registers and transition filters. It clears any pending *WAI, *OPC, and *OPC?.

Example

```
*CLS
```

6.3.1.2 *ESE

Standard Event Status Enable

Sets the enable bits of the standard event enable register. This enable register contains a mask value for the bits to be enabled in the standard event status register. A bit that is set true in the enable register enables the corresponding bit in the status register. An enabled bit will set the ESB (Event Status Bit) in the Status Byte Register if the enabled event occurs.

Command Syntax

```
*ESE <decimal>
```

Parameters

<dec.data> = the sum (between 0 and 255) of all bits that are true.

Event Status Enable Register (1 = enable)		
Bit	Weight	Enables
7	128	PON, Power-on occurred
6	64	URQ, User Request
5	32	CME, Command Error
4	16	EXE, Execution Error
3	8	DDE, Device Dependent Error
2	4	QYE, Query Error
1	2	RQC, Request Control (not used)
0	1	Operation Complete

Returned Format (*ESE?)

<Decimal data>

6.3.1.3 *ESR?

Event Status Register

Reads the contents of the standard event status register. Reading the standard event status register clears the register.

Returned Format

<dec.data> = the sum (between 0 and 255) of all bits that are true. See table for *ESE.

6.3.1.4 *IDN?

Identification query

Reads out the manufacturer, model, serial number, firmware level and options in an ASCII response data element. The query must be the last query in a program message.

Returned Format

<Manufacturer>, <Model>, <Serial Number>, <Firmware Level>, <Options>.

Example

SEND:

```
*IDN?
```

READ:

```
SPECTRACOM,GSG-5,163049,V6.0.3,16 SBAS TRAJ TRG FN NOW INTF MP PPS RSG
RP
```

Options

The first option listed is the maximum number of channels the unit has been licensed for.

- **SBAS** – Satellite Based Augmentation System satellites
- **TRAJ** – Trajectories
- **TRG** – External Trigger
- **FN** – Fixed bandwidth noise
- **VN** – Variable Bandwidth Noise (GSG-55 only)
- **INTF** – Interference channels
- **MP** – Multipath
- **NOW** – Simulate Now
- **PPS** – PPS output (1/10/100/1000)
- **HV** – High Velocity/Altitude, Extended Limits Option
- **L2** – L2 Frequency band, enables P-code for GPS L1 and L2, is required for GLONASS L2
- **L2C** – GPS L2C
- **L5** – L5 Frequency band, enables GPS L5, is required for Galileo E5a/b, BeiDou B2
- **L6** – L6 Frequency band, required for BeiDou B3 / Galileo E6
- **RTK** – Virtual Basestation and RTCM messages
- **GAL** – Galileo, enables Galileo E1, is required for Galileo E5a/b
- **GLO** – GLONASS, enables GLONASS L1, is required for GLONASS L2
- **RSG** – Real-time Scenario Generation
- **BDS** – BeiDou, enables B1, is required for BeiDou B2

- **RP** – Record and Playback
- **JAM** – Jamming Package
- **SEN** – Sensor Simulation
- **QZ** – QZSS
- **VIS** – Visibility/Terrain Obscuration
- **IRN** – IRNSS L5

6.3.1.5 *OPC

Operation Complete

The Operation Complete command causes the device to set the operation complete bit in the Standard Event Status Register when all pending selected device operations have been finished.

`*OPC` and `*OPC?` commands can be used with overlapping commands, i.e., commands which take long time to finish. In GSG such commands are starting/arming scenario execution (`SOUR:SCEN:CONT START`), and starting/arming of the signal generator (`SOUR:ONECHN:CONT START`).

Example

Enable OPC-bit

SEND:

```
*ESE 1
```

Start scenario. *OPC will set the operation complete bit in the status register when the start of scenario is done and it is running.

SEND:

```
SOURce:SCENario:CONTrol start;*OPC
```

Wait 5s for the scenario to start. Then read the event status register.

SEND:

```
*ESR?
```

Check the Operation complete bit (0) in the result. If it is true, the start of scenario is completed and you can ask for example the current position.

SEND:

```
SOURce:SCENario:LOG?
```

Then read the event status register to reset it:

SEND:

```
*ESR?
```

6.3.1.6 *OPC?

Operation Complete Query

The Operation Complete query places an ASCII character 1 into the device's Output Queue when all pending device operations have been finished.

When a scenario is running there will be a pending operation set that is released at the start of each 100 ms epoch. As a consequence of this, an `*OPC?` command will constantly block except for a short period at the start of each 100 msec epoch. The `OPC?` command (and `*WAI` command) can hence be used to synchronize the execution of other SCPI commands with GSG's internal processing loop, with a resolution of 100 ms.

The `OPC?` is the recommended method to synchronize commands as `*OPC?` blocks at the user's application, rather than within the GSG. The use of `*OPC?` over `*WAI` is to be preferred, particularly if several consecutive commands are used as a means to count elapsed epochs.

For example, checking that the ECEF position command is applied on the next 10 Hz (100 ms) epoch:

```
sour:scen:ecefposition IMMEDIATE,1000.00,2000.00,3000.00
*OPC?
sour:scen:ecefposition?
```

Returned format

1

6.3.1.7 *RST

Reset

Resets the device. Any ongoing activity is stopped and the device is prepared to start new operations.

Example

```
*RST
```

6.3.1.8 *SRE

Service Request Enable

Sets the service request enable register bits. This enable register contains a mask value for the bits to be enabled in the status byte register. A bit that is set true in the enable register enables the corresponding bit in the status byte register to generate a Service Request.

Command Syntax

```
*SRE <Decimal data>
```

Parameters

<dec.data> = the sum (between 0 and 255) of all bits that are true.

Service Request Enable Register (1 = enable)		
Bit	Weight	Enables
7	128	OPR, Operation Status
6	64	RQS, Request Service
5	32	ESB, Event Status Bit
4	16	MAV, Message Available
3	8	QUE, Questionable Data/Signal Status
2	4	EAV, Error Available
1	2	Not used
0	1	Device Status

Returned format

<Integer>

Where:

<Integer> = the sum of all bits that are set.

Example

```
*SRE 1
```

6

In this example, the device generates a service request when a message is available in the output queue.

6.3.1.9 *SRE?

Service Request Enable Query

Read the value of the service request enable register.

Returned format

<Integer> = the sum (between 0 and 255) of all bits that are true. See "*SRE" on the previous page for a description of the individual bits.

6.3.1.10 *STB?

Status Byte Query

Reads out the value of the Status Byte. Bit 6 reports the Master Summary Status bit (MSS), not the Request Service (RQS). The MSS is set if the instrument has one or more reasons for requesting service.

Returned format

<Integer> = the sum (between 0 and 255) of all bits that are true.

Status byte Register (1 = true)			
Bit	Weight	Name	Condition
7	128	OPR	Enabled operation status has occurred.
6	64	MSS	Reason for requesting service.
5	32	ESB	Enabled status event condition has occurred
4	16	MAV	An output message is ready
3	8	QUE	The quality of the output signal is questionable
2	4	EAV	Error available
1	2		Not used
0	1		Not used

6.3.1.11 *TST?

Self Test

The self-test query causes an internal self-test and generates a response indicating whether or not the device completed the self-test without any detected errors.

Returned format

<Integer>

Where:

- 0 = No error
- 1 = Error in reference clock

6.3.1.12 *WAI

Wait-to-continue

The Wait-to-Continue command prevents the device from executing any further commands or queries until execution of all previous commands or queries have been completed. It differs from the *OPC? command in that it blocks within the GSG. It also resumes operation and allows synchronous execution of a command sequence within the GSG 10Hz epoch.

Example

```
SOURce:SCENario:CONTrol start;*WAI;SOURce:SCENario:LOG?
```

Wait until scenario is running and then request NMEA position.

RSG Example

SEND:

```
*WAI?
```

SEND:

```
SOURce:SCENario:VELocity 123.400,27.25, 210.8000
```

SEND:

```
SOURce:SCENario:PRYRate 123.400,-2.0000,2.0000,1.0000
```

Wait until the next 100 msec interval and issue the following commands.

6.3.2 SYSTem: Subsystem Commands

6.3.2.1 SYSTem:ERRor?

Function

This SYSTem command queries the error queue for an ASCII text description of the next error and removes it from the queue. The error messages are placed in an error queue, with a FIFO (First In-First Out) structure. This queue is summarized in the Error AVailable (EAV) bit in the status byte.

The System Error command is extended with relevant command and protocol errors. It will allow the user to determine:

- Scheduled commands arrive too late to meet their required pre-processing time based on their timestamps.
- The user can configure protocol to flag error in situation where:
  - More than one position command is received during same epoch, and the commands contradict (not complement) each other. Values are not analyzed to determine contradiction, but only type of data (e.g., two position information commands are deemed to contradict, even if the actual position would not change.). In these situations only the last received information is served (no queue system used). The default configuration is NOT to flag error.
  - No position information is received during the epoch. The default configuration is NOT to flag error.

Command Syntax

```
SYSTem:ERRor[:NEXT]?
```

Note

All `SOURce:SCENario` commands are only available during scenario execution. If scenario is not running these error codes are to be returned (for both set and get functions);

- **-191**,"Execution not in progress".

Returned format

<error number>,"<Error Description String>"

Where:

<Error Description String> = an error description as ASCII text.

6.3.2.2 SYSTem:RESET:FACTory

Function

This SYSTem command performs the factory reset. With parameter restore it only restores the factory default files. With parameter clean it cleans all user data and restores factory default files.

Command Syntax

```
SYSTem:RESET:FACTory <restore|clean>
```

Note

Communication interface is not reset in order to maintain connection to the unit.

Parameter

enum = {restore, clean}

Example

SEND:

```
SYSTem:RESET:FACTory clean
```

6.3.3 SOURce: Subsystem Commands

Commands are available at all times, but note that some commands behave differently depending on the status of the unit. More specifically, commands related to, e.g., power settings will have an immediate effect, but if these commands are called during scenario or signal execution, the original settings will be restored when the execution stops.

In general, to permanently store settings the commands should be called when execution is not running.

6.3.3.1 SOURce:POWer

Function

Sets the transmit power of the device. During scenario execution all signals on all bands will get the new transmit power, and all possible power offsets between different satellite constellations and frequency bands are discarded.

Command Syntax

```
SOURce:POWer <decimal>
```

Note

Setting not stored during scenario or 1-channel mode execution.

If power is inside allowed limits, but other RF parameters need to be modified, such parameters are modified and an error about settings conflict is set.

Parameter

decimal [-160,-65] dBm

Example

SEND:

```
SOURce:POWer -123.2
```

6.3.3.2 SOURce:POWer?

Function

Queries the current transmit power of the unit.

Command Syntax

```
SOURce:POWer?
```

Example

SEND:

```
SOURce:POWer?
```

READ:

```
-121.3
```

6.3.3.3 SOURce:REFPOWer

Function

Changes the absolute power in dBm of the reference signal (GPS L1 C/A).

Command Syntax

```
SOURce:REFPOWer <decimal>
```

Notes

This command can only be used before starting a simulation. The setting is not stored during scenario or 1-channel mode execution.

If power is inside allowed limits, but other RF parameters need to be modified, such parameters are modified and an error about a settings conflict is set.

Parameter

decimal [-160,-65] dBm

Example

SEND:

```
SOURce:REFPOWer -123.2
```

You can use the keyword “default” instead of `<power>`:

```
SOURce:REFPOWer default
```

» Restores default relative power for GPS L1 C/A

6.3.3.4 SOURce:REFPOWer?

Function

Returns current absolute power in dBm of the reference signal (GPS L1 C/A).

Command Syntax

```
SOURce:REFPOWer?
```

Example

SEND:

```
SOURce:REFPOWer?
```

READ:

```
-121.3
```

6.3.3.5 SOURce:ABSPOWer

Function

Changes the absolute power for the given signal type and orbit type.

Command Syntax

```
SOURce:ABSPOWer <signal name>,[<orbit type name>,]<power>
```

Notes

This command can only be used before starting a simulation. The setting is not stored during scenario or 1-channel mode execution.

If power is inside allowed limits, but other RF parameters need to be modified, such parameters are modified and an error about a settings conflict is set.

You can use the keyword “all” instead of `<signal name>`. However, in this case the `<orbit type name>` cannot be specified. The power you entered will be used for all signal types for all constellations.

You can use the keyword “default” instead of `<power>`. The chosen `<signal name>` (and optionally `<orbit type name>`) will use the default power (specified in the corresponding ICD).

Examples

SEND:

```
SOUR:ABSPOWer BDSB1,GEO,-123.2
```

Both keywords “all”, and “default” can be used together:

```
SOURce:ABSPOWer all,default
```

- This command will reset whole power configuration to the default state (reference power and relative power offsets as it is specified in ICDs)

6.3.3.6 SOURce:ABSPOWer?

Function

Returns the current absolute power in dBm for the given signal type and orbit type.

Command Syntax

```
SOURce:ABSPOWer?<signal name>,[<orbit type name>,]<power>
```

Examples

SEND:

```
SOUR:ABSPOW? GLOL1
```

- Returns power for GLONASS L1

```
SOUR:ABSPOW? BDSB1,GEO
```

- Returns power for BEIDOU B1 on geostationary orbit

6.3.3.7 SOURce:RELPOWer

Function

Changes the relative power offset for the given signal type and orbit type.

Command Syntax

```
SOURce:RELPOWer <signal name>,[<orbit type name>,]<power>
```

Notes

This command can only be used before starting a simulation. The setting is not stored during scenario or 1-channel mode execution.

If the power is inside the allowed limits, but other RF parameters need to be modified, such parameters are modified and an error about a settings conflict is set.

You can use the keyword "all" instead of `<signal name>`. However, in this case the `<orbit type name>` cannot be specified. The power you entered will be used for all signal types for all constellations.

You can use the keyword "default" instead of `<power>`. The chosen `<signal name>` (and optionally `<orbit type name>`) will be set to the default relative power offsets.

Examples

SEND:

```
SOUR:RELPOWer BDSB1,GEO,-123.2
```

```
SOURce:RELPOWer all,default
```

» This command will reset the relative power offsets to their default values. The reference power, however, WILL NOT be changed (to change the reference power level also, use the command `SOUR:ABSPOW all, default` instead.)

6.3.3.8 SOURce:RELPOWer?

Function

Returns current relative power offset in dBm for the given signal type and orbit type (offset relative to reference power).

Command Syntax

```
SOURce:RELPOWer? <signal name>,[<orbit type name>]
```

Example

SEND:

```
SOURce:RELPOWer? BDSB1,GEO
```

6.3.3.9 SOURce:EXTREF

Function

Specifies the reference clock source. If set to ON, the external reference clock signal is required and used when scenarios are executed, or the signal generator is running.

Command Syntax

```
SOURce:EXTREF <ON|OFF>
```

Parameter

enum = {ON, OFF}

Example

SEND:

```
SOURce:EXTREF ON
```

6.3.3.10 SOURce:EXTREF?

Function

Get the currently selected clock source.

Command Syntax

```
SOURce:EXTREF?
```

Example

SEND:

```
SOURce:EXTREF?
```

READ:

```
ON
```

6.3.3.11 SOURce:PPSOUTput

Function

Sets the PPS (pulses-per-second) output of the device.

Command Syntax

```
SOURce:PPSOUTput <value>
```

Note

This feature is not available on GSG-52.

Parameter

value = 1, 10, 100, 1000 pulses per second

Example

SEND:

```
SOURce:PPSOUTput 10
```

6.3.3.12 SOURce:PPSOUTput?

Function

Get the current PPS output setting.

Command Syntax

```
SOURce:PPSOUTput?
```

Note

This feature is not available on GSG-52.

Example

SEND:

```
SOURce:PPSOUT?
```

READ:

```
100
```

6.3.3.13 SOURce:EXTATT

Function

Set the external attenuation of the device.

Command Syntax

```
SOURce:EXTATT <decimal>
```

Note

Setting not stored during scenario or 1-channel mode execution.

If the value is inside allowed limits, but other RF parameters need to be modified, they are modified and an error about settings conflict is set.

Parameter

decimal = [0,30] in dB

Example

SEND:

```
SOURce:EXTATT 12.2
```

6.3.3.14 SOURce:EXTATT?

Function

Query the current external attenuation setting of the unit.

Command Syntax

```
SOURce:EXTATT?
```

Example

SEND:

```
SOURce:EXTATT?
```

READ:

```
11.3
```

6.3.3.15 SOURce:NOISE:CONTrol

Function

Set the noise simulation ON or OFF.

Command Syntax

```
SOURce:NOISE:CONTrol <ON|OFF>
```

Note

Setting not stored during scenario or 1-channel mode execution.

Parameter

enum = {ON, OFF}

Example

SEND:

```
SOURce:NOISE:CONTrol ON
```

6.3.3.16 SOURce:NOISE:CONTrol?

Function

Get the noise simulation state.

Command Syntax

```
SOURce:NOISE:CONTrol?
```

Example

SEND:

```
SOURce:NOISE:CONTrol?
```

READ:

```
OFF
```

6.3.3.17 SOURce:NOISE:CNO

Function

Set the maximum carrier-to-noise density of the simulated signals.

Command Syntax

```
SOURce:NOISE:CNO <decimal>
```

Note

Setting not stored during scenario or 1-channel mode execution.

The actual C/N_0 of individual signals may be lower than this setting due to various reasons (distance, elevation, modified by event, etc).

Parameter

C/N_0 in dB·Hz. A decimal number, within the range [0.0 ... 56.0].

Example

SEND:

```
SOURce:NOISE:CNO 44.1
```

6.3.3.18 SOURce:NOISE:CNO?

Function

Get the current maximum carrier-to-noise density of the simulated signals.

Command Syntax

```
SOURce:NOISE:CNO?
```

Example

SEND:

```
SOURce:NOISE:CNO?
```

READ:

```
39.2
```

6.3.3.19 SOURce:NOISE:BW

Function

Set the noise simulation bandwidth. This command is only available with GSG-55 units.

Command Syntax

```
SOURce:NOISE:BW <decimal>
```

Note

Setting not stored during scenario execution or 1-channel mode execution. This command is only available with GSG-55 units.

Parameter

Noise simulation bandwidth in MHz:, Decimal number in range [0.001 ... 20.46].

Example

SEND:

```
SOURce:NOISE:BW 18.001
```

6.3.3.20 SOURce:NOISE:BW?

Function

Get the noise simulation bandwidth. This command is only available with GSG-55 units.

Command Syntax

```
SOURce:NOISE:BW?
```

Example

SEND:

```
SOURce:NOISE:BW?
```

READ:

```
20.2
```

6.3.3.21 SOURce:NOISE:OFFSET

Function

Set the frequency offset of the simulated noise from the GPS L1 center frequency (1.57542 GHz).

For example, if the noise bandwidth is set to be 20 MHz, and offset is 10 MHz, the noise will be simulated on frequency band 1575.42 ... 1595.42 MHz.

Command Syntax

```
SOURce:NOISE:OFFSET <decimal>
```

Note

Setting not stored during scenario or 1-channel mode execution. This command is only available in GSG-55.

Parameter

Noise frequency offset in MHz. A decimal number within the range [-10.23 ... 10.23].

Example

SEND:

```
SOURce:NOISE:OFFSET 2.0
```

6.3.3.22 SOURce:NOISE:OFFSET?

Function

Get the frequency offset (in MHz) of the simulated noise from the GPS L1 center frequency. This command is only available with GSG-55 units.

Command Syntax

```
SOURce:NOISE:OFFSET?
```

Example

SEND:

```
SOURce:NOISE:OFFSET
```

READ:

```
-8.2
```

6.3.3.23 SOURce:ONECHN:CONTrol

Function

Control the execution of the Signal Generator.

Command Syntax

```
SOURce:ONECHN:CONTrol <START|STOP|ARM>
```

Parameter

enum {START,STOP,ARM}

Example

SEND:

```
SOURce:ONECHN:CONTrol start
```

6.3.3.24 SOURce:ONECHN:CONTrol?

Function

Query the current state of the Signal Generator. Meaning of returned values is the following:

- **START**: Signal Generator is started and running
- **STOP**: Signal Generator is stopped and thus not running
- **WAIT**: Signal Generator delays startup for 2 minutes to allow the simulation to load required data. The start time derived from the NTP server is then aligned to the next full GPS minute.ARMED: Signal Generator is armed, all data loading is done, but Signal Generator is not yet running, but waiting for the trigger to start it
- **ARMING**: Signal Generator is loading data to memory, and arming after which it transitions to the ARMED state

Command Syntax

```
SOURce:ONECHN:CONTrol?
```

Returned values

START, STOP, WAIT, ARMED or ARMING

Example

SEND:

```
SOURce:ONECHN:CONTrol?
```

READ:

```
START
```

6.3.3.25 SOURce:ONECHN:SATid

Function

During RF generation[1], modify the current signal mode. While GSG is not generating RF, set & store the 1-channel mode satellite identifier and signal mode.

While GSG is not generating RF, modify the current signal mode.

Command Syntax (while not generating RF)

```
SOURce:ONECHN:SATid <signal_mode_letter><gnss_letter><integer>

SOURce:ONECHN:SATid <gnss_letter><integer>

SOURce:ONECHN:SATid <signal_mode_letter><integer>
```

Command Syntax (during RF generation, effective as of firmware version 7.0.1)

```
SOURce:ONECHN:SATid <signal_mode_letter>
```

Parameters

`<signal_mode_letter>`: [M, P, U]

`<gnss_letter>`: [G, R, E, C, J, I, S]

`<integer>`: [–7 ... 210]

The `gnss_letter` parameter can be:

[1]That is, when GSG is executing a scenario or generating a signal.

- ‘G’ (or ‘g’) for GPS
- ‘R’ (or ‘r’) for GLONASS
- ‘E’ (or ‘e’) for Galileo
- 'C' (or 'c') for BeiDou
- ‘J’ (or ‘j’) for QZSS
- 'I' (or 'i') for IRNSS
- ‘S’ (or ‘s’) for SBAS

The `signal_mode_letter` parameter can be:

- ‘U’ (or ‘u’) for unmodulated GPS signal (carrier only)
- P’ (or ‘p’) for PRN signal (carrier modulated by PRN sequence, no data messages)
- ‘M’ (or ‘m’) for modulated signal (normal signal including data messages)
- The syntax to be used while GSG is not generating any RF allows to omit the `signal_mode_letter` parameter and instead specify only the `gnss_letter` parameter followed by an integer. In such case, a modulated signal mode is assumed.

The `integer` parameter:

- If the signal mode is modulated or PRN, the parameter <`integer`> must be a valid satellite id number.
- For an unmodulated GPS, Galileo, BeiDou, QZSS, or IRNSS signal, the parameter <`integer`> must not be specified. For an unmodulated GLONASS signal the <`integer`> parameter specifies the frequency slot [–7 ... 6].

Additionally, as of firmware version 7.0.1, it is possible to specify only the signal mode and satellite id. All signals that have been enabled by the `source:onechn:signaltype` command (or via the front panel) and are compatible to the specified signal mode and satellite id remain set. Incompatible signals are automatically disabled. This syntax is designed to be used together with the ‘`source:onechn:signaltype`’ command, so that you can first specify the signal mode and satellite ID using the ‘`source:onechn:satid`’ command and later specify particular signals with the ‘`source:onechn:signaltype`’ command.

When GLONASS signals are selected along with the modulated mode, the GLONASS frequency slot is determined by the satellite id from the navigation data (specified either using the `SOURce:ONECHN:EPHemeris` command or from the Ephemeris field in the front panel interface).

For the PRN mode, the frequency slot for GLONASS is determined automatically by a predefined mapping of satellite id to frequency slot.

While the GSG unit is generating RF it is possible to change the signal mode using the reduced command syntax specifying only the desired signal mode. Note that currently it is only possible to change the signal mode to any mode that is ‘lower’ or equal to the initial signal mode used when the signal generator was started. For example, if the signal generator is started in PRN mode it is possible to switch to unmodulated mode and back to PRN mode, but not to modulated mode. If started in modulated mode it is possible to switch between all three signal modes. The signal mode cannot be changed if the signal generator started in unmodulated mode.

Examples (used when the unit is not generating RF)

Set modulated GPS signal with id11:

SEND:

```
SOURce:ONECHN:SATid G11
```

Set unmodulated GLONASS signal with frequency slot -5:

SEND:

```
SOURce:ONECHN:SATid UR-5
```

Examples (used while the unit is generating RF, e.g. executing a scenario)

Set signal mode to unmodulated mode:

SEND:

```
SOURce:ONECHN:SATid U
```

Set signal mode to PRN:

SEND:

```
SOURce:ONECHN:SATid P
```

Set signal mode to modulated mode:

SEND:

```
SOURce:ONECHN:SATid M
```

6.3.3.26 SOURce:ONECHN:SATid?

Function

Query the 1-channel mode satellite identifier. The returned satellite identifier can be:

- Gxx for GPS, for example G12
- Rxx for GLONASS, for example R15

- Exx for Galileo, for example E01
- Cxx for BeiDou, for example C11
- Jxx for QZSS, for example J02
- Ixx for IRNSS, for example I01
- Sxxx for SBAS, for example S120
- UG for unmodulated GPS signal
- UE for unmodulated Galileo signal
- UC for unmodulated BeiDou
- UJ for unmodulated QZSS
- UI for unmodulated IRNSS
- URx for unmodulated GLONASS signal. X is the frequency slot from -7 to 6

Command Syntax

```
SOURce:ONECHN:SATid?
```

Notes

If several signal types are selected with either `SOURce:ONECHN:SIGNALtype` or via menus, then the returned value may have several satellite identifiers separated by comma.

If the transmission of data message is disabled, the satellite identifier is preceded by the letter "P". For example, the identifier is "PG30" for the simulated GPS satellite 30, transmitting only the PRN code.

As of firmware version 7.0.1 this query takes into account possible signal mode modifications made by `SOURce:ONECHN:SATID` command while the signal generator is running.

Example

SEND:

```
SOURce:ONECHN:SATid?
```

READ:

```
G13
```

SEND:

```
SOURce:ONECHN:SATid?
```

READ:

```
G5,R5
```

6.3.3.27 SOURce:ONECHN:STARTtime

Function

Set & store 1-channel mode start time (use this command only while the unit is not generating any RF).

Command Syntax

```
SOURce:ONECHN:STARTtime <string>
```

Note

Seconds are omitted, always starts at 0 seconds.

Parameter

String of format DD/MM/YYYY hh:mm, where:

- DD=day, MM=month, YYYY=year, hh=hour, mm=minutes

Example

SEND:

```
SOURce:ONECHN:STARTtime 23/11/2010 12:45
```

6.3.3.28 SOURce:ONECHN:STARTtime?

Function

Query 1-channel mode start time.

Command Syntax

```
SOURce:ONECHN:STARTtime?
```

Example

SEND:

```
SOURce:ONECHN:STARTtime?
```

READ:

```
23/11/2010 12:45
```

6.3.3.29 SOURce:ONECHN:EPHemeris

Function

Set & store 1-channel mode ephemeris files to be used (use this command only while the unit is not generating any RF).

Ephemeris files may include RINEX v2 or newer navigation message files for GPS and/or GLONASS, “agl” type GLONASS almanac, or EGNOS/WAAS SBAS message files.

Command Syntax

```
SOURce:ONECHN:EPHemeris <string>
```

Parameter

String identifier of filename(s)

Example

SEND:

```
SOURce:ONECHN:EPHemeris brdc0020.09n7
```

SEND:

```
SOURce:ONECHN:EPHemeris Geo133_1736_01
```

6.3.3.30 SOURce:ONECHN:EPHemeris?

Function

Query 1-channel mode ephemeris files.

Command Syntax

```
SOURce:ONECHN:EPHemeris?
```

Example

SEND:

```
SOURce:ONECHN:EPHemeris?
```

READ:

```
Default
```

6.3.3.31 SOURce:ONECHN:FREQuency

Function

Set & store 1-channel mode frequency offset (use this command only while the unit is not generating any RF).

Parameter can also have optional suffix (MHz, kHz or Hz).

Command Syntax

```
SOURce:ONECHN:FREQuency <decimal>
```

Parameter

decimal [-6000000, 6000000] in Hz

Example

SEND:

```
SOURce:ONECHN:FREQuency -54
```

SEND:

```
SOURce:ONECHN:FREQuency 4.345 MHz
```

6.3.3.32 SOURce:ONECHN:FREQuency?

Function

Query 1-channel mode frequency offset in MHz.

Command Syntax

```
SOURce:ONECHN:FREQuency?
```

Example

SEND:

```
SOURce:ONECHN:FREQuency?
```

READ:

```
4.345
```

6.3.3.33 SOURce:ONECHN:SIGNALtype

Function

Sets signal(s) to be simulated (use this command only while the unit is not generating any RF). Signal type consists of comma separated list of signal names, as described under “Parameters” below.

Command Syntax

```
SOURce:ONECHN:SIGNALtype <string>
```

Notes

The satellite system GPS/GLONASS/GALILEO, BeiDou/QZSS/IRNSS and the modulation (signal mode) are set with the `SOURce:ONECHN:satID` command.

In firmware versions before version 7.0.1 if requested signal types were not compatible with the satellite ID selected either by the `SOURce:ONECHN:satID` command or from the front panel those signals were ignored and not set. As of firmware version 7.0.1 any incompatible signal results in whole command failure so that no signal is set.

Parameters

- `<String>` – GPSL1CA,GPSL1P,GPSL1PY, GPSL2P,GPS L2PY for GPS
- `<String>` – GLOL1,GLOL2 for GLONASS
- `<String>` – GALE1,GALE5a,GALE5b for Galileo
- `<String>` – BDSB1, BDSB2 for BeiDou
- `<String>` – QZSSL1CA, QZSSL1SAIF, QZSSL2C, QZSSL5 for QZSS
- `<String>` – IRNSSL5 for IRNSS

Example

SEND:

```
SOURce:ONECHN:SIGNALtype GPSL1CA,GPSL2P
```

SEND:

```
SOURce:ONECHN:SIGNALtype GPSL1CA,GLOL2
```

6.3.3.34 SOURce:ONECHN:SIGNALtype?

Function

Query 1-channel signal type in use. Signal type consists of comma-separated list of the simulated signals.

Command Syntax

```
SOURce:ONECHN:SIGNALtype?
```

Example

SEND:

```
SOURce:ONECHN:SIGNALtype?
```

READ:

```
GPS L1CA,GPSL2P
```

SEND:

```
SOURce:ONECHN:SIGNALtype?
```

READ:

```
GPS L1CA,GLOL1,GALE1,BDSB1,QZSSL1CA,IRNSSL5
```

6.3.3.35 SOURce:ONECHN:LOSDynamics:SETtings

Function

Set the line of sight dynamics parameters for the Signal Generator.

If the profile is running new parameters are memorized, but will be applied only on profile restart.

Command Syntax

```
SOURce:ONECHN:LOSDynamics:SETtings <J>,<A>,<DA>,<DV>
```

Parameters

<J> – absolute jerk value in m/s^3

<A> – maximum absolute acceleration value in m/s^2

<D_A> – duration of movement with constant acceleration, positive value, in seconds;

<D_V> – duration of movement with constant velocity D_V, positive value, in seconds

Introduction to line of sight dynamics profile

This feature supports the simulation of line of sight (LOS) dynamics (velocity profile). This command is effective as of firmware version 7.0.1.

The profile is defined by the following four parameters:

i. Jerk magnitude J [m/s^3]

ii. Maximum acceleration magnitude A [m/s^2]

iii. Duration of movement with constant acceleration D_A [s]

iv. Duration of movement with constant velocity D_V [s]

When the profile is activated the simulator controls the Doppler frequency shift and range variation in accordance with specified parameters as if the receiver was moving along a straight line between it and a satellite(s). Positive velocity corresponds to range increase and negative to Doppler shift.

The profile is defined by a series of jerk pulses (duration of the jerk pulse D_J is given by above parameters J and A: D_J=A/$_J$).

The first jerk pulse is positive with J magnitude and lasts for D_J seconds causing an acceleration increase. After D_J seconds jerk resets to zero resulting in constant acceleration that is maintained for D_A seconds. Then negative jerk pulse with J magnitude follows resulting in acceleration decrease during the next D_J seconds. After that the acceleration returns to zero so that a constant velocity is maintained during the next D_V seconds. The whole process is depicted in the illustration below. The process is repeated until the profile is stopped.

The initial conditions when the profile is started are as follows:

- Start jerk J_0 = 0 m/s^3
- Start acceleration A_0 = 0 m/s^2
- Start velocity = current velocity
- Start range = current range

When the profile is stopped, the simulator resets jerk and acceleration to zero, but continues to simulate the velocity that was present at the moment when the profile was stopped.

Example

SEND:

```
SOURce:ONECHN:LOSD:SET 0.005, 0.1, 20, 20
```

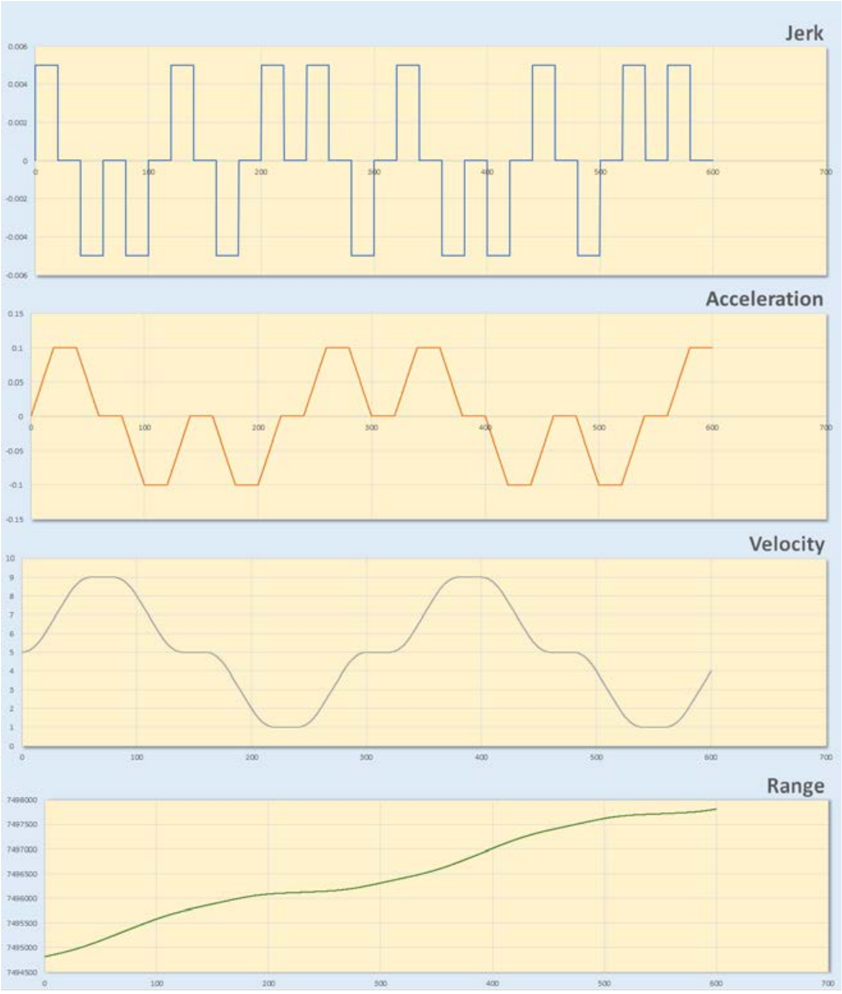


Figure 6-1: Jerk [m/s^3], acceleration [m/s^2], velocity [m/s], and range [m] over time [s]

6.3.3.36 SOURce:ONECHN:LOSDynamics:SETtings?

Function

Queries the line of sight dynamics profile parameters previously set by the `SOURce:ONECHN:LOSDynamics:SETtings` command.

Command Syntax

```
SOURce:ONECHN:LOSDynamics:SETtings?
```

Example

SEND:

```
SOURce:ONECHN:LOSD:SET?
```

READ:

```
0.005,0.1,20,2
```

6.3.3.37 SOURce:ONECHN:LOSDynamics:CONTrol

Function

Starts, restarts or stops the line of sight dynamics profile. This command can only be used while RF is generated. Before starting the profile its parameters must be set using the `SOURce:ONECHN:LOSDynamics:SETtings` command.

After the profile is started, the Doppler frequency shift is controlled automatically and can neither be changed using the `SOURce:ONECHN:FREQuency` command, nor via the front panel.

Command Syntax

```
SOURce:ONECHN:LOSDynamics:CONTrol <START|STOP>
```

Parameters

`<START>` – starts line of sight dynamics profile if the profile is not running or restarts the profile applying new parameters

`<STOP>` – stops line of sight dynamics profile

Example

```
SOUR:ONECHN:LOSD:CONT START
```

6.3.3.38 SOURce:ONECHN:LOSDynamics:CONTrol?

Function

Queries the current status of the line of sight dynamics profile, i.e. whether it is running (active) or not.

Command Syntax

```
SOURce:ONECHN:LOSDynamics:CONTrol?
```

Return values

`<START>` – the profile is currently active

`<STOP>` – the profile is not active

Example

SEND:

```
SOUR:ONECHN:LOSD:CONT?
```

READ:

```
STOP
```

6.3.3.39 SOURce:SCENario:LOAD

Function

Load the scenario as specified by <string>.

Command Syntax

```
SOURce:SCENario:LOAD <string>
```

Note

Calling the command will stop any running scenarios.

Parameter

String identifier of filename

Example

SEND:

```
SOURce:SCENario:LOAD scen01.scen
```

6.3.3.40 SOURce:SCENario:LOAD?

Function

Query the current loaded scenario.

Command Syntax

```
SOURce:SCENario:LOAD?
```

Example

SEND:

```
SOURce:SCENario:LOAD?
```

READ:

```
scen01.scen
```

6.3.3.41 SOURce:SCENario:CONTrol

Function

Control the execution of the scenario.

Command Syntax

```
SOURce:SCENario:CONTrol <START|STOP|HOLD|ARM>
```

Notes

The scenario must be loaded beforehand using `SOURce:SCENario:LOAD`.

Calling a START command will first automatically stop any running scenarios. HOLD can be used to pause and resume trajectory movement, not the entire scenario. HOLD is effective when a scenario is running.

ARMing a scenario means to hold a scenario before it is started.

Parameter

enum {START,STOP,HOLD,ARM}

Example

SEND:

```
SOURce:SCENario:CONTrol start
```

6.3.3.42 SOURce:SCENario:CONTrol?

Function

Query the current state of scenario execution. Meaning of returned values is the following:

- **START**: scenario is started and running
- **STOP**: scenario is stopped and thus not running
- **HOLD**: scenario is running, but the trajectory is on hold
- **WAIT**: scenario delays startup for 2 minutes to allow the simulation to load required data. The start time derived from the NTP server is then aligned to the next full GPS minute.
- **ARMED**: scenario is armed, all data loading is done, but scenario is not yet running but waiting for the trigger to start it
- **ARMING**: scenario is being loaded to memory after which it is in ARMED state

Command Syntax

```
SOURce:SCENario:CONTrol?
```

Returned values

START, STOP, HOLD, WAIT, ARMED or ARMING

Example

SEND:

```
SOURce:SCENario:CONTrol?
```

READ:

```
START
```

6.3.3.43 SOURce:SCENario:PROPenv

Function

Sets built-in propagation environment model.

The parameters `sky_limit`, `obstruction_limit` and `nlos_probability` are optional, either all of them or none should be set.

Notes:

- The scenario must be running.
- Note that 0<= obstruction_limit <= sky_limit <=90.

For additional information, see "Propagation Environment Models" on page 69.

Command Syntax

```
SOURce:SCENario:PROPenv <URBAN|SUBURBAN|RURAL|OPEN> [,<sky_lim-
it>,<obstruction_limit>,<nlos_probability>]
```

Parameter

Decimal [0.0,90.0] `sky_limit`: elevation above which there is no obstruction.

Decimal [0.0,90.0] `obstruction_limit`: elevation below which there is no line-of-sight satellites.

Decimal [0.0,1.0] `nlos_probability`: probability for a satellite with elevation between sky_limit and obstruction_limit to be non-line-of-sight.

Examples

SEND:

```
SOURce:SCENario:PROPenv urban
```

SEND:

```
SOURce:SCENario:PROPenv suburban,50,30,0.2
```

6.3.3.44 SOURce:SCENario:PROPenv?

Function

Query the current propagation model and its parameters.

Command Syntax

```
SOURce:SCENario:PROPenv?
```

Example

SEND:

```
SOURce:SCENario:PROPenv?
```

READ:

```
suburban,50,30,0.2
```

6.3.3.45 SOURce:SCENario:LOG?

Function

Get current position as NMEA data, available only when scenario is running.

Command Syntax

```
SOURce:SCENario:LOG?
```

Example

SEND:

```
SOURce:SCENario:LOG?
```

READ:

```
$GPRMC,181810.000,A,6000.1041,N,2400.0553,E,019.4,284.9,060109,,*0B
$GPGGA,181810.000,6000.1041,N,2400.0553,E,1,15,0.6,587.0,M,0.0,M,,,*0F

$GPGSV,4,1,15,23,77.7,192.3,44,20,52.8,132.7,44,32,31.2,117.3,44,31,2-
4.6,44.0,44*00

$GPGSV,4,2,15,16,9.2,96.3,44,7,1.1,190.7,44,17,0.5,242.4,44,2,17.4,31-
9.9,44*00

$GPGSV,4,3,15,30,6.3,1.2,44,4,46.0,280.1,44,13,51.5,230.8,44,25,19.6,-
184.5,44*2A
$GPGSV,4,4,15,126,22.0,178.8,44,124,21.9,182.9,44,120,14.3,223.6,44*E4
```

6.3.3.46 SOURce:SCENario:ADVLOG?

Function

The Advanced Log feature queries log records of the specified log. This feature is effective as of firmware version 6.7.1.

The following logs can be queried:

- **RSG** log – contains realtime movement parameters of the object being modelled along with time information;
- **SAT** log – contains various data describing modelled satellites movement parameters along with time.
- **NAVMSG** log – contains navigation data messages transmitted by simulated satellite

Command Syntax

```
SOURce:SCENario:ADVLOG? <logID>[,<filter1>,<filter2>,...,
<filterN>]
```

`<logID>` – log identifier that specifies what log data to request

`<filterX>` – optional filter expression that allows to include into response only specified record types

Mechanism

When a scenario is running, the GSG unit internally creates log records at predefined time intervals and puts them into a limited-size queue with a FIFO (First In – First Out) structure where they remain for some time (several seconds). As the scenario continues to run, new records get added into the log and old records get removed either automatically or by request from the user. Each type of log features its own (independent) queue.

RSG log records are created every 100 ms. SAT log records are created every 1 second.

When requested using the `sour:scen:advlog?` command, log records are removed from the queue starting with the oldest available records. They are returned as text lines containing comma-separated fields. These lines can be easily processed by user-developed software to extract specific fields, or they can be stored in a csv-file allowing for further analysis with spreadsheet software.

When the user does not request any log records, the latest log records remain in their queues until the unit is requested to start a new scenario. Hence it is possible to get remaining records even after a scenario was stopped.

One possible approach to using the advanced log feature is to periodically make a series of repetitive queries, waiting for an empty response in each series. An empty response signifies that the most recent data have been received and no new data is available so far.

General response structure

A response to an advanced log command usually contains several lines of text, with each line containing several comma-separated fields. The order of these fields corresponds with the labels order in response to a `SOURce:SCENario:ADVLOG:HEADER?` query.

Each line of the response is terminated with the line end symbol "`\n`" (ASCII code 0x0A). Not all records may be returned at once, so another request may be required. If all records have been returned and no new records are available, an empty response will be returned. When **SCPI raw** mode is enabled, responses are additionally terminated with an empty line (containing only the newline symbol in it), even if no new records are available. The additional newline symbol is intended to simplify end of response detection on the TCP client side.

Since queue sizes are limited, it may be important to ensure that no record is missed due to queue overflow caused by too low a query rate. Log records of RSG and SAT logs contain the `id` field that can be used to detect such an "overflow condition". The `id` field is a 16-bit unsigned counter that counts from 0 to 65535 and then wraps to 0.

Note that several consecutive lines may contain the same `id` field. That is because a group of records is usually created at the same moment, but may still contain different data sets. For example, a group of **SAT** log records usually contains information about several satellites, so there will be several lines of data – one per satellite, and each line will have the same `record id` and time fields. In the RSG log, movement parameters for the body center and the antenna are created at the same moment.

Filter expressions

Each log supports individual filter expressions. Several of them can be combined with commas to specify what record types should be included in the response. If no filter expression is specified, the return response will contain all supported record types.

For the RSG log the supported record types are:

BODY_CENTER CENTER CENT	Includes RSG parameters for body center
ANTENNA ANT	Includes RSG parameters for antenna

Filter expressions example:

```
SOUR:SCEN:ADVLOG? RSG,BODY_CENTER
```

```
SOUR:SCEN:ADVLOG? RSG,CENTER,ANT
```

EXAMPLE:

SEND:

```
sour:scen:advlog? sat
```

READ:

```
17803, SAT, 17803.0, 2014-05-09T19:56:26.000, 503803.0, G9,
13999325.9529469125, -21451840.2281696014, -7614347.5806083838,
23965993.35, -269.19, 1414.63, -0.37
```

6.3.3.47 SOURce:SCENario:ADVLOG:HEADer?

Function

Queries the header for the data of the specified log. This feature is effective as of firmware version 6.7.1.

This command is intended to be used together with the command `SOURce:SCENario:ADVLOG?`, and allows to get a line of comma-separated labels for fields of corresponding log records.

Command Syntax

```
SOURce:SCENario:ADVLOG:HEADer? <logID>
```

`<logID>` — Log identifier that specifies the advanced log for which to obtain the header.

Notes

The position of specific field label within the comma-separated line is the same as the position of that field's value within a response line of the `SOURce:SCENario:ADVLOG` command.

In order to be compatible with future versions of the unit firmware, any user-developed software should not strictly rely on a specific order of fields in responses to the `SOURce:SCENario:ADVLOG?` and `SOURce:SCENario:ADVLOG:HEADer?` commands. Before issuing log record requests user-developed software should:

a. first request the log header once,

b. then determine positions of all fields of interest based on their labels,

c. and then refer to log record fields by their determined positions.

The order of fields is fixed within one firmware version.

When the `logID` argument is `RSG`, the following fields are available:

Field label (in response to a "SOURce:SCENario: ADVLOG:HEADer?" query)	Field meaning	Possible field values (in response to a "SOURce:SCENario:ADVLOG?" query)
id	Record numeric id	Integer in range [0; 65535]
RSG	Log identifier	Always RSG
record_type	Record type	ANTENNA or BODY_CENTER
time	Scenario time in seconds	Non-negative decimal
utc_time	UTC time in ISO 8601 format	e.g., 2016-09-20T20:29:36.100
gps_sow	GPS second of week	Integer in range [0; 604799]
speed_over_ground	Speed over ground, m/s	Non-negative decimal
acceleration_over_ground	Acceleration, m/s^2	Decimal
vertical_speed	Vertical speed, m/s	Decimal
vertical_acceleration	Vertical accel-eration, m/s^2	Decimal
heading	Course, degrees	[0 ; 360)
heading_rate	Course rate, degrees/s	Decimal
pitch	Pitch, degrees	Decimal in range [-180; +180]
pitch_rate	Pitch rate, degrees/s	Decimal
roll	Roll, degrees	Decimal in range [-180; +180]
roll_rate	Roll rate, degrees/s	Decimal
yaw	Yaw, degrees	Decimal in range [-180; +180]
yaw_rate	Yaw rate, degrees/s	Decimal
latitude	Latitude, degrees	Decimal in range [-90; +90]
longitude	Longitude, degrees	Decimal in range [0; +360]
altitude	Altitude, degrees	Decimal
pos_x	ECEF X position, m	Decimal
pos_y	ECEF Y position, m	Decimal
pos_z	ECEF Z position, m	Decimal
vel_x	Speed ECEF X-pro-jection, m/s	Decimal

Field label (in response to a "SOURce:SCENario: ADVLOG:HEADer?" query)	Field meaning	Possible field values (in response to a "SOURce:SCENario:ADVLOG?" query)
vel_y	Speed ECEF Y-projection, m/s	Decimal
vel_z	Speed ECEF Z-projection, m/s	Decimal
acc_x	Acceleration ECEF X-projection, m/s^2	Decimal
acc_y	Acceleration ECEF Y-projection, m/s^2	Decimal
acc_z	Acceleration ECEF Z-projection, m/s^2	Decimal
vel_e	East speed projection, m/s	Decimal
vel_n	North speed projection, m/s	Decimal
vel_u	Vertical speed projection, m/s	Decimal
acc_e	Acceleration east projection, m/s^2	Decimal
acc_n	Acceleration north projection, m/s^2	Decimal
acc_u	Acceleration vertical projection, m/s^2	Decimal

When the `LogID` is `SAT`, the following fields are available:

Field label (in response to a "SOURce:SCENario: ADVLOG:HEADer?" query)	Field meaning	Possible field values (in response to a "SOURce:SCENario:ADVLOG?" query)
id	Record numeric id	Integer in range [0; 65535]
SAT	Log identifier	Always SAT
time	Scenario time, s	>= 0
utc_time	UTC time in ISO 8601 format	e.g. 2016-09-20T20:29:36.000
gps_sow	GPS second of week	Integer in range [0; 604799]
sat_id	Satellite ID	e.g. G11 for GPS satellite 11

Field label (in response to a "SOURce:SCENario: ADVLOG:HEADer?" query)	Field meaning	Possible field values (in response to a "SOURce:SCENario:ADVLOG?" query)
pr_l1	Pseudorange L1, m	Decimal
prr_l1	Pseudorange rate L1, m/s	Decimal
doppler_shift_l1	Doppler shift L1, Hz	Decimal
doppler_shift_rate_l1	Doppler shift rate L1, Hz/s	Decimal
pos_x	Satellite ECEF X position, m	Decimal
pos_y	Satellite ECEF Y position, m	Decimal
pos_z	Satellite ECEF Z position, m	Decimal

As of firmware version 6.7.5, when LogID is NAVMSG, the following fields are available:

Field label (in response to a "SOURce:SCENario: ADVLOG:HEADer?" query)	Field meaning	Possible field values (in response to a "SOURce:SCENario:ADVLOG?" query)
id	Record numeric id	Integer in range [0; 65535]
NAVMSG	Log identifier	Always NAVMSG
time	Scenario time, s	>= 0
utc_time	UTC time in ISO 8601 format	e.g., 2016-09-20T20:29:36.000
sat_id	Satellite ID	e.g., G11 for GPS satellite 11

Field label (in response to a "SOURce:SCENario:ADVLOG:HEADer?" query)	Field meaning	Possible field values (in response to a "SOURce:SCENario:ADVLOG?" query)
signal_type	Signal type of navigation message	For GPS satellites: L1CA, L1P, L1CAP, L2P; "L1CAP" is used when the satellite has both L1CA and L1P signals enabled.
sf_id	Subframe id	Integer per corresponding ICD
pg_id	Page id	Integer per corresponding ICD
msg	Navigation message data	Hexadecimally-encoded binary data of navigation message e.g., 8BFFFC464C7749C005364A923E46B3001EDA4C48A6BEC14EBA60ECD324A90057186FC0133C4. String length depends on corresponding navigation message length. Please note that data length (in bits) is not necessarily multiple of 8 bits, hence the number of hex digits is not necessarily even. It can be even or odd. Usually hex dump represents some amount of bytes, each byte is represented by 2 hex digits. In this case a half byte is possible, therefore there can be an odd number of hex digits.

Currently supported navigation messages are:

- GPS: L1CA, L1P, L2P; QZSS: L1CA; Beidou: B1, B2.

Navigation message decoding functionality is provided by GSG **StudioView** software.

Example:

SEND:

```
sour:scen:advlog:head? sat
```

READ:

```
id, SAT, time, utc_time, gps_sow, sat_id, pos_x, pos_y, pos_z, pr_l1,
prr_l1, doppler_shift_l1, doppler_shift_rate_l1
```

6.3.3.48 SOURce:SCENario:OBServation

Function

Turn on scenario observations.

All parameters are seconds.

Start is the number of seconds from scenario start.

Duration is length of observations from start. Interval is the interval between the individual observations in the resulting Rinex OBS file.

Observations files are created in observations/ with name <scenarioName> <yyyymmddhhmmss>.obs, where the date is the date of the first observation in the file.

Observation files can be retrieved using the MMEMory commands. Maximum length for each file is 1 hour (3600 seconds). If duration is longer than 1 hour, then multiple files are created.

Command Syntax

```
SOURce:SCENario:OBServation <start>,<duration>,<interval>
```

Parameter

Decimal start [-1,nnn] seconds. If '-1' is used the logging will start immediately when a command is received, and this is only available when the scenario is running.

Decimal duration [-1,nnn] seconds. If '-1' is used the logging will continue until the scenario is running

Decimal interval [1,3600] seconds

Example

SEND:

```
SOURce:SCENario:OBS 10,3600,1
```

6.3.3.49 SOURce:SCENario:OBServation?

Function

Query scenario observation parameters.

Command Syntax

```
SOURce:SCENario:OBServation?
```

Example

SEND:

```
SOURce:SCENario:OBS?
```

READ:

```
10,3600,1
```

6.3.3.50 SOURce:SCENario:NAV

Function

Turn ON/OFF RINEX navigation data logging.

The generated files are in RINEX 3.0.2 mixed format, so the information for all the simulated constellations/satellites will be written into one file.

Note that the RINEX data is logged only when the GSG generates new navigation message sets, which is not done often. Therefore, the recommended way to use this command is to turn ON RINEX navigation data logging before a scenario is started. Logging is stopped when scenario stops. See the GSG User Manual for naming of the generated files.

Command Syntax

```
SOURce:SCENario:NAV <ON|OFF>
```

Parameter

ON | OFF

Example

SEND:

```
SOURce:SCENario:NAV ON
```

6.3.3.51 SOURce:SCENario:NAV?

Function

Query status of RINEX navigation data logging.

Command Syntax

```
SOURce:SCENario:NAV?
```

Example

SEND:

```
SOURce:SCENario:NAV?
```

READ:

```
OFF
```

6.3.3.52 SOURce:SCENario:SATid[n]?

Function

Query the current satellite identifier of channel n. The parameter n can be 1-5 for GSG-52/53, 1-8 for GSG-54, 1-16 for GSG-55/GSG-56 and 1-32/48/64 for GSG-62/63/64. The returned satellite identifier can be:

- Gxx for GPS for example G12
- Rxx for GLONASS, for example R15
- Exx for Galileo, for example E01
- Cxx for BeiDou, for example C11
- Jxx for QZSS, for example J02
- Ixx for IRNSS, for example I01
- Sxxx for SBAS for example S120
- UG for unmodulated GPS signal
- UE for unmodulated Galileo signal
- UC for unmodulated BeiDou signal
- UJ for unmodulated QZSS signal
- UI, for unmodulated IRNSS signal
- URx for unmodulated GLONASS signal. X is the frequency slot from -7 to 6

Would the signal be a multipath signal, this is identified by an added character D at the end. The satID is returned with a leading timestamp.

Command Syntax

```
SOURce:SCENario:SATid[n]?
```

Note

Only available during scenario execution.

Example

SEND:

```
SOURce:SCENario:SATid5?
```

READ:

```
123.4,R23
```

6.3.3.53 SOURce:SCENario:SIGNALtype[n]?

Function

Query signal type of satellite. The parameter n can be:

- 1-5 for GSG-52/53
- 1-8 for GSG-54
- 1-16 for GSG-55/GSG-56 and
- 1-32/48/64 for GSG-62/63/64.

The signal type consists of a comma-separated list of frequency bands and codes (CA or P code) for GPS and frequency bands for GLONASS, Galileo, BeiDou, QZSS and IRNSS.

Command Syntax

```
SOURce:SCENario:SIGNALtype[n]?
```

Example

SEND:

```
SOURce:SCENario:SIGNALtype20?
```

READ:

```
GPSL1CA,GPSL2P
```

6.3.3.54 SOURce:SCENario:SIGNALtype?

Function

Query the signal satellite types in the form of comma-separated values.

Command Syntax

```
SOURce:SCENario:SIGNALtype? <satID>
```

Parameter

satID – The format is explained under "SOURce:ONECHN:SATid?" on page 252.

Example

SEND:

```
SOURce:SCENario:SIGNALtype? G2
```

READ:

```
GPSL1CA,GPSL2P
```

6.3.3.55 SOURce:SCENario:NAVBITS

Function

Sets bits in a navigation message.

The endBitPos - startBitPos +1 LSB of the hexstring are used to replace the bits between startBitPos and endBitPos, so that the endBitPos is aligned with the LSB of the hexstring. In case endBitPos - startBitPos +1 > length(hexstring), the hexstring will be used as a repeating pattern to replace the bits between startBitPos and endBitPos.

Multiple commands may be applied to the same message.

Command Syntax

```
SOURce:SCENario:NAVBITS IMM, <satID>, <sigtype>, <sfid>, <pageid>,
<startBitPos>, <endBitPos>, <hexstring>, <repeat>, <crcflag> [,print-
flag]
```

Note

Only available during scenario execution.

Parameter

satID – GPS, Glonass, BeiDou, QZSS and SBAS are supported, the format is explained under "SOURce:ONECHN:SATid?" on page 252.

sigtype – One of the signal types supported by the satellite, allowed values are:

- For **GPS**: L1CA, GPSL1CA, L1P, GPSL1P, L1PY, GPSL1PY, L1CAP, GPSL1CAP, L1CAPY, GPSL1CAPY, L2P, GPSL2P, L2PY, GPSL2PY, L2C, GPSL2C, L5, GPSL5

  > **Note:** The signal types from the same group below share the same navigation bit stream.

  - L1CA, GPSL1CA, L1P, GPSL1P, L1PY, GPSL1PY, L1CAP, GPSL1CAP
  - L2P, GPSL2P, L2PY, GPSL2PY
  - L2C, GPSL2C
  - L5, GPSL5
- For **Glonass**: L1, GLOL1, L2, GLOL2

  > **Note:** The signal types from the same group below share the same navigation bit stream.

  - L1, GLOL1
  - L2, GLOL2
- For **Galileo**: E1, E5a, E5b
- For **BeiDou**: B1, B2,
- For **QZSS**: L1CA, L1SAIF (L1SBAS can be also used for L1SAIF)
- For **SBAS**: L1SBAS or SBASL1, L5SBAS or SBASL5

sfid – For GPS L1 and L2P signals: subframe id

- For GPS L2C and L5 signals: message type
- For Glonass: frame id
- For Galileo E1 and E5b signals: word id
- For Galileo E5a: page id
- For BeiDou: subframe id
- For QZSS L1CA: subframe id
- For QZSS L1SAIF: message type, where 0 means that the modification is applied on the next message independently of its type

- For SBAS: message type, where 0 means that the modification is applied on the next message independently of its type

pageid – For GPS L1 and L2P signals: page id and 0 (not relevant) when subframe id is 1-3

- For GPS L2C and L5 signals: 0 (not relevant)
- For Glonass: string id
- For Galileo E1 and E5b signals: 0/1 (even/odd)
- For Galileo E5a: 0 (not relevant)
- For BeiDou: page id
- For QZSS L1CA: page id and 0 (not relevant) when subframe id is 1-3
- For QZSS L1SAIF: 0 (not relevant)
- For SBAS: 0 (not relevant)

startBitPos, **endBitPos** – positions of bits in a navigation message,

> **Note:** For Glonass the bit count starts from LSB, whereas for other messages the bit count starts from MSB.

hexstring – Bit pattern to be set in the message

repeat – 0 if the modification should be applied only once

- 1 if the modification should be repeated on every message

crcflag – 0 if the CRC/parity does not need to be corrected after the modification

- 1 if the CRC/parity needs to be correct after the bit modification. With SBASL1, SBASL5, and L1SAIF, the preamble will be also maintained.

printflag – 0 if the modified message does not to be logged

- 1 if the modified message needs to be logged in the execution log. Note that the message is logged only once even if the modification is repeated on every message (repeat flag = 1) .
- This parameter is optional, the default value is 0.

Example

Set MSB to 1 in 6 bit health (bits 77-82) in subframe 1 of the GPS L1CA message:

```
SOUR:SCEN:NAVBITS IMM,G23,L1CA,1,0,77,77,1,1,0,1
```

Example message in the execution log:

```
06/10/2013 15:00:24 GPS GPS 23 L1CA repeat 1 sfid 1 pgid 0:
8b0c98374923e24b4108008aaaaab-
f5555550d5555543ffff2b31048ca1600ffe3b780634a8
```

Set all bits to 0 in subframe 3 of GPS L1CA message:

```
sour:scen:navbits IMM,G23,L1CA,3,0,1,300,0,0,0
```

Set bits 16-119 to 1 in the next QZSS L1SAIF message from satellite J3:

```
sour:scen:navbits IMM,J3,L1SAIF,0,2,16,119,FF,0,1
```

6.3.3.56 SOURce:SCENario:FREQuency[n]?

Function

Query the current frequency setting ofn when scenario is running. The parameter n can be 1-8 for GSG-54, 1-16 for GSG-55/56 and 1-32/48/64 for GSG-62/63/64. The frequency is returned with a leading timestamp.

Command Syntax

```
SOURce:SCENario:FREQuency[n]?
```

Note

Only available during scenario execution.

Example

SEND:

```
SOURce:SCENario:FREQuency3?
```

READ:

```
123.4,-480.513
```

6.3.3.57 SOURce:SCENario:FREQuency?

Function

Query the current frequency setting of channel satID when scenario is running. The frequency is returned with a leading timestamp.

Command Syntax

```
SOURce:SCENario:FREQuency? <satID>
```

Note

Only available during scenario execution.

Parameter

For a list of satID satellite identifiers, see "SOURce:ONECHN:SATid?" on page 252.

Example

SEND:

```
SOURce:SCENario:FREQuency? G32
```

READ:

```
123.4,-480.513
```

6.3.3.58 SOURce:SCENario:POWer[n]

Function

Sets the absolute power of channel or switch power ON or OFF when the scenario is running.

The parameter **n**, the channel number, can be:

- 1-5 for GSG-52/53
- 1-8 for GSG-54
- 1-16 for GSG-55/56
- 1-32/48/64 for GSG-62/63/64.

The **freqband** parameter is optional and can be used when only a certain satellite frequency band power is changed.

The value ALL in **freqband** means that the power for all bands is adjusted by the amount indicated via the command.

The command also accepts ON/OFF keywords as an argument, see examples below.

Command Syntax

```
SOURce:SCENario:POWer[n] <decimal>[,<freqband>]
```

```
SOURce:SCENario:POWer[n] ON|OFF
```

Note

Only available during scenario execution.

Parameters

Decimal [-160.0,-65.0] dBm, if **freqband** is not ALL. For ALL, the relative change by which the power setting is to be modified, should be limited to a delta of 100 (e.g., changing a power of -65 dBm to -165 dBm (by -100) and vice versa (+100).

FreqBand [L1, L2, L5, ALL]

Examples

```
SOURce:SCENario:POWer3 -75, ALL
```

- Set absolute power to -75 dBm for all channel #3 freqbands

```
SOURce:SCENario:POWer3 -115, L1
```

- Set absolute power to -115 dBm for the L1 channel #3 freqband

```
SOUR:SCEN:POW2 OFF
```

- Turns OFF power for channel #2

```
SOUR:SCEN:POW3 ON
```

- Turns ON power for channel #3, absolute power restored to the level it had before switching power off

6.3.3.59 SOURce:SCENario:POWer[n]?

Function

Query the current power setting of channel/satellite/satellite system/scenario during scenario execution.

The parameter **n** (the channel number) can be:

- 1-5 for GSG-52/53
- 1-8 for GSG-54
- 1-16 for GSG-55/56
- 1-32/48/64 for GSG-62/63/64.

The absolute power is returned with a leading timestamp.

Freqband is an optional parameter used to specify for which frequency band the power is returned.

If the **freqband** parameter is omitted, the L1 power is returned.

Command Syntax

```
SOURce:SCENario:POWer[n]? [<freqband>]
```

Note

Only available during scenario execution.

Parameter

FreqBand [L1, L2, L5, ALL]

Example

```
SOUR:SCEN:POWer3?
```

- Returns power of channel #3
- Example return value: 123.4, -119.7

```
SOUR:SCEN:POWer2? L5
```

- Returns L5 power of channel #2
- Example return value: 123.4, -119.7

6.3.3.60 SOURce:SCENario:POWer

Function

Set the power of satellite satID when scenario is running. Freqband parameter is optional and can be used when only certain frequency band power of satellite is changed. Value ALL in freqband means that power of all bands are adjusted by the amount indicated by the command.

Command Syntax

```
SOURce:SCENario:POWer ON|OFF

SOURce:SCENario:POWer <satID>,<decimal>[,<freqband>]

SOURce:SCENario:POWer <satID>, ON|OFF

SOURce:SCENario:POWer <SatSystem>, ON|OFF
```

```
SOURce:SCENario:POWer <SatSystem>,<decimal>[,<freqband>]
```

Note

Only available during scenario execution.

Parameter

Decimal [-160.0,-65.0] dBm, if **freqband** is not ALL. For ALL, the relative change by which the power setting is to be modified, should be limited to a delta of 100 (e.g., changing a power of -65 dBm to -165 dBm (by -100) and vice-versa (+100).

For a list of satID satellite identifier, see "SOURce:ONECHN:SATid?" on page 252.

FreqBand [L1, L2, L5, ALL]. If the freqband is not specified, it is assumed to be L1.

Examples

```
SOURce:SCENario:POWer E7, -100
```

» Sets power of L1 freqbands of satellite E7 to -100 dBm

```
SOURce:SCENario:POWer G23,-75, ALL
```

» Sets power of all freqbands of satellite G23 to -75 dBm

```
SOURce:SCENario:POWer R22,-115, L1
```

» Sets power of L1 freqbands of satellite R22 to -120 dBm

```
SOUR:SCEN:POW GPS,-120, L5
```

» Sets power of all GPS L5 channels to -120 dBm

```
SOUR:SCEN:POW GLO,OFF
```

» Turns off power of all GLONASS satellites

```
SOUR:SCEN:POW GAL,ON
```

» Turns on power of all Galileo satellites, absolute power is restored to the level it had before it was turned off

6.3.3.61 SOURce:SCENario:POWer?

Function

Query the current power setting of the satellite satID during scenario execution. The power is returned with a leading timestamp. **Freqband** is an optional parameter used to specify the frequency band whose power is returned. If **freqband** is omitted, the L1 power is returned.

Command Syntax

```
SOURce:SCENario:POWer? <satID>[,<freqband>]
```

```
SOURce:SCENario:POWer? <SatSystem>[,<freqband>]
```

Note

Only available during scenario execution.

Parameter

For a list of satID satellite identifiers see "SOURce:ONECHN:SATid?" on page 252.

FreqBand [L1, L2, L5, ALL]

SatSystem- the name of the satellite system. [GPS, GLONASS, GLO, GALILEO, GAL, BEIDOU, BDS, QZSS, IRNSS, SBAS]

Example

```
SOURce:SCENario:POWer?
```

» Returns OFF when there is no active satellites in scenario (for example after SOUR:SCEN:POW OFF command) on ON when at least one active satellite exists in scenario

```
SOUR:SCEN:POWer? G1,L2
```

» Returns power in L2 freqband of G1 satellite

```
SOUR:SCEN:POWer? GPS,L5
```

» Returns OFF when power is off for all GPS L5 satellites or ON when at least one GPS L5 active satellitesexists in scenario

6.3.3.62 SOURce:SCENario:FREQBAND:POWer

Function

Set the power for a frequency band (all satellites) when scenario is running. **Freqband** is used to specify the frequency band. The **freqband** value ALL means that the power for all bands is adjusted by the amount indicated.

Command Syntax

```
SOURce:SCENario:FREQBAND:POWer <decimal>[,<freqband>]
```

Note

Only available during scenario execution.

Parameter

Decimal [-160.0,-65.0] dBm if **freqband** is not ALL. For ALL, the limits are [-100,100] dB.

FreqBand [L1, L2, L5, ALL]

Examples

SEND:

```
SOURce:SCENario:FREQBAND:POWer -115,L1
```

SEND:

```
SOURce:SCENario:FREQBAND:POWer 10,ALL
```

6.3.3.63 SOURce:SCENario:SVmodel?

Function

Query the satellite's Space Vehicle model. The Space Vehicle model can be:

- Block II, Block IIA, Block IIR, Block IIR-M, Block IIF or Block IIIA for GPS
- Glonass-M or Glonass-K1 for GLONASS

Command Syntax

```
SOURce:SCENario:SVmodel? <satID>
```

Parameter

Decimal [-160.0,-65.0] dBm, if **freqband** is not ALL. For ALL, the limits are [-100,100] dB.

For a list of satID satellite identifiers, see "SOURce:ONECHN:SATid?" on page 252.

Example

SEND:

```
SOURce:SCENario:SVmodel? G11
```

READ:

```
Block IIR-M
```

6.3.3.64 SOURce:SCENario:SVmodel[n]?

Function

Query the satellite's Space Vehicle model.

The parameter **n** can be:

- 1-5 for GSG-52/53
- 1-8 for GSG-54
- 1-16 for GSG-55/GSG-56
- 1-32 for GSG-62.

The Space Vehicle model can be:

- Block II, Block IIA, Block IIR, Block IIR-M, Block IIF or Block IIIA for GPS
- Glonass-M or Glonass-K1 for GLONASS

Command Syntax

```
SOURce:SCENario:SVmodel[n]?
```

Example

SEND:

```
SOURce:SCENario:SVmodel4?
```

READ:

```
Block IIR-M
```

6.3.3.65 SOURce:SCENario:LIST?

Function

List possible models which can be used in the scenarios. Note that for ionomodels, the options are limited to 'ON, OFF'.

Command Syntax

```
SOURce:SCENario:LIST? <antennamodels | tropomodels | ionomodels>
```

Example

SEND:

```
SOURce:SCENario:LIST? antennamodels
```

READ:

```
Zero model, Helix, Patch, Cardioid
```

6.3.3.66 SOURce:SCENario:ANTennamodel

Function

Set the antenna model for the current scenario.

Command Syntax

```
SOURce:SCENario:ANTennamodel <antennamodel>
```

Example

SEND:

```
SOURce:SCENario:ANTennamodel Zero model
```

6.3.3.67 SOURce:SCENario:ANTennamodel?

Function

Query the antenna model of current scenario.

Command Syntax

```
SOURce:SCENario:ANTennamodel?
```

Example

SEND:

```
SOURce:SCENario:ANTennamodel?
```

READ:

```
Zero model
```

6.3.3.68 SOURce:SCENario:TROPOmodel

Function

Set the tropospheric model for the current scenario.

Command Syntax

```
SOURce:SCENario:TROPOmodel <tropomodel>
```

Example

SEND:

```
SOURce:SCENario:TROPOmodel Black model
```

6.3.3.69 SOURce:SCENario:TROPOmodel?

Function

Query the tropospheric model of the current scenario.

Command Syntax

```
SOURce:SCENario:TROPOmodel?
```

Example

SEND:

```
SOURce:SCENario:TROPOmodel?
```

READ:

```
Saastamoinen
```

6.3.3.70 SOURce:SCENario:IONOmodel

Function

Select the ionospheric model to be used in the current scenario. Permitted values are ON and OFF.

Command Syntax

```
SOURce:SCENario:IONOmodel <ionomodel>
```

6.3.3.71 SOURce:SCENario:IONOmodel?

Function

Query whether the Ionospheric model is used in the current scenario. The command returns:

- 'OFF', if the ionospheric model is not used
- 'ON' if the Klobuchar model is used
- a comma-separated list of files, if IONEX files are used.

Command Syntax

```
SOURce:SCENario:IONOmodel?
```

Note

When 'OFF' or 'ON' mode is selected and ionospheric correction can be determined using SBAS satellites, then SBAS satellites information is used.

When IONEX files are used and ionospheric correction cannot be determined using the specified IONEX files e.g., because the IONEX files do not cover the current time or position, then the unit will act as if the 'ON' mode was selected.

Example

SEND:

```
SOURce:SCENario:IONOmodel?
```

READ:

```
ON
```

SEND:

```
SOURce:SCENario:IONOmodel?
```

READ:

```
codg0010.14i,codg0030.14i,codg0020.14i
```

6.3.3.72 SOURce:SCENario:POSition

Function

Set latitude, longitude and altitude for the geodetic position (WGS84) as the start position for the loaded scenario, or the current position if the scenario is running.

Latitude and longitude are defined using decimal degrees. The altitude is given in meters as altitude over an ellipsoid.

For latitude and longitude, the recommended decimal accuracy is 8 digits, with 6 digits being the minimum recommended accuracy. No benefit is achieved at accuracies greater than 10 digits for latitude or longitude.

The altitude can be specified to a resolution down to two digits or centimeter level. No benefit is achieved with altitude accuracies greater than 4 decimal digits.

Note: In order to use this command in real time, OPT-RSG is required.

Command Syntax

```
SOURce:SCENario:POSition TIME,<decimal>,<decimal>,<decimal>
```

Parameter

TIME must be IMMediate.

Decimal Latitude [-89.99999999, +89.99999999] degrees North

Decimal Longitude [-360.00000000, +360.00000000] degrees East

Decimal Altitude [-1000.00, +20,200,000.00] meters

Notes

If a scenario is armed but not running yet, an error is returned.

The maximum altitude for normal operation is 18470 meters. (With Extended Limits it is 20,200 km).

This command changes position of the currently loaded scenario, but does not change the scenario file, so that when you try to edit the scenario, you will see unchanged parameters from the file.

Example

SEND:

```
SOURce:SCENario:POSition IMM,-77.58895432,43.08332157,168.58
```

6.3.3.73 SOURce:SCENario:POSition?

Function

Query the current geodetic position in latitude, longitude and altitude during scenario execution or the start position, if a scenario is loaded and not running yet. A time stamp of the elapsed time into the scenario is also returned.

Command Syntax

```
SOURce:SCENario:POSition?
```

Example

SEND:

```
SOURce:SCENario:POSition?
```

READ:

```
0.0,-77.58895432,43.08332157,168.58
```

6.3.3.74 SOURce:SCENario:ECEFPOSition

Function

Set the ECEF position in X, Y and Z coordinates as the start position for the loaded scenario or the current position, if the scenario is running.

The X, Y, and Z position is given in decimal meters. The recommended decimal accuracy of ECEF is 2 decimal digits. No benefit for ECEF positions is achieved at accuracies greater than 4 digits.

Command Syntax

```
SOURce:SCENario:ECEFPOSition TIME,<decimal>,<decimal>,<decimal>
```

Note

If a scenario is armed and not running yet, an error is returned.

Parameter

Decimal X Position [-26 500 000.00, +26 500 000.00] meters

Decimal Y Position [-26 500 000.00, +26 500 000.00] meters

Decimal Z Position [-26 500 000.00, +26 500 000.00] meters

TIME must be IMMediate.

Note

The maximum altitude for normal operation is 18470 meters. (Altitude for Extended Limits is 20,200 Km.)

This command changes position of the currently loaded scenario, but does not change the scenario file, so that when you try to edit the scenario, you will see unchanged parameters from the file.

Example

SEND:

```
SOURce:SCENario:ECEFPOSition IMM,2920791.72, 1300420.26, 5500650.33
```

6.3.3.75 SOURce:SCENario:ECEFPOSition?

Function

Query the current ECEF position in X, Y and Z coordinates during scenario execution or the start position, if a scenario is loaded and not running yet.

Command Syntax

```
SOURce:SCENario:ECEFPOSition?
```

Example

SEND:

```
SOURce:SCENario:ECEFPOSition?
```

READ:

```
0.0,2920791.72, 1300420.26, 5500650.33
```

6.3.3.76 SOURce:SCENario:DATEtime

Function

Set the scenario start time as GPS time.

Command Syntax

```
SOURce:SCENario:DATEtime <MM-DD-YYYY hh:mm | NTP>
```

Note

If scenario is running or armed, an error is returned.

Parameter

String format:

- MM-DD-YYYY hh:mm

...where MM=Month {01-12}, DD=day of month {01-31}, YYYY=year, hh=hours {00-23}, mm=minutes {00-59}.

For Simulate Now, the string must be "NTP".

This command changes start time of the currently loaded scenario, but does not change the scenario file, so that when you try to edit the scenario, you will see unchanged parameters from the file.

Example

SEND:

```
SOURce:SCENario:DATEtime 11-11-2011 11:11
```

SEND:

```
SOURce:SCENario:DATEtime NTP
```

6.3.3.77 SOURce:SCENario:DATEtime?

Function

Query the Date, Time and Timescale of the running scenario or the start time of the loaded scenario. The default timescale is GPS. However, the user can optionally provide a parameter to convert the current Date and Time of the running scenario to various timescales including GPS, UTC, BeiDou, QZSS, Galileo, GLONASS, EGNOS Network Time and WAAS Network Time. If no argument is provided, GPS time scale is returned. Correct time conversion to a timescale different from GPS can be performed only when a scenario is running, because timescales relation information is loaded only when starting a scenario. The special parameter RUNTIME can be used to get start time together with elapsed scenario time in seconds when scenatio is running.

Command Syntax

When scenario is not running:

- `SOURce:SCENario:DATEtime?`

When scenario is running:

- `SOURce:SCENario:DATEtime? <gps|utc|bds|qzt|gal|glo|glo0|ent|wnt>`
- `SOURce:SCENario:DATETIME? RUNTIME`

Return

When no argument is specified, or the timescale argument is specified, returned string corresponds to one of the following formats:

- MM-DD-YYYY hh:mm:ss.s AAA
- NTP

...where MM=Month {01-12}, DD=day of month {01-31}, YYYY=year, hh=hours {00-23}, mm=minutes {00-59}, ss.s=seconds {00-59} with one decimal of sub-seconds digits.

The Timescale AAA= {GPS, UTC, BDS, QZS, GAL, GLO, GLO0, ENT, WNT} field supports various GNSS timescales. If AAA is not supplied, the default is GPS timescale.

NTP is returned when scenario start time is set to NTP.

When RUNTIME argument specified, returned string is a pair:

- <date -time>
- <elapsed -seconds>

...where date time is date and time in GPS scale as specified above, and elapsed seconds is time in seconds elaped from scenario start.

Example

QUERY:

```
SOURce:SCENario:DATEtime? GLO
```

RESPONSE:

```
05-07-2012 12:34:56.7 GLO
```

QUERY:

```
SOUR:SCEN:DATETIME? RUNTIME
```

RESPONSE:

```
12-31-2012 23:55:00.1 GPS, 60.1
```

6.3.3.78 SOURce:SCENario:RUNtime?

Function

Query the current length of time in seconds elapsed since the start of RF signal generation. The time is returned including 3 digits of sub-seconds. The accuracy is equivalent to the system's internal update rate.

Notes

If no scenario is running, an error is returned.

Currently, the system accuracy is 10 Hz or 100 ms. Only a single digit of accuracy is valid.

Parameter

Decimal time [0, 2678400] Seconds Sub-Second Time [0,999] Milliseconds

Command Syntax/Example

SEND:

```
SOURce:SCENario:RUNtime?
```

READ:

```
123.400
```

6.3.3.79 SOURce:SCENario:ELAPsedtime?

Function

Query the time of a scenario elapsed since the start of RF signal generation. The time returned is in units of days, hours, minutes, seconds, and 3 digits of sub-seconds. The accuracy is equivalent to the system's internal update rate.

Command Syntax

SOURce:SCENario:ELAPsedtime?

Notes

If no scenario is running, an error will be returned. Currently the system accuracy is 10 Hz or 100 ms. Only a single digit of accuracy is valid. For now we only plan to support the GPS rimeframe, bu the UTC time scale is also defined.

Parameter

String format:

- `DDDdhh:mm:ss.xxx`, where DDD=days, hh=hours, mm=minutes, ss=seconds, xxx-x=sub-seconds up to three decimals

Example

SEND:

```
SOURce:SCENario:ELAPsedtime?
```

READ:

```
029d12:34:56.700 GPS
```

6.3.3.80 SOURce:SCENario:RTCM?

Function

Queries for the latest RTCM messages (update rate of 1Hz).

Returns a hexadecimal string of the latest RTCM messages, as configured.

Command Syntax

SOURce:SCENario:RTCM?

Example (1006 message type read)

SEND:

```
SOURce:SCENario:RTCM?
```

READ:

```
D300153EE001038519731F728933157AC40A72ABE4310000061AC0
```

6.3.3.81 SOURce:SCENario:RTCMCFG?

Function

Queries the current RTCM configuration for output.

Returns comma separated RTCM version (i.e., 3x or 2x), followed by the selected message types.

Command Syntax

```
SOURce:SCENario:RTCMCFG?
```

Example

SEND:

```
SOURce:SCENario:RTCMCFG?
```

READ:

```
3x,1002,1006,1033
```

6.3.3.82 SOURce:SCENario:RTCMCFG

Function

Sets the RTCM configuration to use. The arguments given identify the RTCM messages to be outputted.

Command Syntax

```
SOURce:SCENario:RTCMCFG 3x,<string>[,<string>]…
```

Parameter

string - 1002, 1004, 1006, 1010, 1012 and 1033.

Example

SEND:

```
SOURce:SCENario:RTCMCFG 3x,1004,1006
```

6.3.3.83 SOURce:SCENario:RLM

Function

This command supports the Galileo Return Link Acknowledgement Service by sending out a **Return Link Message** to a user in distress, thereby informing him that his distress signal has been detected and located. For more information, see "RLS (Return Link Service)" on page 186.

The SCPI command is available in two variants:

Command Syntax

Short RLM message:

```
SOURce:SCENario:RLM 0,satID,int1,int2,int2,int4
```

Long RLM message

```
SOURce:SCENario:RLM 1,satID,int1,int2,int3,int4,int5,int6,int7,int8
```

Parameters

`RLM [x]`: 0 = short message; 1 = long message

`satID`: The satellite chosen to transmit the message (PRN).

`int1...0`: an unassigned decimal integer, representing 20 bits of RLM (SAR) data transmitted within the INAV page, as illustrated below:

- Short RLM

Part (1/4)			Part (2/4)			Part (3/4)			Part (4/4)			
		SAR RLM data			SAR RLM data			SAR RLM data			SAR RLM data	
Start bit = 1	Short RLM	Beacon ID (1/3)	Start bit = 0	Short RLM	Beacon ID (2/3)	Start bit = 0	Short RLM	Beacon ID (3/3)	Start bit = 0	Short RLM	Message code	Parameters
1	1	20	1	1	20	1	1	20	1	1	4	16
22			22			22			22			

- Long RLM

Part (1/8)			Part (2/8)			Part (3/8)			Part (4/8)			
		SAR RLM data			SAR RLM data			SAR RLM data			SAR RLM data	
Start bit = 1	Long RLM	Beacon ID (1/3)	Start bit = 0	Long RLM	Beacon ID (2/3)	Start bit = 0	Long RLM	Beacon ID (3/3)	Start bit = 0	Long RLM	Message code	Parameters (1/5)
1	1	20	1	1	20	1	1	20	1	1	4	16
22			22			22			22			

Part (5/8)			Part (6/8)			Part (7/8)			Part (8/8)		
		SAR RLM data			SAR RLM data			SAR RLM data			SAR RLM data
Start bit = 0	Long RLM	Parameters (2/5)	Start bit = 0	Long RLM	Parameters (3/5)	Start bit = 0	Long RLM	Parameters (4/5)	Start bit = 0	Long RLM	Parameters (5/5)
1	1	20	1	1	20	1	1	20	1	1	20
22			22			22			22		

For additional information, see the [Galileo Open Service Signal in Space Interface Control Document].

Examples

Short RLM

» SOURce:SCENario:RLM 0,satid,int1,int2,int3,int4

» Satid = Galileo satellite in view in running scenario

» Int1,int2,int3 – beacon id – 3x20bits converted to decimal

» 15 HEX ID -> 60 binary bits (3x20) -> each 20 bit binary converted to decimal

» Int 4 – 4 bit message ID, 16 bit parameter data

» SOURce:SCENario:RLM 0,satid,int1,int2,int3,int4

» Satid = Galileo satellite in view in running scenario

» Int1,int2,int3 – beacon id – 3 x 20bits converted to decimal

» 15 HEX ID -> 60 binary bits (3 x 20) -> each 20 bit binary converted to decimal

» Int 4 – 4 bit message ID, 16 bit parameter data

```
SOUR:SCEN:RLM 0,8,711888,141509,1025,65536
```

A	D	C	D	0	2	2	8	C	5	0	0	4	0	1
1010	1101	1100	1101	0000	0010	0010	1000	1100	0101	0000	0000	0100	0000	0001

Decimal 65536 = Binary 00010000000000000000

4 bit message ID = 0001

Long RLM

- SOURce:SCENario:RLM 1,satid,int1,int2,int3,int4,int5,int6,int7,int8
- Satiid = Galileo satellite in view in running scneario
- Int1,int2,int3: Beacon ID – 3 x 20 bits convertd to decimal
- 15 HEX ID -> 60 binary bits (3 x 20) –> each 20 bit binary converted to decimal
- Int 4-8: 4bit message ID, 96 bit parameter data

```
SOUR:SCEN:RLM 1,8,711888,141509,1025,983040,1048575,1048575,1048575,1-
048575
```

A	D	C	D	0	2	2	8	C	5	0	0	4	0	1
1010	1101	1100	1101	0000	0010	0010	1000	1100	0101	0000	0000	0100	0000	0001

Decimal 983040 = Binary 11110000000000000000

4 bit message ID = 1111

6.3.3.84 SOURce:SCENario:DUPlicate

Function

This command creates a duplicate of the satellite with the given satID using the provided multipath parameters. The parameters include the Range Offset, Range Change, Range Interval, Doppler Offset, Doppler Change, Doppler Interval, Power Offset, Power Change and Power Interval.

The final optional satID can be used to specify which existing satellite is to be replaced by the newly created duplicate. If this satID is not provided, and there are no free channels, the command will fail, and produce an error.

Note

Multipath satellites require 60 seconds to be created and are introduced at modulo 30 second intervals. The GSG can only introduce 4 duplicate satellites at a time and at a maximum rate of one satellite every two seconds. Multipath, SBAS and interference/jamming channels cannot be the duplicated. This command is only available when the scenario is running. Note that excessive changes to Range or Doppler may result in Doppler shifts greater than the system can handle and cause the satellites to shutdown due to exceeding the hardware capabilities.

Command Syntax

```
SOURce:SCENario:DUPlicate <TIME>,<satID>,<decimal>,<decimal>, <decim-
al>,<decimal>,<decimal>,<integer>,<decimal>,<decimal>, <integer>[,<sa-
tID>]
```

Parameter

TIME – As TIME argument only IMMediate is supported.

satID – Satellite identifier of the satellite to duplicate

Decimal [-999.999,999.999] – Range offset in meters

Decimal [-99.999,99.999] – Range Change rate in meters/interval

Decimal [0.0,600.0] – Range Interval in seconds

Decimal [-99.9999,99.9999] – Doppler offset in meters

Decimal [-99.9999,99.9999] – Doppler Change rate in meters/sec/interval

Integer [0,600] – Doppler Interval in seconds

Decimal [-30.0,6.0] – Power offset in meters

Decimal [-30.0,0.0] – Power Change rate in dB/interval

Integer [0,600] – Power Interval in seconds

satID – Optional satellite identifier for which satellite that is to be replaced by the duplicate

Examples

SEND:

```
SOURce:SCENario:DUPlicate IMM,G3,1.0,2.0,3.0,4.0,5.0,6,7.0,-8.0,9,G9
```

6.3.3.85 SOURce:SCENario:DUPlicate[n]

Function

This command creates a duplicate of the satellite in given channel number (the second argument) using the provided multipath parameters. The parameters include the Duplicate Satellite Channel Number, Range Offset, Range Change, Range Interval, Doppler Offset, Doppler Change, Doppler Interval, Power Offset, Power Change and Power Interval.

When the scenario is running, the optional argument n can be used to specified the target channel where the duplicate will be placed. If the target channel already contains a satellite, that satellite is disabled and replaced by the duplicate.

Notes

Multipath satellites require 60 seconds to be created and are introduced at modulo 30 second intervals. The GSG can only introduce 4 duplicate satellites at a time and at a maximum rate of one satellite every two seconds. Multipath, SBAS and interference/jamming channels cannot be the duplicated. Note that excessive changes to Range or Doppler may result in Doppler shifts greater than the system can handle and cause the satellites to shutdown due to exceeding the hardware capabilities.

The command can also be used to alter multipath configuration settings before the scenario has started. The argument n is then mandatory and specifies which multipath configuration is changed. For the command to be successful the scenario configuration must have at least n number of multipath signals defined. Furthermore, the scenario must be started using the `SCPI Scenario:Control Start` command for the modification to be effective, i.e. the altered configuration will not be used if the scenario is started from the front panel. Note also that the changed configuration will not be saved to the scenario configuration file.

Command Syntax

```
SOURce:SCENario:DUPlicate[n] <TIME>,<in-
teger-
>,<-
decim-
al>,<-
decim-
al>,<-
decimal>,<decimal>,<decimal>,<integer>,<decimal>,<decimal>,<integer>
```

Parameter

TIME – As TIME argument only IMMediate is supported.

Integer [1:N] – Satellite index of the satellite to duplicate. Maximum is number of satellites

Decimal [-999.999,999.999] – Range offset in meters

Decimal [-99.99,99.99] – Range Change rate in meters/interval

Decimal [0.0,600.0] – Range Interval in seconds

Decimal [-99.9999,99.9999] – Doppler offset in meters

Decimal [-99.9999,99.9999] – Doppler Change rate in meters/sec/interval

Integer [0,600] – Doppler Interval in seconds

Decimal [-30.0,6.0] – Power offset in meters

Decimal [-30.0,0.0] – Power Change rate in dB/interval

Integer [0,600] – Power Interval in seconds

Example

SEND:

```
SOURce:SCENario:DUPlicate9 IMM,3,1.0,2.0,0.0,4.0,5.0,6,7.0,-8.0,9
```

6.3.3.86 SOURce:SCENario:DUPlicate?

Function

The command returns a comma delimited list of the channel numbers which are duplicates of the satID given.

Command Syntax

```
SOURce:SCENario:DUPlicate? <satID>
```

Parameter

satID – For a list of satellite identifiers, see "SOURce:ONECHN:SATid?" on page 252.

Example Running

SEND:

```
SOURce:SCENario:DUPlicate? G3
```

READ:

```
9
```

6.3.3.87 SOURce:SCENario:DURATION

Function

Changes the scenario duration before starting the simulation.

Command Syntax

```
SOURce:SCENario:DURATION [<mode>,][duration]
```

Parameters

`<duration>` is specified in seconds.

<mode> could be ONCE/FOREVER/LOOPING.

If only <duration> is given, then <mode> is ONCE by default.

If <mode> is FOREVER, <duration> cannot be specified.

Notes

This command changes duration of the currently loaded scenario, but does not change the scenario file, so that when you try to edit the scenario, you will see unchanged parameters from the file.

Examples

SEND:

```
SOUR:SCEN:DURATION ONCE,3600
```

Set scenario duration to 1 hour, executed once.

```
SOUR:SCEN:DURATION FOREVER
```

Set scenario duration to forever.

```
SOUR:SCEN:DURATION 600
```

Set scenario duration to 10 minutes, executed once.

6.3.3.88 SOURce:SCENario:DURATION?

Function

Inquires the duration of the scenario (<duration> specified in seconds).

Command Syntax

```
SOURce:SCENario:DURATION?
```

Return

Returns pair <mode>, <duration>. <mode> can be ONCE/FOREVER/LOOPING.

Example

QUERY:

```
SOUR:SCEN:DURATION?
```

RESPONSE:

```
LOOPING,1800
```

6.3.3.89 SOURce:SCENario:MULtipath[n]

Function

This command sets the multipath parameters for satellite with a satID. The parameters include the Range Offset, Range Change, Range Interval, Doppler Offset, Doppler Change, Doppler Interval, Power Offset, Power Change and Power Interval.

After issuing the command the target satellite becomes a multipath satellite and this is reflected in the satID as multipath satellites have a trailing character 'D' at the end of their satID.

We can have several multipath satellites with the same satID. In such cases the optional parameter *n* can be used to specify that we want to act on the *n*:th instance of these. If the *n* parameter is left out the command acts on the first satellite found with matching satID.

If the satID is left out, the parameter *n* is mandatory and specifies that the command it to act on the *n*:th multipath satellite configured.

Command Syntax

```
SOURce:SCENario:MULtipath[n] <TIME>,<satID>],
<decimal>,<decimal>,<decimal>,<decimal>,<decimal>,
<integer>,<decimal>,<decimal>,<integer>
```

Notes

- By leaving out the satID the command can be executed before the scenario has started to alter the scenario configuration. For this to be successful the scenario configuration must have at least n number of multipath signals already defined. Furthermore the scenario must be started using the `SCPI Scenario:Control Start` command for the modification to be effective. Note also that the changed configuration will not be saved to the scenario configuration file.
- This command cannot be used with SBAS and interference/jamming channels.
- Excessive changes to Range or Doppler may result in Doppler shifts greater than the system can handle and the satellites to shutdown.

Parameter

TIME - As TIME argument only IMMediate is supported.

satID – Satellite identifier of the satellite to update

Decimal [-999.0,999.0] – Range offset in meters

Decimal [-99.0,99.0] – Range Change rate in meters/interval

Decimal [0.0,600.0] – Range Interval in seconds

Decimal [-99.0,99.0] – Doppler offset in meters

Decimal [-99.0,99.0] – Doppler Change rate in meters/sec/interval

Integer [0,600] – Doppler Interval in seconds

Decimal [-30.0,6.0] – Power offset in meters

Decimal [-30.0,0.0] – Power Change rate in dB/interval

Integer [0,600] – Power Interval in seconds

Examples

SEND:

```
SOURce:SCENario:MULTIPath2
 IMM,G9D,1.0,2.0,3,4.0,5.0,6,7.0,-8.0,9
```

SEND:

```
SOURce:SCENario:MULTIPath
 IMM,G9,1.0,2.0,3,4.0,5.0,6,7.0,-8.0,9
```

6.3.3.90 SOURce:SCENario:MULtipath[n]?

Function

This command returns the multipath settings for the satellite with given satID. If we have several multipath satellites with the same satID the optional parameter *n* can be used to specify that we are interested in the *n*:th duplicate of this satellite. If instance *n* is not specified it always defaults to the first duplicate found.

If the satID is not specified the *n* argument is mandatory and the command with return the multipath settings for the *n*:th multipath satellite. This command is also available before the scenario has started to query scenario configuration settings.

In the response, the first parameter will be the satID (when scenario is running) or the satellite index for the satellite that is to be duplicated (when scenario is not running).

Command Syntax

```
SOURce:SCENario:MULtipath[n]? <satID>]
```

Parameter

Integer [1:N] – Maximum is number of defined multipath satellite channels

satID – the satellite identifier of the satellite

Example

Before execution:

SEND:

```
SOURce:SCENario:MULtipath1?
```

READ:

```
3,1.0,2.0,3,4.0,5.0,6,7.0,-8.0,9
```

During execution:

SEND:

```
SOURce:SCENario:MULtipath? G17
```

READ:

```
G17D,1.0,2.0,3,4.0,5.0,6,7.0,-8.0,9
```

6.3.3.91 SOURce:SCENario:DELete[n] <TIME>

Function

This command deletes the satellite at channel n.

Command Syntax

```
SOURce:SCENario:DELete[n] <TIME>
```

Note

Command is allowed only during scenario execution. SBAS and interference channels cannot be deleted.

Parameter

TIME – As TIME argument only IMMediate is supported.

Example

SEND:

```
SOURce:SCENario:DELete17 IMM
```

6.3.3.92 SOURce:SCENario:DELete <TIME>,<satID>[,<satID>]

Function

This command deletes the comma-delimited list of satellites.

Command Syntax

```
SOURce:SCENario:DELete <TIME>,<satID>[,<satID>] …
```

Note

Command is allowed only during scenario execution. SBAS and interference channels cannot be deleted. Only one satellite with the same satID string can be deleted at a time. Satellites which are still valid in the constellation will be restarted 1-2 minutes after deletion.

Parameter

TIME – As TIME argument only IMMediate is supported.

satID – Comma separated list of satellite identifier strings.

Example

SEND:

```
SOURce:SCENario:DELete IMM,G10,G10D,R9D
```

6.3.3.93 SOURce:SCENario:DELete[n] <TIME>,<satID>

Function

This command deletes the satellite specified by the given satID string. The optional n parameter allows the n:th duplicate satellite to be deleted rather than the first found.

Command Syntax

```
SOURce:SCENario:DELete[n] <TIME>,<satID>
```

Note

Command is allowed only during scenario execution. SBAS and interference/jamming channels cannot be deleted.

Parameter

TIME – As TIME argument only IMMediate is supported.

satID – Satellite identifier string.

Example

SEND:

```
SOURce:SCENario:DELete2 IMM,G10D
```

6.3.3.94 SOURce:SCENario:CLKMDL

Note: This SCPI command is only supported if the Spoofing Range Option is installed (OPT-SPF license, see "GSG Series Model Variants and Options" on page 203.)

Function

The **Clock Model** command is used in the context of simulating the spoofing of mobile equipment. This command allows to adjust the time-of-transmission in order to compensate for time-of-flight. Information on the Query Clock Model command can be found under "SOURce:SCENario:CLKMDL?" on the next page.

Command Syntax

```
SOURce:SCENario:CLKMDL <decimal>,<decimal>,<decimal>,<decimal>
```

Parameters

The Clock Model is described by the following parameters:

Table 6-1: Clock Model parameters

Parameter	Unit	Range	Description
t_0	s	>0	Scenario elapsed time when parameters a_0, a_1 and a_2 were measured. When t_0 is set, its value must be within ±10 seconds compared to the current elapsed scenario time.

Parameter	Unit	Range	Description
a_0	m	±10000	Clock bias measured at t_0.
a_1	m/s	±100	Clock drift at t_0.
a_2	m/s^2	±10	Rate of clock drift.

At the time **t**, the clock offset is then calculated as follows:

- $\Delta_t = t - t_0$
- bias (meters) = $a_0 + \Delta_t(a_1 + 0.5 * \Delta_t * a_2)$

Note

SOURce:CLKMDL commands are accepted only while a scenario is running.

Example

```
SOUR:SCEN:CLKMDL 10.000000,2000.000000,-10.000000,0.000000
```

6.3.3.95 SOURce:SCENario:CLKMDL?

Note: This SCPI command is only supported if the Spoofing Range Option is installed (OPT-SPF license, see "GSG Series Model Variants and Options" on page 203.)

Function

The **Clock Model** command is used in the context of simulating the spoofing of mobile equipment.

This command is used to query the Clock Model state and its parameters t, bias, t_0, a_0, a_1, and a_2. See also: "SOURce:SCENario:CLKMDL" on the previous page.

Command Syntax

```
SOURce:SCENario:CLKMDL?
```

Return

- t: scenario elapsed time in seconds when the query was handled (and when the bias was calculated).
- bias: clock bias at time t.
- t_0, a_0, a_1, a_2: the current clock model parameters

Example

```
SOURce:SCENario:clkmdl?
```

Return Format

```
2.010000E+01,1.899000E+03,1.000000E+01,2.000000E+03,-
1.000000E+01,0.000000E+00
```

6.3.3.96 SOURce:FILe:TYPe

Function

This commands are used to transfer a file to the unit. The order of commands is fixed: Type, name, length checksum and data.

`SOURce:FILe:TYPe` sets the type of the file transferred.

Valid files types are:

- CALibration
- FIRMware
- SCENario
- TRAJectory
- RSGTRAJectory
- EPHemeris
- ALManac
- EVEnt
- ENVironmentmodel
- ANTenna

Command Syntax

```
SOURce:FILe:TYPe <file type>
```

Note

Command not allowed during scenario execution, and will result in the error code “-190,"Execution in progress"”.

6.3.3.97 SOURce:FILe:NAMe

Function

This command sends the file name to be used to store the file to the unit. The name shall only contain alphanumeric characters.

Command Syntax

```
SOURce:FILe:NAMe
```

Note

Command not allowed during scenario execution, and will result in the error code “-190,"Execution in progress"”.

6.3.3.98 SOURce:FILe:LENgth

Function

This command sends the file length to the unit.

Command Syntax

```
SOURce:FILe:LENgth
```

Note

This command not allowed during scenario execution, and will result in the error code “-190,"Execution in progress"”.

6.3.3.99 SOURce:FILe:CHECKsum

Function

This command sends the file checksum to the unit. A simple arithmetic checksum is calculated by adding the characters in the file as binary unsigned 8-bit integers. The resulting sum is then negated.

Command Syntax

```
SOURce:FILe:LENgth
```

Note

This command not allowed during scenario execution, and will result in the error code "-190,"Execution in progress"".

The checksum is calculated using the following algorithm, presented here in a Python language example. The array s passed in must be read from a file opened with attributes read and binary (rb).

```
def cksum(s):
    sum = 0
    for c in s:
        sum += ord(c)
    sum &= 255
    sum = -sum
    return sum
```

An example in C is shown below. Again, the char *Data array is read from a file and is a binary array of unsigned 8-bit char values.

```
unsigned char CalcChecksum(const char *Data, unsigned Length)
{
    unsigned char sum = 0;
    unsigned char chksum;
    unsigned i;
    for (i = 0; i < Length; ++i) {
        sum += Data[i];
    }
    chksum = -sum;
    return chksum;
}
```

6.3.3.100 SOURce:FILe:DATA

Function

This command sends the file data to the unit. The file being transferred is divided into multiple data commands. There can be as many data commands as needed to send the whole file. The maximum data in one command is 4000 bytes. At the start of each data block there is a header #800001234 which tells that 8 following digits gives the length of block.

Command Syntax

```
SOURce:FILe:DATA
```

Notes

The example below depicts the transfer of a file. The first DATA command depicts the transfer. The checksum shown cannot be recreated from the file data because the end of

line characters cannot be identified from the text below.
A space must separate the DATA command from the “#” character.
This command not allowed during scenario execution, and will result in the error code “-190,"Execution in progress"”.

Example

Sending a scenario file to the unit:

SEND:

```
SOURce:FILe:TYPe SCEN
```

SEND:

```
SOURce:FILe:NAMe scen02
```

SEND:

```
SOURce:FILe:LENgth 335
```

SEND:

```
SOURce:FILe:CHECKsum 234
```

SEND:

```
SOURce:FILe:DATA #800000335StartTime 01/06/2009 00:00:00
Duration 31 3 46 0
NavigationData Default
EventData None
NumSignals 14
Startpos 60.00000000 degN 24.00000000 degE 10.0000 m
UserTrajectory Circle
TrajectoryParameters 300 10 -1
AntennaModel Zero model
IonoModel 1
TropoModel Saastamoinen
Temperature 15
Pressure 1100
Humidity 50
MinElev 0
NrSBASChannels 2
```

6.3.3.101 SOURce:KEYLOCK:PASSWord

Function

Changes the password of the front panel lock. The password has to contain only numerical characters and has to be 4-8 digits in length to be valid.

Command Syntax

```
SOURce:KEYLOCK:PASSWord <password>
```

Parameter

4-8 numerical characters.

Example

SEND:

```
SOURce:KEYLOCK:PASSWord 123456
```

6.3.3.102 SOURce:KEYLOCK:PASSWord?

Function

Queries the current password used in front panel lock.

Command Syntax

```
SOURce:KEYLOCK:PASSWord?
```

Example

SEND:

```
SOURce:KEYLOCK:PASSWord?
```

READ:

```
123456
```

6.3.3.103 SOURce:KEYLOCK:STATus

Function

Sets the state of the front panel lock.

Command Syntax

SOURce:KEYLOCK:STATus <ON|OFF>

Parameter

enum = {ON, OFF}

Example

SEND:

```
SOURce:KEYLOCK:STATus ON
```

6.3.3.104 SOURce:KEYLOCK:STATus?

Function

Queries the state of the front panel lock.

Command Syntax

```
SOURce:KEYLOCK:STATus?
```

Example

SEND:

```
SOURce:KEYLOCK:STATus?
```

READ:

```
ON
```

6.3.4 Mass Memory Subsystem Commands

All Mass Memory Subsystem commands and queries are not allowed during scenario execution, and will result in the error code *"-190,"Execution in progress""*.

6.3.4.1 MMEMory:CATalog?

Function

This command lists the content of directory <dirname>, or the current directory if the parameter is omitted.

The response contains first used bytes then free bytes on device and then list of the files in format <name>,<type>,<size>.

Command Syntax

```
MMEMory:CATalog? <dirname>
```

Example

SEND:

```
MMEMory:CATalog? events
```

READ:

```
3145728,72351744,AGPS1e,ASCII,208,AGPS2e,ASCII,110,AGPS3e,
ASCII,208,EventAGPS1,ASCII,59,EventAGPS2,ASCII,29,EventAGPS3,
ASCII,29,EventAGPS4,ASCII,180,EventAGPS5,ASCII,250,EventAGPS6,
ASCII,29,event0,ASCII,146,event007,ASCII,146,event01,ASCII,
1,eventAGPS1,ASCII,61,eventAGPS2,ASCII,30,eventAGPS3,ASCII,
30,eventAGPS4,ASCII,186,eventAGPS5,ASCII,256,eventAGPS6,
ASCII,30,events1,ASCII,874,events2,ASCII,384,events3, ASCII,122
```

6.3.4.2 MMEMory:CDIRectory

Function

Change current directory on the device. The <dirname> must be/start with navigationData, events, trajectories or scenarios.

Command Syntax

```
MMEMory:CDIRectory <dirname>
```

Example

SEND:

```
MMEMory:CDIRectory scenarios
```

6.3.4.3 MMEMory:CDIRectory?

Function

Get current directory on the device.

Command Syntax

```
MMEMory:CDIRectory?
```

Example

SEND:

```
MMEMory:CDIRectory?
```

READ:

```
events
```

6.3.4.4 MMEMory:DATA?

Function

Get contents of file. At the start of the response is the header e.g., #800001234, containing the information about the length of the file. The first digit after "#" symbol tells how many next symbols are used to encode the file size. So, in the example above, 8 digits are used to encode the file size, which is 1234 bytes. The file data follow immediately after the header.

Command Syntax

```
MMEMory:DATA? <filename>
```

Example

SEND:

```
MMEMory:CDIRectory scenarios
```

SEND:

```
MMEMory:DATA? Scen02
```

READ:

```
#800000337StartTime 01/06/2009 00:00:00
Duration 31 23 44 0
NavigationData Default
EventData None
NumSignals 16
Startpos 60.00000000 degN 24.00000000 degE 587.0000 m
```

```
UserTrajectory Circle
TrajectoryParameters 400 10 -1
AntennaModel Zero model
IonoModel 1
TropoModel Saastamoinen
Temperature 15
Pressure 1100
Humidity 50
MinElev 0
NrSBASChannels 2
```

6.3.4.5 MMEMory:DELete

Function

Delete a file in device. If <dirname> is omitted, file is assumed to be in current directory otherwise the file is deleted from <dirname>.

Command Syntax

```
MMEMory:DELete <filename>[,<dirname>]
```

Example

SEND:

```
MMEMory:DELete scen02,scenarios
```

6.3.4.6 MMEMory:COPY

Function

Copy a file in current directory or directory <srcdir>. Note that copying between directories is forbidden, so <srcdir> must be equal to <dstdir>.

Command Syntax

```
MMEMory:COPY <srcfile>[,<srcdir>],<dstfile>[,<dstdir>]
```

Example

SEND:

```
MMEMory:COPY scen02,scenarios,scen02_copy,scenarios
```

6.3.4.7 MMEMory:MOVE

Function

Move a file in current directory or directory <srcdir>. Note that moving between directories is forbidden, so <srcdir> must be equal to <dstdir>.

Command Syntax

```
MMEMory:MOVE <srcfile>[,<srcdir>],<dstfile>[,<dstdir>]
```

Example

SEND:

```
MMEMory:MOVE scen02,scenarios,scen022,scenarios
```

6.3.5 Network Subsystem Commands

6.3.5.1 NETwork:MACaddress?

Function

Reads out the Ethernet Network Port's MAC Address. If none is found, an error is returned.

Command Syntax

```
NETwork:MACaddress?
```

Returned Format

<String>

Example

SEND:

```
NETwork:MACaddress?
00:1A:F1:01:68:2D
```

6.3.6 STATus: Subsystem Commands

6.3.6.1 STATus:OPERation:CONDition?

Function

Reads out the contents of the operation status condition register. This register reflects the state of the GSG operation.

Command Syntax

```
STATus:OPERation:CONDition?
```

Returned Format

<Decimal data> = the sum (between 0 and 97) of all bits that are true. See table below:

Bit	Weight	Condition
6	64	Waiting for bus arming
5	32	Waiting for triggering and/or external arming
0	1	Calibrating

6.3.6.2 STATus:OPERation:ENABle

Function

Enables operation status reporting by setting the enable bits of the Operation Status Enable register.

This register contains a mask value for the bits to be enabled in the Operation Status Event register. A bit that is set True in the enable register enables the corresponding bit in the status register.

An enabled bit will set bit #7, OPR (Operation Status Bit), in the Status Byte Register if the enabled event occurs.

Command Syntax

STATus:OPERation:ENABle <Decimal data>

Parameters

<decimal data> = the sum (between 0 and 96) of all bits that are true. See table below:

Bit	Weight	Condition
6	64	Waiting for bus arming
5	32	Waiting for triggering and / or external arming

Returned Format

<Decimal data>

Example

SEND:

```
STAT:OPER:ENAB 32
```

In this example, waiting for triggering, bit 5, will set the OPR-bit of the Status Byte.

6.3.6.3 STATus:OPERation[:EVENt]?

Function

Read out the contents of the Operation Event Status register. Reading the Operation Event Register clears the register.

Command Syntax

```
STATus:OPERation[:event]?
```

Returned Format

<Decimal data> = the sum (between 0 and 97) of all bits that are true.

6.3.6.4 STATus:QUEStionable:CONDition?

Function

Read out the contents of the Status Questionable Data/Signal Condition register.

Command Syntax

```
STATus:QUEStionable:CONDition?
```

Returned Format

<decimal data> = the sum (between 0 and 16384) of all bits that are true. See table below:

Bit	Weight	Condition
14	16384	Unexpected command parameter

6.3.6.5 STATus:QUEStionable:ENABle

Function

Enable the Questionable Data/Signal Status Reporting by setting the enable bits of the status questionable enable register.

This enable register contains a mask value for the bits to be enabled in the status questionable event register. A bit that is set true in the enable register enables the corresponding bit in the status register. An enabled bit will set bit #3, QUE (Questionable Status Bit), in the Status Byte Register if the enabled event occurs.

Command Syntax

```
STATus:QUEStionable:ENABle <decimal data>
```

Parameters

<decimal_data> = the sum (between 0 and 16384) of all bits that are true. See the table on previous chapter.

Returned format

<Decimal data>

Example

SEND:

```
STAT:QUES:ENAB 16384
```

In this example 'unexpected parameter' bit 14 will set the QUE-bit of the Status Byte when a questionable status occurs.

6.3.6.6 STATus:QUEStionable[:EVENt]?

Function

Reads out the contents of the Questionable Data/Signal Event Register. Reading this register clears it.

Command Syntax

```
STATus:QUEStionable[:EVENt]?
```

Returned Format

<decimal data> = the sum (between 0 and 16384) of all bits that are true. See the table for STATus:QUEStionable:CONDition

6.3.6.7 STATus:PRESet

Function

Enables Device Status Reporting. This command has an SCPI standardized effect on the status data structures. The purpose is to precondition these toward reporting only device-dependent status data.

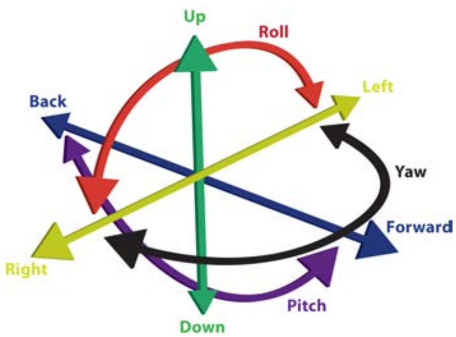
- It only affects enable registers. It does not change event and condition registers.
- The IEEE-488.2 enable registers, which are handled with the common commands *SRE and *ESE remain unchanged.
- The command sets or clears all other enable registers. Those relevant for this device are as follows:
- It sets all bits of the Device status Enable Registers to 1.
- It sets all bits of the Questionable Data Status Enable Registers and the Operation Status Enable Registers to 0.
- The following registers never change in the GSG-5x, but they do conform to the standard :STATus:PRESet values.
- All bits in the positive transition filters of Questionable Data and Operation status registers are 1.
- All bits in the negative transition filters of Questionable Data and Operation status registers are 0.

Command Syntax

```
STATus:PRESet
```

6.4 Sensors Command Reference

As the GSG unit simulates a user's movement along a given trajectory, it can also be configured to output sensor data generated by the user dynamics. The generated sensor output data is a result of the user exercising his six degrees of freedom:



- forward/backward
- left/right
- up/down , as well as the rotations around the three perpendicular axes:
- pitch
- yaw
- roll.

All sensors are initially mounted so that at start of the simulation the sensor's coordinate system XYZ is aligned with the user's ENU system (East, North, Up).

- The X axis has a positive direction towards the right side of the sensor.
- The Y axis has a positive direction towards the front of the sensor.
- The Z axis has a positive direction towards the top of the sensor.

At the start of a scenario, the X axis corresponds to the east/west axes of the ENU system while the front of the sensor—positive direction on the Y axis—is pointing to the north.

6.4.1 Supported Sensor Types

The supported sensor types and they keywords are listed in the table below, with each sensor described in the subsections.

Sensor	SENSOR_TYPE keyword
Accelerometer	ACCelerometer
Linear Accelerometer	LINearaccelerometer
Gravimeter	GRAvimeter

Sensor	SENSOR_TYPE keyword
Gyroscope	GYRoscope
Odometer	ODOmeter
3D Odometer	ODOMETER3D, ODO3D

6.4.1.1 Accelerometer

The accelerometer outputs acceleration in the XYZ axis. The typical case where the device is flat relative to the surface of the Earth appears as -STANDARD_GRAVITY in the Z axis, and X and Y values as zero.

Sensor data

values[0] – Acceleration along the x-axis, in g

values[1] – Acceleration along the y-axis, in g

values[2] – Acceleration along the z-axis, in g

6.4.1.2 Linear Accelerometer

The linear accelerometer outputs acceleration force to XYZ axis, excluding force of gravity. In all other aspects it is like the accelerometer above.

6.4.1.3 Gravimeter

The gravimeter outputs the gravity force against the XYZ axis. In all other aspects it is like the accelerometer above.

6.4.1.4 Gyroscope

The gyroscope sensor measures the rate of rotation around the X, Y and Z axis. Unlike the accelerometer, the gyro is not affected by gravity. The coordinate system is the same as is used for the acceleration sensor. Rotation is positive in the counter-clockwise direction for pitch and roll (not yaw). That is, an observer looking from some positive location on the x, y. or z axis at a device positioned on the origin would report positive rotation if the device appeared to be rotating counter clockwise.

Sensor data

values[0] – Angular speed around the x-axis, in radians/second

values[1] – Angular speed around the y-axis, in radians/second values[2] – Angular speed around the z-axis, in radians/second

6.4.1.5 Odometer

The odometer sensor keeps track of the total traveled distance.

Sensor data

values[0] – Traveled distance in meters

6.4.1.6 Odometer 3D

The 3D odometer sensor keeps track of the total traveled distance in a 3D ENU vector form.

Sensor data

values[0] – Traveled distance along the x-axis, in meters

values[1] – Traveled distance along the y-axis, in meters

values[2] – Traveled distance along the z-axis, in meters

6.4.2 Sensor Commands

6.4.2.1 SOURce:SCENario:SENSor:REGister

Function

This command registers a sensor of a given type. Once registered, the output from all registered sensors can be retrieved using the `SOURce:SCENario:SENSor:DATa?` command.

Only one sensor of each type can be registered.

Command Syntax

```
SOURce:SCENario:SENSor:REGister <SENSORTYPE>
```

6.4.2.2 SOURce:SCENario:SENSor:REGister?

Function

Queries if a given sensor is registered.

Command Syntax

```
SOURce:SCENario:SENSor:REGister? <SENSORTYPE>
```

6.4.2.3 SOURce:SCENario:SENSor:UNREGister

Function

This command unregisters a sensor of a given type, after which the sensor data is no longer output.

Command Syntax

```
SOURce:SCENario:SENSor:UNREGister <SENSORTYPE>
```

6.4.2.4 SOURce:SCENario:SENSor:DATa?

Function

The command queries for the output of all registered sensors of a running scenario. The data is updated at a 10Hz rate.

Command Syntax

```
SOURce:SCENario:SENSor:DATa?
```

6.4.2.5 SOURce:SCENario:SENSor:NORMalize SENSOR_TYPE

Function

The command specified that the output of a given sensor should be normalized. This is not applicable to all types of sensors and before the max range is set (see below) the command has no effect. The default setting is OFF.

Command Syntax

```
SOURce:SCENario:SENSor:NORMalize SENSOR_TYPE <ON|OFF>
```

6.4.2.6 SOURce:SCENario:SENSor:NORMalize? SENSOR_TYPE

Function

Queries if a sensor of a given type is normalized or not.

Command Syntax

```
SOURce:SCENario:SENSor:NORMalize? SENSOR_TYPE
```

6.4.2.7 SOURce:SCENario:SENSor:MAXrange SENSOR_TYPE

Function

The command specified the max range of a sensor. The minrange equals –maxrange.

Command Syntax

```
SOURce:SCENario:SENSor:MAXrange SENSOR_TYPE <decimal>
```

6.4.2.8 SOURce:SCENario:SENSor:MAXrange? SENSOR_TYPE

Function

The command returns the max range for a specified sensor.

Command Syntax

```
SOURce:SCENario:SENSor:MAXrange? SENSOR_TYPE
```

6.5 RSG Command Reference

6.5.1 Data Types

The Real-time Scenario Generation (RSG) commands transfer the data as ASCII strings. However, coordinate systems, units of measure, Earth Models, base data types and accuracy limits are required to implement this in the software. These attributes and values are listed in this section.

Coordinate Systems

- Geodetic (Cartesian)
- Earth Centered Earth Fixed (ECEF)

Earth Model

- WGS-84

Timestamp

- Time into scenario is given in second and 100 millisecond accuracy.

Field	Type	Default Units
Latitude	<DOUBLE>	decimal degrees
Longitude	<DOUBLE>	decimal degrees
Altitude	<DOUBLE>	meters.<2digit centimeters>
ECEF X	<DOUBLE>	meters.<2digit centimeters>
ECEF Y	<DOUBLE>	meters.<2digit centimeters>
ECEF Z	<DOUBLE>	meters.<2digit centimeters>
VelocityNS	<DOUBLE >	meters/second
VelocityEW	<DOUBLE >	meters/second
VelocityUD	<DOUBLE >	meters/second
AcelNW	<DOUBLE >	meters/second/second
AcelEW	<DOUBLE >	meters/second/second
AcelUD	<DOUBLE >	meters/second/second
Heading (psi)	<DOUBLE >	+/- degrees
Heading Rate	<DOUBLE >	+/- degrees/second
Pitch (theta)	<DOUBLE >	+/- radians

Field	Type	Default Units
Roll (phi)	<DOUBLE >	+/- radians
Pitch Rate	<DOUBLE >	radians/sec
Roll Rate	<DOUBLE >	radians/sec
Yaw Rate	<DOUBLE >	radians/sec

6.5.2 TIME Parameter

In all cases where the TIME parameter is allowed, it can be specified as:

- IMMediate, which indicates that the command is to be applied in REAL time

or

- <decimal>, indicating in seconds from Scenario start time when the information is to be applied when using uploaded Scenario/Trajectory files.

All commands issued in real-time must use IMM for the TIME parameter.

6.5.3 RSG Commands

6.5.3.1 SOURce:SCENario:POSition

Function

Set latitude, longitude and altitude for the geodetic position (WGS84) as the start position for the loaded scenario, or the current position if the scenario is running.

Latitude and longitude are defined using decimal degrees. The altitude is given in meters as altitude over an ellipsoid.

For latitude and longitude, the recommended decimal accuracy is 8 digits, with 6 digits being the minimum recommended accuracy. No benefit is achieved at accuracies greater than 10 digits for latitude or longitude.

The altitude can be specified to a resolution down to two digits or centimeter level. No benefit is achieved with altitude accuracies greater than 4 decimal digits.

Note: In order to use this command in real time, OPT-RSG is required.

Command Syntax

```
SOURce:SCENario:POSition TIME,<decimal>,<decimal>,<decimal>
```

Parameter

TIME must be IMMediate.

Decimal Latitude [-89.99999999, +89.99999999] degrees North

Decimal Longitude [-360.00000000, +360.00000000] degrees East

Decimal Altitude [-1000.00, +20,200,000.00] meters

Notes

If a scenario is armed but not running yet, an error is returned.

The maximum altitude for normal operation is 18470 meters. (With Extended Limits it is 20,200 km).

This command changes position of the currently loaded scenario, but does not change the scenario file, so that when you try to edit the scenario, you will see unchanged parameters from the file.

Example

SEND:

```
SOURce:SCENario:POSition IMM,-77.58895432,43.08332157,168.58
```

6.5.3.2 SOURce:SCENario:POSition?

Function

Queries the current geodetic position in Latitude, Longitude and Altitude. A time stamp of the time into the scenario is also returned.

As an optional argument one can specify the antenna position, as an effect of a specified lever arm, or the body center position. If the argument is not given, the body center position will be returned.

Command Syntax

```
SOURce:SCENario:POSition? [<ANTenna|BODYcenter>]
```

Example

SEND:

```
SOURce:SCENario:POSition?
```

READ:

```
123.4,-77.58895432,43.08332157,168.58
```

6.5.3.3 SOURce:SCENario:ECEFPOSition TIME

Function

Sets the ECEF position in X, Y and Z coordinates. The X, Y, and Z position is given in decimal meters. The decimal accuracy of ECEF is recommended as 2 decimal digits. No benefit is achieved for ECEF positions at accuracies greater than 4 digits.

Command Syntax

```
SOURce:SCENario:ECEFPOSition TIME,<decimal>,<decimal>,<decimal>
```

Parameter

Decimal X Position [-26 500 000.00, +26 500 000.00] meters

Decimal Y Position [-26 500 000.00, +26 500 000.00] meters

Decimal Z Position [-26 500 000.00, +26 500 000.00] meters

Note

The maximum altitude for normal operation is 18470 meters. (The altitude for Extended Limits is 20200 km.)

Example

SEND:

```
SOURce:SCENario:EPOSition 123.4,2920791.72, 1300420.26, 5500650.33
```

6.5.3.4 SOURce:SCENario:ECEFPOSition?

Function

Queries the current ECEF position in X, Y and Z coordinates.

As an optional argument, the antenna position can be specified, as an effect of a specified lever arm, or the body center position. If the argument is not given, the body center position will be returned.

Command Syntax

```
SOURce:SCENario:ECEFPOSition? [<ANTenna|BODYcenter>]
```

Example

SEND:

```
SOURce:SCENario:ECEFPOSition?
```

READ:

```
123.4,2920791.72, 1300420.26, 5500650.33
```

6.5.3.5 SOURce:SCENario:SPEed TIME

Function

Sets the vehicle's speed over ground (WGS84 ellipsoid).

Command Syntax

```
SOURce:SCENario:SPEed TIME,<decimal>
```

Parameter

Decimal 1D Speed [0.00 to +20000.00] m/s

Note

The maximum allowed speed for normal operation is 520 m/s. If you want to reverse direction, change heading or use the velocity command. (For Extended Limits it is limited by interface above.)

Example

SEND:

```
SOURce:SCENario:SPEed 123.4,30.10
```

6.5.3.6 SOURce:SCENario:SPEed?

Function

Query the current speed expressed in m/s.

Command Syntax

```
SOURce:SCENario:SPEed? [<ANTenna|BODYcenter>]
```

Example

SEND:

```
SOURce:SCENario:SPEed?
```

READ:

```
123.4,30.10
```

6.5.3.7 SOURce:SCENario:HEADing TIME

Function

Sets the vehicle's true heading. The heading is expressed in clockwise direction from the true north (WGS84 ellipsoid) representing 0 degrees, increasing to 359.999 degrees.

Command Syntax

```
SOURce:SCENario:HEADing TIME,<decimal>
```

Parameter

Decimal Heading [0, 359.999] true heading in decimal degrees

Example

SEND:

```
SOURce:SCENario:HEADing 123.4, 90.000
```

6.5.3.8 SOURce:SCENario:HEADing?

Function

Returns the vehicle's true heading expressed as described above.

Command Syntax

```
SOURce:SCENario:HEADing? [<ANTenna|BODYcenter>]
```

Example

SEND:

```
SOURce:SCENario:HEADing?
```

READ:

```
123.4, 90.000
```

6.5.3.9 SOURce:SCENario:RATEHEading TIME

Function

Sets the heading change rate. Rate is expressed as degrees per second. Heading will be updated each epoch according to the specified constant rate. Next position is calculated using direct rhumb line method (movement with constant heading). Pay attention that specifying constant heading rate results in non-constant curvature radius, thus it is not suitable for creation of closed-circle trajectories.

Command Syntax

```
SOURce:SCENario:RATEHEading TIME,<decimal>
```

Parameter

Decimal RateHeading [-180.000, 180.000] true heading change in decimal degrees per second. Positive value correspond to right turn, negative – left turn.

Example

SEND:

```
SOURce:SCENario:RATEHEading 123.4,5.500
```

6.5.3.10 SOURce:SCENario:RATEHEading?

Function

Returns the vehicle's heading rate, which was previously set using the command described above.

Command Syntax/Example

SEND:

```
SOURce:SCENario:RATEHEading?
```

READ:

```
123.4, 5.500
```

6.5.3.11 SOURce:SCENario:TURNRATE TIME

Function

Sets the rate of turning. Rate is expressed as degrees per second. Next position is calculated using direct orthodromic method (moving along shortest path with non-constant heading). Use this command to simulate movement along arc of circle or closed circle trajectory with constant velocity. Heading rate is varying each epoch, but overall average rate along single full closed circle will be equal to the value specified.

Command Syntax

```
SOURce:SCENario:TURNRATE TIME,<decimal>
```

Parameter

Decimal TurnRate [-180.000, 180.000] desired average heading rate (over single full closed circle) in decimal degrees per second. Positive value correspond to right turn, negative – left turn.

Example

SEND:

```
SOURce:SCENario: RATEHEading 123.4,5.500
```

6.5.3.12 SOURce:SCENario:TURNRATE?

Function

Returns the vehicle's rate of turning, which was previously set using the command described above.

Command Syntax/Example

SEND:

```
SOURce:SCENario:TURNRATE?
```

READ:

```
123.4, 5.500
```

6.5.3.13 SOURce:SCENario:TURNRADIUS TIME

Function

Sets the radius of turning. Radius is expressed in meters. The next position is calculated using direct orthodromic method (moving along shortest path with non-constant heading). Use this command to simulate movement along arc of circle regardless of velocity changes. Heading rate is varying each epoch, but radius of turning will be constantly equal to value specified.

Command Syntax

```
SOURce:SCENario:TURNRADIUS TIME,<decimal>
```

Parameter

Decimal TurnRadius [-5 000 000.000, 5 000 000.000] radius of turning in meters. Positive value correspond to right turn, negative – left turn.

Example

SEND:

```
SOURce:SCENario:TURNRADIUS 123.4,500 – start right turn with radius of 500 meters
```

6.5.3.14 SOURce:SCENario:TURNRADIUS?

Function

Return the vehicle's radius of turning previously set using command described above.

Command Syntax/Example

SEND:

```
SOURce:SCENario:TURNRATE?
```

READ:

```
123.4, 500.0
```

6.5.3.15 SOURce:SCENario:VELocity TIME

Function

Sets the vehicle's speed over ground (WGS84 ellipsoid) and heading in degrees.

Command Syntax

```
SOURce:SCENario:VELocity TIME,<decimal>,<decimal>
```

Parameter

Decimal 1D Speed [0.000 to +20000.000] m/s

Decimal Bearing [0, 359.999] true bearing in decimal degrees

Note

The maximum allowed speed for normal operation is 520 m/s. (For Extended Limits it is limited by interface above.)

Example

SEND:

```
SOURce:SCENario:VELocity 123.4,27.25, 210.800
```

6.5.3.16 SOURce:SCENario:VELocity?

Function

Queries the vehicle's velocity.

Command Syntax

```
SOURce:SCENario:VELocity?
```

Example

SEND:

```
SOURce:SCENario:VELocity?
```

READ:

```
123.4,27.25,210.800
```

6.5.3.17 SOURce:SCENario:VSPEed TIME

Function

Sets the vehicle's vertical speed.

Command Syntax

```
SOURce:SCENario:VSPEed TIME,<decimal>
```

Parameter

Decimal 1D Speed [-20000.00 to +20000.00] m/s

Note

The maximum allowed speed for normal operation is 520 m/s. (For Extended Limits it is limited by interface above.)

Example

SEND:

```
SOURce:SCENario:VSPEed 123.4,3.15
```

6.5.3.18 SOURce:SCENario:VSPEed?

Function

Get the vehicle's vertical speed.

Command Syntax

```
SOURce:SCENario:VSPEed? [<ANTenna|BODYcenter>]
```

Example

SEND:

```
SOURce:SCENario:VSPEed?
```

READ:

```
123.4,3.15
```

6.5.3.19 SOURce:SCENario:ENUVELocity TIME

Function

Sets the velocity expressed in ENU coordinates when scenario is running. The Velocity terms are defined in m/s.

Command Syntax

```
SOURce:SCENario:ENUVELocity TIME,<decimal>,<decimal>,<decimal>
```

Note

The local plane of the coordinates will always be re-aligned with ellipsoid surface, meaning the Up-Down velocity can be seen as a velocity with respect to ellipsoid (and not the local plane formed by the position the user was at TIME).

Parameter

Decimal Velocity East [-20000.00, +20000.00] m/s

Decimal Velocity North [-20000.00, +20000.00] m/s

Decimal Velocity Up [-20000.00, +20000.00] m/s

Note

The maximum allowed speed for normal operation is 520 m/s. (For Extended Limits it is limited by interface above.)

Example

SEND:

```
SOURce:SCENario:ENUVELocity 123.4,-4.00,3.00,0.00
```

6.5.3.20 SOURce:SCENario:ENUVELocity?

Function

Queries the current velocity during scenario execution, expressed as ENU coordinates.

Command Syntax

```
SOURce:SCENario:ENUVELocity? [<ANTenna|BODYcenter>]
```

Example

SEND:

```
SOURce:SCENario:ENUVELocity?
```

READ:

```
123.4,-4.00,3.00,0.00
```

6.5.3.21 SOURce:SCENario:ECEFVELocity

Function

Sets the current ECEF velocity in X, Y and Z coordinates when the scenario is running. The Velocity terms are defined in m/s.

Command Syntax

```
SOURce:SCENario:ECEFVELocity TIME,<decimal>,<decimal>,<decimal>
```

Parameter

Decimal Velocity X [-20000.00, +20000.00] m/s

Decimal Velocity Y [-20000.00, +20000.00] m/s

Decimal Velocity Z [-20000.00, +20000.00] m/s

Note

The maximum allowed speed for normal operation is 520 m/s. (Velocity for Extended Limits is not limited.)

Example

SEND:

```
SOURce:SCENario:ECEFVELocity 123.4,-4.00,3.00,1.00
```

6.5.3.22 SOURce:SCENario:ECEFVELocity?

Function

Queries the current ECEF velocity in 3 dimensions as X, Y and Z coordinates during scenario execution.

Command Syntax

```
SOURce:SCENario:ECEFVELocity? [<ANTenna|BODYcenter>]
```

Example

SEND:

```
SOURce:SCENario:ECEFVELocity?
```

READ:

```
123.4,-4.00,3.00,1.00
```

6.5.3.23 SOURce:SCENario:ACCeleration TIME

Function

Sets the 1D acceleration expressed in m/s^2 when scenario is running.

Command Syntax

```
SOURce:SCENario:ACCeleration TIME,<decimal>
```

Parameter

Decimal 1D Acceleration [-981 to +981] m/s^2, equivalent to [-100G to +100G]

Example

SEND:

```
SOURce:SCENario:ACCeleration 123.4,0.50
```

6.5.3.24 SOURce:SCENario:ACCeleration?

Function

Queries the 1D acceleration.

Command Syntax

```
SOURce:SCENario:ACCeleration? [<ANTenna|BODYcenter>]
```

Example

SEND:

```
SOURce:SCENario:ACCeleration?
```

READ:

```
123.4,0.50
```

6.5.3.25 SOURce:SCENario:VACCel TIME

Function

Sets the vehicle's vertical acceleration.

Command Syntax

```
SOURce:SCENario:VACCel TIME,<decimal>
```

Parameter

Decimal 1D Acceleration [-981 to +981] m/s^2, equivalent to [-100G to +100G]

Example

SEND:

```
SOURce:SCENario:VACCel 123.4,0.50
```

6.5.3.26 SOURce:SCENario:VACCel?

Function

Query the vehicle's vertical acceleration.

Command Syntax

```
SOURce:SCENario:VACCel? [<ANTenna|BODYcenter>]
```

Example

SEND:

```
SOURce:SCENario:VACCel?
```

READ:

```
123.4,0.50
```

6.5.3.27 SOURce:SCENario:ENUACCel TIME

Function

Sets the acceleration expressed in ENU coordinates when scenario is running. The acceleration terms are defined in m/s^2.

Command Syntax

```
SOURce:SCENario:ENUACCel TIME,<decimal>,<decimal>,<decimal>
```

Note

The local plane of the coordinates will always be re-aligned with ellipsoid surface, meaning the Up-Down velocity can be seen as a velocity with respect to ellipsoid (and not the local plane formed by the position the user was at TIME).

Parameter

Decimal Acceleration East [-981, +981] m/s^2, equivalent to [-100G to +100G]

Decimal Acceleration North [-981, +981] m/s^2, equivalent to [-100G to +100G]

Decimal Acceleration Up [-981, +981] m/s^2, equivalent to [-100G to +100G]

Example

SEND:

```
SOURce:SCENario:ENUACCel 123.4,-2.83,2.83,0.00
```

6.5.3.28 SOURce:SCENario:ENUACCel?

Function

Queries the current acceleration expressed as ENU coordinates during scenario execution.

Command Syntax

```
SOURce:SCENario:ENUACCel? [<ANTenna|BODYcenter>]
```

Example

SEND:

```
SOURce:SCENario:ENUACCel?
```

READ:

```
123.4,-2.83,2.83,0.00
```

6.5.3.29 SOURce:SCENario:ECEFACCel TIME

Function

Sets the ECEF acceleration in 3-dimensions as Acceleration X, Y, and Z when scenario is running. The Acceleration terms are defined in m/s^2.

Command Syntax

```
SOURce:SCENario:ECEFACCel TIME,<decimal>,<decimal>,<decimal>
```

Parameter

Decimal Acceleration X [-981, +981] m/s^2, equivalent to [-100G to +100G]

Decimal Acceleration Y [-981, +981] m/s^2, equivalent to [-100G to +100G]

Decimal Acceleration Z [-981, +981] m/s^2, equivalent to [-100G to +100G]

Example

SEND:

```
SOURce:SCENario:EACCel 123.4,-2.83,2.83,1.00
```

6.5.3.30 SOURce:SCENario:ECEFACCel?

Function

Queries the current ECEF acceleration in 3-dimensions as Acceleration X, Y, Z during scenario execution.

Command Syntax

```
SOURce:SCENario:ECEFACCel? [<ANTenna|BODYcenter>]
```

Example

SEND:

```
SOURce:SCENario:ECEFACCeleration?
```

READ:

```
123.4,-2.83,2.83,1.00
```

6.5.3.31 SOURce:SCENario:PRYattitude TIME

Function

Sets the Vehicle Attitude in 3-dimensions about the center of mass as Pitch, Roll, and Yaw when scenario is running. The terms are defined in Radians.

The pitch argument will be positive when pitching from forward to up. The roll argument is positive when rotating from up to right. The yaw argument is positive when rotating from forward to right. The angles are applied in the order of pitch, roll and finally yaw. The user cannot impact this order by applying the pitch, roll, and yaw as separate calls.

Command Syntax

```
SOURce:SCENario:PRYattitude TIME,<decimal>,<decimal>,<decimal>
```

Parameter

Decimal Pitch [-π, +π] Radians

Decimal Roll [-π, +π] Radians

Decimal Yaw [-π, +π] Radians

Example

SEND:

```
SOURce:SCENario:PRYattitude -2.0000,2.0000,1.0000
```

6.5.3.32 SOURce:SCENario:PRYattitude?

Function

Query the current Vehicle Attitude in 3-dimensions about the center of mass as Pitch, Roll, and Yaw during scenario execution.

Command Syntax/Example

SEND:

```
SOURce:SCENario:PRYattitude?
```

READ:

```
123.4,-2.0000,2.0000,1.0000
```

6.5.3.33 SOURce:SCENario:DPRYattitude TIME

Function

Sets the Vehicle Attitude in 3-dimensions about the center of mass as Pitch, Roll, and Yaw when scenario is running. The terms are defined in Degrees.

The pitch argument will be positive when pitching from forward to up. The roll argument is positive when rotating from up to right. The yaw argument is positive when rotating from forward to right. The angles are applied in the order of pitch, roll, and finally yaw. The user cannot impact this order by applying the pitch, roll, and yaw as separate calls.

Command Syntax

```
SOURce:SCENario:DPRYattitude TIME,
<decimal>,<decimal>,<decimal>
```

Parameter

Decimal Pitch [-180, +180] Degrees

Decimal Roll [-180, +180] Degrees

Decimal Yaw [-180, +180] Degrees

Example

SEND:

```
SOURce:SCENario:DPRYattitude -2.0000,2.0000,1.0000
```

6.5.3.34 SOURce:SCENario:DPRYattitude?

Function

Queries the current Vehicle Attitude in 3-dimensions about the center of mass as Pitch, Roll, and Yaw during scenario execution. Returned values are defined in Degrees.

Command Syntax

```
SOURce:SCENario:DPRYattitude?
```

Example

SEND:

```
SOURce:SCENario: DPRYattitude?
```

READ:

```
123.4,-2.0000,2.0000,1.0000
```

6.5.3.35 SOURce:SCENario:PRYRate TIME

Function

Sets the rate of change in Vehicle Attitude in 3-dimensions about the center of mass as Pitch Rate, Roll Rate, and Yaw Rate when scenario is running. The Rate of Attitude change terms are defined in Radians per second.

When the PRY rate is active the changes will be applied in the order of pitch, roll, and yaw. Note that this order matters and can't be controlled by user, but angle arguments will have to adapt to this order.

Command Syntax

```
SOURce:SCENario:PRYRate TIME,<decimal>,<decimal>,<decimal>
```

Parameter

Decimal Pitch Rate [-π, +π] Radians per second

Decimal Roll Rate [-π, +π] Radians per second

Decimal Yaw Rate [-π, +π] Radians per second

Example

SEND:

```
SOURce:SCENario:PRYRate 123.4,-2.0000,2.0000,1.0000
```

6.5.3.36 SOURce:SCENario:PRYRate?

Function

Queries the current rate of change in Vehicle Attitude in 3-dimensions about the center of mass as Pitch, Roll, and Yaw during scenario execution.

Command Syntax/Example

SEND:

```
SOURce:SCENario:PRYRate?
```

READ:

```
123.4,-2.0000,2.0000,1.0000
```

6.5.3.37 SOURce:SCENario:DPRYRate TIME

Functions

Sets the rate of change in Vehicle Attitude in 3-dimensions about the center of mass as Pitch Rate, Roll Rate, and Yaw Rate when scenario is running. The Rate of Attitude change terms are defined in Degrees per second.

When the PRY rate is active the changes will be applied in the order of pitch, roll, and yaw. Note that this order matters and can't be controlled by user, but angle arguments will have to adapt to this order.

Command Syntax

```
SOURce:SCENario:DPRYRate TIME,<decimal>,<decimal>,<decimal>
```

Parameter

Decimal Pitch Rate [-3600, +3600] Degrees per second

Decimal Roll Rate [-3600, +3600] Degrees per second

Decimal Yaw Rate [-3600, +3600] Degrees per second

Example

SEND:

```
SOURce:SCENario:DPRYRate 123.4,-2.0000,2.0000,1.0000
```

6.5.3.38 SOURce:SCENario:DPRYRate?

Function

Queries the current rate of change in Vehicle Attitude in 3-dimensions about the center of mass as Pitch, Roll, and Yaw during scenario execution. Returned values are defined in Degrees per second.

Command Syntax/Example

SEND:

```
SOURce:SCENario:DPRYRate?
```

READ:

```
123.4,-2.0000,2.0000,1.0000
```

6.5.3.39 SOURce:SCENario:KEPLER TIME

Function

Sets the Kepler orbit parameters.

If a position, speed or acceleration command is sent after the Kepler orbit command, they will overwrite the movements along the Kepler orbit. PRY commands can be applied while the Kepler orbit is active.

Command Syntax

```
SOURce:SCENario:KEPLER TIME,<-
decimal>,<decimal>,<decimal>,<decimal>,<decimal>,<decimal>
```

Parameter

Decimal Mean anomaly [−π] Radians

Decimal Eccentrity

Decimal Semi-major axis

Decimal Ascension of ascending node [-π, +π] Radians

Decimal Inclination [-π, +π] Radians

Decimal Argument of perigee [-π, +π] Radians

Example

SEND:

```
SOURce:SCENario:KEPLER 0,1.30280292873,0.995806301944E-
03,0.075377837181E+08,-
0.159728922636E+01,0.957334107483E+00,0.296123313943E+01
```

6.5.3.40 SOURce:SCENario:KEPLER?

Function

Queries the Kepler orbit parameters in the same order as set and the current true anomaly. If Kepler orbit is not in use, the return value is an empty string.

Command Syntax/Example

SEND:

```
SOURce:SCENario:KEPLER?
```

READ:

```
1618.6,1.302803E+00,3.130653E+00,9.958063E-04,7.537784E+06,-
1.597289E+00,9.573341E-01,2.961233E+00
```

6.5.3.41 SOURce:SCENario:RSGUNDERflow

Function

Enables or disable RSG underflow detection. It is active once an RSG command comes in. Underflow detection is disabled by default.

Command Syntax

```
SOURce:SCENario:RSGUNDERflow <integer>
```

Parameter

Integer – Enable or disable {1,0}, respectively.

Example

SEND:

```
SOURce:SCENario:RSGUNDERflow 1
```

6.5.3.42 SOURce:SCENario:RSGUNDERflow?

Function

Queries RSG underflow detection status, whether enabled or disabled.

Command Syntax/Example

SEND:

```
SOURce:SCENario:RSGUNDERflow?
```

READ:

```
0
```

6.5.3.43 SOURce:SCENario:DOPPler?

Function

Queries a satellite's Doppler for a specific signal supported by that satellite. The signals supported vary based on the constellation and scenario configuration.

Command Syntax

```
SOURce:SCENario:DOPPler? <satID>,<sigtype>
```

Notes

If no scenario is running, an error is returned.

If the satellite does not support the signal type, an error is returned.

Parameters

satID – GPS, Glonass, Galileo, BeiDou, QZSS, IRNSS and SBAS are supported. For more information on the format, see "SOURce:ONECHN:SATid?" on page 252.

sigtype – One of the signal types supported by the satellite, allowed values are:

- For GPS: L1CA, GPSL1CA, L1P, GPSL1P, L1PY, GPSL1PY, L1CAP, GPSL1CAP, L1CAPY, GPSL1CAPY, L2P, GPSL2P, L2PY, GPSL2PY, L2C, GPSL2C, L5, GPSL5 Note that the signal types from the same group below share the same navigation bit stream.
  - L1CA, GPSL1CA, L1P, GPSL1P, L1PY, GPSL1PY, L1CAP, GPSL1CAP,
  - L2P, GPSL2P, L2PY, GPSL2PY
  - L2C, GPSL2C
  - L5, GPSL5
  - For Glonass: GLOL1 (or L1), GLOL2 (or L2)
  - For Galileo: E1, E5a, E5b
  - For Beidou: BDSB1 (or B1), BDSB2 (or B2)
  - For QZSS: QZSSL1CA (or L1, or L1CA), L1SAIF (or L1SBAS), QZSSL2C (or L2C), QZSSL5 (or L5)
  - For IRNSS: IRNSSL5 (or L5)
  - For SBAS: L1SBAS or SBASL1 and L5SBAS and SBASL5

Example

SEND:

```
SOURce:SCENario:DOPPler? G27,L1CAP
```

READ:

```
-320.51
```

6.5.3.44 SOURce:SCENario:PRANge?

Function

Queries a satellite's range for a specific frequency band supported by that satellite for the simulated user position or optionally an RTK base station position. The signals supported vary based on the constellation and scenario configuration.

Command Syntax

```
SOURce:SCENario:PRANge? <satID>,<sigtype>,<location>
```

Notes

If no scenario is running, an error is returned.

If the satellite does not support the signal type, an error is returned.

If the base station location is not enabled, 0 values are returned.

Parameters

satID – GPS, Glonass, Galileo, BeiDou, QZSS, IRNSS and SBAS are supported. For more information on the format, see "SOURce:ONECHN:SATid?" on page 252.

sigtype – one of the signal types supported by the satellite, allowed values are:

- For GPS: L1CA, GPSL1CA, L1P, GPSL1P, L1PY, GPSL1PY, L1CAP, GPSL1CAP, L1CAPY, GPSL1CAPY, L2P, GPSL2P, L2PY, GPSL2PY, L2C, GPSL2C, L5, GPSL5
  Note that the signal types from the same group below share the same navigation bit stream
  - L1CA, GPSL1CA, L1P, GPSL1P, L1PY, GPSL1PY, L1CAP, GPSL1CAP,
  - L2P, GPSL2P, L2PY, GPSL2PY
  - L2C, GPSL2C
  - L5, GPSL5
- For Glonass: L1, GLOL1, L2, GLOL2,
- For Galileo: E1, E5a, E5b
- For BeiDou: B1, B2,
- For QZSS: L1CA, L1SAIF (L1SBAS can be also used for L1SAIF)
- For IRNSS: L5, IRNSSL5
- For SBAS: L1SBAS or SBASL1 and L5SBAS or SBASL5

Location – user or base

Example

SEND:

```
SOURce:SCENario:PRANge? G19,L1CA
```

READ:

```
24241628.51
```

6.5.3.45 SOURce:SCENario:CHINview?

Function

Queries a comma separated list of values ranging from 1 to 64 which indicate which satellite index values are active in view in the simulated sky. Duplicate and interference channels are ignored.

Command Syntax

```
SOURce:SCENario:CHINview? <ALL|GPS|GLO|GAL|BDS|QZSS|IRNSS|SBAS>
```

Note

If no scenario is running, an error is returned.

Parameter

constellation – ALL returns all active channels, while a constellation value returns satellite index values for that constellation only. No argument is the same as ALL.

Example

SEND:

```
SOURce:SCENario:CHINview? GLO
```

READ:

```
3,5,9,12,14,17
```

6.5.3.46 SOURce:SCENario:SVINview?

Function

Queries a comma-separated list of SatID values which indicate which satellites are in view in the simulated sky. Duplicate and interference channels are ignored.

Command Syntax

```
SOURce:SCENario:SVINview? <ALL|GPS|GLO|GAL|BDS|QZSS|IRNSS|SBAS>
```

Note

If the scenario is not running, an error is returned.

Parameter

constellation – ALL returns all active channels, while a constellation value returns satellite IDs for that constellation only. No argument is the same as ALL.

Example

SEND:

```
SOURce:SCENario:SVINview? GLO
```

READ:

```
R2,R5,R9,R11,R12,R17
```

6.5.3.47 SOURce:SCENario:SVPos[n]?

Function

Queries a satellite's ECEF position using channel number.

Command Syntax

```
SOURce:SCENario:SVPos[n]?
```

Note

If no scenario is running, an error is returned.

Parameter

Integer [1:N] – Satellite index of the satellite channel. Maximum is number of satellites

Example

SEND:

```
SOURce:SCENario:SVPos8?
```

READ:

```
13802999.54,18312013.72,13305242.14
```

6.5.3.48 SOURce:SCENario:SVPos[n]?

Function

Queries a satellite's ECEF position using a Satellite ID. The user can specify all satellite types supported including their multipath duplicates by satID. An optional location argument is specified to allow use with GSG's simulating user position or with systems using base station operation. If no location is specified, the user value is assumed.

Command Syntax

```
SOURce:SCENario:SVPos[n]? <satID>,<location>
```

Note

If no scenario is running, an error is returned.

If the base station location is not enabled, 0 values are returned.

Parameters

Integer [1:N] – Satellite index of the satellite. Maximum is number of satellites

satID – GPS, Glonass, Galileo, BeiDou, QZSS, IRNSS and SBAS are supported. The format is explained under "SOURce:ONECHN:SATid?" on page 252.

Location – user or base

Example

SEND:

```
SOURce:SCENario:SVPos? G20
```

READ:

```
13802999.54,18312013.72,13305242.14
```

6.6 Programming

6.6.1 Usage Recommendations

6.6.1.1 Communication Interface

It is strongly recommended to use USB in conjunction with RSG. USB is more reliable due to being a dedicated interface as opposed to Ethernet which can be more susceptible to network traffic. Ethernet should hence be avoided if attempting advanced steering using high message rates or requiring synchronization at the GSG 10Hz epoch rate.

GPIB can be used as an alternative but as there are synchronisation issues with GPIB, USB remains as the number one choice.

6.6.1.2 Synchronization

It is possible to synchronize SCPI commanding with GSG's internal processing loop, with a resolution of 100 ms. This can be achieved by using the `*WAI` and/or `*OPC?` commands.

For example, checking that the ECEF position command is applied on next 10 Hz epoch:

```
sour:scen:ecefposition IMMEDIATE,1000.0,2000.0,3000.0
*OPC?
sour:scen:ecefposition?
```

This synchronization can happen irrespective of whether an RSG command comes in.

For example, to see elapsed time "ticking" in 100 ms epochs

```
*OPC?
sour:scen:elapsedTime?
*OPC?
sour:scen:elapsedTime?
```

In addition, this synchronization mechanism can be used to consecutively to achieve any desired synchronization rate (max resolution of 100 ms, ie. at 10 Hz). For this purpose only `*OPC?` should be used. To use `*WAI` for this purpose the user would need to insert a small micro sleep, or perform suitable actions, between consecutive `*WAI` commands.

For example, to see elapsed time "ticking" every half a second the following commands can be looped:

```
...
*OPC?
sour:scen:elapsedTime?
*OPC?
*OPC?
*OPC?
*OPC?
*OPC?
```

```
sour:scen:elapsedTime?
*OPC?
...
syst:err?
```

6.6.1.3 Underflow and Overflow

Underflow and overflow errors are signaled by the GSG unit. The possible errors which can be retrieved with the command `SYSTem:ERRor[:NEXT]?`.

The relevant error codes are:

- **-193** "RSG command overflow occurred."
- **-194** "RSG command underflow detected."

The GSG unit will flag the overflow error in a situation where redundant or conflicting information is given during the same epoch, i.e., giving position both using the `SOURce:SCENario:ECEFPOSition` as well as the `SOURce:SCENario:POSition` command would trigger an overflow error. The overflow error will always trigger by default in such situations. Would redundant data come in the later commands will overwrite the earlier information.

The underflow error detection is by default not used but has to be explicitly set ON using the command listed in previous chapter. When in usage GSG will require at least one RSG command to come in every epoch (100 ms). Would there be an out take in this command stream GSG will set the error flag that indicates, e.g., problems in communication with host.

6.6.1.4 Best Practices

In a high rate control setup it is recommended that queries are avoided or kept to a minimum. The reason for this is to reserve the maximum time for the controlling commands.

The user must pay attention that the actual data sent in is smooth. The signal tracking in GNSS receivers are very sensitive to high dynamics and won't be able to track signals if position changes with several meters during one epoch. Hence it should be preferred to change user position using the more dynamic speed and acceleration commands, as a 'blunt' position change has to be smooth and small for the receivers to be able to follow. Hence using position/speed commands you only need to send commands when these parameter values changes. Relying on position commands you are recommended/forced to utilize 10 Hz commands to make the movements smooth enough for receivers to follow.

6.6.1.5 Limitations

- Communication over GPIB is not currently working for RSG commands – synchronization fails.

» Communication over GPIB is not currently working for RSG commands – synchronization fails.

6.6.2 Trajectory FILE Format (.traj)

Files in the `.traj` file format can be created with the StudioView RSG Trajectory Editor, but an RSG license is not required to use these commands in a file.

All positioning commands can be written to the file as:

```
SOURce:SCENario:POSition TIME,<decimal>,<decimal>,<decimal>
```

...

or, without the `SOURce:SCENario:` part as:

```
POSition TIME,<decimal>,<decimal>,<decimal>
```

...

In the trajectory file format TIME must be a decimal number.

The resolution of the time stamps is 0.1 seconds (100 ms).

6.6.3 Trajectory Two-Line Element Format (TLE)

The two-line element, or TLE trajectory format has been supported by GSG since 2016. Two-line elements sets are used as a coordinate system, as utilized by State Vector descriptions of e.g., satellite positions and velocities.

While there is no editor for TLE-based trajectories in StudioView, you can select a TLE trajectory in a scenario, once you created it in a text editor.

The file extension must be `.tle`, and the file must contain 3 lines:

» First line: Header <the satellite text name>

» Second line: <data>

» Third line: <data>

The format is standardized and follows the definition outlined here:

http://spaceflight.nasa.gov/realdata/sightings/SSapplications/Post/JavaSSOP/SSOP_Help/tle_def.html

To use a TLE-based trajectory:

1. Create the above-mentioned 3-line file in any text editor.
2. Name the file `<any_name>*.tle`.

3. Select the file for use in a scenario the same way as any trajectory is selected.

6.7 Revision History (SCPI Guide)

SCPI Guide Revision History			
Rev	ECN	Description	Date
1.0draft	N/A	Initial issue.	4/2/2011
1.0	N/A	Minor comments & layout changes	11/2/2011
1.1	N/A	Added *SRE? and details about overlapping commands	8/3/2011
A	2673	Changes in support of the 2.06 software release.	June 2011
B	2702	Updated address information.	October 2011
C	2769	Added compliance section and updated regulatory information, additional minor document maintenance.	November 2011
D	2832	Updated title (SCPI Handbook), added information regarding GSG-52/56 models, GLONASS, new command information & updates.	March 2012
E	2929	Added GSG-53 model and various updates. Added change to SatId message and replaced 1ch with 1-channel.	May 2012
F	2990	Added GSG-62 model and set Start Time based on NTP time. Minor corrections.	July 2012
G		Updated for Real-time Scenario Generation commands. Minor correction.	December 2012
H	3150	Updates corresponding with latest software release & product enhancements.	February 2013
J	3179	Added factory reset command.	March 2013
K	3197	Minor corrections & updates.	April 2013
L	3254	Supports latest hardware & software revision.	June 2013
M	3347	New commands and updates to support latest firmware release	March 2014
N	3458	New commands and updates to support latest firmware release	May 2014
15	000073	New commands/Sensor option reference/support for 6.1.1 firmware	July 2014
16	000194	New/updated commands to support 6.2.1 firmware release	October 2014
17	000293	New/updated commands to support 6.3.1 firmware release	February 2015

SCPI Guide Revision History			
Rev	ECN	Description	Date
18	000421	New/updated commands to support 6.4.1 firmware release	May 2015
19	000587	New/updated commands (mainly Propagation Environment) to support 6.5.1 firmware release. New layout due to carry-over into new Authoring tool. Integration of SCPI Guide into GSG User Manual: Future revision history tracking see GSG User Manual revision table (see Appendix)	Sept 2015

BLANK PAGE.

APPENDIX

Appendix

The following topics are included in this Chapter:

7.1 Lists of Tables and Images

Tables in this document:

Images in this document:

7.2 GSG User Manual Revision History

Rev	ECO	Description	Date
0.1	N/A	First release.	November 2010
1.0	N/A	Updated to include GSG-55.	March 2011
A	2673	Changes in support of the 2.06 software release.	June 2011
B	2702	Updated address information.	October 2011
C	2769	Added compliance section with updated regulatory information. Additional minor document maintenance.	November 2011
D	2832	Updates including information supporting GSG-56 product, GLONASS support, new software features. Additional document maintenance.	March 2012
E	2929	Added support for GSG-53 product, additional corrections.	May 2012
F	2990	Added support for GSG-62 product features and NTP Server as a source for Start Time.	August 2012
G	2999	Minor updates.	August 2012
H	3015	Minor corrections & specification updates.	September 2012
J	3128	General updates coinciding with latest software release: newly released GSG-62 product & features, added information regarding new platform software feature enhancements.	December 2012
K	3150	Updates coinciding with latest software release. Added information regarding product feature enhancements.	February 2013
L	3179	Updates related to addition of new platform software feature enhancements and clarified existing documentation regarding NMEA file length.	March 2013
M	3197	Minor corrections and updates.	April 2013

Rev	ECO	Description	Date
N	3254	Updated to support latest software & software release modifications	June 2013
P	3347	Updated to support latest software & software release modifications	March 2014
Q	3458	Updated to support latest software & software release modifications	April 2014
18	000073	Updated to support latest software & new features	July 2014
19	000194	Updated to support latest software & new features	October 2014
20	000293	Updated to support latest software & new features	February 2015
21	000421	Updated to support latest software & new features	May 2015
22	000587	Updated to support latest software & new features (mainly Propagation Environment functionality) to support 6.5.1 firmware release. New layout due to carry-over into new Authoring tool. Integration of SCPI Guide into GSG User Manual.	Sept 2015
23	000856	Added/changed content following SW release 6.6.1 (SCPI Clock Model), new options (TLM, HPWR, SPF). Changes to Trajectories topic, Encryption topic. Ongoing document maintenance.	April 2016
24		New RSG ADVLOG SCPI commands. Content improvements, power setting changes. Errata changes.	Oct 2016
25		Added StudioView instructions, some functionality descriptions for Galileo -related options, spoofing, new SCPI commands. Errata.	May 2017
26	DOC-000153	Changes to Transmit Power adjustment. Added several SCPI power commands. Errata.	Jan 2018
27		Added commands corresponding to latest firmware. Branding rework.	Oct 2019

BLANK PAGE.

INDEX

N

O

P

R

S