R&S®PULSE-K32/-K39 Pulse Sequencer Digital, Standard Version and Expert Upgrade User Manual

1179387302
Version 07

ROHDE&SCHWARZ
Make ideas real

This document describes the following software options:

- R&S®PULSE-K32 standard version (1414.7077.22)
- R&S®PULSE-K39 expert upgrade (1414.7125.22)

This manual describes software version V2.8 and later of the R&S®Pulse Sequencer Digital.

Muehldorfstr. 15, 81671 Muenchen, Germany
Phone: +49 89 41 29 - 0
Email: info@rohde-schwarz.com
Internet: www.rohde-schwarz.com

1179.3873.02 | Version 07 | R&S®PULSE-K32/-K39

Throughout this document, R&S® is indicated as R&S.

Contents

1 Welcome to the R&S Pulse Sequencer Digital

The R&S Pulse Sequencer Digital is an application that generates complex pulse signals by using predefined, configurable test scenarios with different complexity. You can simulate the signals of different emitter and receiver configurations, including antennas and scan types.

The R&S Pulse Sequencer Digital software is a standalone, PC-based application that creates simulated signals that you can export to a customized plugin. You can use R&S Pulse Sequencer Digital for example to perform radar receiver tests and for research and development.

Licenses

A license is not required for the demo version of the application.

To obtain a license provider (R&S PULSE-LPA) and license activation codes for R&S PULSE-K32 or R&S PULSE-K39, contact Rohde & Schwarz support.

For details, see:

- Description of the key features: Chapter 1.1, "Key features", on page 11.
- Licenses: Chapter 2.3, "Licensing", on page 18.

Installation

For details on installation and activating licenses, see:

- Installation instructions are included in the delivery of the option.
- Instructions are described in Chapter 2.2, "Installing the software", on page 16.

1.1 Key features

R&S PULSE-K32 standard features:

- Pulse definition
- Modulation on pulse with all major formats like chirps, Barker codes, polyphase codes, PSKs, AM, FM
- Single pulse, pulse train generation with repetition count per pulse
- Powerful sequencing tool with loops, nested loops, and fillers
- Interpulse modulation of amplitude, phase, frequency, etc. values from pulse to pulse
- Calculation of signal under consideration of one-way free space propagation according to emitter and receiver location on the 2D map
- Import of PDW files
- A "Platform" element that represents a real-world vehicle that can contain multiple emitters.

- The realization is similar to the DF receiver model.
- Platforms are mapped to a certain inertia profile (car, ship or plane).
- Platforms have a tactical icon in dynamic scenarios.
- Three-dimensional placement of emitters on the platform via 3-axis concept (X, Y and Z).
- A 2D and 3D preview of the emitter placement.
- Emitter definition by signal, antenna diagram, antenna scan, attitude information, EIRP, and carrier frequency
- Antenna diagram definition and antenna scan definition
- Antenna diagrams like pencil beams, cosecant beams, Gaussian diagrams, user-defined antenna diagrams, phased array antenna diagrams
- Antenna scan types like helical scans, circular scans, conical scans
- Receiver definition by antenna diagram, antenna scan, and attitude information

R&S PULSE-K39 expert features:

- Allows merging of multiple PDW lists in the "PDW list (Collection)" scenario type or multiple emitters in the "Emitters (Collection)" scenario into a single output file.
  The feature uses a priority scheme for dropping or just merges the pulses even if they overlap in time
- Enhances the localized and direction finding scenario types by movement profiles for emitters and receivers
- Predefined line and arc movements for simple movement traces
- Waypoint generation from user-defined "Traces" in the GUI
- Waypoint import interface for complex movement traces
- WGS84 waypoint interface and import of NMEA waypoints
- Import of Google Earth and Google Maps `.kml` files
- East-North-Up (ENU) 2D vector trajectory interface (line, arc) for automatic waypoint generation
- Motion interface for dynamics input (velocity vector or velocity magnitude) in ENU and WGS84
- Predefined waypoint files for land vehicles, ships, aircraft and spacecraft
- User-definable and predefined vehicle description files for land vehicles, ships, aircraft and spacecraft
- Smoothing of waypoints using vehicle description files
- Use georeferenced maps as visual backgrounds for 2D and 3D map views.
- Import the following map formats:
  – GeoTIF
- Create trajectories using the "Trace" function.
  This function uses a series of user-selected points on the map (i.e. a user-defined trace) to generate waypoints, automatically.
- Dedicated direction finding scenario and receiver element with multiple antennas and individual positioning

- Direction finding receiver definition with up to 20 antennas with individual positioning and pointing

1.2 Accessing the R&S Pulse Sequencer Digital

To open the application

► On your PC, select "Start > R&S Pulse Sequencer Digital > R&S Pulse Sequencer Digital".

The software opens and per default displays the last opened workspace.

1.3 What's new

This manual describes software version V2.8 and later of the R&S®Pulse Sequencer Digital.

Compared to the previous version, it provides the following new features:

- Added the sequence selection in a sequence collection scenario to the signal generation block, see "Sequence (Collection) Scenario Settings" on page 85.
- Added RX information and velocity to the 2D live preview stats, see "2D Live Preview Stats" on page 299.
- Editorial changes.

1.4 Documentation overview

This section provides an overview of the R&S Pulse Sequencer Digital user documentation. Unless specified otherwise, you find the documents on the R&S Pulse Sequencer Digital product page at:

https://www.rohde-schwarz.com/manual/pulse-sequencer/

1.4.1 User manuals and help

Introduces the R&S Pulse Sequencer Digital and describes how to start working with the product. Includes basic operations, typical examples, general information, and the description of all software modes and functions. It also provides a complete description of the remote control commands with programming examples.

The contents of the user manuals are available as help in the R&S Pulse Sequencer Digital. The help offers quick, context-sensitive access to the complete information.

All user manuals are also available for download or for immediate display on the Internet.

1.4.2 Specifications and product brochures

The specifications document, also known as the data sheet, contains the technical specifications of the R&S Pulse Sequencer Digital. It also lists the options and their order numbers as well as optional accessories.

The brochure provides an overview of the R&S Pulse Sequencer Digital software and deals with the specific characteristics.

See https://www.rohde-schwarz.com/brochure-datasheet/pulse-sequencer/.

1.4.3 Release notes and open source acknowledgment (OSA)

The release notes list new features, improvements and known issues of the current firmware version, and describe the firmware installation.

The open source acknowledgment document provides verbatim license texts of the used open source software.

See https://www.rohde-schwarz.com/software/pulse-sequencer/.

1.4.4 Application notes, application cards, white papers, etc.

These documents deal with special applications or background information on particular topics.

See https://www.rohde-schwarz.com/application/pulse-sequencer/.

1.5 Scope

This user manual contains a description of the functionality that the software provides, including remote control operation.

1.6 Notes on screenshots

When describing the functions of the product, we use sample screenshots. These screenshots are meant to illustrate as much as possible of the provided functions and possible interdependencies between parameters. The shown values may not represent realistic usage scenarios.

The screenshots usually show a fully equipped product, that is: with all options installed. Thus, some functions shown in the screenshots may not be available in your particular product configuration.

2 Getting started

This section describes the basic steps to be taken when starting up the R&S Pulse Sequencer Digital for the first time.

2.1 Required options and licences

The R&S Pulse Sequencer Digital software is a standalone, PC-based application that creates digital simulation data for the verification of software algorithms, digital training and digital validation of ELINT systems.

The R&S Pulse Sequencer Digital is available in three operation modes:

- Demo
  A demo version that enables you to try out some of the main features of the application including access to the visualizations. However, you cannot export simulation data.
  This option does not require a license.
- Standard (R&S PULSE-K32)
  A standalone application offering basic simulation features. Provides a range of standard features and functionality, including exporting of simulated signal data
  Requires a license.
- Expert (R&S PULSE-K39)
  An optional extension to R&S PULSE-K32 that provides expert functionality.
  Requires a license.

Any of the licensed modes require a license dongle. The license provider R&S PULSE-LD is available as a separate product and *must be ordered in addition to the software licenses*.

For information about licenses, see Chapter 2.3, "Licensing", on page 18.

2.2 Installing the software

Software updates and the release notes describing the improvements and modifications are available for download at the product page https://www.rohde-schwarz.com/software/pulse-sequencer/.

The following software components must be installed to run the R&S Pulse Sequencer Digital successfully:

- R&S License Server
- VISA (Virtual Instrument Software Architecture)

The R&S License Server is installed automatically during installation of the R&S Pulse Sequencer Digital. VISA can be installed directly during installation of the R&S Pulse Sequencer Digital, or manually, independently of the R&S Pulse Sequencer Digital installation. We recommend that you use the R&S VISA driver. The R&S VISA driver is supplied with the R&S Pulse Sequencer Digital installation file, and can be installed together with the R&S Pulse Sequencer Digital software.

Software and hardware requirements

Minimum requirements:

- **Software**
  – Microsoft®Windows 10, 64-Bit operating system (x64-based or x86-based processor)
  – Microsoft®Windows 11, 64-Bit operating system (x64-based or x86-based processor)
- **Hardware**

Table 2-1: Hardware requirements

	Minimum requirements	Recommended hardware
AMD or Intel CPU	Dual Core, 2 GHz	Intel 8 Core i7 10700, AMD Ryzen series
RAM	≥ 2 GB	64 GB
Video	NVIDIA Quadro 128 MB or ATI Radeon	NVIDIA Quadro 128 MB or ATI Radeon
Video resolution	≥ 1280 x 1024 pixels	≥ 1920 x 1200 pixels
Network	LAN 1 GB/s	LAN 1 GB/s

To install the software

Download the R&S Pulse Sequencer Digital software form the Rohde & Schwarz website.

The R&S Pulse Sequencer Digital software consists of the file
`PS-DIGITAL-Install <major>.<minor>.<build>.<release>-x64.msi`.

1. Start the
   `PS-DIGITAL-Install <major>.<minor>.<build>.<release>-x64.msi`
   file.

   The filename follows the naming conventions:
   - `<major>.<minor>` is the software version
   - `<build>` is the build number.
     This is the day elapsed since 1 January 2000
   - `<release>` indicates the release on the build date.
     This is 1/10 of the number of seconds elapsed since midnight.
   - `x64` indicates that this is a 64-bit installation file.
     You can run the software version with the extension `x64` only on a 64-bit operating system, see "Software and hardware requirements" on page 16.
2. Depending on your user rights, select one of the following:
   a) Installation for all users (default)
      - Requires administrator rights
      - Is a per-machine installation
      - Project data and program data are common to all users
      - Users keep their individual settings

   b) Installation for current user only
      - Does not require administrator rights
      - Is a per-user installation
      - Installation in `%HomePath%`
3. Choose the setup type.
4. Follow the installation instructions.

   The installer performs the following actions:
   - If enabled, installs the R&S VISA software on the PC
   - Installs the R&S License Server
   - Installs the R&S Pulse Sequencer Digital software including an uninstall tool
   - Creates a shortcut on the desktop
   - Creates the folder structure listed in Table 2-2.

Table 2-2: Default file location (software installation for all users)

File type	File location
Program data	`%PROGRAMFILES(X86)%\Rohde-Schwarz\Pulse Sequencer Digital` (`Path`)
Project data (Repository database) SDK files	`%PUBLIC%\Public\Documents\Rohde-Schwarz\Pulse Sequencer Digital` (`DataPath`)

File type	File location
Report files, startup log file, calculated data, volatile data	`%HOMEPATH%` (`HomePath` or `ReportPath`)
User settings (Workspace, etc.)	`%HOMEPATH%\AppData\Roaming\Rohde-Schwarz\Pulse Sequencer Digital`

2.3 Licensing

A license is not required for the demo version of the R&S Pulse Sequencer Digital.

Running the R&S Pulse Sequencer Digital in standard mode requires a valid core license R&S PULSE-K32, and, depending on the desired features, additional licenses for options, like R&S PULSE-K39 for expert mode are required.

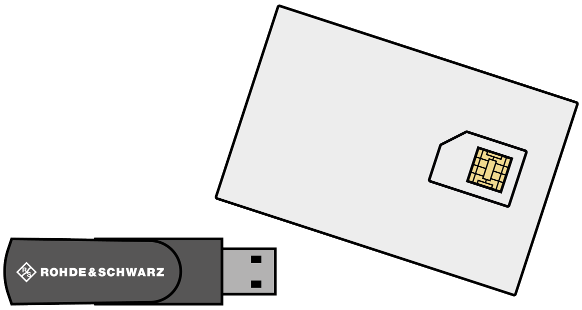
R&S PULSE-LD

The R&S Pulse Sequencer Digital software requires a smart card containing the software license to be connected to the PC when you are using the software. The R&S PULSE-LD license provider contains the purchased software license and consists of a smart card and a USB dongle. The R&S PULSE-LD is available as a separate product and must be ordered in addition to the software.

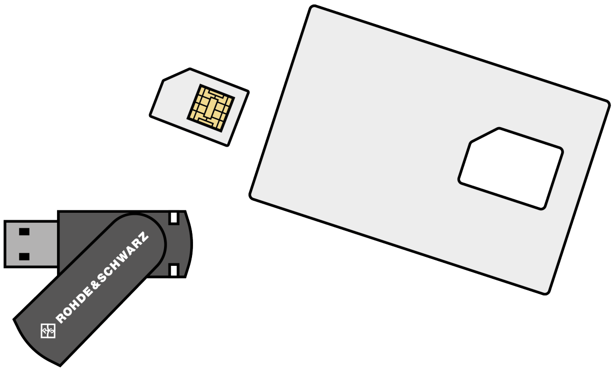
To obtain the license provider R&S PULSE-LD and license activation codes for R&S PULSE-K32 or R&S PULSE-K39, contact Rohde & Schwarz support.

Using the license provider R&S PULSE-LD

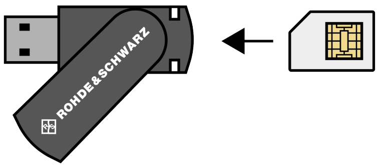
1. The R&S PULSE-LD license provider consists of a smart card in full format and a USB smart card reader (dongle).



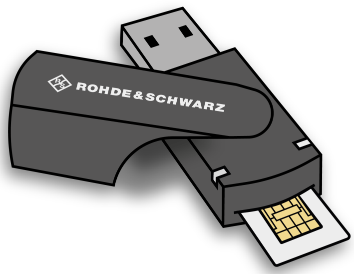
2. Break out the smart card in SIM format.



3. Twist out the upper part of the smart card reader.
4. Insert the smart card with the chip facing upwards and the angled corner facing the USB dongle, whose "Rohde & Schwarz" label is also facing upwards.



Insert the smart card as far as possible.



5. Twist the smart card reader back into its original state.

The license provider is ready for use on any USB interface.
6. Connect the license provider to the PC the R&S Pulse Sequencer Digital is running on.

The R&S License Server, that is installed automatically with the R&S Pulse Sequencer Digital installation, detects the license provider and the software licenses.
The software is ready to use.

Activating licenses

When you order a license and a license provider at once, the purchased licenses are already registered, activated and are delivered on the license provider.

If you purchase a license, for example R&S PULSE-K39, at a latter time, you receive a license key, registered for your license provider and need to activate the license.

1. Connect the license provider to the PC the R&S Pulse Sequencer Digital is running on.
2. Open the "R&S License Server".
3. Select "Activate".
4. In the License Key field, enter the registered license key you have received.
5. Follow the instructions.
6. Restart the "R&S License Server".
7. Start the R&S Pulse Sequencer Digital.

Missing licenses

If the R&S Pulse Sequencer Digital complains about missing licenses, run the R&S License Server Manager browser application.

1. On the R&S Pulse Sequencer Digital PC, select "Windows" > "Start" > "R&S License Server" > "R&S License Server Manager".
2. Check if the required R&S PULSE-LD license is visible in the "Licenses > Licence providers" tab.
3. Check if the required R&S PULSE-K32 or R&S PULSE-K39 licenses are visible in the "Licenses > Licences" tab.

   For more information, on the R&S License Server:
   - In the "R&S License Server Manager" web interface, click the "Help" icon.
   - Open Windows start menu and select "Open License Server Manual" to access the R&S License Server user manual in PDF format.

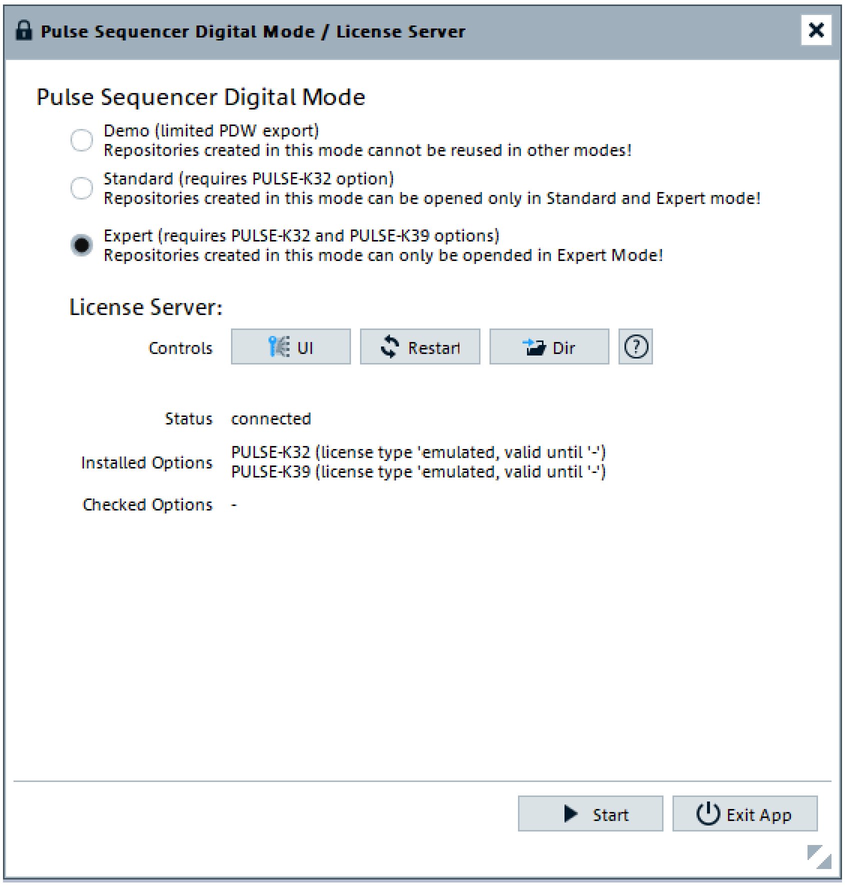
2.4 Starting the R&S Pulse Sequencer Digital for the first time

1. Check that you have a valid license registered on your PC.

   Note: If you do not have a license provider and license registered yet, you can run the software in demonstration mode. Continue with this procedure and select "Demo Mode" from the list of options.
2. On your PC:

a) Select "Start > R&S Pulse Sequencer Digital > R&S Pulse Sequencer Digital" or
b) Double click the shortcut icon on the desktop.

The "Pulse Sequencer Digital Mode / License Server" dialog opens.



Note: Dialog options correspond to registered licenses. You cannot select an option in the dialog unless the required license is registered. "Installed Options" displays the available licenses.

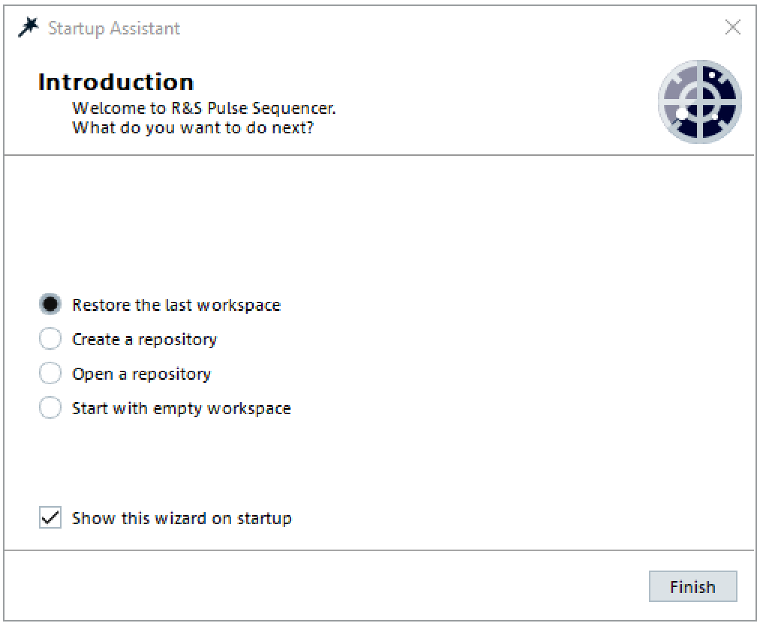
3. In the "Pulse Sequencer Digital Mode / License Server" dialog, select the required mode from the available options.

   a) For example, select "Pulse Sequencer Digital Mode = Demo".
   This option is always available, with or without license.
   Select this option if you want to experiment with the R&S Pulse Sequencer Digital functionality without producing PDWs.

b) If the respective license or combination there of are available and registered, select one of the following options:
   - "Pulse Sequencer Digital Mode = Standard (requires R&S PULSE-K32)"
     Select this mode if you want to create PDWs from scenarios that do not require advanced functions such as DF and maps.
   - "Pulse Sequencer Digital Mode = Expert (requires R&S PULSE-K32 and R&S PULSE-K39)"
     Select this mode if you require advanced functions such as DF, interleaving, movements and maps.

4. Select "Start".

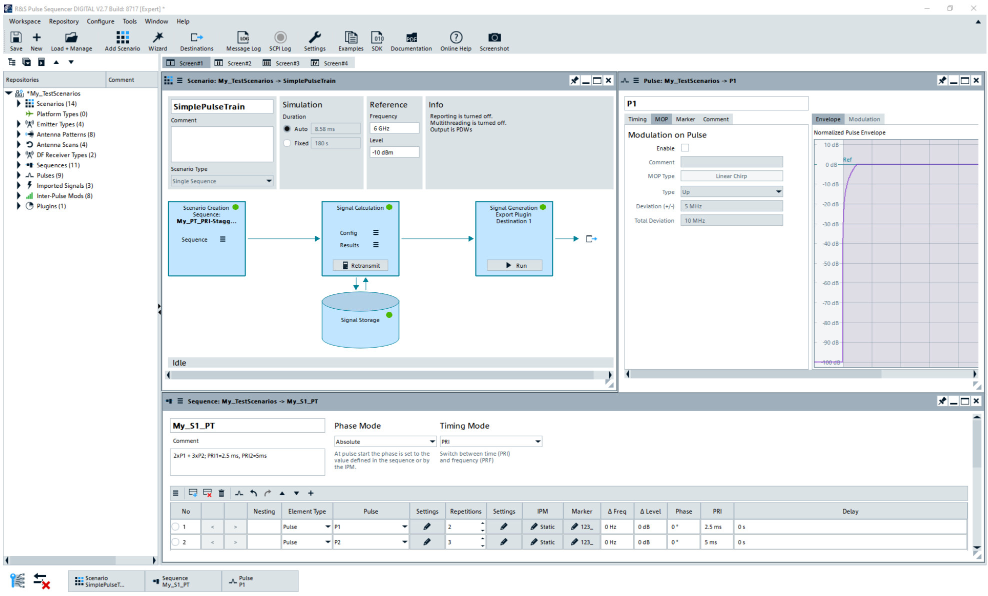
   The "Startup Assistant" opens.



5. Select "Create a repository" > "Finish".

6. Follow the instructions. Use the default settings when starting R&S Pulse Sequencer Digital for the first time.

   A new repository ("New_<Year>-<Month>-<Day>T<Time>") is created that contains a simple scenario ("SimplePulseTrain").
   Your workspace resembles the one shown in the following figure.



See also:

- Figure 2-1 (in Chapter 2.5, "Understanding the displayed information", on page 23)
- Chapter 2.6, "Software concept in brief", on page 29
- Chapter 2.8, "Trying out the software", on page 34

2.5 Understanding the displayed information

The following figure shows an example test scenario during software operation. The different information areas are labeled. They are explained in more detail in the following sections.

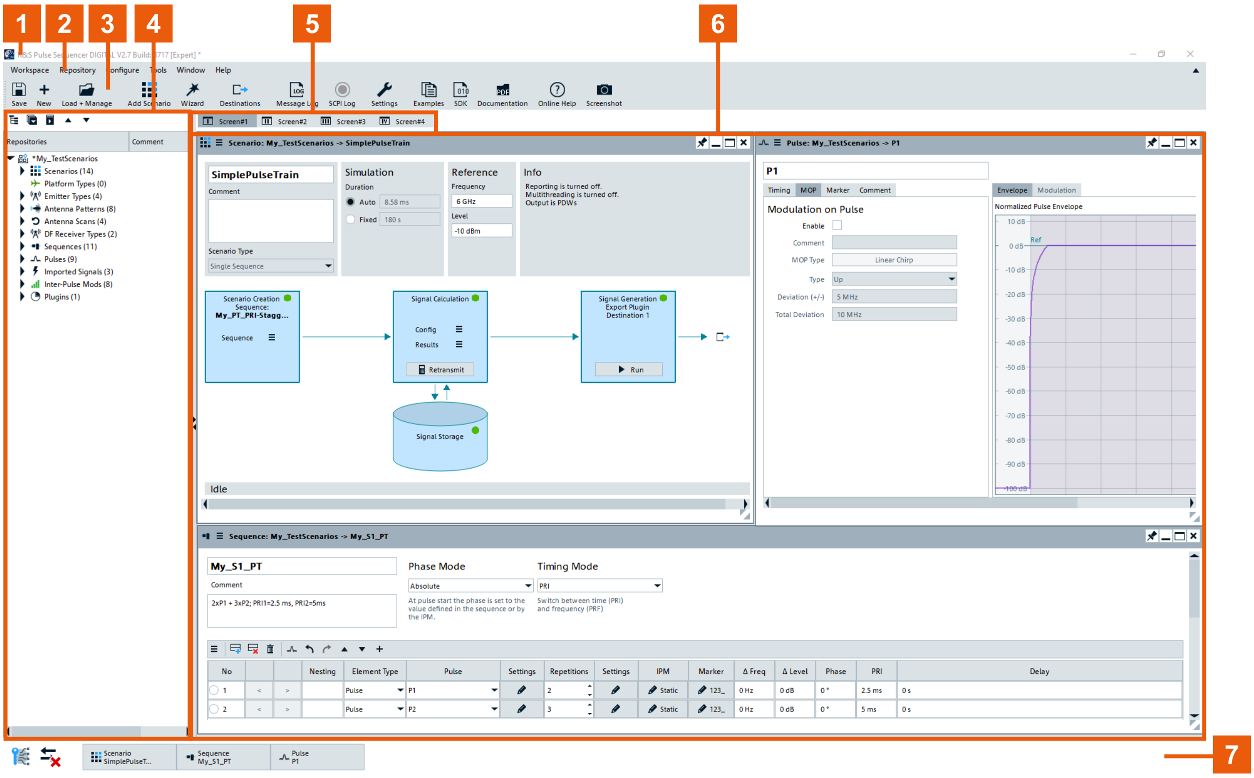


Figure 2-1: R&S Pulse Sequencer Digital default workspace: understanding the displayed information

1 = Title bar
2 = Menu bar with standard functions, e.g. save/open file etc.
3 = Toolbar
4 = Project tree (repository tree)
5 = Screens for individual scenarios
6 = Work area
4+6 = Workspace
7 = Taskbar with indication of enabled remote control incl. details like IP address (if enabled), information on the required highest security level (if enabled) and access to active dialogs

The user interface of the R&S Pulse Sequencer Digital offers an intuitive operating concept. Operation is similar to any operating system. All menus and dialogs use known elements, e.g., selection lists, checkboxes, and entry fields.

Menu bar

Table 2-3: Content of the menu bar

Workspace	Repository	Configure		Tools		Window	Help
• Clear • Open • Save • Save As • Exit	• New • Load or Manage • Save • Import • Export • Edit	• Destinations • Remote Control • Colors • Settings		• Script • Console • Wizard		• Message Log • SCPI Log • MinSize All Windows • Cascade • Close All • Screen 1 .. 4	• Online Help • Documentation • Examples • SDK • Tutorials • Debug Information • About

Toolbars

The R&S Pulse Sequencer Digital offers two toolbars.

Standard functions can be performed via the icons in the main toolbar at the top of the screen. Click the arrow on the right-hand side of the menu bar to expand and collapse the toolbar.

Figure 2-2: Standard functions in the main toolbar

1 = Save as: stores current workspace configuration
2 = Repository: Creates a repository/Opens "Manage Repository" dialog for loading, discarding or exporting repositories
3 = Project tree: Creates a scenario/Starts a wizard to create an emitter or pulse train in current repository
4 = Opens the "Destinations" dialog
5 = Logs: Opens the "Message Log" window/"SCPI Log" window
6 = Opens the "Program Settings" dialog
7 = Help: Opens the "Example Files" folder/Opens the "SDK" folder/Opens the help documentation folder containing help documents/Displays context-sensitive help topic for currently selected element
8 = Hardcopy: Copies current dialog to the clipboard

Functions relating to the "Project Tree" can be performed via the icons in the toolbar below the main toolbar.

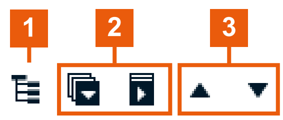


Figure 2-3: Functions in the Project Tree toolbar

1 = Switches between tree view and table mode, see Chapter 4.3, "Repository view", on page 67
2 = Expand/collapse all nodes in the "Project Tree"
3 = Moves the elements within a node up/down in the "Project Tree"

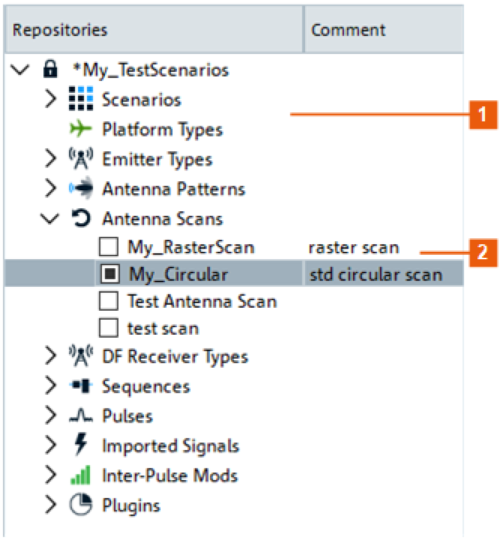
Virtual screens

Using the R&S Pulse Sequencer Digital you can handle several scenarios with even comprehensive settings at the same time. You can arrange the dialogs across multiple screens logically, so that you can easily switch between them. You can, for example, assign the subset of antenna dialogs to one screen and the pulse and modulation dialogs to another.

Repository tree

The repository tree shows the content of all currently loaded repositories. Repository elements are indicated with the assigned icons, their names, and the first comment line.

You can also switch the tree view to table mode, see Chapter 4.3, "Repository view", on page 67.

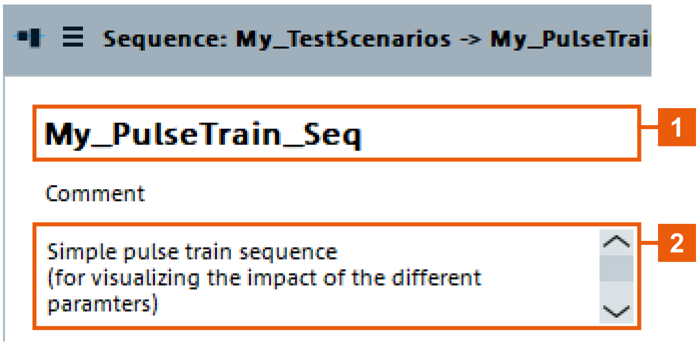


1 = Name, see "Name and comment" on page 26
2 = First comment line

Name and comment

Each repository element is identified with its unique name and optional description entered in the comment field. The description can contain several lines; new line is added by pressing [Shift+Enter].

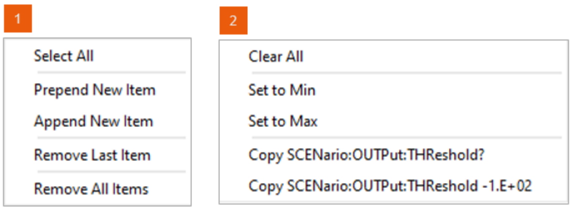
See also "Dialog names" on page 27.



1 = A unique name to indicate the element in the repository
2 = Description; the first line is displayed in the repository tree

Context-sensitive menus

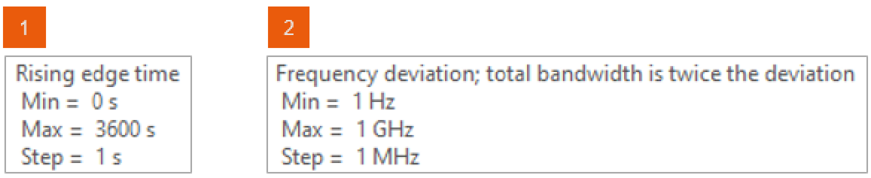
All user interface controls provide a context-sensitive menu. Context-sensitive menus are accessed with a right mouse click the control.



1 = Context menu of an icon
2 = Context menu of a parameter

Tooltips

Tooltips indicate the possible value range of a parameter or list overview information on current settings.

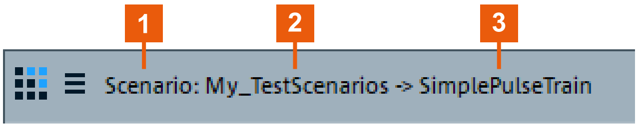


1 = Tooltip of a setting parameter, indicating the allowed value range
2 = Tooltip with detailed information on a setting

Dialog names

The name of a dialog indicates its content or the functionality this dialog covers. The used naming convention helps you to recognize which database element you are configuring, to which scenario it belongs and what its current name is.

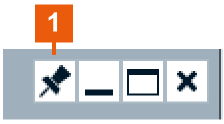
"<Repository Element>: <Scenario Name> -> <Function Name>"



1 = Repository element
2 = Scenario name
3 = Function name

Undocking dialogs

If your workspace contains several simultaneously opened dialogs, you can undock dialogs and distribute them also outside the workspace, for example place them on a second monitor.

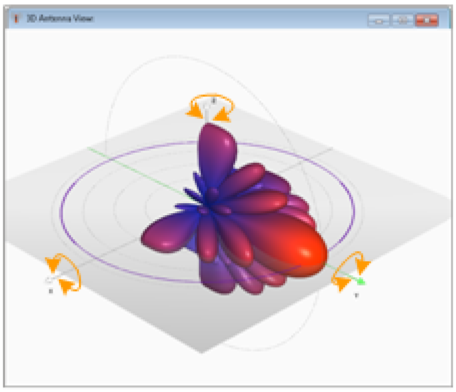


1 = Undocks the dialog

Interactive 3D displays

Several dialogs display the configured settings as 2D or as interactive 3D models or full 3D antenna radiation patterns.

Turning the mouse wheel zooms in and out on these dialogs. Moving the mouse while holding the left mouse key rotates the 3D view around its origin.



Interactive 2D map

Drag&drop elements into the 2D map or pan and zoom to change the displayed map area.

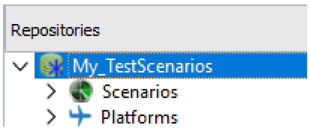
Keyboard shortcuts

Keyboard shortcuts are available for the most common functions in the software.

2.6 Software concept in brief

The following is brief introduction to the software concept and the way it handles files and settings.

Repository



The R&S Pulse Sequencer Digital organizes the data in repositories.

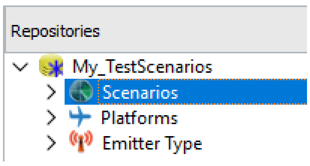
A repository is a **file-based database** located on the local hard drive or on a network storage drive.

Repositories are entirely managed by the R&S Pulse Sequencer Digital. They can be password protected against unauthorized access. Repositories can be exported as archives and shared among multiple users if they are located on a network drive. Several users can have read permission to load and use the same repository simultaneously. However, only one user can have a write permission at a time.

When you create a repository in one operational mode, it is not always compatible with the other modes. For a description see "Create" on page 65.

For more information, see Chapter 4, "Organizing the project data in repositories", on page 60.

Scenario



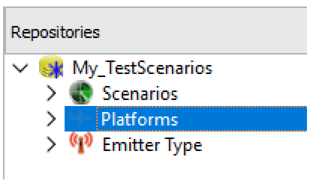
The scenario is the **top-level description** of the signals to be generated.

The type of the scenario determines which kind of signal is calculated and how this signal is processed. A more complex scenario combines several signals. Scenarios can also contain information about the time variation of the signals.

Starting a scenario creates the simulated data and exports them to user-written plugins.

For more information, see Chapter 5, "Selecting a suitable scenario and creating scenarios", on page 73.

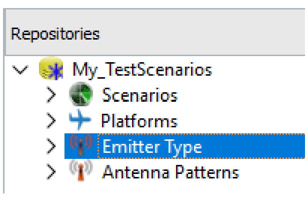
Platform



A platform is a **group of up to 8 emitters** assigned to a single vehicle. A platform can be static or moving. A platform is characterized by its icon, list of emitters, position and trajectory. The emitters on a platform can be individually configured.

For more information, see Chapter 12, "Creating platforms", on page 247.

Emitter



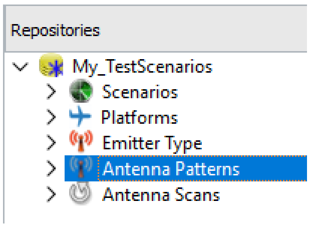
Emitters emulate **radar systems**. The emitter combines a sequence, an antenna scan, and an antenna pattern to one logical unit.

An emitter can work in more than one operating mode and switch between them. The *operating mode* is a description of the radar purpose, such as scanning, searching or tracking. For example, an airport surveillance radar can switch between different scan types to observe its airspace.

Each mode can have individual antenna and signal configuration. Multiple beams are possible within a mode.

For more information, see Chapter 16, "Creating complex 2D scenarios with receiver and TX items", on page 289.

Antenna

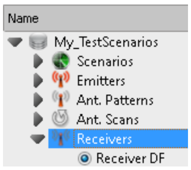


The antenna characterizes the **radiation pattern of the emitter**. Typical radar antennas like the phase array antennas can adapt their radiation pattern dynamically, depending on the operating mode the radar is working in. For example, an antenna can use a cosecant beam in scan mode and pencil beam in track mode.

Antennas can have various scan types. The scan type describes the movement of the antenna. For example, circular, sector and custom.

For more information, see Chapter 9, "Defining antenna patterns and antenna scans", on page 170.

Receiver

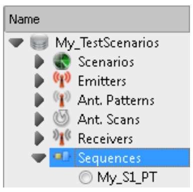


Receivers are used to model **direction finding systems**. They comprise of up to 20 individual antennas with individual patterns and positioning relative to a direction finding system.

Option:R&S PULSE-K39 required for direction finding.

For more information, see Chapter 13, "Emulating receivers", on page 259.

Sequence

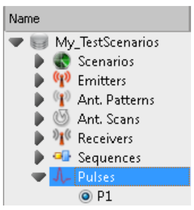


A sequence describes how pulses are arranged to form a waveform. The most simple sequence comprises one pulse that is repeated with a constant PRI (pulse repetition interval). Typical sequences however are rather complex; they can contain pulse definitions, waveforms, and sequencing elements such as repetitions and loops.

The sequence is a **logical description**. It cannot directly be represented as an I/Q waveform.

For more information, see Chapter 7, "Building pulse sequences", on page 119.

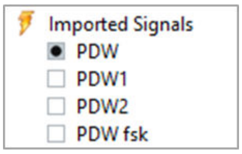
Pulse



In the R&S Pulse Sequencer Digital, pulses are the **fundamental building block** of any signal. Pulses are described mathematically with their amplitude (envelope), the applied modulation on pulse (MOP) and the enabled marker signals. One single pulse description is however not sufficient to calculate (and process) a signal; pulses have to be organized in a sequence, and assigned to an emitter.

For more information, see Chapter 6, "Creating a pulse library", on page 93.

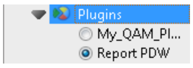
Imported signals



In addition to creating pulse signals you can also load PDW files and use them in sequences.

For more information, see Chapter 18, "Working with PDWs", on page 355.

Plugins



A plugin is an **extension** to the standard functions of the R&S Pulse Sequencer Digital.

For even more flexibility, plugins can contain variables.

To export the simulated signal from the R&S Pulse Sequencer Digital, you can write "export" plugins that are customized to your interfaces. The export plugin enables you to extract the digital simulation data from the application in a defined format and export it to different formats or directly stream the data.

For more information, see "How to import and assign user defined plugins" on page 377.

2.7 Means of users interaction

This chapter provides an overview on how to work with the R&S Pulse Sequencer Digital and describes the manual operation of the software and also the alternative ways of operation.

There are two ways to operate the R&S Pulse Sequencer Digital:

- Manual operation:
  Run the software on your PC and use the mouse and/or keyboard.
- Remote control:
  Create programs to automate settings, tests and measurements. The software is controlled by a program, running on the same or on another computer.
  This way of operation is described Chapter 23, "Automation of R&S Pulse Sequencer Digital", on page 408.

Basic operation concept

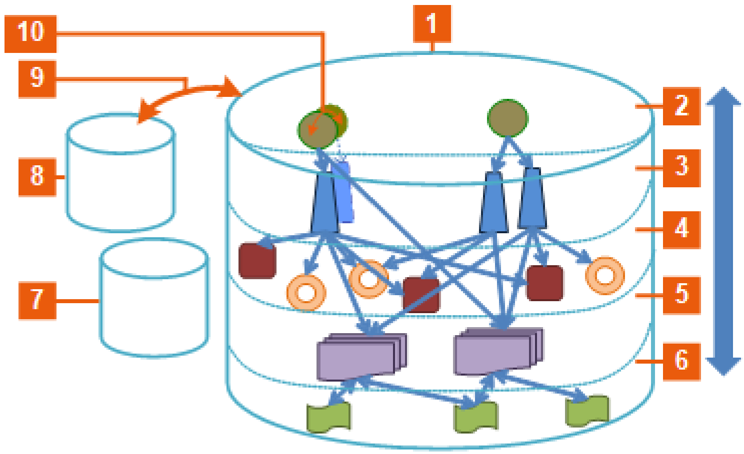
This section lists settings and functions that are common to the dialogs and are performed in the same manner throughout the software. You can use them alternatively or complementary.

As described in Chapter 2.6, "Software concept in brief", on page 29, the R&S Pulse Sequencer Digital software is based on a relational database that organizes database elements within repositories. Database elements are indicated by their names; within one repository, element names must be unique (see "Name and comment" on page 26).

Creating elements

Once created, database elements can be used by other elements within the same repository. Elements can be created in both the bottom-up and the top-down direction, or in a combination of these two methods:

- *Bottom-up:* you can create your pulse and waveform library first, then build sequences, create scenarios, and use the created sequences or assign them to emitters.
- *Top-down:* create your scenario, create emitter, define the antenna pattern/scan, define the used sequence, and define the pulses and waveforms.



1, 7, 8 = Repository, one or more within the database
2 = Scenario level; scenarios describe one or more emitters and one or more sequences
3 = Emitter level; emitters are described among others with antenna patterns and antenna scans
4 = Antenna patterns and antenna scans
5 = Sequence level; sequences are built of one or more pulses or waveforms
6 = Pulse and waveform level
9, 10 = Creating elements at any level with clone and copy

For information on the naming conventions and description of the database elements, see Chapter 2.6, "Software concept in brief", on page 29.

Handling database elements

While creating database elements, consider also the following possibilities:

- Change
  Files describing elements are stored once but can be used several times. When one file is changed, its parent elements are also updated.
- Clone
  Elements can be cloned to create a deep copy of the database element.
- Delete
  Elements can be deleted if they are not referenced by other elements.
- Copy
  If two repositories are opened in the project tree, a drag and drop operation creates a copy of the selected element and all referenced elements.

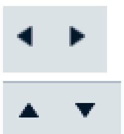
Standard function in the context menus

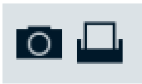
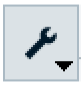
Each dialog with settings belonging to a "higher-level" element provides standard functions to:

- Assign existing elements
- Create elements
- Edit the existing assigned element.

You can access these functions from the context-sensitive menus of the icons listed in Table 2-4.

Table 2-4: Overview of the used icons and their functions

Icon	Name	Function/Description
	<Function> Menu (the name changes depending on the situation)	Smart menu with several options, depending on the current element and situation: • New (Item) • Append item • Prepend item • Insert Item • Insert Item Before/After • Delete Item • Clear (Selection) • Edit • Select (from a list) • Rename • Import/Export
	More	Select Edit New
	Pulse, Waveform	Creates repository element
	Antenna Pattern, Antenna Scan, Sequence	Accesses a dialog with detailed settings
	Emitter, receiver	
	3D, 2D diagram	
	Append item, Remove item, Delete	Appends or removes item, deletes all items
	Left/Right , Up/Down	Reorders items
	Copy and append	Copies the selected line of the table and appends it at the end of the table

Icon	Name	Function/Description
	Select	Selects a table row or an item for editing
	Enable/Disable	Activates or deactivates settings
	Hardcopy, Print	Copies current screen or dialog to the clipboard; Prints current diagram
	Name depends on contents	Opens a settings dialog. For example, "Visualization settings" or "Program settings".

Standard functions for elements handling

The following standard direct interaction functions are supported:

- Double-click
  Opens the selected element for editing
- Clone
  Creates a copy of the selected element
- Drag and Drop

  A drag and drop operation:
  - Creates a copy of the selected element, together with all referenced elements
  - Assigns an element to a 2D view
  - Imports a repository
  - Imports waveform files into the repository
- Mouse wheel
  Turning the mouse wheel zooms in and out.

Description of alternative methods

This manual describes the manual interaction with the software via the menus. Many elements that can be accessed form menus can also be accessed by clicking an icon or using the context menu. Alternative procedures are only described if they deviate from the standard operating procedures.

Throughout the manual, the term "select" refers to any of the described methods.

2.8 Trying out the software

This chapter introduces the most important functions and often used settings of the R&S Pulse Sequencer Digital step by step. The complete description of all functions and their usage is provided in the corresponding main chapters of this user manual.

Prerequisite

The software is running and started up as described in Chapter 2.4, "Starting the R&S Pulse Sequencer Digital for the first time", on page 20.

The software is manually operated. Try out the following:

2.8.1 Completing the scenario that was automatically created upon start up

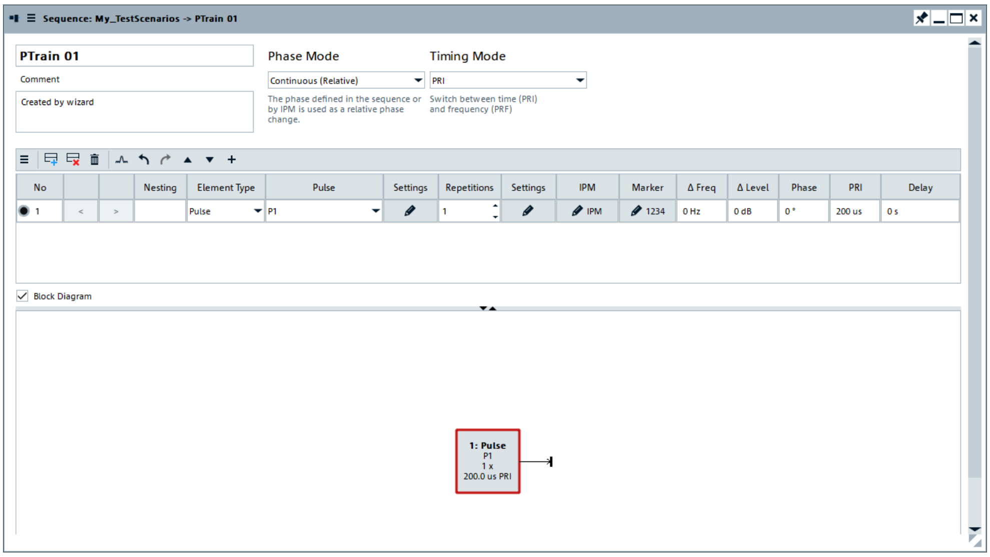
We assume that you have started the software as described in Chapter 2.4, "Starting the R&S Pulse Sequencer Digital for the first time", on page 20. Your workspace resembles the one shown on Figure 2-1 (see section Chapter 2.5, "Understanding the displayed information", on page 23).

Your repository contains:

- "P1": an unmodulated pulse with pulse width of 100 us and rise and fall times set to zero
- "PTrain 01": a sequence containing one single pulse, "P1". This pulse is processed once ("Rep. Count = 1") and has a pulse repetition interval "PRI = 200 us".

To create a sequence with 10 pulses, each having a PRI of 1 ms

1. In the repository tree, select "Scenario > Sequences > PTrain 01".
2. In the "Sequence Description" table, select:
   a) "Rep. Count = 10"
   b) "PRI = 1 ms"



For information on the provided settings, see:
- Chapter 7.2, "Sequence settings", on page 120
- Chapter 6.2.2, "Pulse timing settings", on page 94

To assign the sequence to the scenario

When created by the "Wizard", the sequence is automatically assigned to the scenario. If not, proceed as follows:

1. In the repository tree, select "Scenario > PTrain 01".

   The "Scenario" dialog opens and displays a block diagram of the signal flow.
2. In the block diagram, select "Sequence" and select the icon to open the context menu.
3. Select "Select > PTrain 01".

   The status indicator in the "Sequence" block is green.
   The status indicator in the "Signal Calculation" block is still red.

2.8.2 Generating a simulated signal

The initial situation is the configuration described in "To start the signal calculation" on page 37.

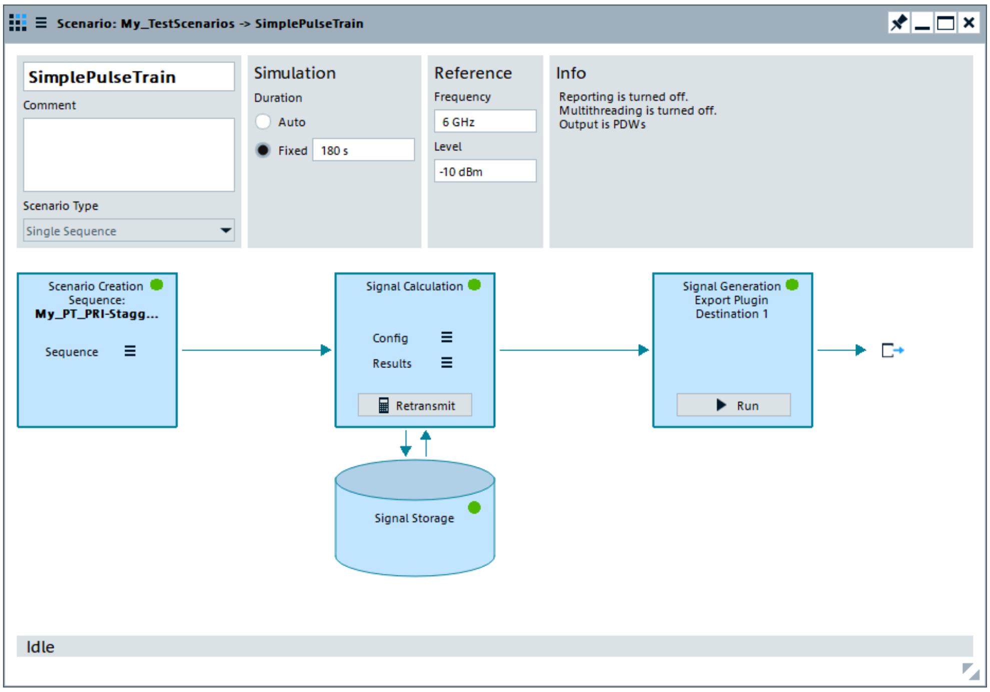
To set the output target for the created signal

1. In the repository tree, select "Scenario > SimplePulseTrain".

   The "Scenario" dialog opens and displays a block diagram of the signal flow.
2. In the block diagram, select "Signal Generation" and select the icon to open the context menu.
3. Select "Config > Target > Destination".

   A list of the available Destinations appears.
   A check mark indicates the currently selected destination.
4. Select a destination from the list.

   The block name changes to indicate the selected destination.
5. The block diagram shows the updated settings.



The "Info" area shows a status indication and displays information and warnings about the scenario. For example, if the destination does not support elements used in the scenario, the information displays here. The created signal is output is PDWs to the export plugin.
You can now change some of the calculation and output settings.

6. Select "Signal Calculation > Config > Config".
7. In the "Signal Calculation Settings" dialog, select "Output".

If all preconditions for signal generation are fulfilled, the "Calculate" button is active and you can start signal calculation.

To start the signal calculation

► In the "Scenario" dialog, select "Calculate".

The R&S Pulse Sequencer Digital calculates the simulated signal according to the selected settings and shows the current progress.
A green indicator in the "Signal Calculation" block confirms that the calculation is completed. The line between the "Signal Calculation" and the "Signal Generation" blocks confirms that the calculated signal can be transmitted to the selected target.

For information on the provided settings, see:
- Chapter 5.2, "Scenario settings", on page 74

2.8.3 Verifying the generated signal in the signal preview

It is often useful to check the generated signal visually, before you transfer it to the Destinations. The R&S Pulse Sequencer Digital provides a built-in function, the "Signal Preview", to represent the generated signal in a graphical form.

The initial situation is the configuration described in "To start the signal calculation" on page 37.

To open the "Signal Preview" and visualize the generated signal

► In the "Scenario" dialog, select "Signal Calculation > Results > View" and select the name of the calculated waveform.

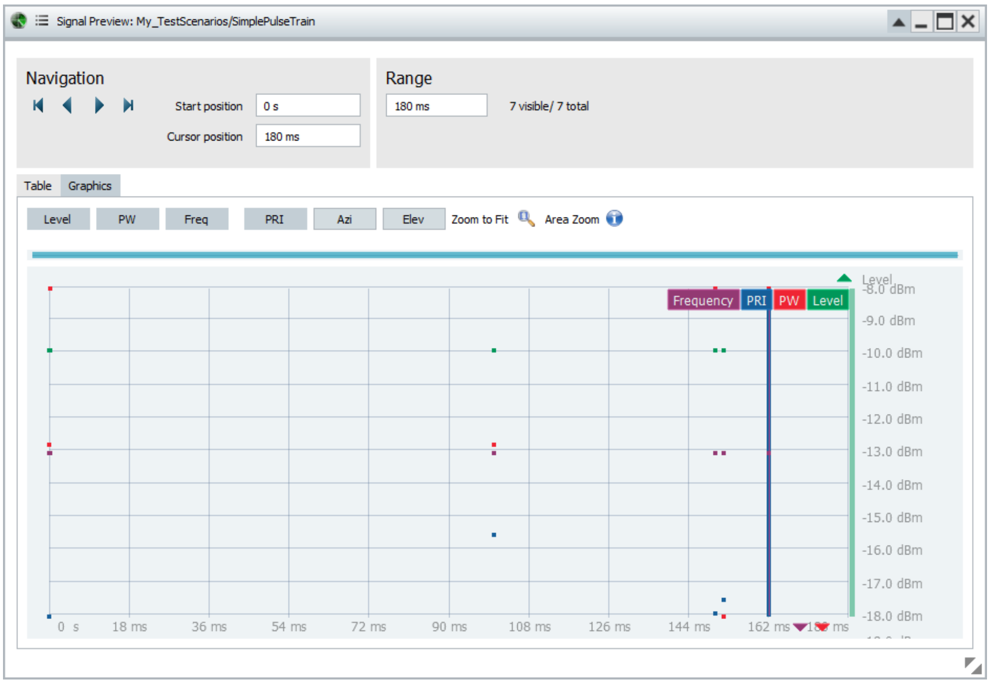


Figure 2-4: Signal Preview

2.8.4 Launching the built-in wizard

To help you get familiar with the software, the R&S Pulse Sequencer Digital provides a built-in startup assistant, the "Wizard".

You can access this "Wizard":

- **On an application start**:
  That is, each time you start the software
- **On demand**:
  That is, any time during operation when you need a fast configuration of few settings.

To start the built-in wizard

► In the tool bar, select "Tools > Wizard".

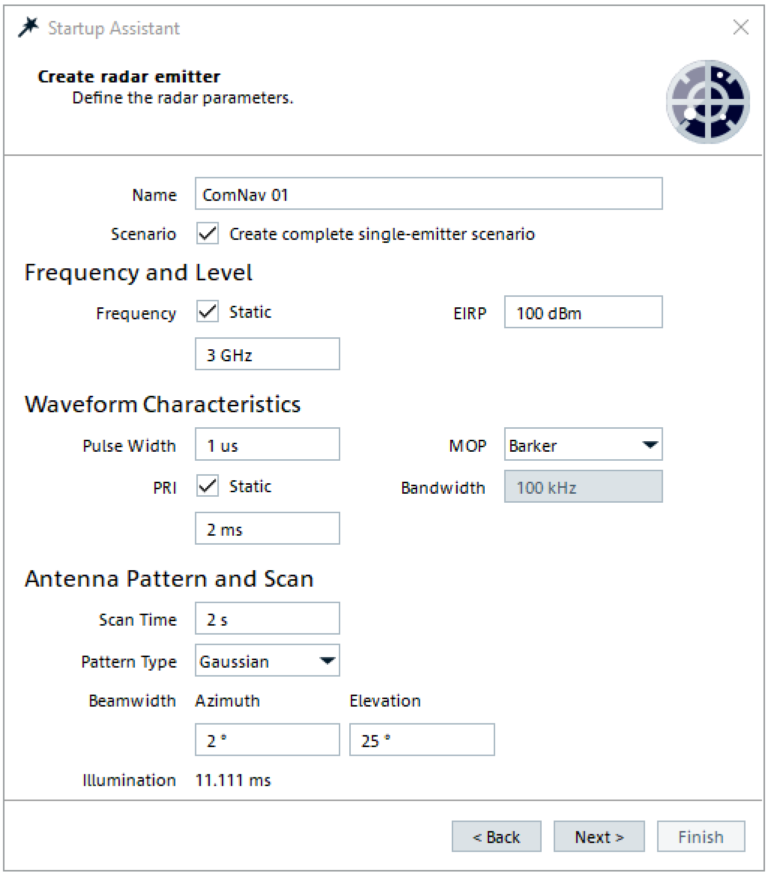
The "Startup Assistant" opens.

2.8.5 Using the wizard to create a complex scenario

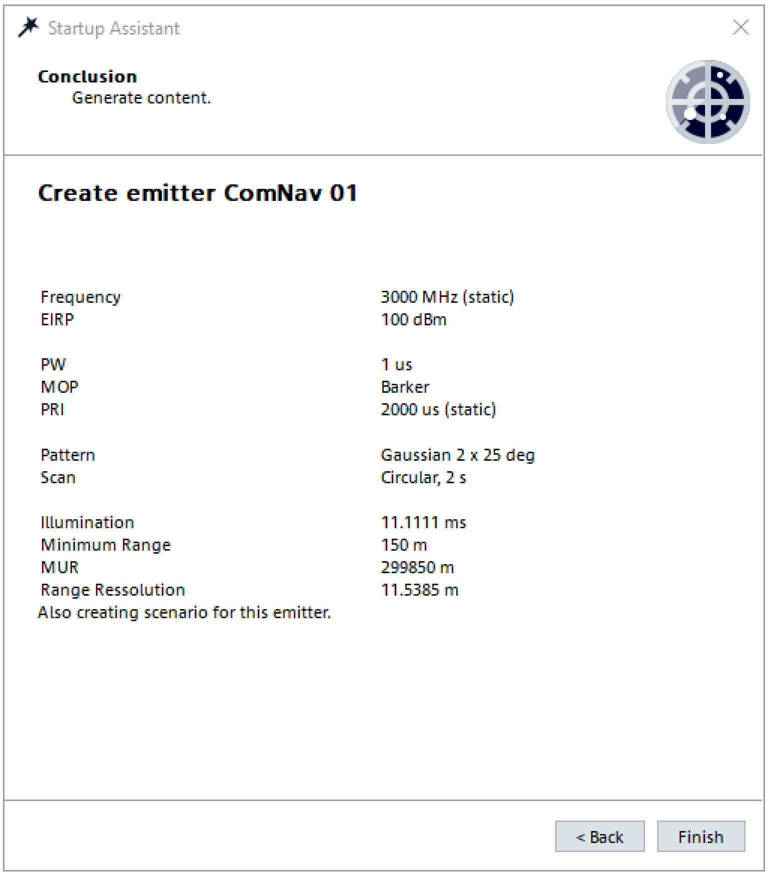
The main application field of the R&S Pulse Sequencer Digital is the generation of pulsed signals. This example uses the "Wizard" to introduce the way you can create and define antenna patterns, antenna scans, emitters, etc. and interact with the software.

To create a new single emitter scenario

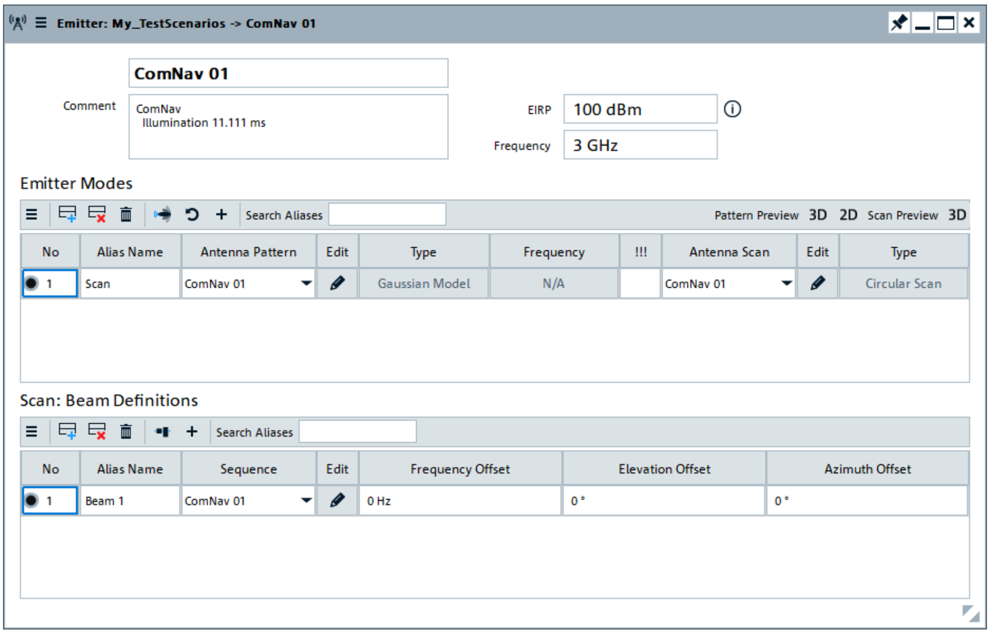
1. Start the "Wizard", see Chapter 2.8.4, "Launching the built-in wizard", on page 38.
2. In the assistant dialog, select "Create Emitter".
   Click "Next".



3. Use the default settings and select "Waveform Characteristics > MOP > Barker". Click "Next".



4. Click "Finish".



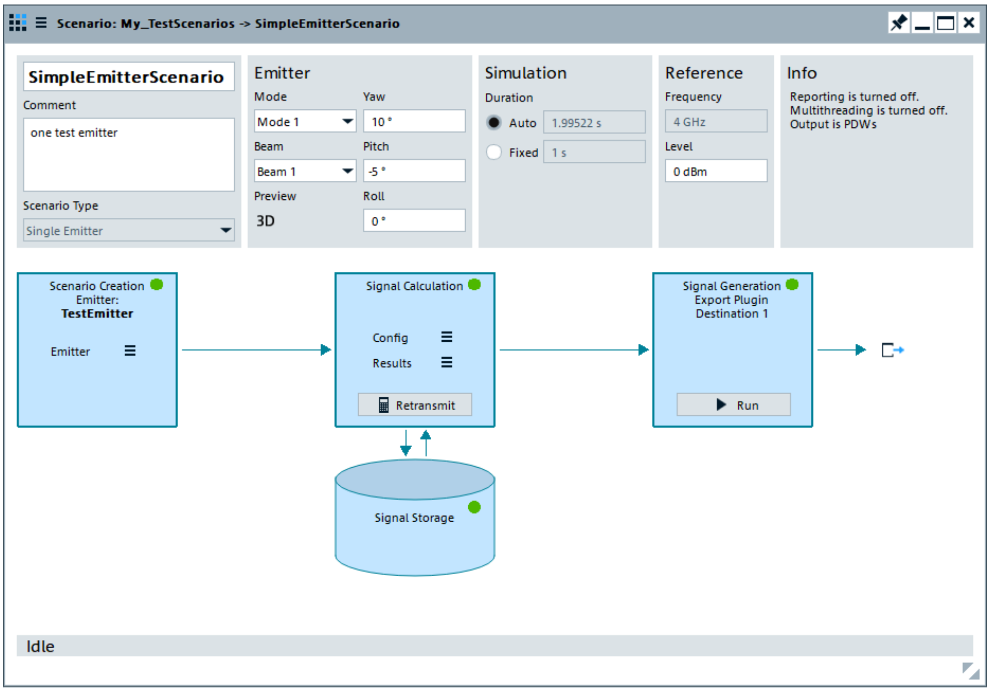
The software creates the scenario automatically with all required elements, e.g. one new emitter ("ComNav 01"), one new sequence ("ComNav 01"), new pulse ("ComNav 01"), etc.

For information on the provided settings, see:

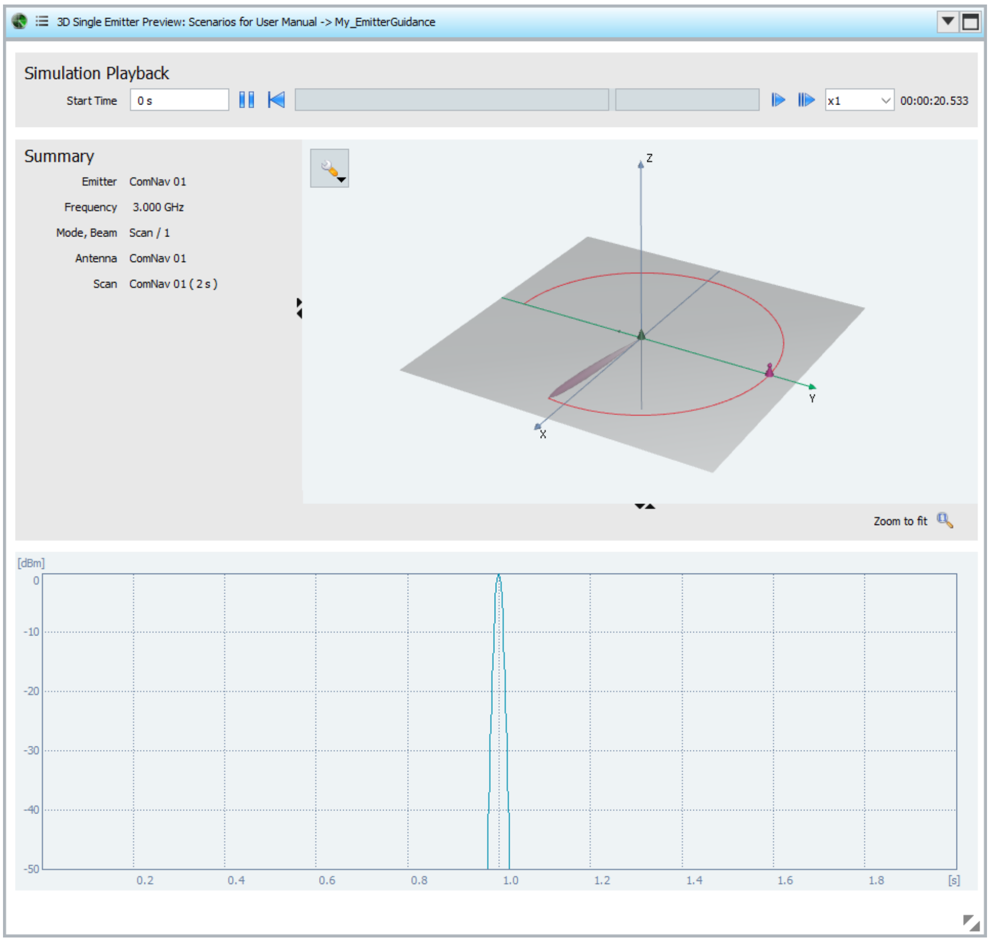
- Chapter 10.1, "Emitter settings", on page 220
- Chapter 9.2, "Antenna pattern settings", on page 174
- Chapter 9.3, "Antenna scans settings", on page 193
- Chapter 6.2.3, "Modulation on pulse (MOP) settings", on page 96

To visualize the signal received by a static receiver

1. Open the automatically created emitter-based scenario ("ComNav 01").



2. In the "Emitter" section on the block diagram, select the "3D" icon.



The "3D Single Emitter" dialog displays a live plot of the normalized signal power level at the receiver and an interactive 3D view of the emitter antenna pattern and scan.

3. Use the mouse wheel to zoom in and out.
4. To rotate the 3D view around its origin, left mouse click. Keep the key pressed and move the mouse.

   For information, see Chapter 16, "Creating complex 2D scenarios with receiver and TX items", on page 289

2.8.6 Saving and recalling settings

Repositories are automatically saved when you exist the application. We recommend that you set a meaningful repository name before closing the application.

To rename the repository

Your workspace shows one automatically created repository, "New_yyyy_mm_tttt".

1. In the repository tree, select "Repository > New_yyyy_mm_tttt".

2. Change the name. Optionally, enter a comment.
   For example, enter "General > Name > K32 and K39 Tests" or a "Comment > Repository with test files".
   To enter a new line in the comment field, press [Shift+Enter].
3. Close the dialog.

To save the repository

A yellow data base icon in the repository tree indicates that there are unsaved changes.

► In the menu bar, select "File > Save Repository > _Repository name_".

  The repository is stored in the data path set during the program installation.

To exit the software

► Close the application software.

  If the current repository contains unsaved changes, you are prompt to store or ignore them.
  Note: Unsaved changes are lost.

For more information, see:
- Chapter 2.9.1, "Customizing your workspace", on page 45
- Chapter 4.4, "How to manage the project data", on page 69

2.8.7 Advanced features and examples

With these first steps, you have gained an impression of the provided functionality.

You find a comprehensive description of the full range of capabilities in the following sections:
- Chapter 6.3, "How to create a new pulse and adjust its settings", on page 116
- Chapter 7.3, "How to create sequences and use the control elements", on page 131
- Chapter 5.3, "How to select and create a test scenario", on page 90
- Chapter 21.2, "How to create test reports", on page 392
- Chapter 9.4, "How to create a library with antenna patterns and scans", on page 206
- Chapter 10.4, "How to create and configure emitters", on page 233
- Chapter 16.1, "How to create scenarios with receiver and TX items", on page 290
- Chapter 8.2, "How to create IPM profiles and vary pulse parameters", on page 160
- Chapter 20.2, "How to configure and visualize markers", on page 382
- "How to import and assign user defined plugins" on page 377

2.9 Customizing the software

You can change default application settings to adapt the user interface to your needs.

2.9.1 Customizing your workspace

A workspace comprises the repository tree on the left side with one or more loaded repositories, and the working area with one or more screens. You can create several workspaces with different repositories assigned to them. You can save and load a user-defined workspace and switch between the different workspaces.

To save a workspace

We assume that you have opened or created a repository and opened one or more dialogs in one of the screens.

1. In the menu bar, select "File > Save Workspace/Save Workspace As" or use the save icon in the toolbar.
2. In the explorer, enter a filename, e.g. `MyWorkspace.pswk` and click "Ok".

   Per default, the `*.pswk` file is stored in the application data directory.

To load a workspace

1. In the menu bar, select "File > Open Workspace".

   Existing default and user-defined workspaces are listed.
2. Select a workspace.

   The workspace opens. All dialogs and repositories associated with the workspace are displayed.

To close a workspace

You can close a workspace if you do not need it no longer or you need to switch to a different one.

► In the menu bar, select "File > Clear Workspace".

The workspace is closed.

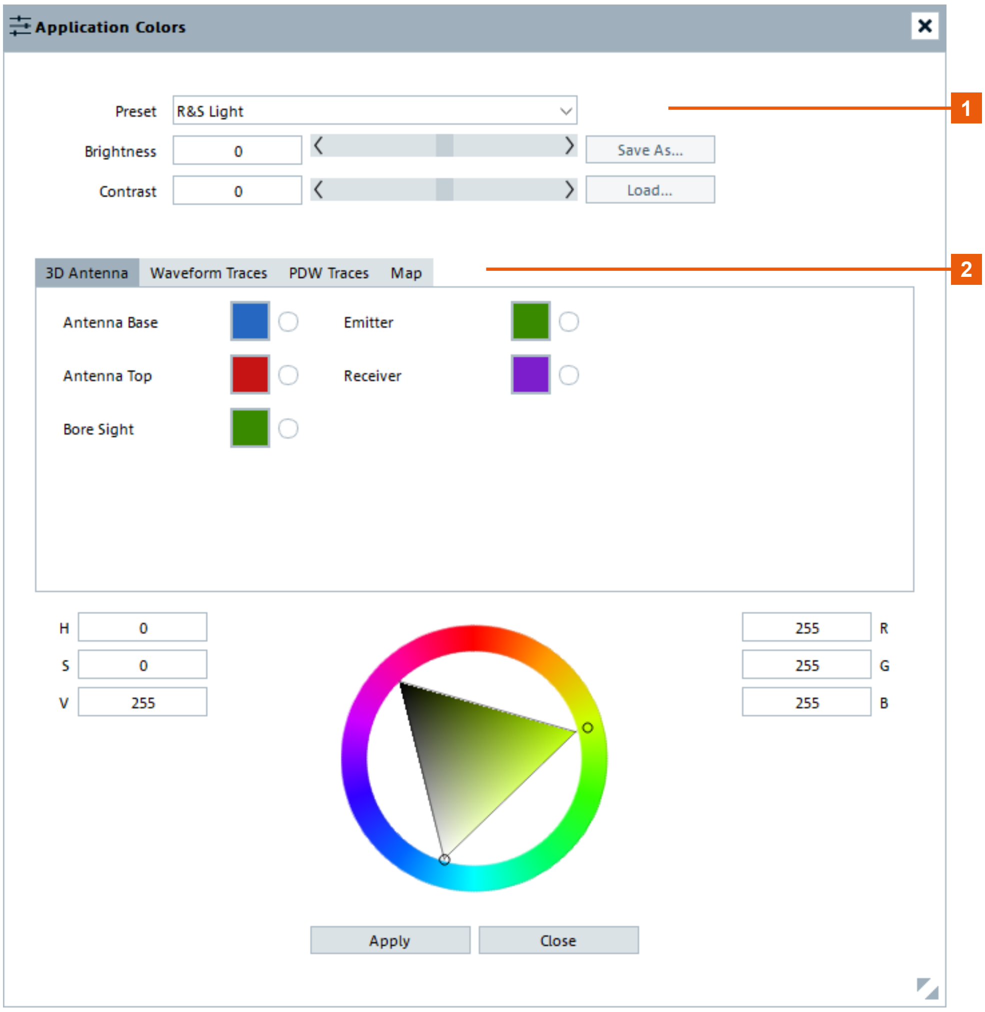
Closing the workspace unloads all repositories and closes all dialogs.
You can open a workspace again, see "To load a workspace" on page 45.

2.9.2 Changing colors and default configuration

You can change the default application settings, for example, change the display colors or change default settings of the R&S Pulse Sequencer Digital.

To customize the display colors

1. In the menu bar, select "Configure > Colors".

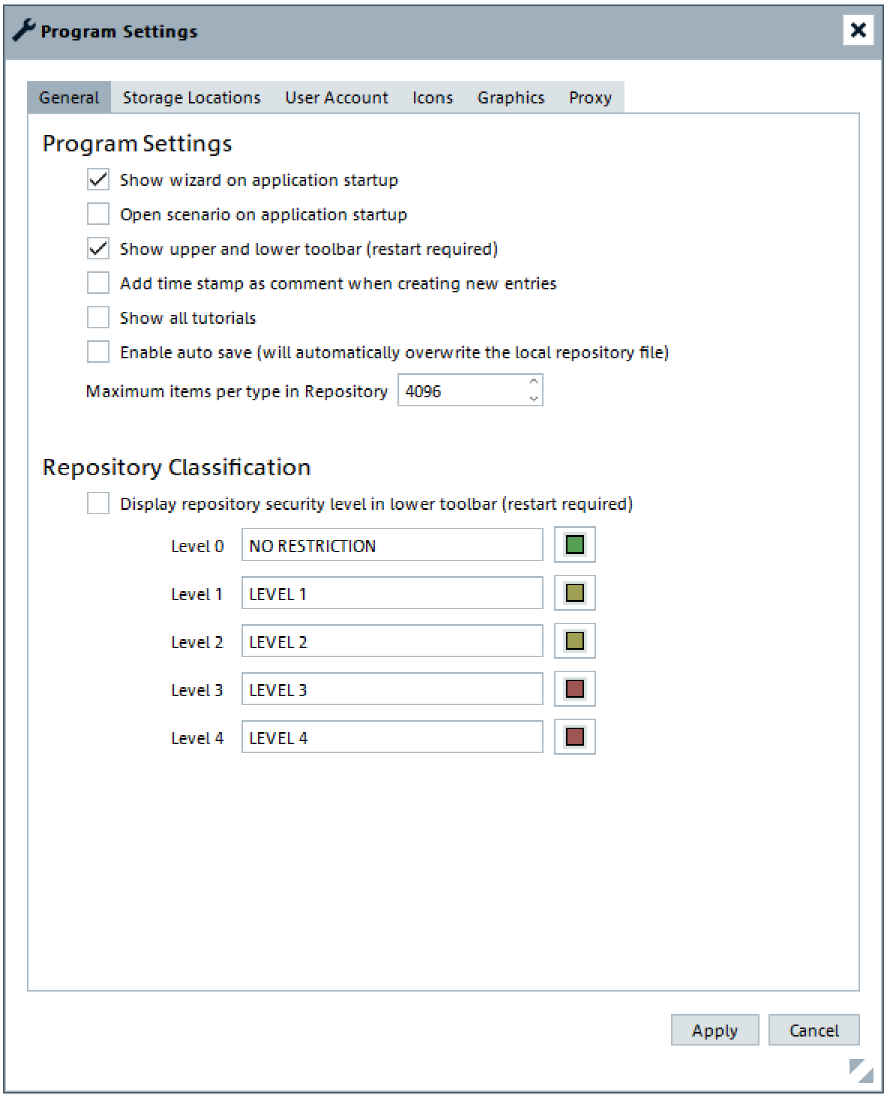


1 = Color scheme
2 = Assigns colors to the different display elements

2. Select a predefined color scheme and, if necessary, adjust the display colors individually.
3. Adjust the display colors of the individual elements, e.g select "Traces > Magnitude" and change the default color.
4. Select "Save As" to store your custom color profile in a file, e.g. `MyColors.col`.

To change the general program settings

1. In the menu bar, select "Configure > Settings".

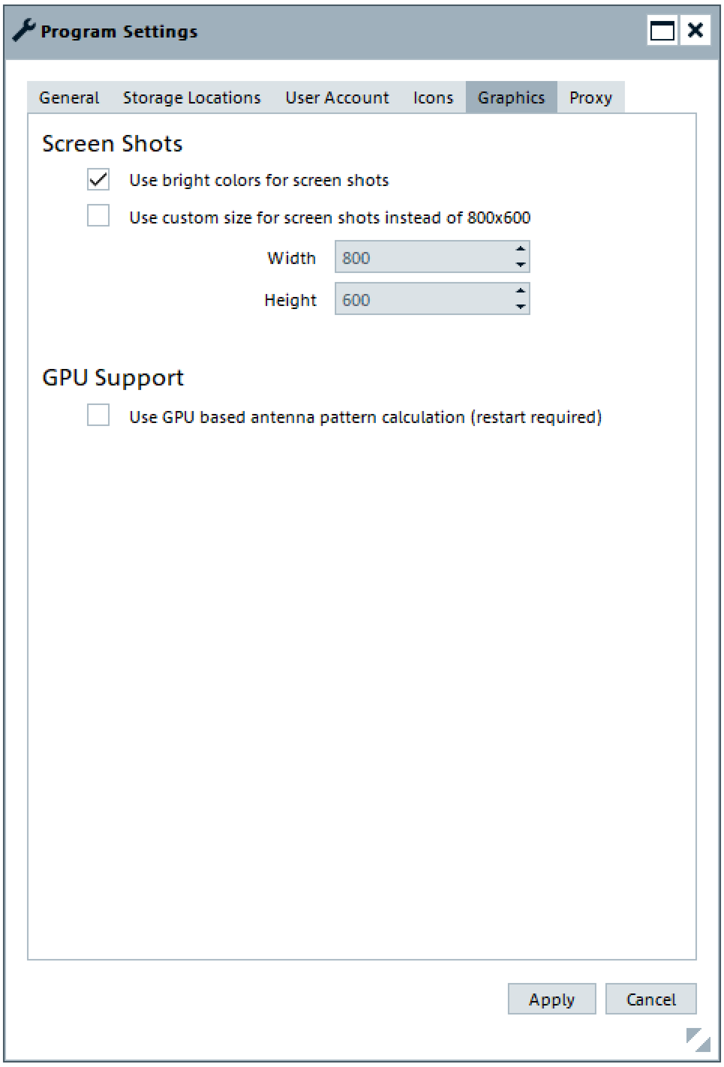


2. Change the settings, select the colors indicating the different security levels. For information on the security levels, see "General Repository Settings" on page 60.
3. Click "Apply".

To set the size and color scheme of the hardcopies

For documentation purposes, you can create a hardcopy or screenshot of the current screen or dialog. Hardcopies can be sent to the printer or stored to the clipboard.

1. To adjust the hardcopy size, in the menu bar, select "Configure > Settings > Graphics".



2. Select "Use bright colors for screenshots > On" e.g. for ink-saving printing.

   Brighter colors are used for all graphical screens instead of the ones set in the current color scheme.

3. Select "Use custom size for screenshots instead of 800x600 > On" and set the "Width" and "Height" in number of pixels.

4. Click "Apply".

The settings provided on the other tabs in the "Program Settings" dialog are explained in:

- Chapter 22, "Performing administration tasks", on page 395

2.10 Getting information and help

If any questions or problems concerning the R&S Pulse Sequencer Digital arise, an extensive online help system is provided in the software and can be consulted at any time.

The integrated help system provides the description of all functions. The help system is context-sensitive and provides information specifically for the current operation or setting to be performed. In addition, general topics provide an overview on complete tasks or function groups and background information.

Calling up Help

► To display the "Help" dialog for the currently focused screen element, e.g. a setting in an opened dialog, select the "Help" icon on the toolbar or press F1.

The "Help" dialog is displayed. A topic containing information about the focused element is displayed. If no context-specific help topic is available, a more general topic or the "Contents" tab is displayed.

The "Help" toolbar provides buttons with standard navigation functions. The "Help" window contains several tabs:

- "View" - shows the selected help topic
- "Contents" - contains a table of help contents
- "Index" - contains index entries to search for help topics
  The index is sorted alphabetically. You can browse the list, or search for entries in the list
- "Bookmarks" - contains custom bookmarks
- "Search" - provides text search

To close the Help window

► Select the "Close" icon in the upper right corner of the help window.

3 Preparing for use

Overview

The R&S Pulse Sequencer Digital runs as a standalone, PC-based application that creates simulated PDW datastreams that you can export to a customized plugin.

The software hands the simulated data to a so-called "destination", which wraps a user-written plugin that receives the data.

You can add as many destinations as you need. To enable you to export a PDW data-stream, you need to assign an export plugin to a destination.

R&S Pulse Sequencer Digital plugins are user-configurable, enabling you to define the export content and format.

See:

- Chapter 3.1, "Creating and using export plugins", on page 50
- Chapter 3.2, "PDW export plugin interface specification", on page 52
- Chapter 3.3, "Destinations settings", on page 56
- Chapter C, "Plug-in programming API", on page 650.

3.1 Creating and using export plugins

To use export plugins in R&S Pulse Sequencer Digital, you need to load a plugin of this type and map it to a destination.

The following chapters explain how to perform these steps.

For a detailed description of the programming API for the plugin, see Chapter C.2, "Export plugin", on page 655.

Creating an export plugin

R&S Pulse Sequencer Digital is delivered with the necessary resources to enable you to create plugins to export the simulated signals.

We recommend using an application such as Visual Studio to open the resource files and create the plugin. However, you can use an alternative compiler if you prefer. The plugin exports an extern "C" interface which ensures the compatibility of the DLL.

Access:

1. Perform one of the following:
   - In the menu bar, select "Help > SDK > Export_PDW > src"
   - On the main toolbar, select the icon "Open SDK folder > Export_PDW > src".

   This folder contains example resource files that you can modify as required.
2. Open the solution file, `PulseSequencer_Plugin.sln`.
3. Change the build configuration to "Release".

4. Rebuild the solution.

   The export plugin is in the "Export_PDW > bin" folder and ready for loading into R&S Pulse Sequencer Digital.

Debugging the export plugin

To debug the DLL

1. Load the plugin project in Visual Studio or your preferred compiler.
2. Set the configuration to "Debug".
3. Set the target path to the location of the R&S Pulse Sequencer Digital executable file.
4. Start the debugger.

   `PS-DIGITAL.exe` starts.
5. Ensure that the PDB debug file is in the same folder as the DLL so that the debugging symbols load correctly and you can set the breakpoints.
   Normally the file location selected automatically by the debug build is correct.
6. When the DLL is loaded into R&S Pulse Sequencer Digital, you can start debugging the plugin code in Visual Studio.

Loading an export plugin

To load an export plugin in to R&S Pulse Sequencer Digital, follow the same steps as for other plugin types.

For details, see "To import a plugin" on page 377.

Navigate to the "SDK > Export_PDW > bin" folder to select the export plugin.

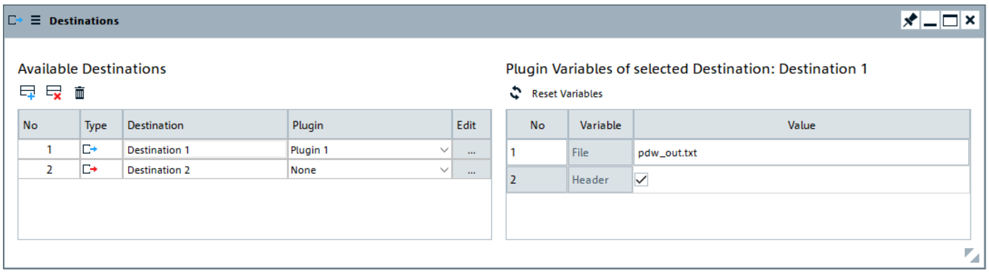
Mapping plugins to destinations

When an export plugin is loaded as described in Loading an export plugin, you can map it to a destination in R&S Pulse Sequencer Digital.

Access:

1. Perform one of the following:
   - In the menu bar, select "Configure" > "Destinations"
   - In the block diagram of a scenario, select "Signal Generation" > "Config" > "Destinations".

   The "Destinations" dialog opens.



2. To add a new destination, in "Available Destinations" select "Add a new destination".

   A new entry displays at the end of the list, assigned with the next free number.
3. To edit the destination name, double-click in the "Destination" field.
4. Select an available plugin from the "Plugin" drop-down list.
5. To edit the name or comment parameters, or to change the selected DLL, select "Edit".

   The "Plugin" dialog opens.
6. Edit the plugin variables on the right as required.

   The plugin is mapped to the specified destination and ready for use in a scenario.

For a description of how to assign signals to destinations, see Chapter 17.2, "Assign signals to destination", on page 351.

3.2 PDW export plugin interface specification

The export plugin that R&S Pulse Sequencer Digital uses for the simulated signals is user-configurable. You can adapt the plugin for streaming or transferring the data as you require, including the content of the data.

The format of the PDW data at the export plugin interface function is a Rohde & Schwarz proprietary format as described in this chapter. In the plugin, the user is free to extract any needed information and adapt the data formats or representation of the values to map it to any needed format.

Depending on the scenario you are using and your export needs, some of the following parameters might not be necessary.

Parameters that are not set represent an invalid value. Parameters that do not have an invalid representation are guaranteed to be valid.

The serialized data uses little endian byte order.

Data type

The invalid value depends on the data type:

- **uint8/16/32**

The invalid case is defined as all 1's in bit representation, which means 0xFF, 0xFFFF, ... in hex.
For unsigned integers, it is the definition of the maximum value that is possible, therefore the valid data range is limited by 1.
For example, for uint8 the data range is [0, 254] in decimal representation, as 255 is used to indicate the invalid case.

- **int16**
  The invalid case is defined as 0x8000, which represents the minimum value -32768. Therefore the valid data range for int16 is [-32767, 32767].
- **float32**
  The invalid case is defined as 0x7F7FFFFF, which represents the maximum available float number regarding IEEE-754.

Table 3-1: PDW export plugin interface parameters

Data type	Parameter	Byte size	Range	Invalid	Unit	Description
uint8	**version**	1	[0x00, 0xFF]	-	-	PDW format version. Initial version = 0x00.
uint16	**tx_platform_id**	2	[1, 65534]	0xFFFF	-	User configured platform ID
uint16	**tx_id**	2	[1, 65534]	0xFFFF	-	User configured emitter ID (per emitter mode)
uint8	**tx_platform_emit_id**	1	[1, 254]	0xFF	-	Emitter index inside platform (if platform emitter)
uint8	**tx_list_id**	1	[1, 254]	0xFF	-	List index in multi-emitter scenario: • Emitter index for single emitter in list • Platform (parent) index if emitter is platform emitter
uint8	**rx_id**	1	[1, 254]	0xFF	-	Database ID of the receiver (list index in repository tree view)
uint8	**rx_antenna_id**	1	[1, 254]	0xFF	-	Antenna ID of receiver
uint64	**toa**	8	[0, 18.446.744.073, ...] (s)	-	ps	TOA
uint64	**freq**	8	[0, $1.884*10^{19}$] (Hz)	-	Hz	Absolute frequency
uint64	**pulse_width**	8	[0, 18.446.744, ...] (s)	-	ps	Pulse width
int32	**level**	4	[-200, 200] (dBm)	-	0.01 dBm	Absolute level
uint8	**modulation**	1	[0x00, 0xFF]	-	-	Modulation flag See Modulation flags

Data type	Parameter	Byte size	Range	Invalid	Unit	Description
uint32	**bandwidth**	4	[0, 42.94] (GHz)	0xFFFF	10 Hz	Bandwidth
uint64	**rise_time**	8	[0, 18.446.744, ...] (s)	-	ps	Pulse rise time
uint64	**fall_time**	8	0, 18.446.744, ...] (s)	-	ps	Pulse fall time
uint8	**marker_mask**	1	[0x00, 0x0F]	-	-	Marker bits (4321)
uint16	**azimuth**	2	[0, 360] (deg)	0xFFFF	0.1 deg	Tx azimuth
int16	**elevation**	2	[-90, 90] (deg)	20x8000	0.1 deg	Tx elevation
float32	**distance**	4	[-1e9, +1e9] (m)	0x7F7FFFFF	m	Tx distance to Rx origin
uint16	**rx_antenna_azimuth**	2	[0, 360] (deg)	0xFFFF	0.1 deg	Rx antenna azimuth
int16	**rx_antenna_elevation**	2	[-90, 90] (deg)	0x8000	0.1 deg	Rx antenna elevation
float32	**longitude**	4	[-180, 180] (deg)	0x7F7FFFFF	deg	Tx longitude coordinate in deg (7-8 positions after decimal point)
float32	**latitude**	4	[-90, 90] (deg)	0x7F7FFFFF	deg	Tx latitude coordinate in deg (7-8 positions after decimal point)
float32	**altitude**	4	[-1e9, +1e9] (m)	0x7F7FFFFF	m	Tx altitude coordinate
float32	**east**	4	[-1e9, +1e9] (m)	0x7F7FFFFF	m	Tx east coordinate (ENU). Reference coordinate is Rx position.
float32	**north**	4	[-1e9, +1e9] (m)	0x7F7FFFFF	m	Tx north coordinate (ENU). Reference coordinate is Rx position.
float32	**up**	4	[-1e9, +1e9] (m)	0x7F7FFFFF	m	Tx up coordinate (ENU).
uint16	**yaw**	2	[0, 360] (deg)	0xFFFF	0.1 deg	Tx yaw
int16	**pitch**	2	[-90, 90] (deg)	0x8000	0.1 deg	Tx pitch
uint16	**roll**	2	[0, 360] (deg)	0xFFFF	0.1 deg	Tx roll
	reserved_field	2				Reserved field for future additions
uint32	**speed**	4	[0, 5999] (m/s)	0xFFFFFFFF	0.01 m/s	Absolute speed of Tx movement
	not yet implemented	36				Parameters not yet implemented

Data type	Parameter	Byte size	Range	Invalid	Unit	Description
	reserved_field	113				Reserved field for future additions
Total byte size		256				

Modulation flags

Table 3-2 shows the 8-bit encoding of the pulse modulation type.

Table 3-2: Modulation flags

modulation	hex code	bin code
unmodulated	00	00000000
fm	01	00000001
fm-step	02	00000010
2-fsk	03	00000011
4-fsk	04	00000100
8-fsk	05	00000101
16-fsk	06	00000110
32-fsk	07	00000111
64-fsk	08	00001000
msk	09	00001001
chirp-generic	10	00010000
chirp-linear-up	11	00010001
chirp-linear-down	12	00010010
chirp-linear-triangular	13	00010011
chirp-linear-piecewise	14	00010100
chirp-sine	15	00010101
barker-2	20	00100000
barker-3	21	00100001
barker-4a	22	00100010
barker-4b	23	00100011
barker-5	24	00100100
barker-7	25	00100101
barker-11	26	00100110
barker-13	27	00100111
polyphase-frank	30	00110000
polyphase-p1	31	00110001

modulation	hex code	bin code
polyphase-p2	32	00110010
polyphase-p3	33	00110011
polyphase-p4	34	00110100
bpsk	35	00110101
qpsk	36	00110110
8psk	37	00110111
phase-generic	3F	00111111
am-standard	40	01000000
am-lsb	41	01000001
am-usb	42	01000010
am-lsb-usb	43	01000011
am-step	44	01000100
ask	45	01000101
16-qam	50	01010000
32-qam	51	01010001
64-qam	52	01010010
128-qam	53	01010011
256-qam	54	01010100
noise 60	60	01100000
unknown	FE	11111110
EOF	FF	11111111

See also:

- Chapter C.2, "Export plugin", on page 655

3.3 Destinations settings

Access:

► Perform one of the following:
  - In the menu bar, select "Configure" > "Destinations"
  - In the block diagram of a scenario, select "Signal Generation" > "Config" > "Destinations".

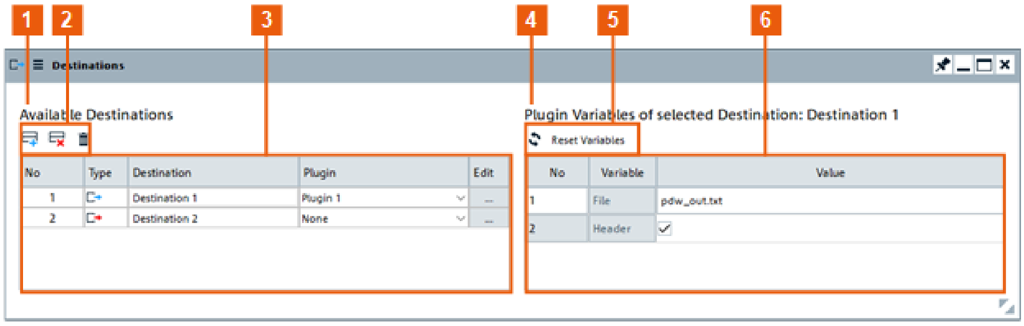


Figure 3-1: Destinations: understanding the displayed information

1 = Displays all available destinations
2 = Adds/removes destinations
3 = Information and parameters for available destinations
4 = Plugin variables for currently selected destination
5 = Resets all variables for selected plugin
6 = Information and parameters of variables for selected plugin

Settings

Available destinations
Displays all available destinations in a repository. Destinations displayed here are available in all scenarios.

Add, delete, delete all ← Available destinations
Standard functions for items handling.

You cannot delete a destination that is assigned in a scenario. First remove the assignment, then delete the destination from the list.

Remote command:
`DESTination:ADD` on page 472
`DESTination:DELete` on page 472
`DESTination:CLEar` on page 472

No ← Available destinations
Displays the destination number assigned automatically when added.

If you delete a destination from the list, the numbers below it are automatically reassigned.

Remote command:
`DESTination:SELect` on page 427

Type ← Available destinations
Plugin type icon assigned automatically for export plugins.

Destination ← Available destinations
User-configurable name of the destination. To edit the name, select the field.

Remote command:
`DESTination:NAME` on page 427

Plugin ← Available destinations
Selects a plugin from the drop-down list.

The list contains all export plugins available in "Project tree > Plugins".

Remote command:
`DESTination:PLUGin:NAME` on page 472

Edit ← Available destinations
Opens the "Plugin" dialog where you can edit the name and comment fields or load a new DLL file.

For detailed information on importing plugins, see "How to import and assign user defined plugins" on page 377.

Plugin variables
Displays the variables for the current destination.

Remote command:
`DESTination:PLUGin:VARiable:CATalog` on page 503
`DESTination:PLUGin:VARiable:SELect` on page 503

Reset variables ← Plugin variables
Resets the variable parameters to the default, removing customizations. The reset applies to all parameter values, including values not currently selected in the list.

Remote command:
`DESTination:PLUGin:VARiable:RESet` on page 504

No ← Plugin variables
Displays the automatically assigned plugin variable number.

Remote command:
`DESTination:PLUGin:VARiable:SELect:ID` on page 504

Variable ← Plugin variables
Displays the variable type.

This parameter is for information only.

Value ← Plugin variables

Displays the value of the variable. To edit the values, double-click in the field.

Remote command:

`DESTination:PLUGin:VARiable:VALue` on page 504

4 Organizing the project data in repositories

This section explains the basic repository concept and informs you briefly how the software database is built.

An introduction to the concept of elements handling is provided in "Creating elements" on page 32.

4.1 Repository settings

Access:

► In the project tree, double-click the repository name.

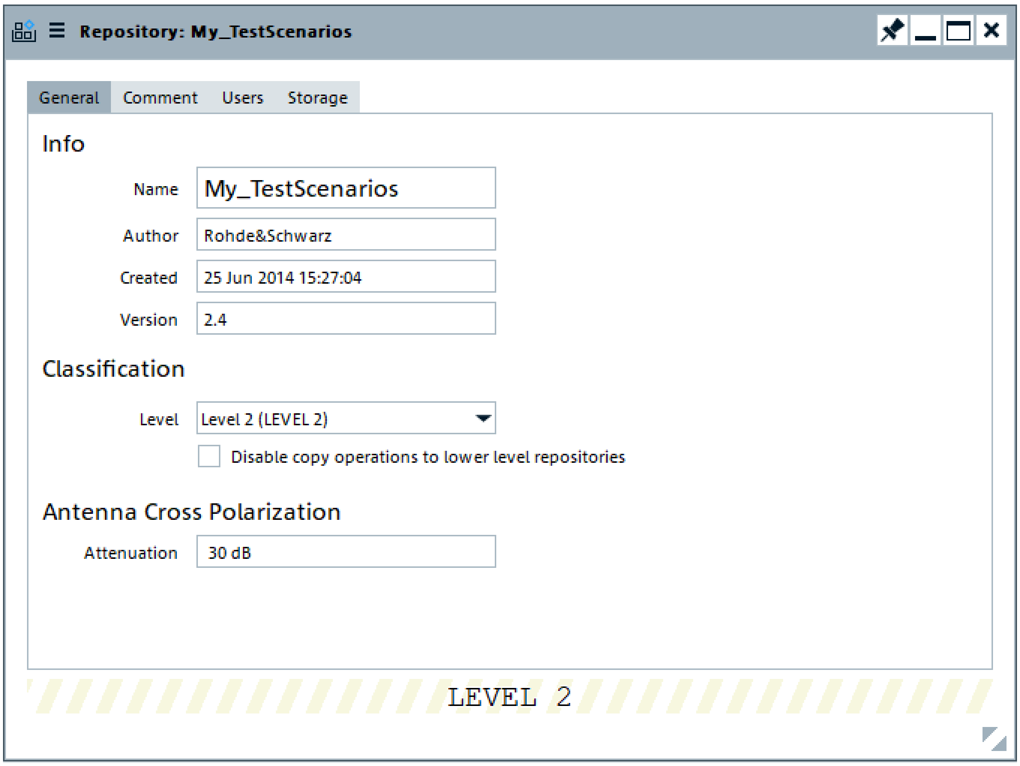
The repository tree shows the content of the currently loaded repositories.

Settings:

General Repository Settings

Provides general information related to repository management.

The "General" tab.



Info ← General Repository Settings

Provides information like the name of the author and the date of creation.

"Name"
When a new repository is created, it is named automatically but you can rename it afterwards.
The automatically assigned names follow the syntax:
`New_<yyyy-mm-dd><T><hhmmss>`, where `<yyyy-mm-dd>` and `T<hhmmss>` indicate the current date and time respectively.

"Author"
Indicates the author.

"Created"
Automatically assigned date and time indication.

"Version"
Repository version.

Remote command:
`REPository:CATalog?` on page 426
`REPository:CREate` on page 426
`REPository:SELect` on page 427
`REPository:REMove` on page 428
`REPository:AUTHor` on page 528
`REPository:DATE` on page 528
`REPository:VERSion` on page 529

Classification ← General Repository Settings

Classification information is assigned to each repository to set the security level.

"Level" "Level 0" is the lowest level that means no restrictions and "Level 4" is the highest one. Elements belonging to a repository with higher level cannot be copied to a repository with lower one.
If more than one repository is loaded, the workspace displays the highest required classification level, see Chapter 2.5, "Understanding the displayed information", on page 23.
See also "To change the general program settings" on page 46.

"Disable copy operations to lower level repositories"
If enabled, you cannot copy repository elements for repositories with higher classification level to repositories with lower one.

Remote command:
`REPository:SECurity` on page 529

Antenna Cross Polarization > Attenuation ← General Repository Settings
Sets the attenuation level used to calculate the cross-polarized antenna patterns, see "Polarization" on page 178.

Remote command:
`REPository:XPOL:ATTenuation` on page 529

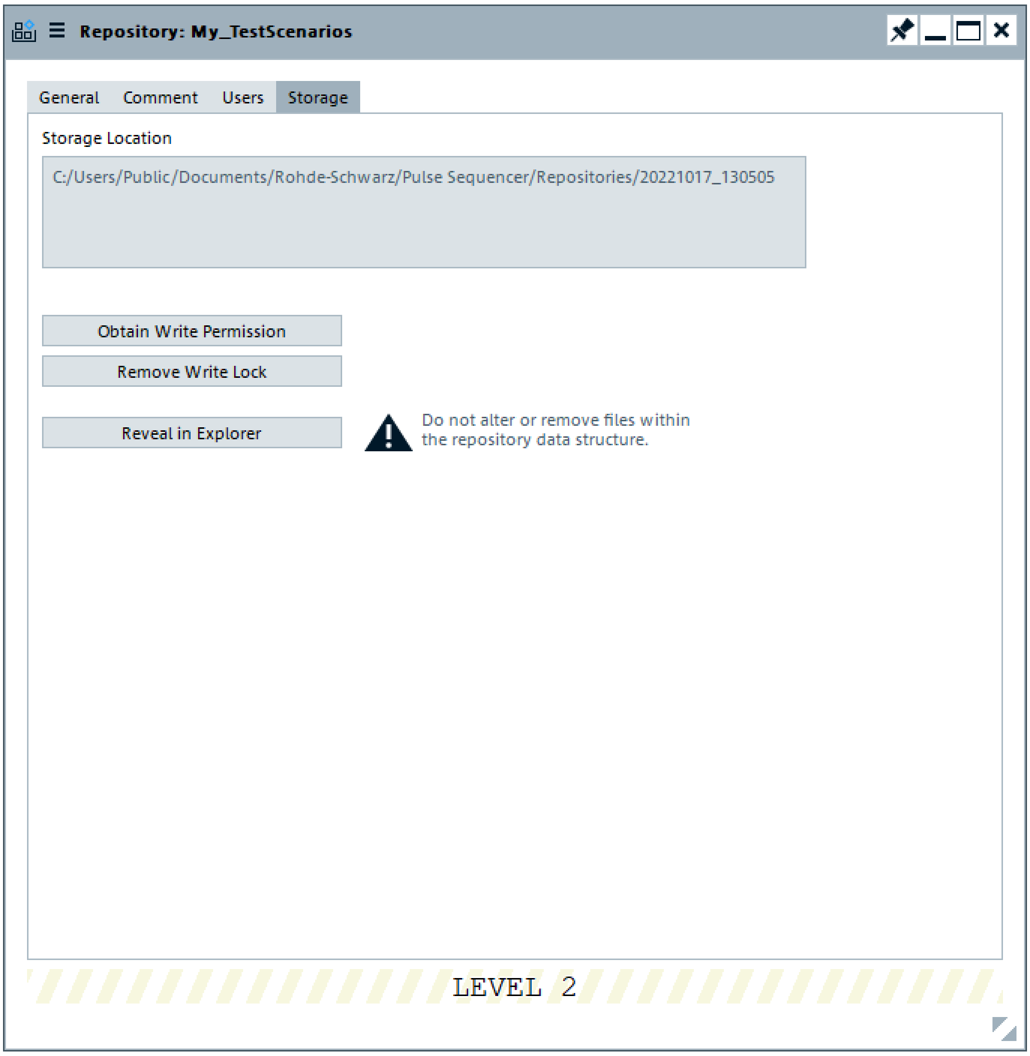
Comment
Enter a short description of the repository.

Remote command:
`REPository:COMMent` on page 428

Users
Provides settings related to user management.

For detailed information, see Chapter 22.2, "User administration", on page 396.

Storage
This tab provides settings necessary to configure the default storage folder for the repositories.



"Path"

Displays the location (network or local folder) the repository is stored in.

Remote command:
`REPository:PATH?` on page 528

"Obtain Write Permission"

Repositories allow the simultaneous access from several users. If a user with write permission has opened a repository, this write permission is exclusive. Another user will be granted with the write permission only after the first user had closed the repository or removed it from its workspace.
You can request an exclusive write permission only on a repository that is not opened by another user with write permission.

See also:
- "Discovered Repositories on the Mass Storage" on page 65
- "To obtain write permission on a repository" on page 72

"Remove Write Lock"

The R&S Pulse Sequencer Digital locks any improperly or unexpectedly closed repository and indicates this situation with a red lock symbol in the *"Repository Tree"*. A locked repository can be unlocked only from the same PC and by the same user with write permission that had opened the repository before.

See "To remove write lock that results from a previously crashed session" on page 622.

"Reveal in Explorer"

You can open the repository file structure in the Windows Explorer.

Note: Do not change the repository content by changing the file structure in the Windows Explorer.
To move or share repositories with other users, always use the "Export/Import Repository Archive" function.

See:
- "To create and export a repository archive" on page 71
- "To import a repository archive via the menu bar" on page 72

Remote command:

`REPository:FILename?` on page 528
`REPository:SAVE` on page 529

4.2 Repository manager settings

Access:

► In the menu bar, select "File > Load and Manage Repository".

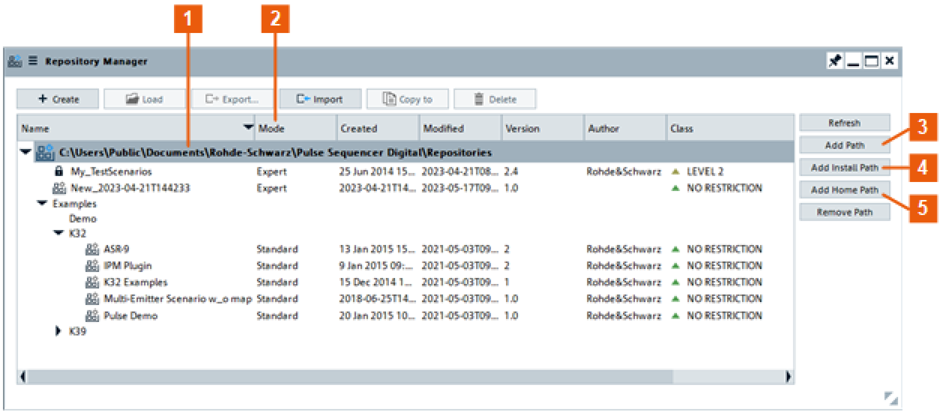


Figure 4-1: Repository Manager: understanding the displayed information

1 = Default repository path (DataPath), see also Table 2-2
2 = Mode/license type of repository, see "Create" on page 65
3 = Add custom path for repository discovery for saving and possibly sharing repositories, alternative to the DataPath and the HomePath
4 = Add the home path set during installation
5 = Add user's home path

The "Repository Manager" comprises the setting for managing the repository files and archives on the mass storage.

Settings:

Discovered Repositories on the Mass Storage

Lists all found directories and the repository files that they contain.

You can change the column width and order.

A lock symbol on the database icon in the "Repository Manager" indicates one of the following situations:
- Repository is opened by another user with explicit write access.
- Insufficient user rights for the particular repository.
- Insufficient user rights on the file system (directory) the repository is stored in.

Try to:
- Wait until the repository has been closed and try to obtain exclusive write permission, see "To obtain write permission on a repository" on page 72.
- Change your current user or use a user with different role, see "To change the default user used to access repositories" on page 399.
- Log in as user with sufficient (write) access rights on the file system.

Remote command:
`REPManager:CATalog?` on page 530

Create

Adds a repository in the selected directory.

The "Mode" ("Demo" / "Standard" / "Expert") indicates in which operation mode you created the repository (see Chapter 2.1, "Required options and licences", on page 15).

The following rules apply to repositories:
- Repositories created in the R&S PULSE-K32 can be opened with R&S PULSE-K39
  The repository is irreversibly updated to R&S PULSE-K39 complexity.
- Repositories created in R&S PULSE-K39 cannot be opened with R&S PULSE-K32.
- Repositories created in the demo version cannot be opened with R&S PULSE-K32 or R&S PULSE-K39.
- Repositories created in the original R&S Pulse Sequencer (RF) application can be imported into R&S Pulse Sequencer Digital.

See also Chapter 22.3, "Defining storage locations", on page 401.

Load
Loads and displays the selected repository in the workspace.

See "To load a repository" on page 70.

Remote command:
`REPManager:LOAD` on page 530

Export
Exports the selected repository to an archive file.

See "To create and export a repository archive" on page 71.

Remote command:
`REPManager:EXPort` on page 531

Import
Imports a repository from a directory.

Remote command:
`REPManager:IMPort` on page 531

Copy to
Copies the selected repository to a new location.

The R&S Pulse Sequencer Digital automatically copies all relevant files and database elements.

Delete
Deletes the entire repository from the permanent mass storage, if:
- The current user has write permission, see Table 22-1.
- The repository is not opened by another user with write permission.

Remote command:
`REPManager:DELete` on page 531

Refresh
Refreshes the list of found repository files.

Add Path
Opens the "Select Folder" dialog that is similar to the Windows Explorer.

Navigate to the required directory and load it. Found repository files are displayed in the list.

See also "To set the storage location for repositories" on page 402.

Remote command:
`REPManager:PATH:ADD` on page 532
`REPManager:CATalog?` on page 530

Add Install Path, Add Home Path
Adds and loads the directory automatically. Found repository files are displayed in the "Discovered Repositories on the Mass Storage" on page 65.

Where:
- "Add Install Path" opens the `DataPath` directory.
  Per default, this directory is the same displayed in "Repository Manager" (see Figure 4-1).
- "Add Home Path" opens the `HomePath` directory.

For information on the default file paths, see Table 2-2.

See also Chapter 22.3, "Defining storage locations", on page 401.

Remote command:
`REPManager:PATH:ADD` on page 532
`REPManager:CATalog?` on page 530

Remove Path
Removes the selected path. The repository files within this directory are also removed from the list.

Remote command:
`REPManager:PATH:DELete` on page 532

4.3 Repository view

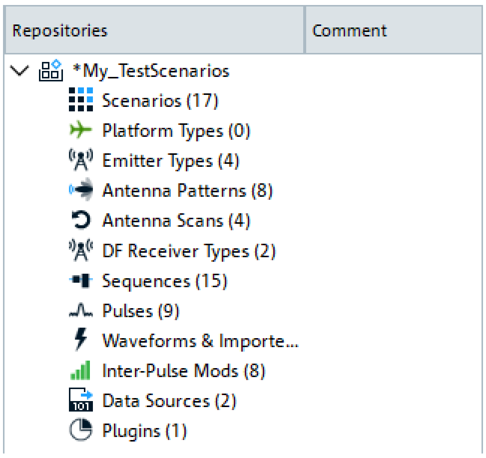
By default, the repository view is set to the tree view. In the tree view, every item of the repository element, e.g. "Scenarios", is displayed.

See "Repository tree" on page 26.

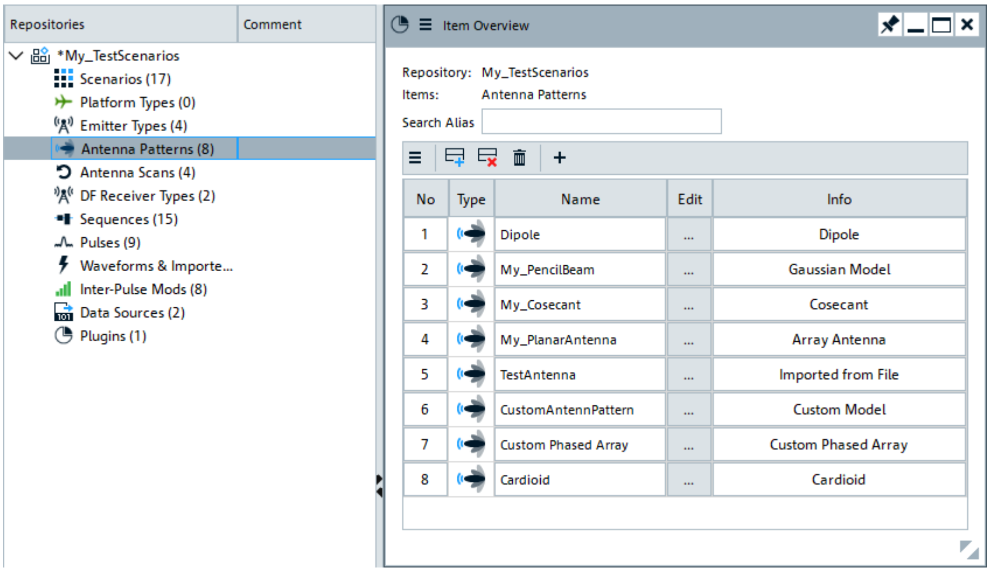
You can switch the tree view to a list view, which displays the number of items in the repository element in brackets. If you select a repository element in the list view, the "Item Overview" dialog opens and displays all items of the selected repository element.

Access:

1. In the project tree toolbar, click .



2. Click the repository element to open the item overview.



Settings:

Repository

Displays the name of the repository.

Items

Displays the name of the selected repository element related to the items.

Search Alias

Filters the list according to the string entered in the dialog.

Edit, Append, Remove, Remove all, Copy and append

Standard functions for items handling, see "Standard function in the context menus" on page 33.

Item overview table

The table shows the following information:

No ← Item overview table

Displays the number of the item.

Type ← Item overview table
Displays the type of the item, e.g. "Antenna Patterns".

Name ← Item overview table
Displays the name of the item.

Edit ← Item overview table
Opens the dialog to edit the selected item.

Info ← Item overview table
Displays important parameters of the item.

4.4 How to manage the project data

See:

- "To create a repository" on page 69
- "To rename the repository" on page 43
- "To save the repository" on page 44
- "To delete a repository" on page 70
- "To remove a repository from the workspace" on page 70
- "To load a repository" on page 70
- "To create and export a repository archive" on page 71
- "To import a repository archive per drag&drop" on page 71
- "To share a repository with other users" on page 72
- "To protect a repository from accidental changes" on page 398
- "To remove the password protection on a repository" on page 399
- "To change the default user used to access repositories" on page 399

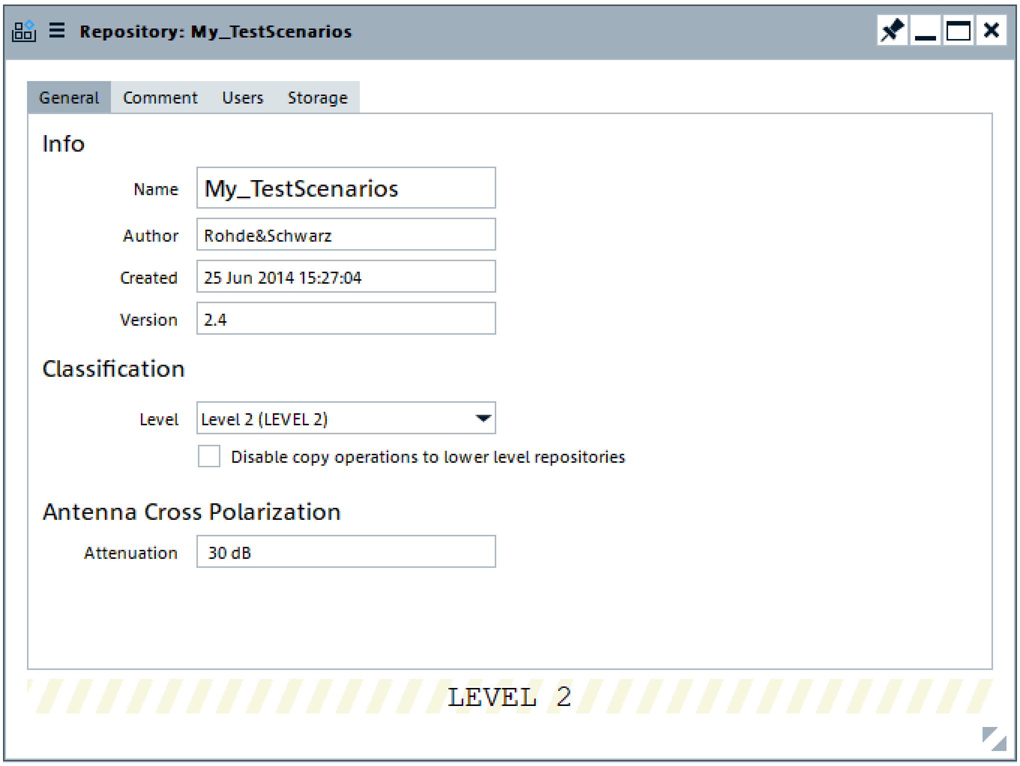
To create a repository

1. In the menu bar, select "File > New Repository".

   A new empty repository is created and added to the project tree.
   Repositories are named automatically. The repository names follow the syntax: `New_<yyyy-mm-dd><T><hhmmss>`, where `<yyyy-mm-dd>` and `T<hhmmss>` indicate the current date and time respectively.

2. In the project tree, double-click the repository name.

   The "Repository" dialog opens.



3. Change the settings. For example, change the automatically assigned "Name" and "Comment".
4. Set the "Classification > Level >" (see "Classification" on page 61).
5. Close the dialog.

To delete a repository

You can delete repositories that you no longer need, if this repository is not opened by another user with write permission.

1. Use a user with write access rights, e.g. "Admin" or "Creator" (see Table 22-1).
2. In the menu bar, select "File > Load and Manage Repository".
3. From the listed files, select a repository file, e.g. "My_TestScenarios".
4. Select "Delete".

To remove a repository from the workspace

► In the project tree, select the repository e.g. "My_TestScenarios". In the context menu, select "Unload from Workspace".

The repository is removed from the workspace but not deleted. You can open it again, see "To load a repository" on page 70.

To load a repository

1. In the menu bar, select "File > Load and Manage Repository".

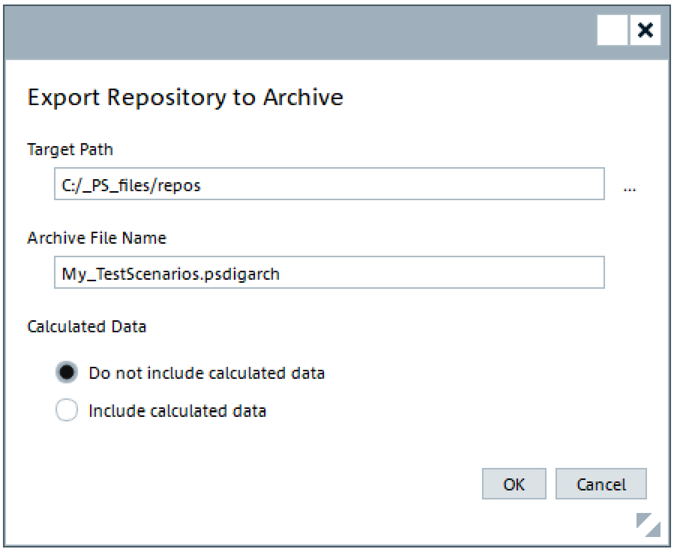
2. If necessary, for example by loading and exchanging repositories with another user, change the directory in that the software searches for repository files.
3. From the listed files, select a repository file, e.g. "K32 Examples".
4. Click "Load".

   The repository is opened and displayed in the project tree.

To create and export a repository archive

We assume that you have created a repository and configured at least one scenario.

1. Perform one of the following:
   a) In the repository tree, select the repository.
      In the context menu, select "Export".
   b) In the menu bar, select "File > Export Repository".
      From the listed files, select a not used repository file.



2. In the "Export" dialog, select "Target Path" to change the directory in which the repository archive file is stored.
3. Per default, the software saves repository archives as `Export.psdigarch` file. Select "Archive File Name" to change the filename.
4. Define whether the archive file includes the calculated data or not. Including calculated date increases the size of the archive.
5. Click "Ok".

   The repository archive is stored in a file with the selected filename. The file extension `*.psdigarch` is assigned automatically.

To import a repository archive per drag&drop

1. In your file explorer, navigate to the network directory with the repository archive.

2. Select a repository file, drag&drop it in the R&S Pulse Sequencer Digital.

   The repository is opened and displayed in the project tree.

To import a repository archive via the menu bar

1. In the menu bar, select "File > Import Repository".
2. Navigate to the network directory. Select a `*.psdigrch` file.

   The "Load and Manage Repository" dialog opens.
3. Select a repository file and click "Load".

   The repository is opened and displayed in the project tree.

To share a repository with other users

To move or share repositories, use the "Export/Import Repository Archive" function. Do not access the repositories from your Windows Explorer.

1. See "To create and export a repository archive" on page 71.
2. See "To import a repository archive via the menu bar" on page 72.

To obtain write permission on a repository

Repositories can be opened with executive write permission by one user at a time.

If you have sufficient user rights, you can request an exclusive write permission on a repository that is not opened by another user with write permission.

1. In the project tree, double-click the repository name.
2. In the "Repository" dialog, select "Storage".
3. Select "Obtain Write Permission".

5 Selecting a suitable scenario and creating scenarios

This section explains the provided scenario types and helps you select the scenario most fitting to your current task.

5.1 Overview of the available scenarios and their complexity

The R&S Pulse Sequencer Digital supports six scenarios with different complexity. Select the scenario most fitting your test case:

- **Single sequence**
  The single sequence scenario is used to create sequences from individual pulses. Pulse parameters include the pulse envelope, modulation on pulse (MOP), carrier frequency, and level. The sequence editor defines the order in which the pulses are generated and also adds control elements such as loops and fillers. Inter-pulse modulation profiles can be applied to vary pulse parameters.
- **Sequences (collection)**
  The sequences (collection) scenario is used to create multiple sequences which can be selected in arbitrary order.
  Pulse parameters include the pulse envelope, modulation on pulse (MOP), carrier frequency, and level. The sequence editor defines the order in which the pulses are generated and also adds control elements such as loops and fillers. Inter-pulse modulation profiles can be applied to vary pulse parameters.
- **Single emitter**
  The emitter scenario combines a pulse train with an antenna pattern and a particular antenna scan type. An emitter comprises several modes. Each mode is defined by an antenna pattern and an antenna scan, and multiple beams that contain the pulse train.
- **Emitters (collection)**
  The emitter collection scenario provides a choice of multiple emitters that can be selected manually. An emitter comprises several modes. Each mode is defined by an antenna pattern and an antenna scan, and multiple beams that contain the pulse train.
  Multiple emitters can be interleaved into multiple output files using groups and a priority-based dropping algorithm.
  If pulses are overlapping, the pulses with the lower priority are dropped.
- **Localized emitters**
  The localized emitters scenario combines multiple emitters and one receiver on a 2D map. An emitter comprises several modes. Each mode is defined by an antenna pattern and an antenna scan, and multiple beams that contain the pulse train. The receiver is defined by a single antenna pattern and scan.
  Multiple emitters can be interleaved into multiple output files using groups and a priority-based dropping algorithm.

If pulses are overlapping, the pulses with the lower priority are dropped.

- **Direction finding**
  Option:R&S PULSE-K39
  The direction finding scenario simulates multiple emitters and a single receiver with up to 10 antennas on a 2D map.
  The individual signals for each receive antenna are simulated to enable the receiver to locate the position of a signal source.
  Multiple emitters can be interleaved into multiple output files using groups and a priority-based dropping algorithm.
  If pulses are overlapping, the pulses with the lower priority are dropped.
- **PDW list (collection)**
  The PDW list collection finding scenario provides a choice of multiple PDW lists that can be selected manually. This scenario generates a Rohde & Schwarz proprietary PDW-based signal from a list of imported PDWs in custom format.
  Imported PDW lists are read from a waveform object.
  Multiple emitters can be interleaved into multiple output files using groups and a priority-based dropping algorithm.
  Option:R&S PULSE-K39 required for interleaving.
  If PDWs are overlapping, the PDWs with the lower priority are dropped.
  See Chapter 18, "Working with PDWs", on page 355.

5.2 Scenario settings

Access:

► Select "Repository Tree > Scenario".

The "Scenario" dialog provides the settings, necessary to configure the signal processing.
Settings include: settings to select the signal source, to create the waveform, to select the destination, and to transfer the waveform. The dialog also provides basic simulation control functions that are required for signal preview and are specific to each scenario type.
The dialog also shows an interactive block diagram of the main processing blocks, together with their current settings and state.

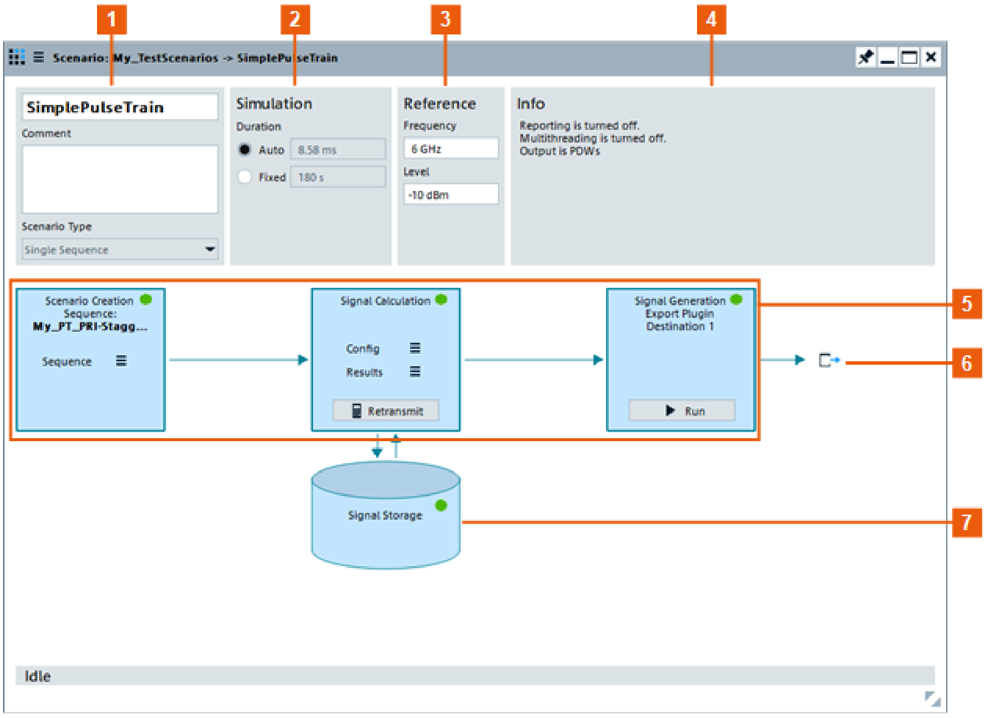


Figure 5-1: Scenario dialog: Understanding the displayed information

1 = Scenario overview
2 = Simulation duration settings (see "Duration" on page 349)
3 = Selected frequency information
4 = Info: report generation, scenario setup, warnings about missing/incompatible assignments or configurations when red LEDs display
5 = Block diagram of the processing chain (see "Scenario creation" on page 79, "Signal Calculation" on page 80, Chapter 5.2.1, "Common scenario settings", on page 75
6 = Selects and indicates the destination for the generated signal (see Chapter 3.3, "Destinations settings", on page 56)
7 = Represents the storage that the simulation data is stored in
Green LED = Indicates suitable source and destination
Yellow LED = Indicates sequence or export errors
Red LED = Indicates that there is no source, an incompatible sequence component or no destination selected
Dotted line = Interrupted signal flow, meaning there is no calculated waveform or no destination selected

The provided settings depend on the selected scenario type and on the target for signal generation (destination plugin).

5.2.1 Common scenario settings

The following settings are common to all scenario types:

Scenario Name

Enter a unique name to identify your current scenario in the repository.

Remote command:
`SCENario:CREate` on page 426
`SCENario:CATalog?` on page 426
`SCENario:NAME` on page 427
`SCENario:SELect` on page 427
`SCENario:REMove` on page 428

Comment

Enter a short description of the scenario. The first line is displayed in the repository tree.

Remote command:
`SCENario:COMMent` on page 428

Scenario Type

Displays the scenario type.

Remote command:
`SCENario:TYPE` on page 544

Simulation
Contains settings relating to the scenario duration and interleaving in appropriate scenarios.

Interleaving ← Simulation
Option:R&S PULSE-K39.

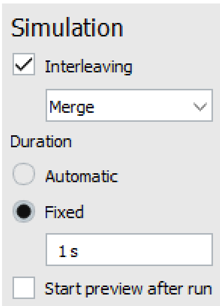
Enables interleaving in scenarios that support it.

For a description of interleaving, see:
- "About the interleaving" on page 238
- "Interleaving" on page 82.

Remote command:
SCENario:LOCalized:INTerleaving on page 545
SCENario:DF:INTerleaving on page 545
SCENario:CEMit:INTerleaving on page 545
SCENario:CPDW:INTerleaving on page 545

Mode ← Interleaving ← Simulation
Enabled if "Interleaving > On".

Defines the interleaving mode.



Select "Drop" or "Merge" mode.

In "Drop" mode, interleaves the PDWs using a priority-based algorithm that drops overlapping pulses. Set the priority in the Config dialog.

In "Merge" mode, merges all PDWs without dropping.

When using "Merge" mode, the "Priority" column in Config is hidden.

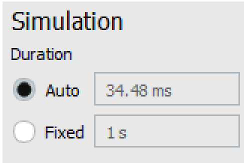
For a description, see "About the interleaving" on page 238 and "Interleaving" on page 82.

Remote command:
SCENario:LOCalized:INTerleaving:MODE on page 545
SCENario:DF:INTerleaving:MODE on page 545
SCENario:CEMit:INTerleaving:MODE on page 545

Duration ← Simulation
Sets the simulation duration and works like the "Duration" settings in the "Signal Calculation > Config > Config > Duration" dialog. If you change the settings here, they update automatically in the "Signal Calculation > Config > Config > Duration" dialog.

The generated signal can:

- Last a specified duration.
  Short sequences are repeated, longer once are truncated.
- Have an automatically determined content and duration, that best fits the current scenario and configuration. After the signal calculation, the duration is displayed.



  Short sequences result in waveform files with small size.
  Short sequences are repeated, longer sequences are truncated.

See also "Duration" on page 349.

Remote command:
`SCENario:OUTPut:DURation:MODE` on page 562
`SCENario:OUTPut:DURation:TIME` on page 562
`SCENario:OUTPut:DURation:AUTO?` on page 562

Start preview after run ← Simulation
Available if a destination is selected: "Signal Calculation > Config > Destinations".

Opens the "2D Movement Preview" automatically after you select "Signal Generation > Run".

Reference
Contains settings relating to the selected frequency.

Frequency ← Reference
Sets the carrier frequency of the instrument on which the generated waveform is modulated on.

Emitters use their current operating frequency, see "EIRP" on page 221.

Remote command:
`SCENario:OUTPut:FREQuency` on page 563

Level ← Reference
Sets the reference level.

This value is used by the calculation of the pulse envelope. The reference level corresponds to the "Ref." line, displayed on the "Pulse Envelope Graph" as a relative ratio. For example, "Ref. = 1" corresponds to 100% * "Level".

For unattenuated pulses, the "Level" sets the pulse top power.

Example: Pulse top power of an unattenuated pulse

If:

- "Level = -30 dBm"
- "Pulse > Level > Attenuation > Top Power = 0 dB"

Then the pulse top power would be -30 dB (at the RF output).

Example: Pulse top power of an attenuated pulse

If:
- "Level = -30 dBm"
- "Pulse > Level > Attenuation Top Power = 10 dB"

Then the pulse top power would be -40 dB (at the RF output).

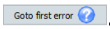
If:
- "Level = -30 dBm"
- "Pulse > Level > Attenuation > Top Power = 0 dB"
- Δ Level = 10 dB

Then the pulse top power would be -20 dB (at the RF output).

Remote command:
`SCENario:OUTPut:LEVel` on page 563

Info

Displays status information on the current scenario setup:

- Report generation status, type and file location
  To enable report generation, select "Signal Calculation > Config > Config > Reporting > Reporting > Enable"
  See also "Reporting" on page 351 and Chapter 21, "Creating reports and documenting measurement results", on page 386.
- Warnings about incomplete or incompatible settings.
  For troubleshooting, open the settings dialog directly by clicking 

.

Scenario creation

Groups and accesses settings related to the signal source.

Emitter/Emitters ← Scenario creation

Access settings related to emitters.

"Emitter" Available in single emitter scenarios.
Opens a context menu that enables you to edit, add or clear an emitter. You can also select an available emitter for the scenario from the menu.

"Emitters" Available in multiple emitter scenarios.
Opens the "Emitters (Collection)" dialog where you can change the settings for multiple emitters.
For more details, see single or multiple emitters.

Map ← Scenario creation

Opens the 2D map in map-based scenarios.

See Chapter 16.2, "2D map settings", on page 298.

PDW lists ← Scenario creation

Opens the "PDW Lists (Collection)" dialog where you can edit, add and remove lists for the scenario.

See Chapter 18, "Working with PDWs", on page 355

Sequence/Sequences ← Scenario creation
Access settings related to the pulse sequence.

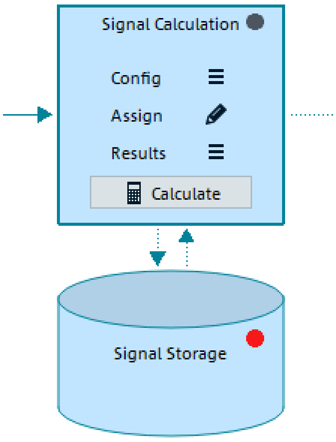
"Sequence" Opens a context menu that enables you to edit, add or clear a sequence. You can also select an available sequence for the scenario from the menu.
Available in "SimplePulseTrain" and "PDW" scenarios.

"Sequences" Opens the "Sequences (Collection)" dialog where you can change the settings for multiple sequences.
Available in "PT Collection" scenarios.
For more details, see single or multiple pulse sequences, a waveform sequence.

Signal Calculation
Groups and accesses settings related to the signal generation and the calculated output signal.

Config ← Signal Calculation
In general, the software calculates a waveform most suitable for the target currently selected in the "Signal Generation" block.

If you prefer to change these settings, however, select "Config".



"Output Format"
Accesses a dialog where you can change the settings influencing the waveform generation, see Chapter 17.1, "Signal calculation settings", on page 348.

"Open Report Folder"
Opens the report folder in Windows Explorer, see Chapter 21, "Creating reports and documenting measurement results", on page 386.

"Target" Allows you to select your target "Destination" from a list.

"Clear Target"
Removes the selected target and sets the selection to "None".

"Destinations"
Accesses the "Destinations" dialog, see Chapter 3.3, "Destinations settings", on page 56.

Remote command:
`SCENario:STARt` on page 546
`SCENario:STOP` on page 546
`SYSTem:PROGress?` on page 594
`SCENario:STATe?` on page 546

Assign ← Signal Calculation
In a "Scenario Type = Localized Emitters/Emitters (Collection)/Direction Finding/PDW List (Collection)", accesses the dialog for mapping of signals to Destinations.

Assigns signals to an existing plugin in the scenario.

See Chapter 17.2, "Assign signals to destination", on page 351.

See Chapter 3.3, "Destinations settings", on page 56.

Results ← Signal Calculation
To observe the calculation results, select "Results".

Depending on your user rights, you have read-only or full access to the repository. Users granted with full access rights are allowed to perform all the following functions:

"View"
Selects a generated file and visualizes it.
See Chapter 15, "Visualizing and analyzing signals", on page 284.

Remote command:
`SCENario:VOLatile:SEL` on page 565
`SCENario:VOLatile:VIEW` on page 565

"Clear calculated results"
Removes the calculated results from the scenario but not from the storage. Change the required settings and calculate again.

Remote command:
`SCENario:CACHe:VOLatile:CLEar` on page 547

"View signal statistics"
Opens the "Signal Statistics" dialog where you get information about the signal parameters, e.g. frequency, level and bandwidth. The results are displayed in a table.

Calculate ← Signal Calculation
Select to start the signal calculation.

Starts signal calculation, if the prerequisites are fulfilled.

The software informs you about the estimated file size. A "Busy" indication in the status bar and detailed progress information indicates that the calculation is in progress.

"Stop" Aborts the current calculation process.

Remote command:
`SCENario:CALCulate` on page 546
`SCENario:STOP` on page 546

Retransmit ← Signal Calculation
After calculating, you can retransmit the calculated data to all destinations.

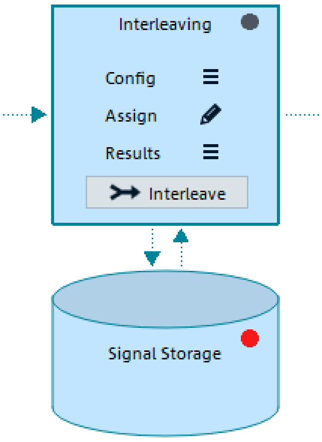
Interleaving
Option:R&S PULSE-K39

Displayed in "PDW Lists (Collection)", "Emitters (Collection)", "Localized Emitters" and "Direction Finding" scenarios, if "Simulation > Interleaving > On". See "Simulation" on page 77.

Groups and accesses settings related to the interleaving process and the analysis of interleaved and dropped PDWs.

Enables you to combine the emitters or PDW lists into multiple output files using groups, "Merge" mode, and a priority-based dropping algorithm, "Drop" mode.

The interleaving is a separate optional post-processing step in the signal calculation. If this step is required, it can be triggered separately (see "Simulation" on page 77).



See also:

- "Emitters (Collection) scenario settings" on page 86
- "Localized emitters scenario settings" on page 88
- "Direction finding scenario settings" on page 88
- "PDW list scenario settings" on page 89

Config ← Interleaving
Allows you to configure the interleaving settings.

See:

- Chapter 11.1, "Emitters (Collection) and localized emitters interleaving settings", on page 239
- Chapter 18.6, "PDW list (Collection) interleaving settings", on page 370

"Interleaving Groups"
Accesses a dialog where you can change the settings influencing the waveform generation, see Chapter 17.1, "Signal calculation settings", on page 348.

"Destinations"
Accesses the "Destinations" dialog, see Chapter 3.3, "Destinations settings", on page 56.

Assign ← Interleaving
In a "Scenario Type = Localized Emitters/Emitters (Collection)/Direction Finding/PDW List (Collection)", accesses the dialog for mapping of signals to Destinations.

Assigns signals to an existing plugin in the scenario.

See Chapter 17.2, "Assign signals to destination", on page 351.

See Chapter 3.3, "Destinations settings", on page 56.

Results ← Interleaving
To observe the statistical information on the interleaved and dropped PDWs or pulses, select "Results".

Depending on your user rights, you have read-only or full access to the repository. Users granted with full access rights are allowed to perform all the following functions:

"View" > "Analyze"
: Opens the "Analyze Interleaving" dialog.
See Chapter 18.9, "Analyze interleaving", on page 375.

"View" > "Interleaved/Dropped PDWs"
: Opens the "Dropped PDWs" dialog presents a visualization of the results.
See Chapter 18.8, "Signal preview with interleaved and dropping statistics", on page 374.

"Clear calculated results"
: Removes the calculation results and enables you to edit the scenario settings and recalculate.

Remote command:
`SCENario:CACHe:VOLatile:CLEar` on page 547

"View signal statistics"
: Opens the "Signal Statistics" dialog where you get information about the signal parameters, e.g. frequency, level and bandwidth. The results are displayed in a table.

Interleave ← Interleaving
Option:R&S PULSE-K39

Select to start interleaving.

Available in "PDW List (Collection)" and "Emitter (Collection)" scenarios, if "Interleaving > On" is used.

Triggers the calculation of a single output file that comprises the individual PDWs or pulses, where overlapping PDWs or pulses are dropped, based on a defined priority.

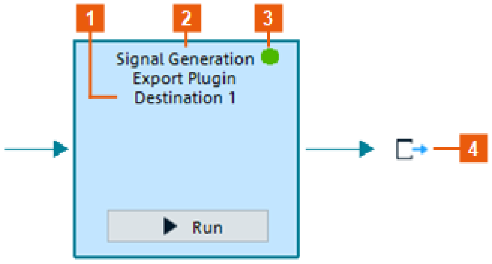
The "Interleaving progress" dialog visualizes the number of PDWs or pulses that are included in the output file and the percentage of other that are dropped.

See "Interleaving" on page 82.

Remote command:
`SCENario:INTerleave` on page 546

Retransmit ← Interleaving
After calculating, you can retransmit the calculated data to all destinations.

Signal Generation
Select "Signal Generation" > "Config > Destinations..." to transfer the simulated data to an export plugin.



1 = Destination, as selected with "Config > Destination"
2 = Current setup name, as selected in the dialog "menu bar > Configure > Destinations", see Chapter 3.3, "Destinations settings", on page 56
3 = Status indication
4 = Simulation export destination as selected in Chapter 3.3, "Destinations settings", on page 56

Run/Stop ← Signal Generation
Provides the following functions:

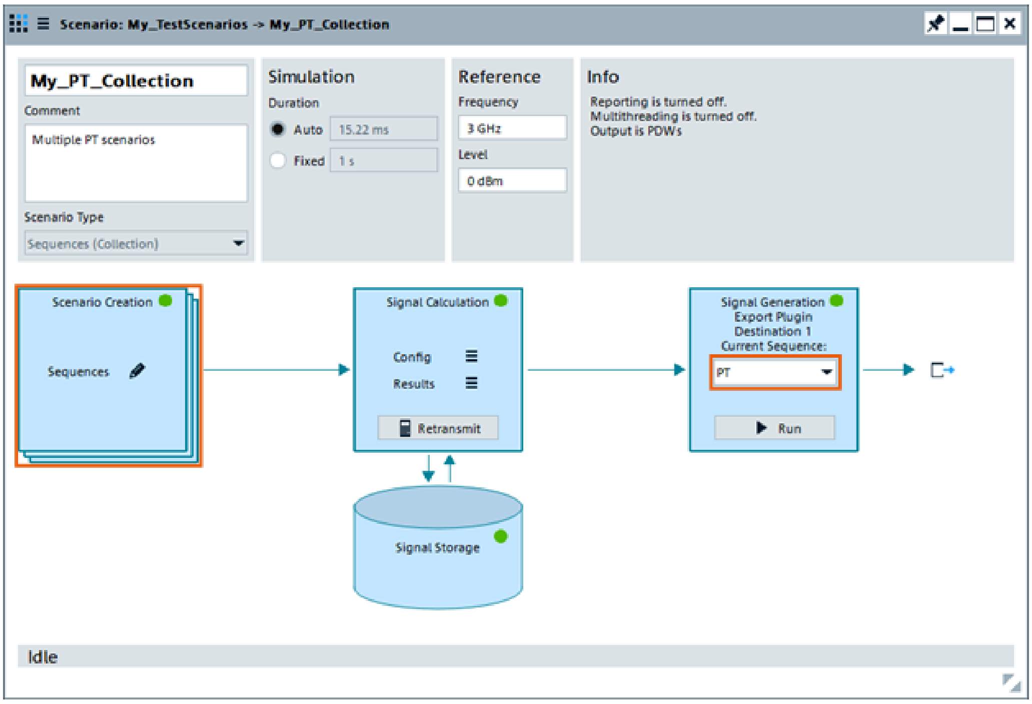
"Run" Calls the run method of the export plugin.
Use this function to perform a task like streaming all PDWs.

"Stop" Calls the stop method of the export plugin if needed.

Remote command:
`SCENario:STARt` on page 546
`SCENario:STOP` on page 546
`SYSTem:PROGress?` on page 594
`SCENario:STATe?` on page 546

5.2.2 Scenario-specific settings

Sequence (Collection) Scenario Settings



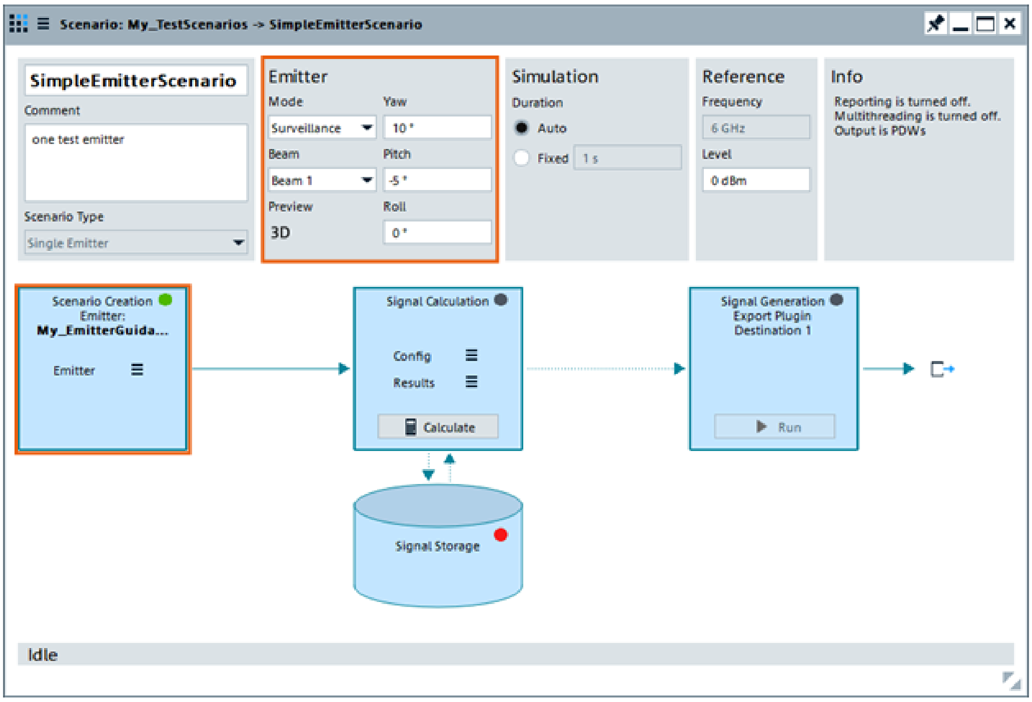
A scenario with multiple sequences requires a list of sequences. You can switch between these sequences using "Scenario Creation" menu and select the one to be transmitted.

The "Current Sequence" field in the "Signal Generation" block indicates the alias name of the sequence currently selected.

See:
- Chapter 7.2.5, "Lists with multiple sequences", on page 129
- Chapter 5.2.1, "Common scenario settings", on page 75

Remote command:
SCENario:CSEQuence on page 556
SCENario:CSEQuence:CURRent on page 548

Single emitter scenario settings



Use the context menu in the "Scenario Creation" block to select and configure the emitter.

You can switch between the emitter modes, define the current beam, set the attitude, and visualize the signal received by a static receiver on a 3D preview.

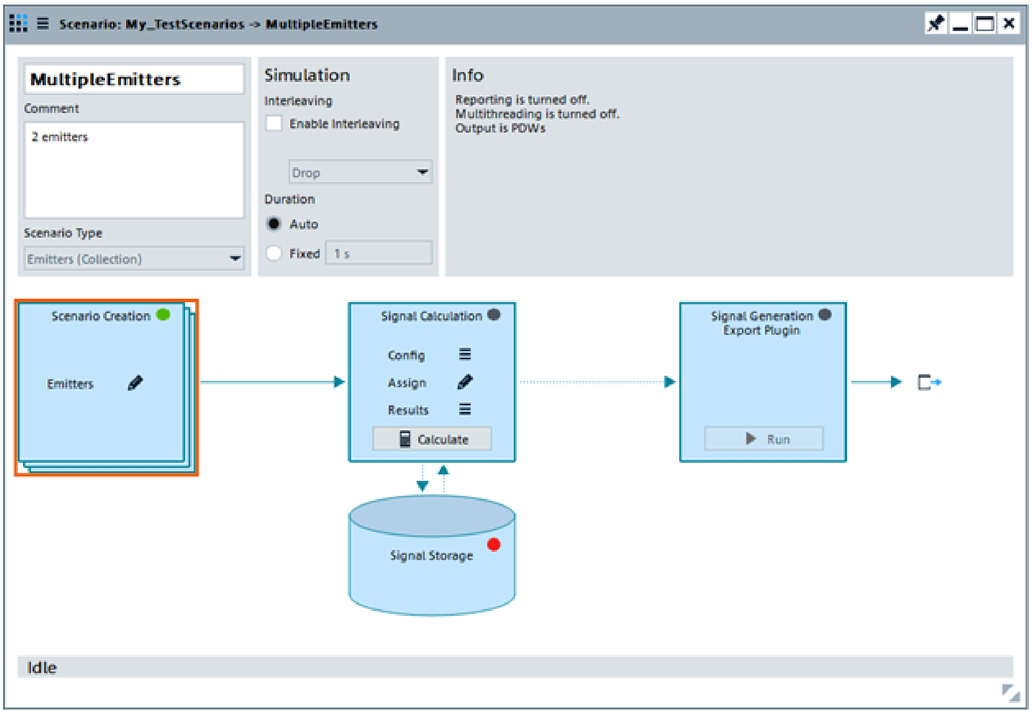
In this scenario, the "Frequency" is a read-only parameter that displays the Frequency value of the current emitter (as selected in the "Emitter" dialog).

See:
- "Emitter attitude" on page 218
- Chapter 10.4, "How to create and configure emitters", on page 233
- Chapter 10.1, "Emitter settings", on page 220
- Chapter 5.2.1, "Common scenario settings", on page 75

Remote command:
`SCENario:EMITter` on page 549
`SCENario:EMITter:MODE` on page 550
`SCENario:EMITter:MODE:BEAM` on page 550
`SCENario:EMITter:CLEar` on page 434
`SCENario:EMITter:DIRection:PITCh` on page 549
`SCENario:EMITter:DIRection:YAW` on page 548
`SCENario:EMITter:DIRection:ROLL` on page 549

Emitters (Collection) scenario settings

You can use "Emitter (Collection)" scenario to test the receiver's ability to detect the signal from different static emitters. In this dedicated scenario, you can configure several emitters and switch between them sequentially. You can also configure the receiver and change its position in the scanning beam of the emitters.



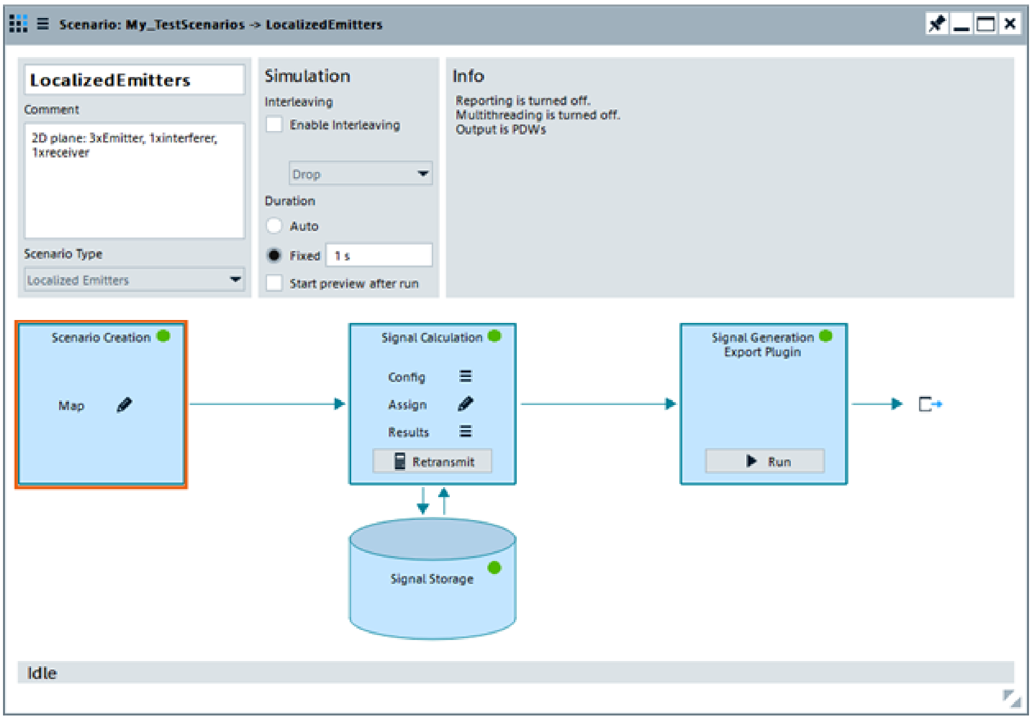
In addition to processing the emitters sequentially, if the selected destination supports **interleaving**, you can also combine them into multiple output files using groups and a priority-based dropping algorithm. During the interleaving process, overlapping pulses with lower priority are dropped. You can configure the priority of the individual emitters.

If "Simulation > Interleaving > On", the block diagram shows the dedicated Interleaving block.

See:
- Chapter 11, "Combining the signals of multiple emitters", on page 238
- "Interleaving" on page 82
- "To configure complex scenarios with several emitters" on page 235
- Chapter 10.3, "Emitters (Collection) settings", on page 225

Remote command:
`SCENario:CEMit:INTerleaving` on page 545

Localized emitters scenario settings



You can use the "Localized Emitter" scenario to test the receiver's ability to detect the sum signal from different static emitters. In this dedicated scenario, you configure the signal of one or more emitters that would be received by a receiver with defined characteristics. You can also configure the receiver and change its position in the scanning beam of the emitters.

Select "Scenario Creation > Map" to display the 2D view of the receiver and currently configured emitters, together with their main characteristics.

In this preview, you see the current position and attitude of the emitters.

This feature is only available after a destination has been set via "Signal Generation > Config".

In addition to processing the emitters sequentially, if the selected destination supports interleaving, you can also combine them into multiple output files using groups and a priority-based dropping algorithm. During the interleaving process, overlapping pulses with lower priority are dropped. You can configure the priority of the individual emitters.

If "Simulation > Interleaving > On", the block diagram shows the dedicated Interleaving block.

See:

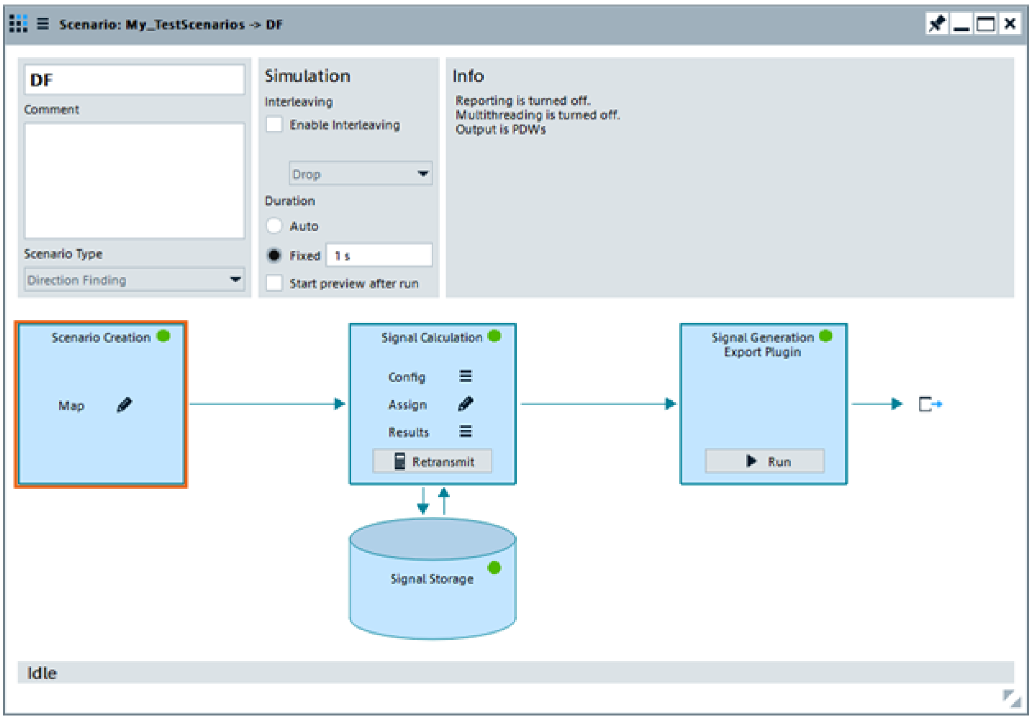
- Chapter 16.1, "How to create scenarios with receiver and TX items", on page 290
- Chapter 17.2, "Assign signals to destination", on page 351
- Chapter 5.2.1, "Common scenario settings", on page 75

Remote command:
`SCENario:LOCalized:INTerleaving` on page 545

Direction finding scenario settings
Option:R&S PULSE-K39

The "Direction Finding" scenario simulates multiple emitters and a single receiver with up to 20 antennas on a 2D map. This scenario is suitable to test direction finding systems and to detect interfering signals or transmissions in unauthorized bands.

The generated files are the individual signals of each receive antenna.



Select "Scenario Creation > Map" to display the 2D view of the receiver and the configured emitters, together with their main characteristics.

In addition to processing the emitters sequentially, if the selected emitters supports interleaving, you can also combine them into multiple output files using groups and a priority-based dropping algorithm. During the interleaving process, overlapping pulses with lower priority are dropped. You can configure the priority of the individual emitters.

If "Simulation > Interleaving > On", the block diagram shows the dedicated Interleaving block.

See:

- Chapter 13.2, "Receiver settings", on page 264
- Chapter 16.1, "How to create scenarios with receiver and TX items", on page 290
- Chapter 17.2, "Assign signals to destination", on page 351

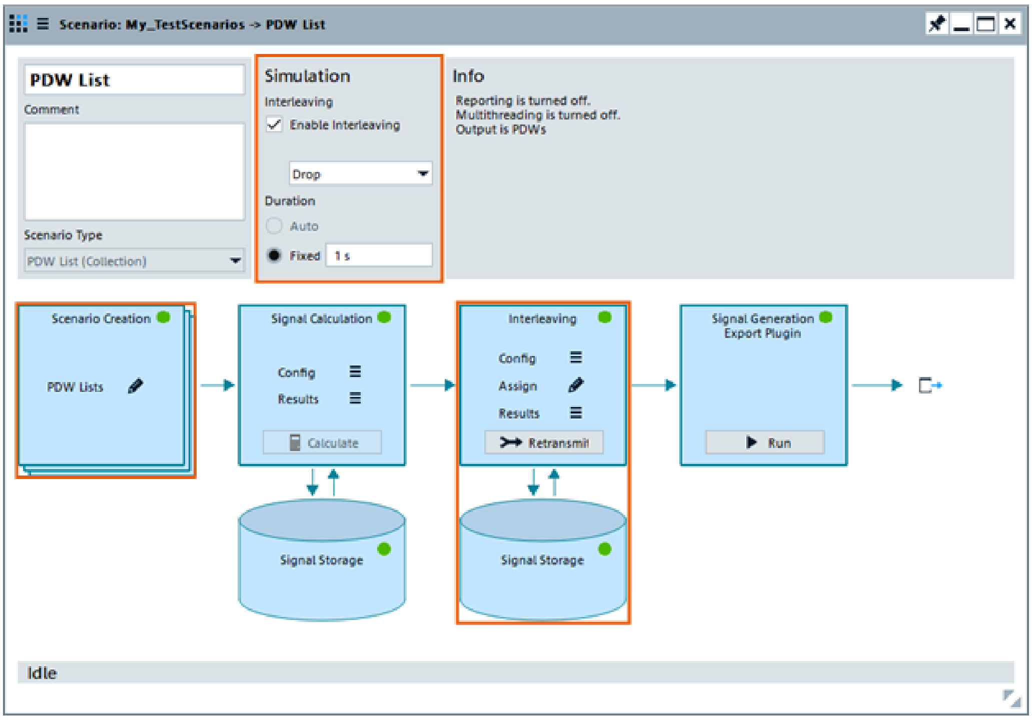
Remote command:
`SCENario:DF:INTerleaving` on page 545

PDW list scenario settings
Option:R&S PULSE-K39

The "PDW List (Collection)" scenario is dedicated for importing custom PDW lists from legacy test cases and importing them into the application.

Moreover, you can use the "PDW List (Collection)" scenario to configure several PDW lists and switch between them sequentially. You can also combine them into multiple output files using groups and a priority-based dropping algorithm. During the interleaving process, overlapping PDWs with lower priority are dropped, where you define the priority of the individual PDW lists.

If "Simulation > Interleaving > On", the block diagram shows the dedicated Interleaving block.



See:
- Chapter 18, "Working with PDWs", on page 355

Remote command:
`SCENario:CPDW:CURRent` on page 548
`SCENario:CPDW:INTerleaving` on page 545

5.3 How to select and create a test scenario

The overview in Table 5-1 helps you decide which scenario type is suitable for a certain test case.

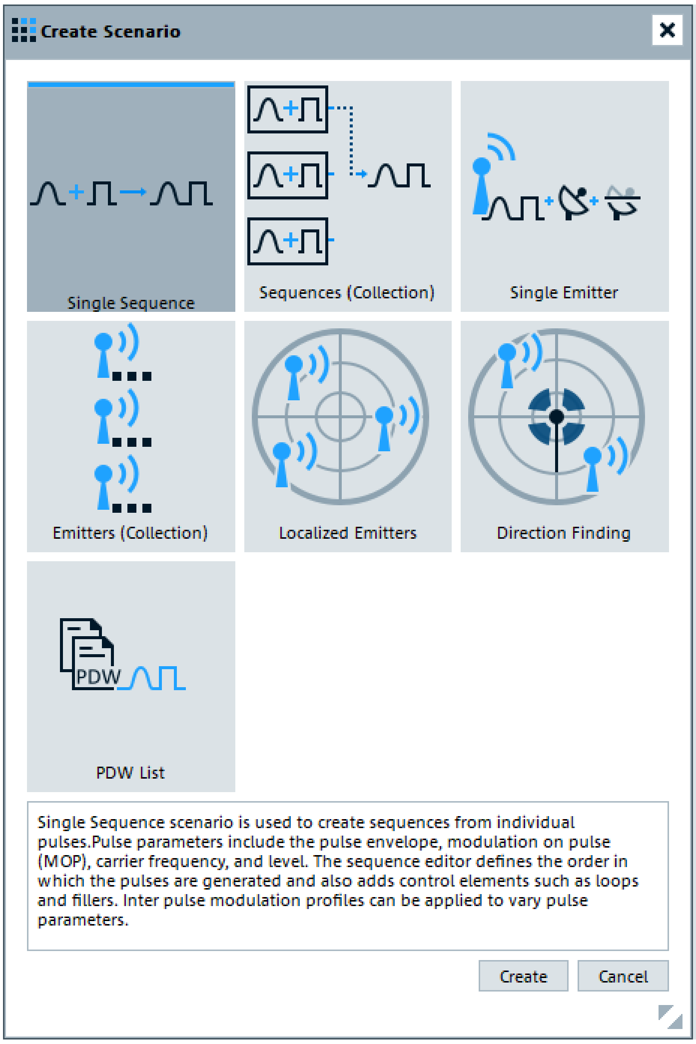
Table 5-1: Typical test cases with their suitable scenario types

Test case	Scenario type
Low-level receiver tests Receiver tests with IPM profiles and modulation on pulse	Single sequence Sequences (collection)
Receiver stress tests, where the receiver antenna is an omnidirectional antenna. The emitted signal follows the amplitude changes of antenna patterns and scans Simulation of the signal at the receiver antenna output The receiver power is normalized to maximum	Single emitter Emitters (collection)
Advanced receiver tests, like reception of multiple signals at the same time. The received signals can be the signals of emitters also in the presence of the background noise of many other emitters Emitters and receiver can be moving objects The emitter and the receiver antenna patterns and antenna scans are considered Receiver power is calculated according to free space loss between emitter and receiver	Localized emitters
Receiver tests for direction finding applications Option:R&S PULSE-K39	Direction finding
Import of custom PDW list Merging of multiple PDW lists into a single output file using a priority scheme for dropping	PDW list
Merging of multiple emitters into multiple output files using groups and a priority-based dropping algorithm.	Emitters (collection) Localized emitters Direction finding

To create a single pulse train scenario

1. To access the scenario settings dialog, perform one of the following:
   a) Select "Repository Tree > Scenario > New".
   b) In the "Toolbar", select the scenario icon .

   A scenario wizard opens that helps you to select a predefined scenario configuration.



2. Select "Scenario Type > Single Sequence".
   Click "Create"

   Created is a new scenario with default name `Scenario <n>`, where `n` is a number starting at one.
   The provided settings depend on the selected scenario type.
   See Chapter 5.2, "Scenario settings", on page 74 for description.

3. Add information to describe and identify the scenario, like a name or a comment.
   Use the standard functions in the context menus to select or create sequences, select the destination, etc.
   See Table 2-4.

   Perform further configurations, see:
   - Chapter 7.3, "How to create sequences and use the control elements", on page 131

6 Creating a pulse library

This section provides background information on pulse generation and pulse characteristics, explains the provided settings and how to use them to create a pulse library.

6.1 Basics on pulse signals and pulse generation

Some background knowledge on basic terms and principles used in pulse generation is provided here for a better understanding of the required configuration settings.

The pulse parameters to be configured are based primarily on the IEEE 181 Standard 181–2003. For detailed descriptions refer to the standard documentation ("IEEE Standard on Transitions, Pulses, and Related Waveforms", from the IEEE Instrumentation and Measurement (I&M) Society, 7 July 2003).

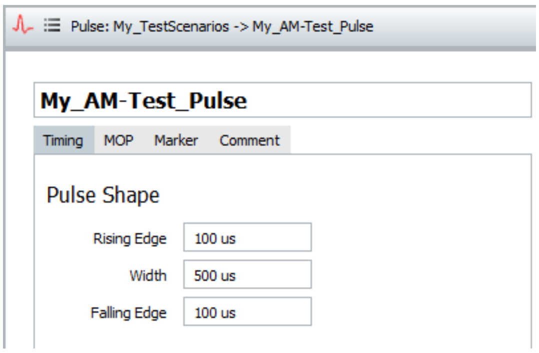
6.2 Pulse settings

Access:

► Select "Repository Tree > Pulse > New".

Pulses are characterized by the following settings:

6.2.1 Pulse name and comment



Settings:

Pulse Name

Displays the name of the current pulse.

Remote command:
`PULSe:CREate` on page 426
`PULSe:NAME` on page 427
`PULSe:CATalog?` on page 426
`PULSe:SELect` on page 427
`PULSe:REMove` on page 428

Comment

General pulse information.

Remote command:
`PULSe:COMMent` on page 428

6.2.2 Pulse timing settings

Access:

► Select "Pulse > Timing".

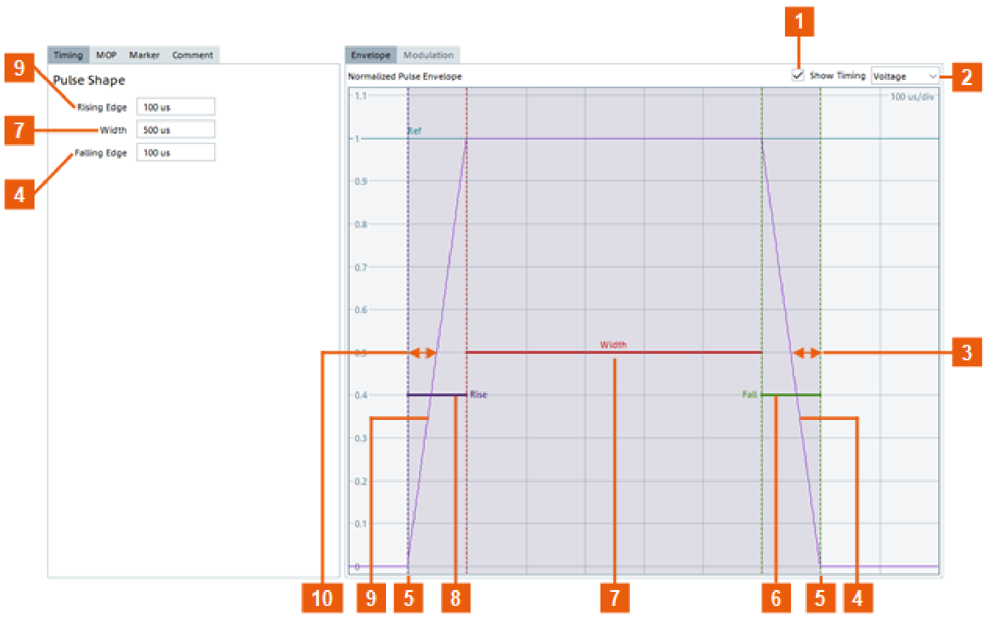


Figure 6-1: Pulse timing: understanding the displayed information

1 = "Show Timing > On" indicates the pulse timing parameters on the envelope graph
2 = "Pulse Envelope Units > Voltage"
3 = Time period between end of pulse and the end of falling edge = 62.5 us, i.e. the time it takes the voltage to fall from 50% to 0% of the top level
4, 6 = "Falling Edge = 100 us", i.e. the time it takes the voltage to fall from 90% to 10% of the top level
5 = Beginning of the rising and falling edge
7 = "Width = 500 us", i.e. the duration the voltage is above 50% of the top level
8, 9 = "Rising Edge = 100 us", i.e. the time it takes the voltage to rise from 10% to 90% of the top level
10 = "Time period between beginning of the rising edge and pulse start = 62.5 us". I.e. the time it takes the voltage to rise from 0% to 50% of the top level

Settings:

Use the following parameters to define the pulse in the time domain:

Pulse Shape Settings

These parameters define the pulse shape:

- Rising/Falling edge
- Width
- Rising/Falling slope

The current pulse shape is displayed on the Envelope graph. To visualize the timing parameters on the graph, select "Envelope > Show Timing > On".

The total pulse duration and the pulse off time are calculated automatically from the selected pulse width, rise and fall time, and the PRI (see PRI/PRF).

"Rising/Falling Edge"
Transition time of the rising/falling edge.

Remote command:
`PULSe:TIME:RISE` on page 520
`PULSe:TIME:FALL` on page 520

"Width" Pulse duration.

Remote command:
`PULSe:TIME:WIDTh` on page 520

Time period between beginning of rising edge and pulse start / Time period between end of pulse and end of falling edge

Display information on the time periods during which the RF power is attenuated or suppressed but per default there is modulation or data content present.

Note: Avoiding data truncation.
Per default, modulation is already present during the rising and falling phases.

6.2.3 Modulation on pulse (MOP) settings

Access:

1. Select "Pulse > MOP".
2. Select "Enable Modulation on Pulse (MOP)".

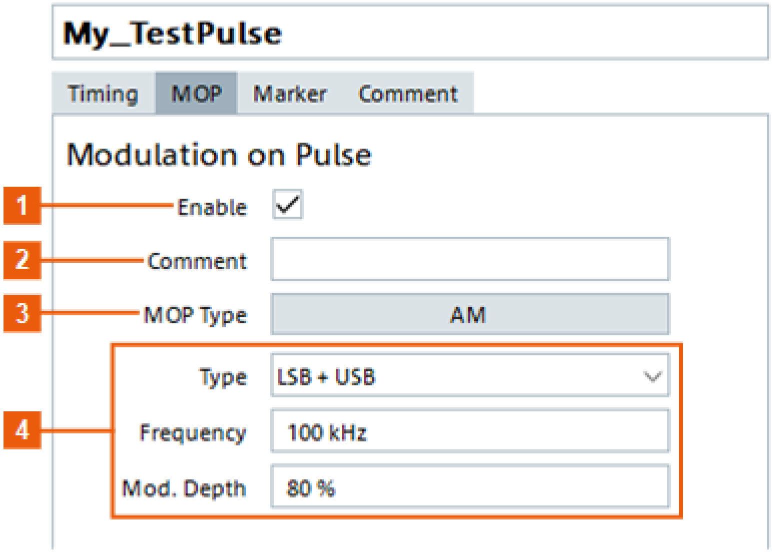


Figure 6-2: MOP settings: understanding the displayed information

1 = Enables pulse signal modulation
2 = "Comment"
3 = "MOP Type" selects the modulation scheme
4 = Modulation parameters, depending on the selected "MOP Type"

Further available settings depend on the selected modulation.

- AM and ASK modulation
- FM and FSK modulation
- Chirp modulation
- Phase modulation
- Vector modulation
- Noise

6.2.3.1 Common MOP parameters

Enable
Enables/disables that the pulse signal is modulated. If the MOP is disabled, the software generates an envelope signal based on the pulse level and timing parameters.

Remote command:
`PULSe:MOP:ENABle` on page 516

Comment

Optional description.

Remote command:
`PULSe:MOP:COMMent` on page 428

MOP Type

Selects a modulation scheme.

Further available settings depend on the selected modulation.

Remote command:
`PULSe:MOP:TYPE` on page 517

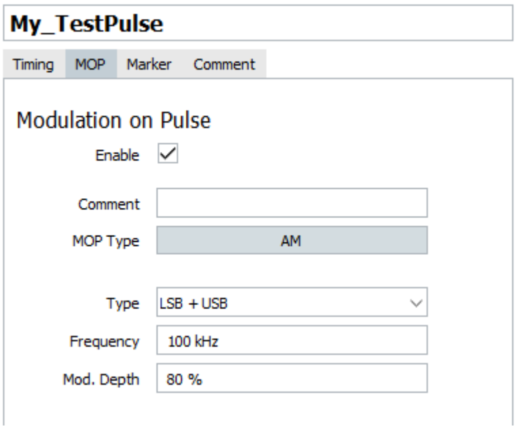
6.2.3.2 AM and ASK modulation

The amplitude modulation (AM) and amplitude shift key modulation (ASK) are modulation schemes that vary the amplitude of the signal.

Settings:

AM

Enables an AM with a single tone.



"Type" Modulation method, like a double-sideband ("Standard"), a low sideband ("LSB"), an upper sideband ("USB"), or a double sideband with suppressed carrier ("LSB+USB") amplitude modulation.

"Frequency" Modulation frequency.

"Mod. Depth"
Depth of the modulation signal in percent, i.e. the peak change in the RF amplitude from its unmodulated value to the amplitude of the unmodulated carrier.

Remote command:
`PULSe:MOP:AM:TYPE` on page 510
`PULSe:MOP:AM:FREQuency` on page 510
`PULSe:MOP:AM:MDEPth` on page 510

AM Step
Defines a modulation as a sequence of discrete states (table rows), each described with its "Duration" and "Level".

Use the standard "Append", "Remove Last", or "Delete All" functions to add or remove a row. The provided settings are self-explanatory.

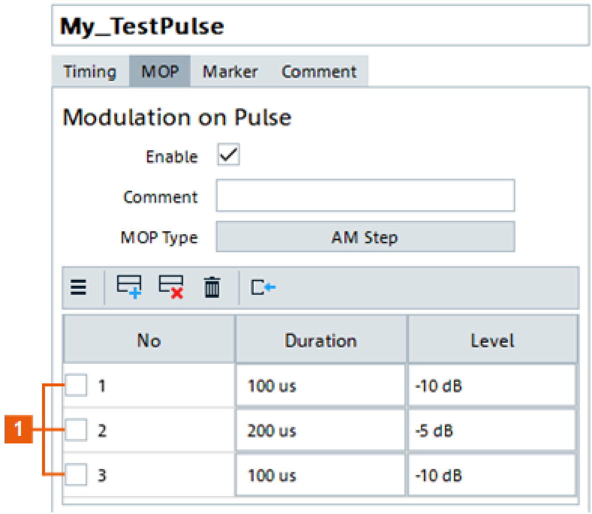


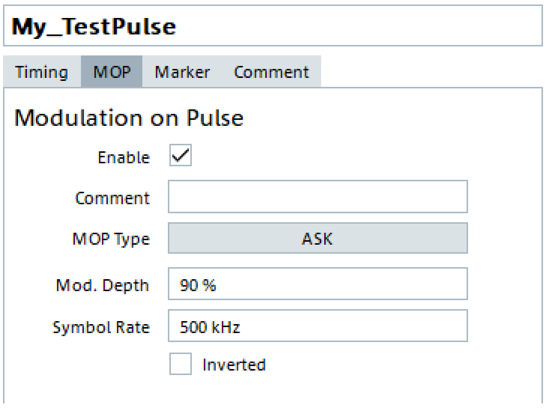
Figure 6-3: MOP Type > AM Step: understanding the displayed information (Pulse Width = 500 us, Restrict MOP to = Pulse Width)

1 = Three discrete states with different attenuation values and a total duration of 400 us

If the total time of the defined states is shorter than the duration of the pulse area on which the MOP is applied, an attenuation of 0 dB is assumed for the remaining time.

Remote command:
`PULSe:MOP:AMSTep:ADD` on page 429
`PULSe:MOP:AMSTep:COUNt?` on page 430
`PULSe:MOP:AMSTep:SELect` on page 431
`PULSe:MOP:AMSTep:INSert` on page 432
`PULSe:MOP:AMSTep:LEVel` on page 511
`PULSe:MOP:AMSTep:DURation` on page 511
`PULSe:MOP:AMSTep:CLEar` on page 433
`PULSe:MOP:AMSTep:DELete` on page 432

ASK
Enables an ASK where the amplitude of the RF carrier is attenuated for a symbol value of zero and remains at full level for symbol values of one. The level of attenuation is specified as depth in percent.



"Mod. Depth"
Depth of the modulation signal in percent.
$m = (Amplitude_{max}-Amplitude_{min}) / (Amplitude_{max}+Amplitude_{min})$

"Symbol Rate" Symbol rate of the modulating signal.

"Inverted" Inverts the amplitude-shift keying (ASK) modulation.
Meaning the RF carrier is attenuated for a symbol value of one.

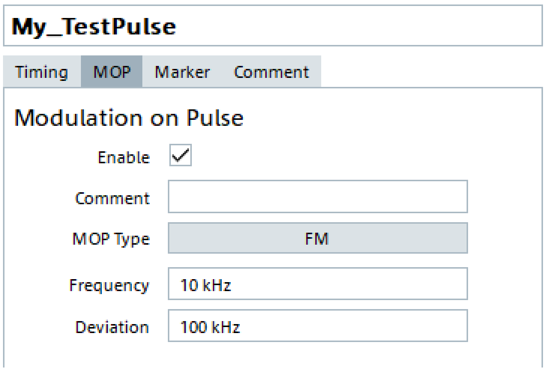
Remote command:
`PULSe:MOP:ASK:MDEPth` on page 512
`PULSe:MOP:ASK:INVert` on page 511
`PULSe:MOP:ASK:SRATe` on page 512

6.2.3.3 FM and FSK modulation

The frequency modulation (FM) and frequency shift keying (FSK) modulation are modulation schemes that vary the frequency of the signal.

Settings:

FM
Enables an FM with a single tone.



"Frequency" Modulation frequency

"Deviation" FM frequency deviation

Remote command:
PULSe:MOP:FM:FREQuency on page 517
PULSe:MOP:FM:DEViation on page 517

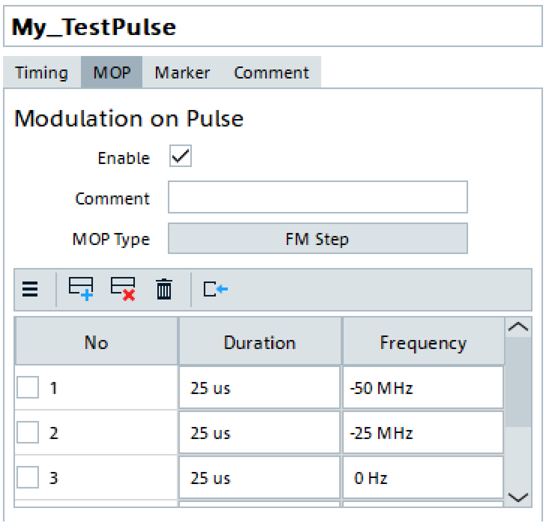
FM Step

The FM step modulation is a common pulse compression technique. It defines a modulation as a sequence of discrete states (table rows), each described with its "Duration" and "Frequency".

Use the standard "Append", "Remove Last", or "Delete All" functions to add or remove a row. The provided settings are self-explanatory.

Example: Sequence composed of four pulses with "MOP = FM Step"

- Configure an FM step with f_{start} = -50 MHz, f_{end} = 50 MHz, Step = 25 MHz, #Steps = 5.

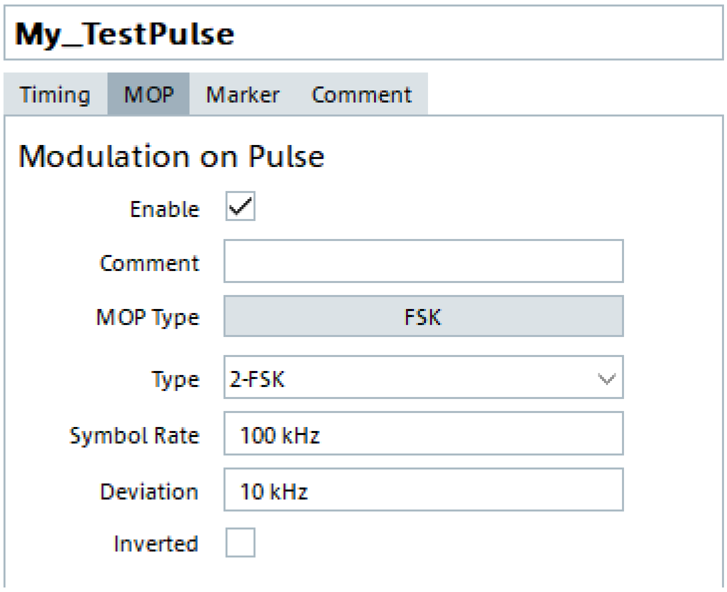


- Build a sequence and use the "Signal Preview" dialog to visualize the signal.

Remote command:
PULSe:MOP:FMSTep:ADD on page 429
PULSe:MOP:FMSTep:COUNt? on page 430

PULSe:MOP:FMSTep:SELect on page 431
PULSe:MOP:FMSTep:INSert on page 432
PULSe:MOP:FMSTep:FREQuency on page 511
PULSe:MOP:FMSTep:DURation on page 511
PULSe:MOP:FMSTep:CLEar on page 433
PULSe:MOP:FMSTep:DELete on page 432

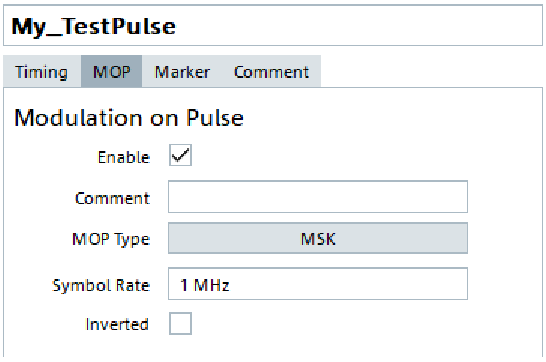
FSK, 4FSK, 8FSK, 16FSK, 32FSK, 64FSK
Enables an FSK. High symbols set the frequency to "+Deviation" and low symbols to "–Deviation".



"Type"	Selects the FSK modulation type.
"Symbol Rate"	Symbol rate of the modulating signal.
"Deviation"	FSK frequency deviation
"Inverted"	Inverts the FSK modulation.

Remote command:
PULSe:MOP:FSK:TYPE on page 515
PULSe:MOP:FSK:DEViation on page 517
PULSe:MOP:FSK:SRATe on page 518
PULSe:MOP:FSK:INVert on page 511
PULSe:MOP:4FSK:DEViation on page 517
PULSe:MOP:4FSK:SRATe on page 518
PULSe:MOP:8FSK:DEViation on page 517
PULSe:MOP:8FSK:SRATe on page 517

MSK
Enables an MSK.



"Symbol Rate" Symbol rate of the modulating signal.

"Inverted" Inverts the MSK modulation.

Remote command:
`PULSe:MOP:MSK:SRATe` on page 512
`PULSe:MOP:MSK:INVert` on page 511

6.2.3.4 Chirp modulation

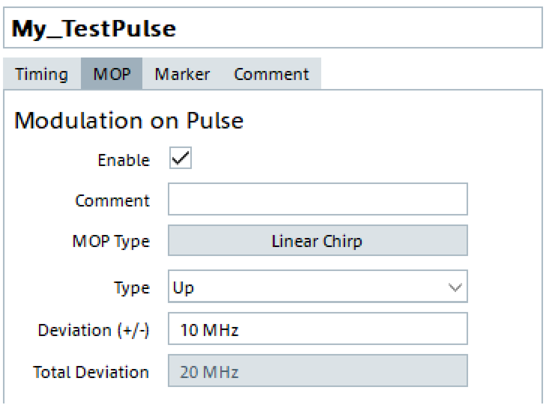
A chirp is a signal in which the frequency varies over the time. The chirp sweeps the RF signal across a set frequency range.

The chirp modulation scheme is used for pulse compression.

Settings:

Linear Chirp

The linear chirp varies the frequency linearly over time.



"Type" Defines the frequency variation:
- Linearly increasing ("Up")
- Linearly decreasing ("Down")
- Following a full sine wave ("Sine")
- Ascending and then descending ("Triangular")

"Deviation" Chirp deviation.
The "Total Deviation" value updates automatically if you change the "Deviation".

Remote command:
`PULSe:MOP:CHIRp:TYPE` on page 514
`PULSe:MOP:CHIRp:DEViation` on page 514

Piecewise Linear Chirp

Define the chirp as a sequence of chirp intervals with user-definable length and frequency deviation.

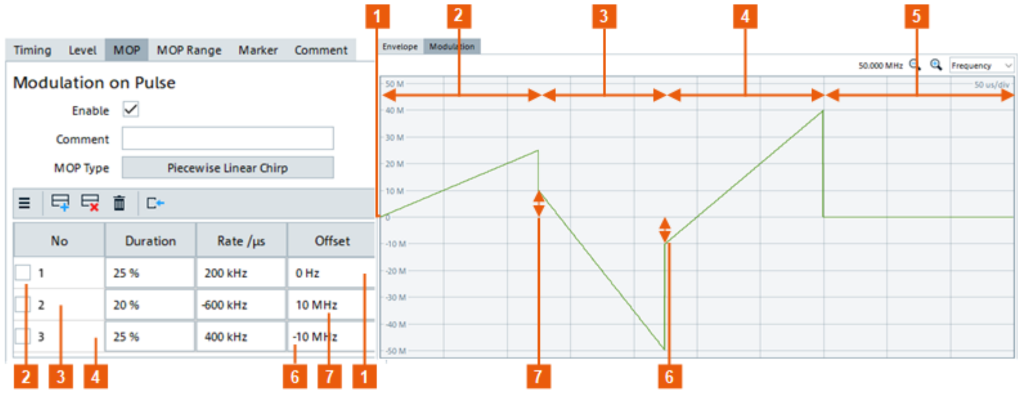


Figure 6-4: Piecewise Linear Chirp: understanding the displayed information

1 = Offset = 0 Hz; the start chirp frequency is 0 Hz
2, 3, 4 = Three chirp intervals with total duration of 70% of the whole time of which the MOP is applied*)
5 = In the remaining 30% of the chirp time, the frequency is kept constant
6 = Negative offset; the start chirp frequency in the third interval is -10 MHz
7 = Positive offset; the start chirp frequency in the second interval is 10 MHz
* = Theoretical example, in practice you rather vary the frequency during the whole time

"Duration" Set the length of each chirp interval as a percentage of the whole duration to which the MOP is applied.
For example, if a pulse width of 100 us and rising/falling edge of 0 us is used and the MOP is applied without restrictions, the whole duration is 100 us.
If the sum of all intervals is less than 100%, constant frequency is used in the remaining part.
If the sum of the defined intervals exceeds 100%, the parts outside the 100% are ignored.

Remote command:
`PULSe:MOP:PIECewise:DURation` on page 515

"Rate/µs" Set the chirp rate in Hz/µs.

Example:

If the MOP is applied on 100 us and a chirp with "Rate = 400 KHz/us" is applied for 25 us ([Duration = 25%]), then the frequency deviation during this interval is 10 MHz.

Remote command:
PULSe:MOP:PIECewise:RATE on page 515

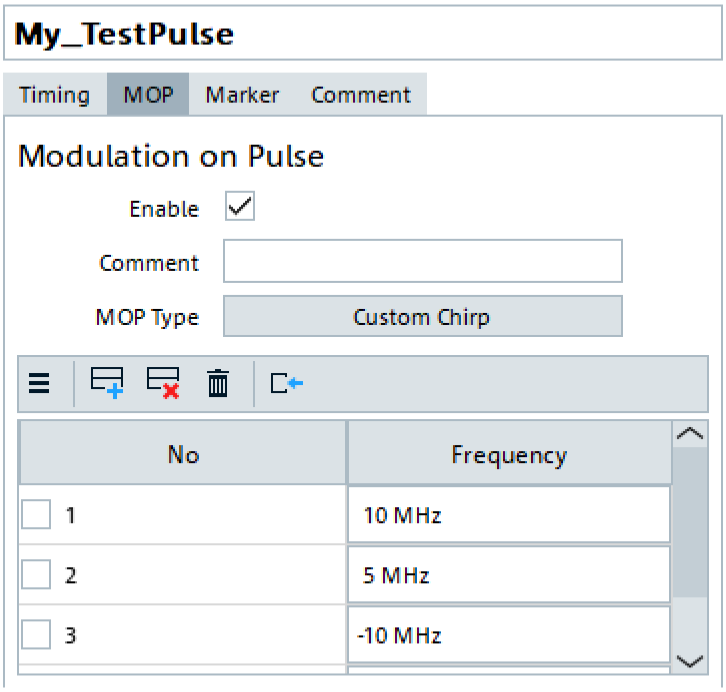
"Offset" Offsets the start frequency of the chirp in the selected interval. Meaning within the interval, the chirp is shifted with the defined value.

Remote command:
PULSe:MOP:PIECewise:OFFSet on page 515

Remote command:
PULSe:MOP:PIECewise:ADD on page 429
PULSe:MOP:PIECewise:COUNt? on page 430
PULSe:MOP:PIECewise:INSert on page 432
PULSe:MOP:PIECewise:CLEar on page 433
PULSe:MOP:PIECewise:DELete on page 432
PULSe:MOP:PIECewise:SELect on page 431
PULSe:MOP:PIECewise:DURation on page 515
PULSe:MOP:PIECewise:RATE on page 515
PULSe:MOP:PIECewise:OFFSet on page 515

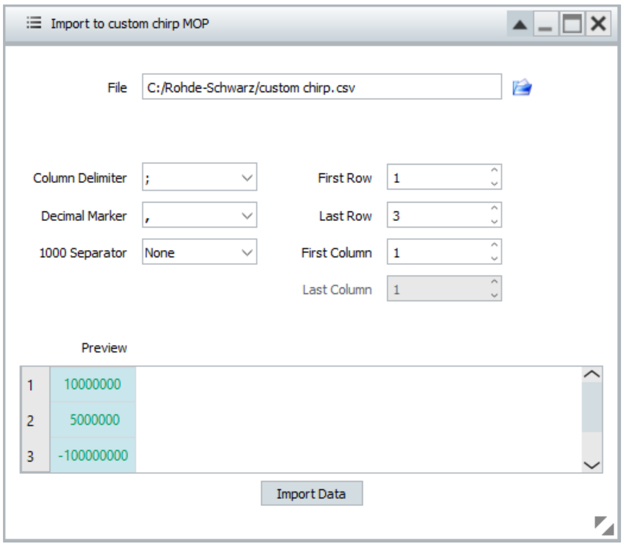
Custom Chirp

The custom chirp modulation is defined as a sequence of discrete "Frequency" stages. The duration of each stage depends on the pulse duration to which the MOP is applied and the number of defined frequency stages.



"New/Select, Append, Insert, Delete, Clear"
: Icons that access standard functions, like add or delete row (see also Table 2-4).
Use the functions in the context-sensitive menu to append new item or reorder the existing item.

"Import data from file"
: Standard file management function to load a custom chirp from and as an ASCII file with `*.txt` or `*.csv` file extension.
When a file is loaded, the "Data Import" dialog assists you to select the used delimiter and separator to interpret the file content.



The provided settings are self-explanatory.

Remote command:
PULSe:MOP:CCHirp:ADD on page 429
PULSe:MOP:CCHirp:INSert on page 432
PULSe:MOP:CCHirp:COUNt? on page 430
PULSe:MOP:CCHirp:SELect on page 431
PULSe:MOP:CCHirp:FREQuency on page 516
PULSe:MOP:CCHirp:DELete on page 432
PULSe:MOP:CCHirp:CLEar on page 433

Polynomial Chirp

Define the chirp as a polynomial with one or more "Term"/"Coefficient" value pairs.

The instantaneous frequency versus time *f(t)* is calculated according to the equation:

$f(t) = \sum (a_n * t^n)$, where:

- $n = 1 .. 32$ is the term
- $a_n = -10^{22}$ *to* 10^{22} are the coefficients.

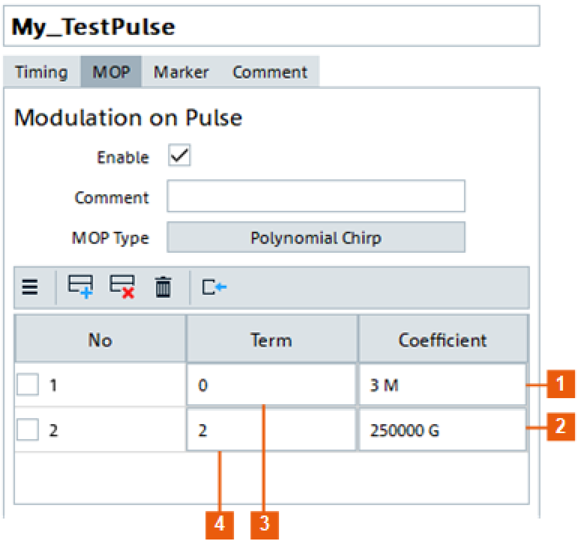


Figure 6-5: Polynomial Chirp: understanding the displayed information

1 = Coefficient a_0 = 3e6 = 3 MHz
2 = Coefficient a_2 = 2.5e14
3 = First term n = 0
4 = Second term n = 2 (the coefficient a_1 is not used)

The polynomial chirp defined with the settings on Figure 6-5 is:

$f(t) = a_0 + a_2 * t^2 = 3e6 + 2.5e14 * t^2$.

Remote command:
PULSe:MOP:PCHirp:ADD on page 429
PULSe:MOP:PCHirp:COUNt? on page 430
PULSe:MOP:PCHirp:SELect on page 431
PULSe:MOP:PCHirp:INSert on page 432
PULSe:MOP:PCHirp:TERM on page 518
PULSe:MOP:PCHirp:COEFficient on page 518

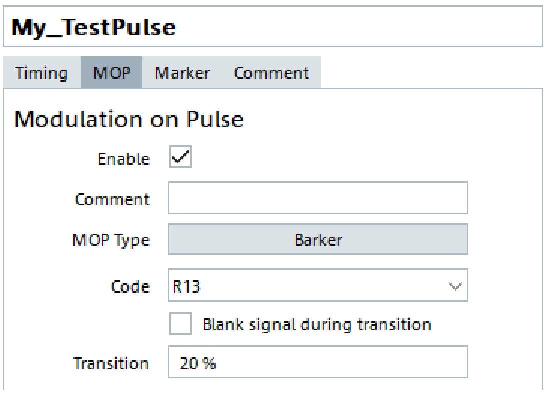
PULSe:MOP:PCHirp:CLEar on page 433
PULSe:MOP:PCHirp:DELete on page 432

6.2.3.5 Phase modulation

Provided are the following phase modulation technics:

Barker

Phase modulation with Barker codes results in signals with low autocorrelation properties.

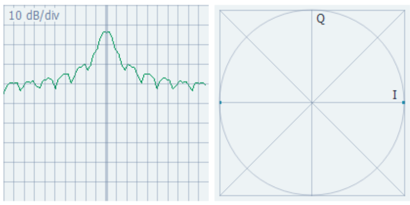
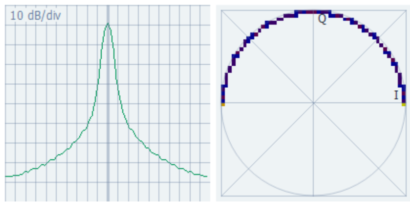


Barker codes are used for pulse compression and are defined by:

"Code" A Barker code is a finite sequence of N values of +1 and −1; R13, for example, is the '+1+1+1+1+1-1-1+1+1-1+1-1+1' sequence.

"Blank signal during transition"
If "Transition > 0%", blanks out the signal during the transition time.

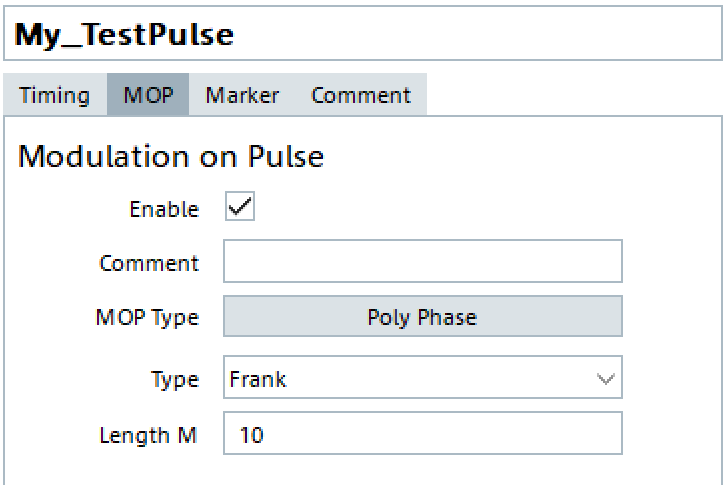
"Transition" Sets the transition duration for constant envelope operation.
A "Transition = 0%" is the border case, where the change for +1 to -1 or conversely is performed immediately. The transition skips (switches back and forth on a direct line) between the constellation points.
With a "Transition > 0%", the two constellation points are still exactly opposite, but the transition follows a circular arc. The values on the transition curve are uniformly distributed.
Observe the effect of a "Transition > 0%" on the modulation envelope, the displayed spectrum, and on the I/Q constellation diagram. The greater the value, the slower the transition, the longer the transition time.

"Transition = 0%"	"Transition = 20%"
	

Remote command:
`PULSe:MOP:BARKer:CODE` on page 512
`PULSe:MOP:BARKer:TTIMe` on page 512
`PULSe:MOP:BARKer:BLANk` on page 513

Poly Phase

A polyphase code modulation is suitable for pulse compression and is used in low probability of intercept (LPI) radars.



"Type" Selection of polyphase compression codes: the default Frank code, and four modified versions of the Frank code, the P1, P2, P3 and P4 codes.

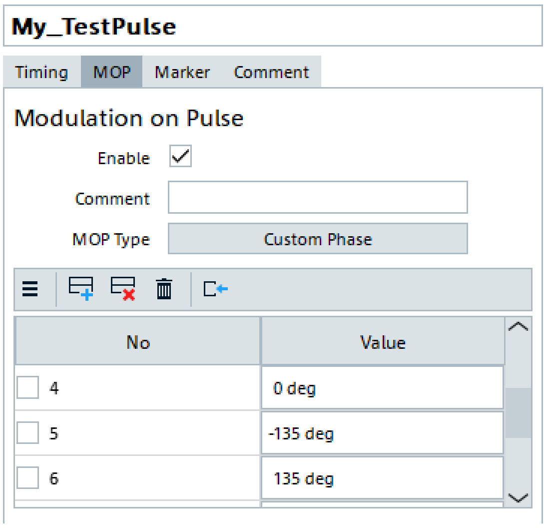
"Length M" Sets the code order, i.e. the number of elements.

Remote command:
PULSe:MOP:POLY:TYPE on page 519
PULSe:MOP:POLY:LENGth on page 519

Custom Phase

Defines a sequence of phase values. The phase changes are distributed evenly.

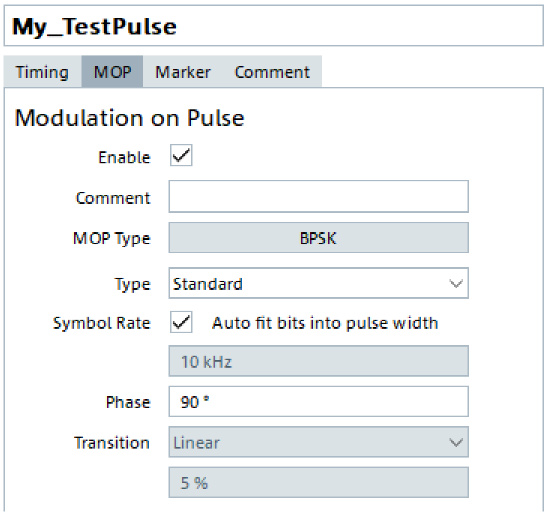
Use the standard "Append", "Remove Last", or "Delete All" functions to add or remove a row. The provided settings are self-explanatory.



Remote command:
PULSe:MOP:PLISt:ADD on page 429
PULSe:MOP:PLISt:COUNt? on page 430
PULSe:MOP:PLISt:SELect on page 431
PULSe:MOP:PLISt:INSert on page 432
PULSe:MOP:PLISt:VALue on page 518
PULSe:MOP:PLISt:CLEar on page 433
PULSe:MOP:PLISt:DELete on page 432

BPSK

A binary phase shift keying (BPSK) modulation applies a phase shift with a selected offset ("Phase") at each constellation point.



"Type" Selects the BPSK modulation method, like a standard BPSK modulation or a BPSK modulation with a constant envelope (C-BPSK).

"Symbol Rate" Symbol rate of the modulating signal, user defined or automatically estimated.

"Phase" Phase shift between the constellation point.

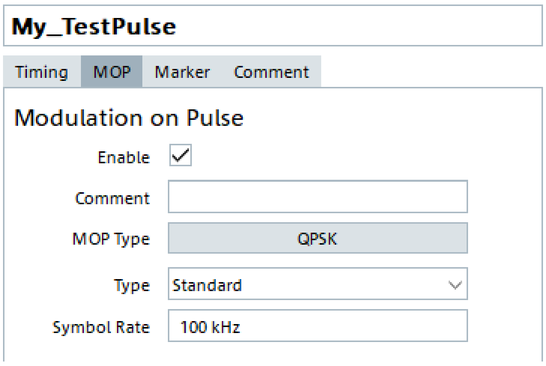
"Transition"

These parameters are important, when a BPSK modulation with a constant envelope is used.
A "Transition = 0%" is a border case, where the change between two constellation points is performed immediately. The transition skips (switches back and forth on a direct line) between the constellation points.
The greater the value, the slower the transition. With a "Transition > 0%", the two constellation points are still exactly opposite, but the transition follows a circular arc.

Remote command:
`PULSe:MOP:BPSK:TYPE` on page 514
`PULSe:MOP:BPSK:SRATe` on page 513
`PULSe:MOP:BPSK:SRATe:AUTO` on page 513
`PULSe:MOP:BPSK:PHASe` on page 513
`PULSe:MOP:BPSK:TTYPe` on page 514
`PULSe:MOP:BPSK:TTIMe` on page 514

QPSK

A quadrature phase shift keying (QPSK) modulation applies a phase shift at each constellation point.



"Type" Selects the QPSK modulation method.

- "Standard"
  A regular QPSK modulation.
- "OQPSK, DQPSK"
  An offset QPSK ("OQPSK"), or a differential QPSK "DQPSK" modulation.
- "SOQPSK-A, SOQPSK-B, SOQPSK-TG"
  Three variants of the shaped-offset QPSK ("SOQPSK") modulation.
  For the aeronautical telemetry "SOQPSK-TG" modulation, you can enable differential encoding according to the telemetry standard IRIG 106-04.

Remote command:
`PULSe:MOP:QPSK:TYPE` on page 519
`PULSe:MOP:QPSK:SOQPsk:IRIG` on page 519

"Symbol Rate"

Symbol rate of the modulating signal.

Remote command:
`PULSe:MOP:QPSK:SRATe` on page 519

8PSK
An 8 phased shift keying modulation applies a phase shift at each constellation point.

"Symbol Rate"

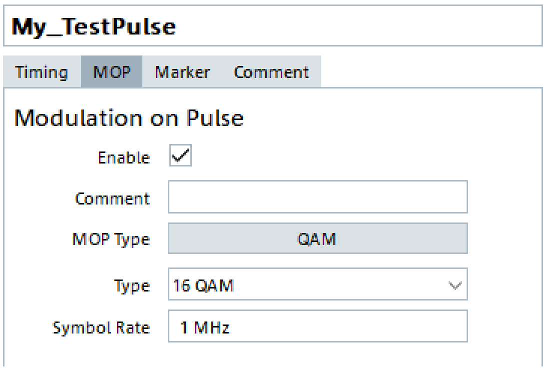
Symbol rate of the modulating signal.

Remote command:
`PULSe:MOP:8PSK:SRATe` on page 517

6.2.3.6 Vector modulation

Provided are the following QAM modulations:

16QAM, 32QAM, 64QAM, 128QAM, 256QAM
A quadrature amplitude modulation QAM applies an amplitude shift at each constellation point.



"Type" Selects the QAM type.

"Symbol Rate"

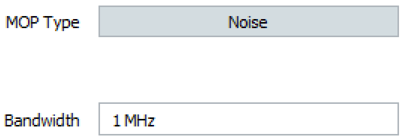
Symbol rate of the modulating signal.

Remote command:
`PULSe:MOP:QAM:TYPE` on page 516
`PULSe:MOP:QAM:SRATe` on page 518

6.2.3.7 Noise

Generates an AWGN noise signal with selected "Bandwidth".

The noise is superimposed during all pulse phases.



Remote command:

`PULSe:MOP:NOISe:BWIDth` on page 518

6.2.4 Envelope and modulation graphs

Envelope graph
Displays the current pulse shape depending on the following:

- The Pulse Shape Settings settings

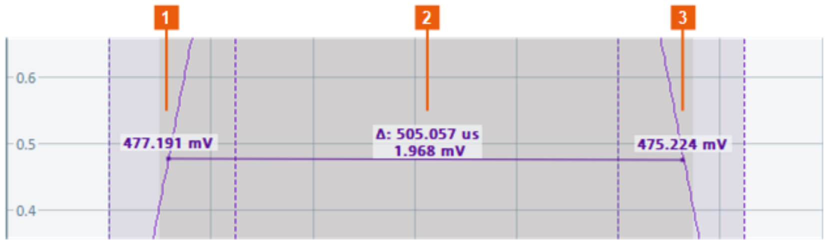
For example, see the envelope graph on Figure 6-1.

"Show Timing" Set "Show Timing > On" to visualize the pulse width, rise and fall time on the graph.
This visualization is useful if you are not familiar with the definition of the pulse parameters.

"Normalized Pulse Envelope > Pulse envelope unit"
Changes the used units.
The options "Voltage", "Watt" and three logarithmic scales with different resolution ("10 dB/Div", "5 dB/Div" and "1 dB/Div") are available.
To display the effect of large overshoots or ripples for example, use the "5 dB/Div" or "1 dB/Div" scale.

"Temporary delta information"
Drag the mouse cursor on the graph.
The delta information is displayed temporarily above the line.

Example: Measuring the pulse width with the help of delta marker



1 = Start y value [units depending on the selected "Pulse Envelope Units"]
2 = Delta information: upper value is the Δx [s], lower value is the Δ y [units depending on the selected "Pulse Envelope Units"]
3 = End y value [units depending on the selected "Pulse Envelope Units"]

Remote command:
`PULSe:PREView:MODE` on page 520

Modulation graph

The MOP is visualized on three diagrams: the time domain view, the spectrum, and the I/Q constellation diagrams.

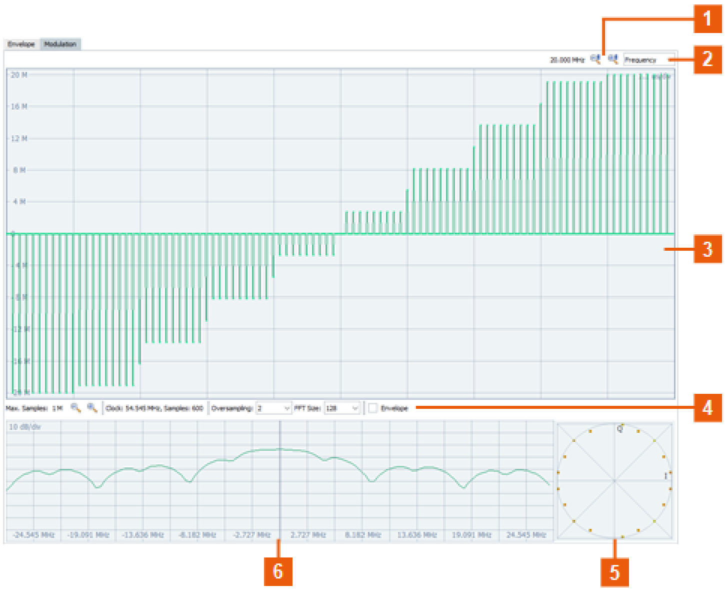


Figure 6-6: Modulation graph: Understanding the displayed information (MOP Type > Polyphase code, Type = P2, Length M = 10)

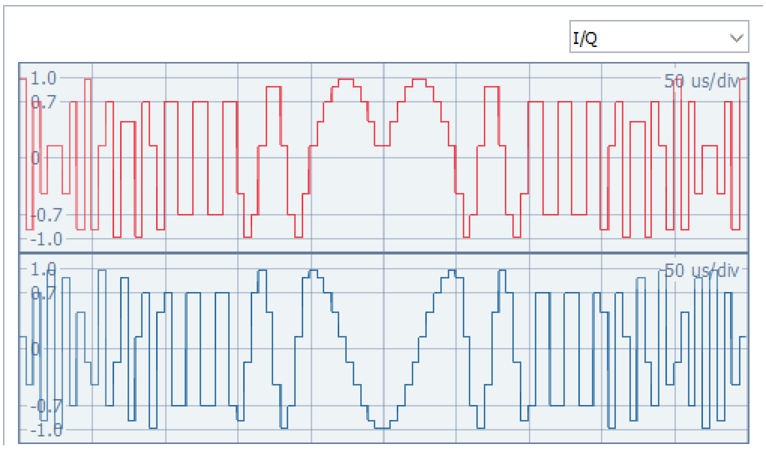
1 = Zoom in/out on the x axis; y axis zoom is supported for the frequency display
2 = Indicated is the "Frequency" variation
3 = Time domain view
4 = FFT preview settings
5 = I/Q constellation diagram
6 = Spectrum

Time domain display ← Modulation graph

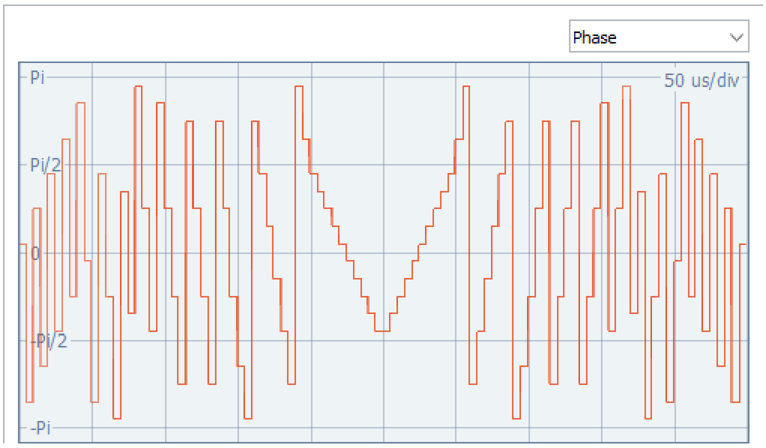
Use the "Modulation" parameter to change the displayed modulation characteristics.

Available are:

"I/Q diagram" Displays the I and Q signal versus time. Both signals use a linear scale in the range between -1.0 and +1.0.



"Phase diagram"
Displays the phase angle of the signal versus time. The diagram shows the phase changes in the range -π and +π.



A phase change is equivalent to a rotation at a constant radius in the constellation diagram.

"Frequency diagram"
Displays the instantaneous signal frequency versus time.

Remote command:
`PULSe:PREView:MOP` on page 521

FFT preview settings ← Modulation graph
Displays information about the FFT preview and allows to set parameters:

"Max. Samples"
Displays the maximum value of samples possible.

"Zoom in/Zoom out"
Changes the resolution on the x and y axis.

"Clock, Samples"
Displays clock frequency and number of samples.

"Oversampling"
Sets the oversampling factor in the modulation preview.

"FFT Size" Sets the FFT size in the modulation preview.

"Envelope" Activates the envelope in the modulation preview.

I/Q constellation diagram ← Modulation graph
Displays the probability at which I/Q points occur.

The color scale is logarithmic; the most probable points are indicated with bright yellow color.

Spectrum ← Modulation graph
Displays the FFT spectrum. The FFT uses a logarithmic scale between 0 dB and -100 dB.

6.3 How to create a new pulse and adjust its settings

Pulses are the fundamental building elements of any sequence and therefore have to be created before you can create a sequence.

See:

- "To create a pulse" on page 116
- "To define the timing parameters of a pulse" on page 116
- "To define and apply modulation on the pulses (MOP)" on page 117
- "To visualize the pulse characteristics" on page 118

To create a pulse

1. Perform one of the following:
   a) Select "Repository Tree > Pulse > New".
   b) In the "Sequence" dialog, select the pulse icon.

   A new pulse with default settings is created and added to the repository tree.
   New pulses are named `Pulse <n>` where n is a number starting at one.
   You can add information to describe and identify the pulse, like a name or a comment.

2. Change the pulse name, e.g. "P1".

   A pulse is described by its timing and level parameters, applied modulation on pulse (MOP) and enabled marker signals.
   For background information on the pulse parameters, see Chapter 6.1, "Basics on pulse signals and pulse generation", on page 93.

   For description of the required settings, see:
   - Chapter 6.2.2, "Pulse timing settings", on page 94
   - Chapter 6.2.4, "Envelope and modulation graphs", on page 112

To define the timing parameters of a pulse

1. In the repository tree, select "Pulse > PulseName e.g. P1 > Timing".
2. Set the timing-related parameters, such as:
   - Delay
   - Rise/fall
3. To configure the pulse repetition interval (PRI), use the parameter PRI/PRF.

4. Use the "Envelope" graph to visualize the current envelope profile.

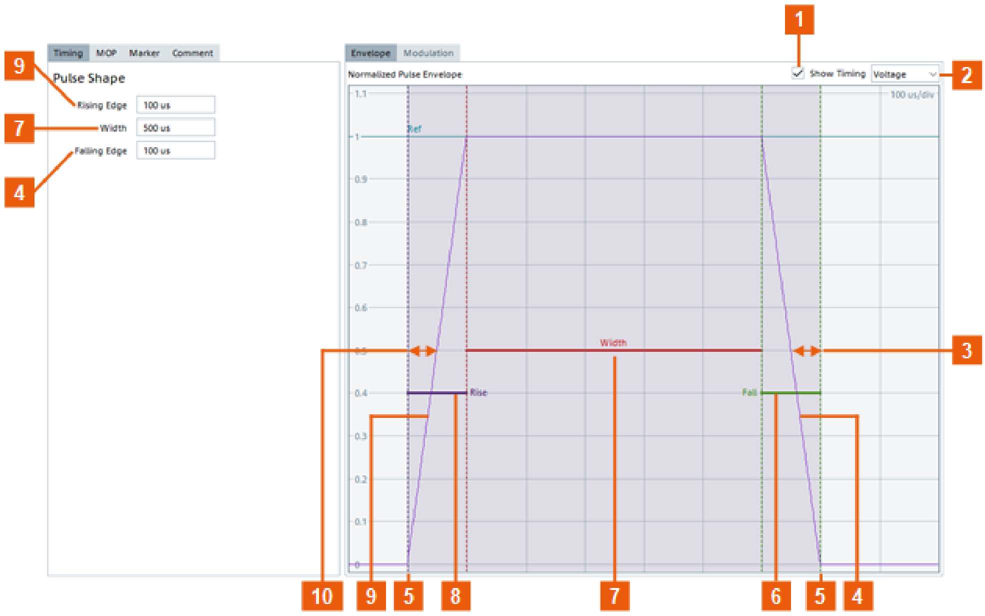


Figure 6-7: Pulse timing: understanding the displayed information

1 = "Show Timing > On" indicates the pulse timing parameters on the envelope graph
2 = "Pulse Envelope Units > Voltage"
3 = "Time period between end of pulse and the end of falling edge = 62.5 us". I.e. the time it takes the voltage to fall from 50% to 0% of the top level
4, 6 = "Falling Edge = 100 us", i.e. the time it takes the voltage to fall from 90% to 10% of the top level
5 = Beginning of the rising and falling edge
7 = "Width = 500 us", i.e. the duration the voltage is above 50% of the top level
8, 9 = "Rising Edge = 100 us" i.e. the time it takes the voltage to rise from 10% to 90% of the top level
10 = "Time period between beginning of the rising edge and pulse start = 62.5 us". I.e. the time it takes the voltage to rise from 0% to 50% of the top level

Per default, the pulses within a pulse sequence that is built by repeating the same pulse, have constant on time and PRI.

You can define IPM profiles to vary the pulse parameters on a pulse-to-pulse basis. See Chapter 8.2, "How to create IPM profiles and vary pulse parameters", on page 160.

To define and apply modulation on the pulses (MOP)

The "MOP" settings define the intra-pulse modulation, i.e. the modulation applied on the pulse rise, on, and off time. The R&S Pulse Sequencer Digital provides a wide range of commonly used modulation schemes, such as AM, FM, or Chirp. You can also add your custom pulse content, Chapter 19, "Defining complex modulation schemes and IPM profiles", on page 377.

To access the "MOP" settings:

1. In the repository tree, select "Pulse > PulseName > MOP".

   The available modulation settings depend on the selected "MOP Type".

See Chapter 6.2.3, "Modulation on pulse (MOP) settings", on page 96.

2. Select for example "MOP > MOP Type > Chirp > Linear Chirp".
3. Select "MOP > Enable > On".
4. Select "Timing > Rising Edge = Falling Edge = 25%".

   Use the "Signal Preview" dialog to observe the effect of the enabled modulation, see Chapter 15, "Visualizing and analyzing signals", on page 284.

To visualize the pulse characteristics

1. Use the "Envelope Graph" to display the envelope shape.
   See Figure 6-1.
2. Set "Show Timing > On" to visualize the pulse width, rise and fall time on the graph.

7 Building pulse sequences

This section explains the sequencing principles and signal processing, like creating loops.

7.1 About the sequencing principles

Sequences are described in table form where the rows are processed sequentially and appended to the final simulated signal, see Figure 7-1.

Figure 7-1: Default segments processing: One segment after the other

Each row is repeated once or as defined with the repetition count parameter (see "Repetition" on page 124).

See also:

- "To define and enable pulse repetition" on page 132
- Chapter 7.2.2, "Pulse repetition settings", on page 125

Complex sequences can also include the following:

- **Loop**
  A loop spans one or more rows (pulses) and processes these rows in the same order several times. Nested loops are complex loops that span rows and loops.
  The loop duration is defined by the loop repetition number. The loop repetition number is a fixed value or a randomly selected value within a value range, defined with its minimum, maximum and step values.

  See also:
  - "To define and apply a loop" on page 133
  - Chapter 7.2.3, "Loop settings", on page 127
- **Filler**
  Fillers are unmodulated signals used to fill in certain period in a sequence.

  Signals can be of type:
  - Blank

  There are two ways to set the duration of a filler: by defining the time duration to be filled in (total duration) or by defining of a certain moment of time (time synchronization).
  The filler duration can be a fixed value or defined as an equation. If the filler is configured within a loop, the equation uses the loop variables.

  See also:
  - Chapter 7.2.4, "Fillers settings", on page 128

7.2 Sequence settings

Access:

► In the repository tree, select "Sequence > New".

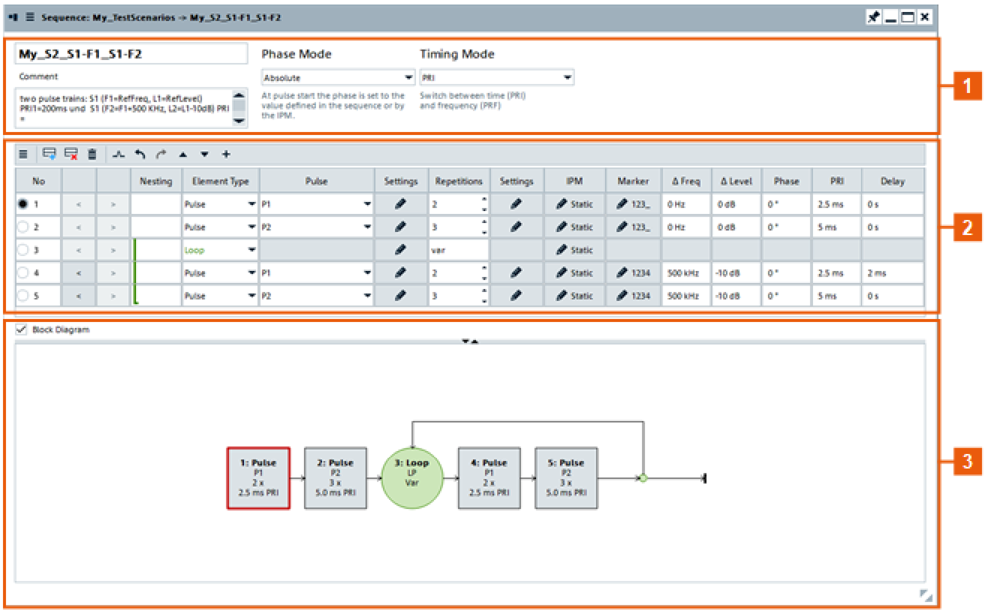


Figure 7-2: Sequence Settings

1 = Sequence name, comment, type, and phase and timing mode
2 = Sequence description table, where each row represents one item
3 = Interactive graphical representation of the current sequence, displayed if "Block Diagram > On"

For step-by-step descriptions, see Chapter 7.3, "How to create sequences and use the control elements", on page 131.

Settings:

7.2.1 Sequence description settings

Sequence Name, Comment, Type, Sequence Description Table, Block Diagram

A sequence is defined with its unique name, an optional description and with information on the sequence content, waveforms, or pulses.

A sequence is defined in table form where each row represents one item.

If enabled, an interactive block diagram displays the current sequence configuration.

- Left-click to select the item (see "Select, No." on page 123).
- Right-click to access the context menu (see "New, Insert, Append, Remove, Clear Items, Undo/Redo" on page 123).
- Double click an item works like the "Settings (...)" icon in the "Sequence Description Table" and accesses the settings of the item directly.

Remote command:
`SEQuence:CREate` on page 426
`SEQuence:CATalog?` on page 426
`SEQuence:SELect` on page 427
`SEQuence:NAME` on page 427
`SEQuence:COMMent` on page 428
`SEQuence:TYPE` on page 586
`SEQuence:REMove` on page 428

Phase Mode

Defines how the phase is set at each pulse start.

"Absolute" The pulse start phase is set to a value defined in the sequence description table (Phase) or by the IPM.

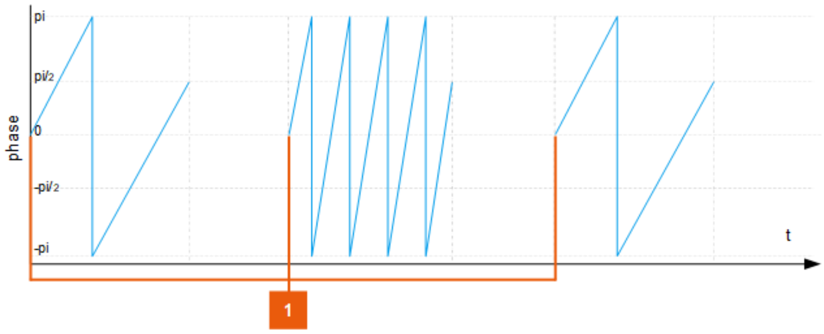


Figure 7-3: "Phase Mode = Absolute" and "Sequence Description (table) > Phase = 0 deg" (IPM not defined)

1 = Start phase of all pulses is 0 deg

"Continuous (Relative)"
The phase defined in the sequence description table (Phase) or by IPM is used as a relative phase change. The phase of the last valid sample of a pulse is held; the specified phase value is applied as a phase offset on it.

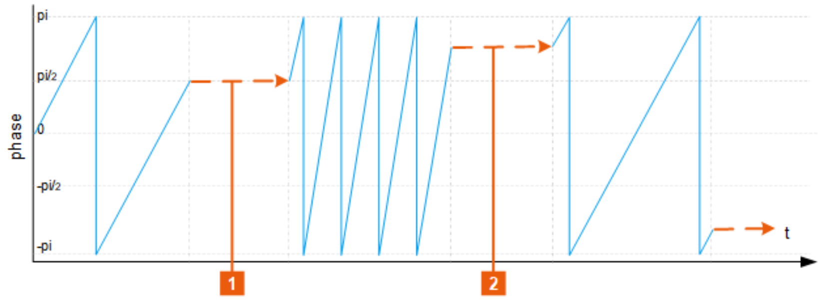


Figure 7-4: "Phase Mode = Continuous (Relative)" and "Sequence Description (table) > Phase = 0 deg" (IPM not defined)

1, 2 = With "Phase = 0 deg", the phase offset is also 0 deg. The phase of last sample of a pulse is also the start phase of the next one

"Memory" If there are frequency steps, the phase is set to a state as if the frequency has never changed.

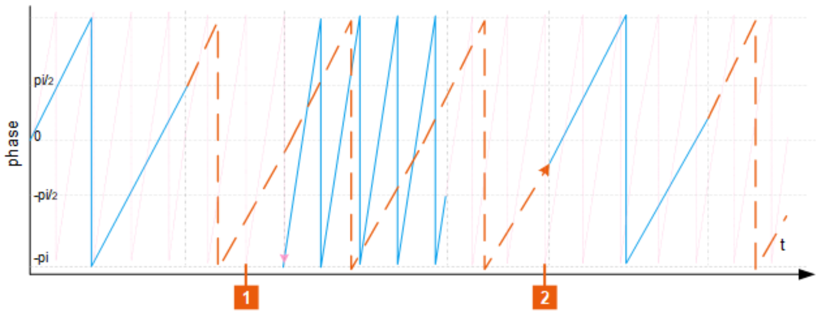


Figure 7-5: "Phase Mode = Memory" and "Sequence Description (table) > Phase = 0 deg" (IPM not defined)

1, 2 = Alternating switching between the outputs of frequency synthesizers (two signals in the example). At the beginning of each pulse, the phase is set to a state the frequency would have had if it had been continuously running

Remote command:
SEQuence:PHASe:MODE on page 586

Timing Mode
Switches between time (PRI) and frequency (PRF), see PRI/PRF.

Remote command:
SEQuence:TIME:MODE on page 586

New, Insert, Append, Remove, Clear Items, Undo/Redo
Standard functions for items handling, see "Standard function in the context menus" on page 33.

Select a row, for example, to delete it or to insert a row before it.

Remote command:
SEQuence:ITEM:ADD on page 429
SEQuence:ITEM:CLEar on page 434
SEQuence:ITEM:DELete on page 433

Select, No.
Select an item to perform any row-based actions.

Items are indicated by consecutive number; also used for indication in remote control.

Remote command:
SEQuence:ITEM:SELect on page 431
SEQuence:ITEM:COUNt? on page 430

Nesting
Indents table rows to include them in a loop or build parallel branches.

Remote command:
SEQuence:ITEM:INDent on page 588

Element Type
Defines the type of content of the current item. Available are:

"Pulse"	To select a pulse, use the Pulse.
"Filler"	Creates dummy data, like a blank segment. See Chapter 7.2.4, "Fillers settings", on page 128.
"Loop"	Creates a loop See Chapter 7.2.3, "Loop settings", on page 127.

Remote command:
`SEQuence:ITEM:TYPE` on page 592

Pulse
Lists all available pulses.

- To access the settings of an existing pulse, select "... (Settings)".
- To create a new pulse, select the icon.
  See also Chapter 6, "Creating a pulse library", on page 93.

Remote command:
`SEQuence:ITEM:PULSe` on page 590

Repetition
Sets the number of times the item is repeated. For more complex repetition patterns, select "... (more)".

See Chapter 7.2.2, "Pulse repetition settings", on page 125.

Remote command:
`SEQuence:ITEM:REP:COUNt:FIXed` on page 591
`SEQuence:ITEM:LOOP:COUNt:FIXed` on page 588

Settings
Accesses the "Rep: Line Item" dialog with further repetition settings.

IPM
Enables and defines an inter-pulse modulation.

See Chapter 8, "Defining and enabling inter-pulse modulation effects", on page 139.

Marker
Defines the active markers per pulse phase, see Chapter 20, "Defining and enabling marker signals", on page 380.

Δ Freq.
Enables a frequency offset relative to the reference frequency.

Frequency-dependent parameter values are defined relative to this value.

See also "To define and apply a loop" on page 133.

Remote command:
`SEQuence:ITEM:FREQuency:OFFSet` on page 588

Δ Level
Defines a level offset per row.

A negative level offset simulates a pulse attenuation. For example, a "Level = -3 dB" means that the pulses of the current row are processed with power level that is the half of the power.

Level-dependent parameter values are defined relative to this value.

See also "To define and apply a loop" on page 133.

Remote command:
`SEQuence:ITEM:LEVel:OFFSet` on page 588

Phase
Sets a phase offset at the beginning of the row. Phase-dependent parameter values are defined relative to this value.

For example, a "Phase = 30 deg" shifts the phase in the first pulse of the current row; enabled modulation would further change the phase. The phase offset is not compensated at the row end.

See also "Phase Mode" on page 121.

Remote command:
`SEQuence:ITEM:PHASe:OFFSet` on page 590

PRI/PRF
The column name changes depending on the selected Timing Mode.

Sets the pulse repetition interval (PRI), i.e. the pulse period or the pulse repetition frequency (PRF).

Use the "PRI" parameter to define pulses with constant total pulse cycle duration and enabled timing variations like jitter for example.

Remote command:
`SEQuence:ITEM:PRI` on page 590
`SEQuence:ITEM:PRF` on page 590

Delay
Enables a start delay.

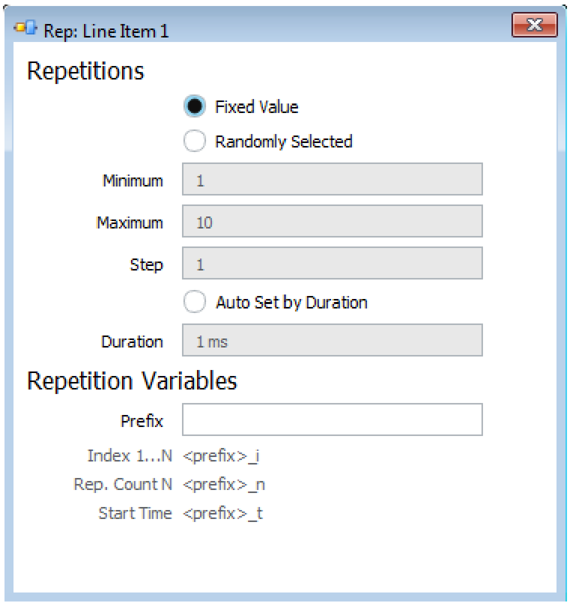
A delay within an overlay shifts the start time of the overlay elements relative to each other. Time-dependent parameter values are defined relative to this value.

See also "To define and apply a loop" on page 133.

Remote command:
`SEQuence:ITEM:PDELay` on page 590

7.2.2 Pulse repetition settings

Per default, a table row in the sequence description table, i.e. a pulse or a waveform, is processed once. To repeat an item several times, enable a repetition.



For a step-by-step instruction, see "To define and enable pulse repetition" on page 132.

Repetition Number

Defines how many times a sequence item, for example a pulse or loop, is repeated.

The repetition number can be:

- A fixed value
- A randomly selected value within a value range, defined with its minimum, maximum and step values ("Randomly Selected")
- Defined to fit in a selected duration ("Auto Set by Duration"), where the repetition number can be rounded up or down

Remote command:
`SEQuence:ITEM:REP:TYPE` on page 592
`SEQuence:ITEM:REP:COUNt:MINimum` on page 591
`SEQuence:ITEM:REP:COUNt:MAXimum` on page 591
`SEQuence:ITEM:REP:COUNt:STEP` on page 592
`SEQuence:ITEM:REP:COUNt:DURation` on page 591
`SEQuence:ITEM:REP:COUNt:ROUNding` on page 591

Repetition Variables

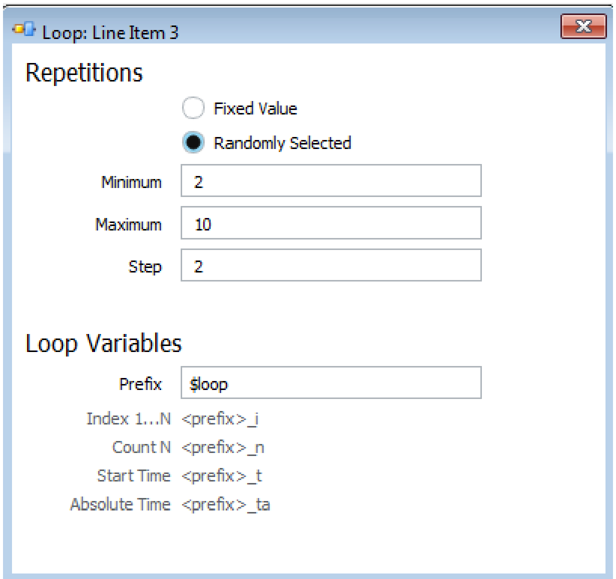
You can define repetition variables that are used by the pulse calculation within one row.

"Prefix"	Indicates the repetition the variable belongs to. If empty, no variables are used. The variable prefix is a case-sensitive entry.

Remote command:
`SEQuence:ITEM:REP:VARiable` on page 592

7.2.3 Loop settings

Per default, table rows in the sequence description table are processed sequentially. To process a subset of pulses or waveforms several times, create a loop.



For a step-by-step instruction, see "To define and apply a loop" on page 133.

Loop Repetition

The loop repetition number is a fixed value or a randomly selected value within a value range, defined with its minimum, maximum and step values.

Remote command:

SEQuence:ITEM:LOOP:TYPE on page 589
SEQuence:ITEM:LOOP:COUNt:MINimum on page 589
SEQuence:ITEM:LOOP:COUNt:MAXimum on page 589
SEQuence:ITEM:LOOP:COUNt:STEP on page 589

Loop Variables

You can define loop variables that are used by the pulse calculation within the loop. When you use a variable, always indicate the prefix and the extension.

"Prefix" Indicates the loop to that the variable belongs. If empty, no loop variables are used.
The variable prefix is a case-sensitive entry.

"Loop Index 1 .. N"
"<prefix>_i" indicates the current repetition number and is a value in the range 1 .. N, where N is the repetition number.

"Loop Count N"
"<prefix>_n" indicates the repetition number, fixed or randomly selected as defined with Loop Repetition.

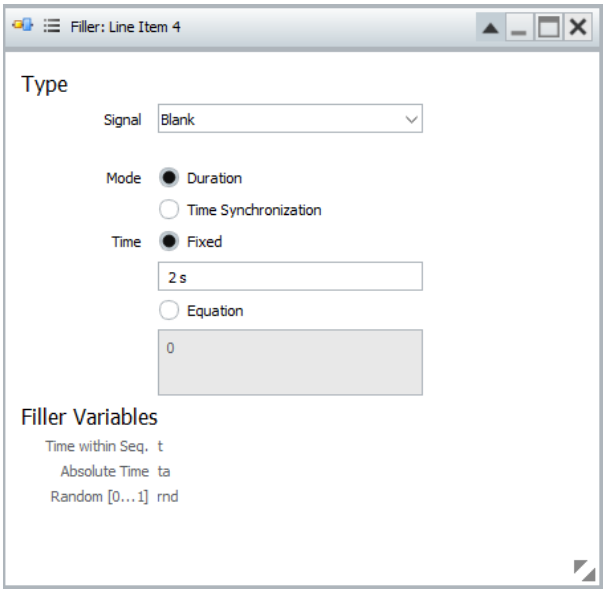
"Start Time" "<prefix>_t" indicates the start time of each subsequent loop

"Absolute Time"
"<prefix>_ta" indicates the absolute time

Remote command:
SEQuence:ITEM:LOOP:VARiable on page 589

7.2.4 Fillers settings

You can enable and add fillers.



Signal
Selects the type of signal to be generated for filling in the dummy segment.

"Blank" A blank segment with the selected duration is appended to the sequence.

Remote command:
SEQuence:ITEM:FILLer:SIGNal on page 587

Mode
Filling segments are commonly used for synchronization purposes. This parameter defines the way the filler duration is determined:

"Duration" Enables a dummy signal for a given period of time.

"Time Synchronization"
In this mode, signal data is generated up to the defined point in time. If the selected time has already elapsed, an error is generated and the calculation is stopped.
Use this mode if a signal has to start at a defined point in time and the length of the previous signals vary or is not known.

Remote command:
SEQuence:ITEM:FILLer:MODE on page 587

Time

Defines the timing of the filler.

"Fixed" Uses a fixed time

"Equation" Defines the time point as an equation.

Remote command:

`SEQuence:ITEM:FILLer:TIME` on page 587

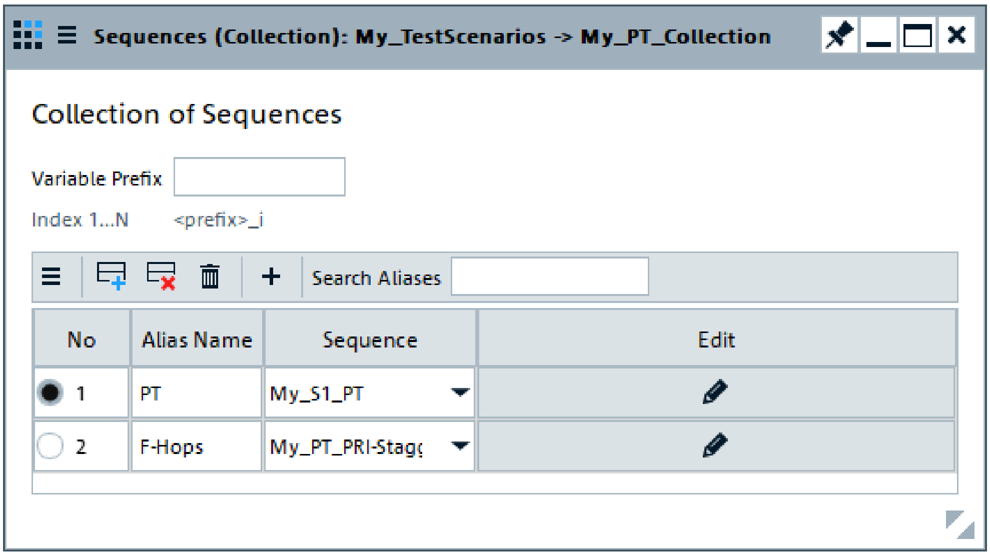
`SEQuence:ITEM:FILLer:TIME:FIXed` on page 587

`SEQuence:ITEM:FILLer:TIME:EQUation` on page 587

7.2.5 Lists with multiple sequences

Access:

1. In a "Sequences (Collection)" scenario, select "Sequences > Edit".



   Switching between multiple sequences is possible in the pulse train sequential scenario.

2. Select "Edit > ..." to access the properties of the individual sequence.
   See Chapter 7.2, "Sequence settings", on page 120.

Variables

You can define collection variables and use them, e.g., in the formula editors, in the reports, etc.

"Prefix" Indicates the sequence to that the variable belongs. If empty, no variables are used.
The variable prefix is a case-sensitive entry.

"Index 1 .. N"
"<prefix>_I" indicates the current number and is a value in the range 1 to N.
Enter the `<prefix>` only; the extension `_I` is appended automatically.

"Count N"
"<prefix>_n" is the number of variables.
Enter the `<prefix>` only; the extension `_I` is appended automatically.

Remote command:
`SCENario:CSEQuence:VARiable` on page 557

List of sequences
Displays a list of sequences.

To add a sequence, select the "Data Source" icon and select "Insert After".

To reorder sequences (items in the list), use the "Up/Down" functions.

Use the standard "Append", "Remove Last", "Delete All" or "Copy and append" functions to add or remove a sequence. The provided settings are self-explanatory.

Remote command:
`SCENario:CSEQuence:ADD` on page 429
`SCENario:CSEQuence:SELect` on page 431
`SCENario:CSEQuence:CURRent` on page 548
`SCENario:CSEQuence:CLEar` on page 434
`SCENario:CSEQuence:DELete` on page 433

Alias Name ← List of sequences
Enters a name.

Remote command:
`SCENario:CSEQuence:ALIas` on page 548

Sequence ← List of sequences
Select a sequence from the list of available sequences.

Remote command:
`SCENario:CSEQuence` on page 556

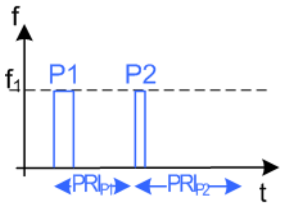
Edit ← List of sequences
Opens the "Sequence" dialog for the selected sequence, see Chapter 7.2, "Sequence settings", on page 120.

7.3 How to create sequences and use the control elements

See:
- "To create a simple sequence" on page 131
- "To define and enable pulse repetition" on page 132
- "To define and apply a loop" on page 133
- "To create a scenario with several pulse train sequences" on page 135

To create a simple sequence

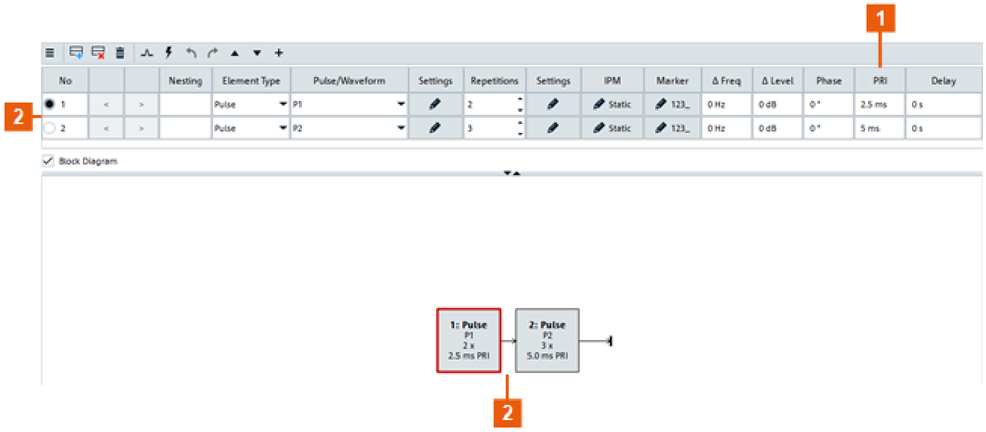
This example explains how to create a simple sequence with two pulses, P1 and P2.



We assume that the required pulses have been created, see Chapter 6.3, "How to create a new pulse and adjust its settings", on page 116.

1. In the repository tree, select "Sequence > New".
2. In the "Sequence" dialog, enter a "Name" and a "Comment", e.g. `My_S1_PT`.
3. For each sequence item, perform the following steps:
   a) Select the "Append New Item" icon.
   b) In the "Sequence Description" table, select "Type > Pulse". Select an existing pulse, e.g. "Pulse > P1".
   c) If the list is empty, select the pulse icon to create a pulse first.
   d) See Chapter 6.3, "How to create a new pulse and adjust its settings", on page 116
   e) Set the PRI_{P1}, e.g. "PRI = 2.5 ms".

4. Enable "Show Block Diagram".



1 = Pulse with different PRIs, i.e items with different duration
2 = Sequence with two items

The items are processed once and one after the other (see also Figure 7-1). You can change the default processing, see:

- "To define and enable pulse repetition" on page 132
- "To define and apply a loop" on page 133

To define and enable pulse repetition

This example explains how to repeat pulses several times, for example to configure a pulse sequence as illustrated on Figure 7-6.

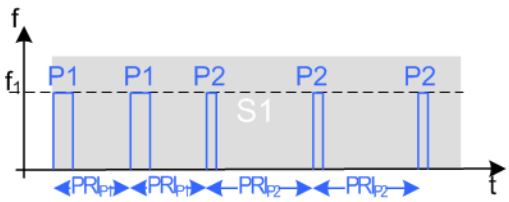
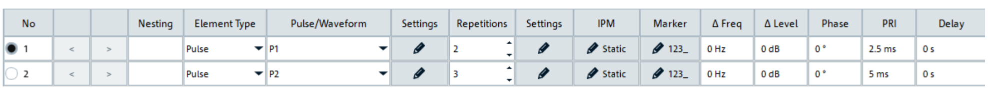


Figure 7-6: Example of pulse sequence composed of 2xP1 and 3xP2, where the pulses use different PRIs

PRI_{P1} = 2.5 ms
PRI_{P2} = 5 ms

We assume that a sequence with two items has been created, see "To create a simple sequence" on page 131.

1. Open this sequence.
2. In the "Sequence Description" table, for the first pulse, select "Rep. Count = 2"
3. For the second item, set "Rep. Count = 3".



The pulse sequence S1 consists of 5 pulses and has a PRI_{S1} = 20 ms.

Use the"Signal Preview" display to visualize the result, see Chapter 15.1, "Signal preview settings", on page 284.

To define and apply a loop

This example explains how to create a loop with pulses and to repeat it several times, for example to configure a pulse sequence as illustrated on Figure 7-7.

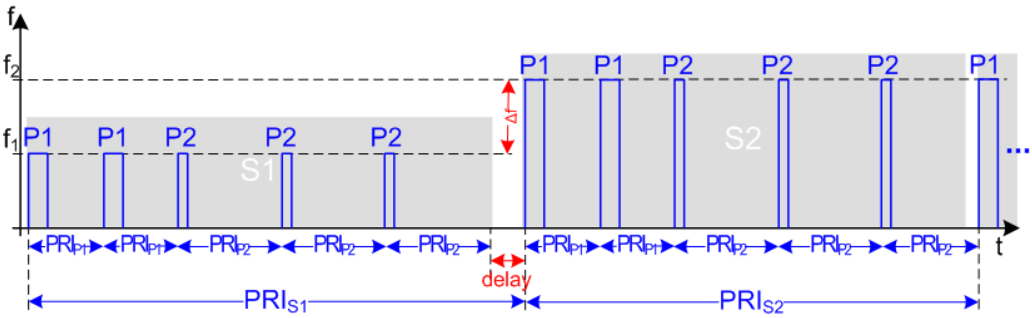
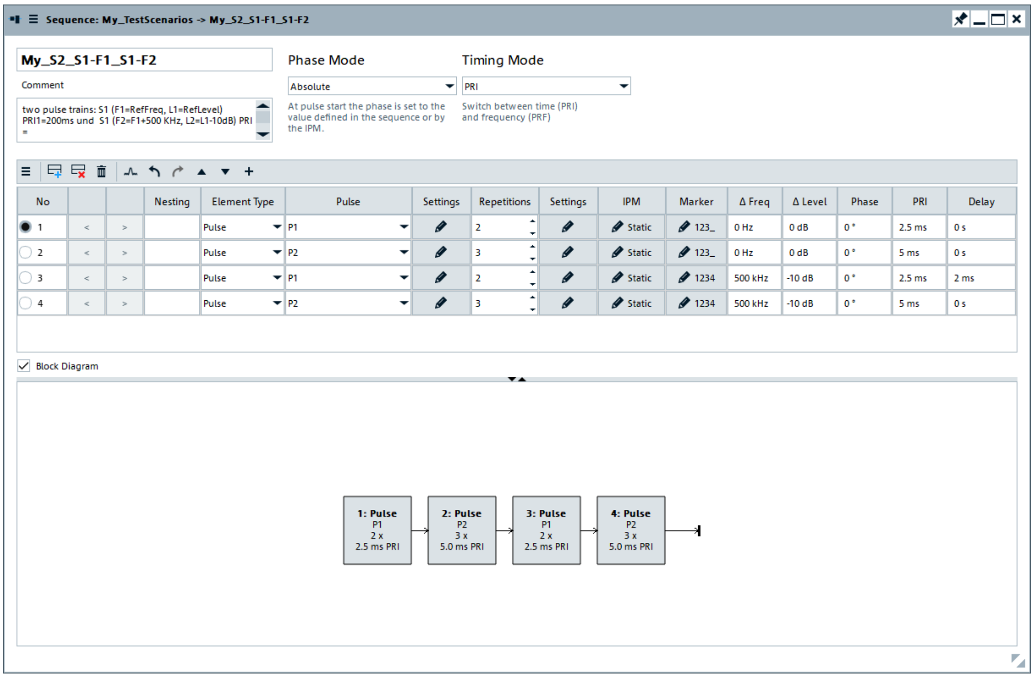


Figure 7-7: Example of pulse sequence composed of two subsequences, S1 and S2

S1, S2 = Both sequences follow the same pulse pattern (2xP1 and 3xP2)
PRI_{S1} = PRI_{S2} + "Delay"
f_2 = f_1 + "ΔFreq"

We assume that a sequence with four items has been created, as shown on the following figure.

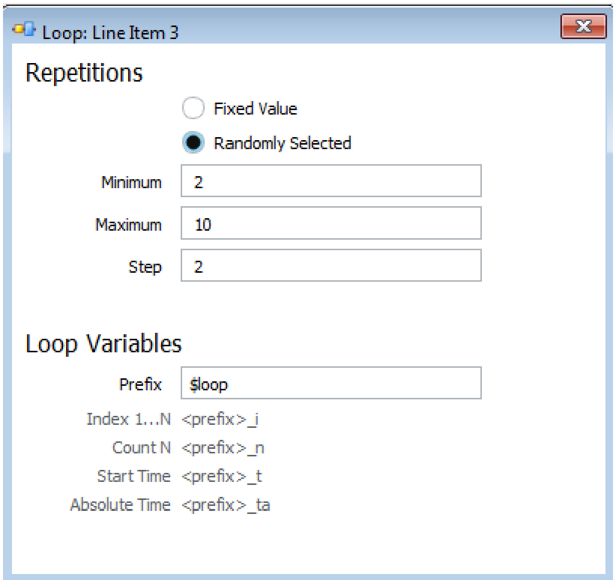


1. In the "Sequence Description" table:

a) Select the third item. Select "Insert Item Before Selection"

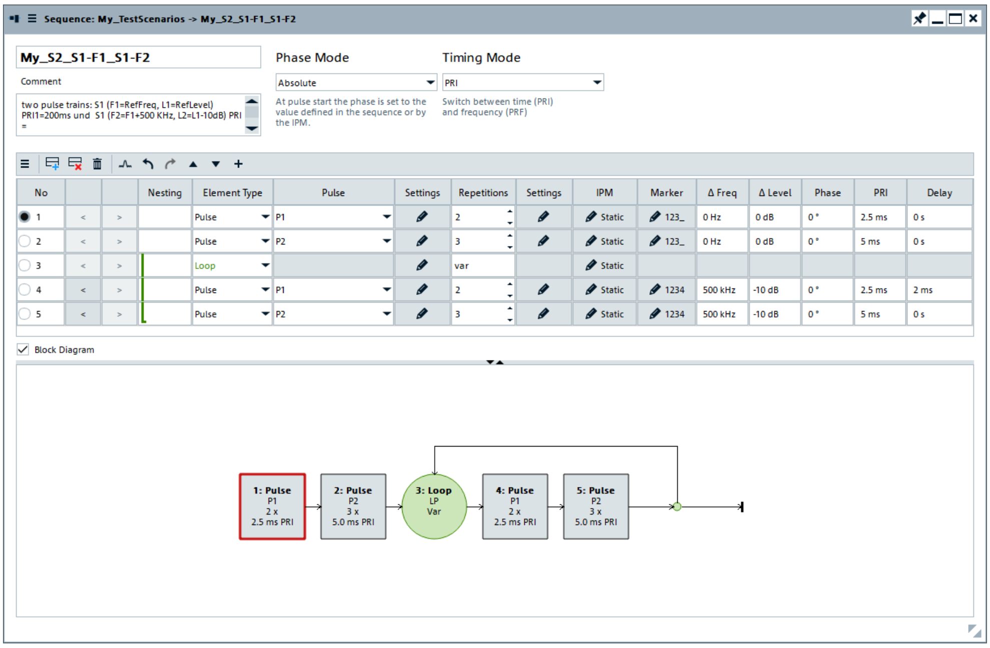
b) Select "Element Type > Loop"

c) Select the "... (more)" icon and enable a random loop repetition.



The loop is repeated 2, 4, 6, 8 or 10 times.

d) To include the last two items (Line items 3 and 5) in the loop, select the "Increase indentation level >" icon.



The sequence graph in the "Sequence Description" confirms that the two items are included in the loop. The loop is repeated a random number of times.

2. To enable a different PRI for the looped items, select the first pulse in the loop and set "Delay = 2 ms".

$PRI_{S1} = PRI_{S2}$ + 2 ms = 22 ms.

3. For the pulses in the loop, set the "Δ Freq = 500 KHz" and "Δ Level = -10 dB".
4. Assign the sequence `My_S2_S1-F1_S1-F2` to a pulse train scenario.
   Start the calculation. Select "Results > View".

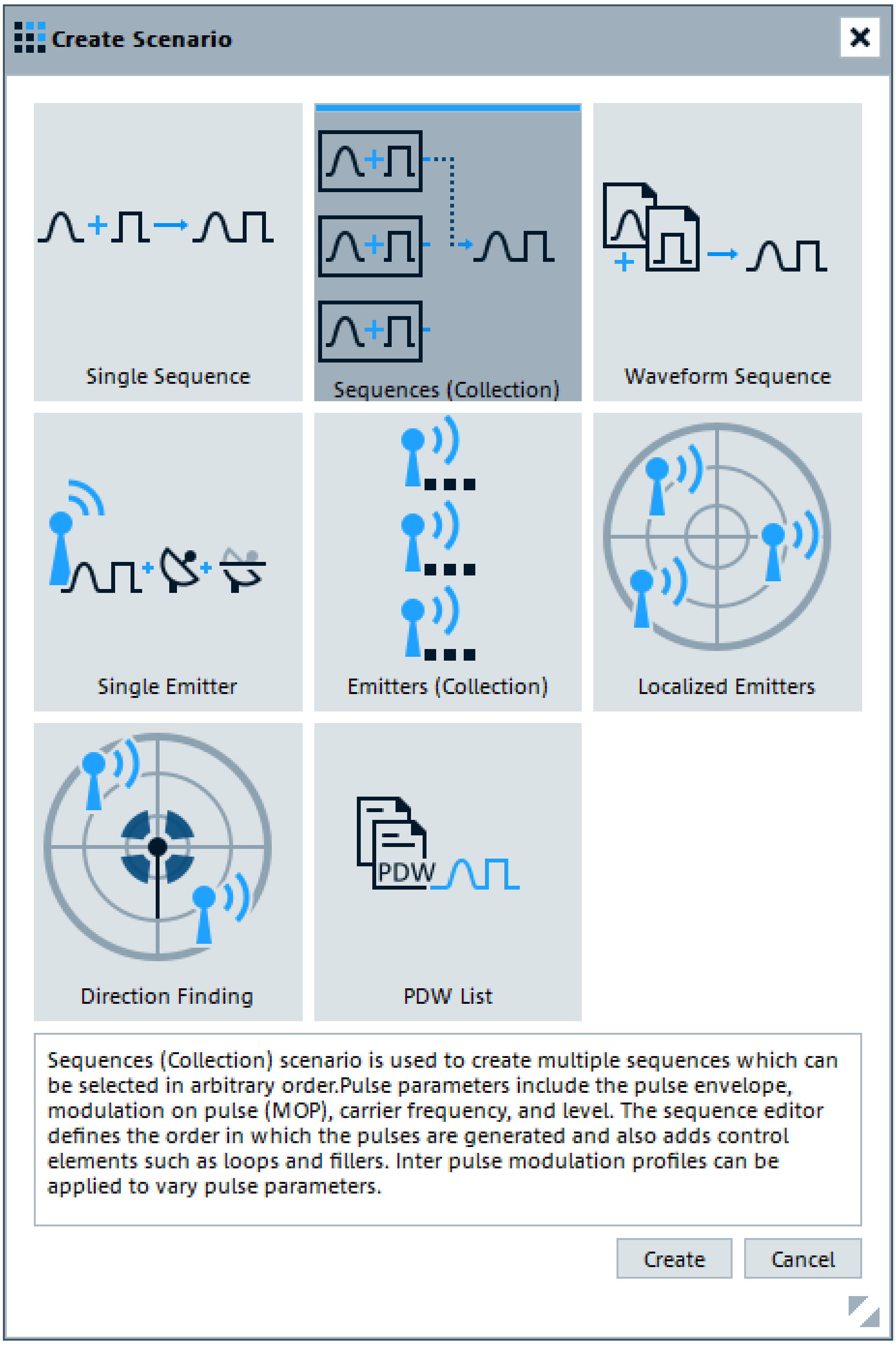
   The "Signal Preview" dialog displays the results.
   Select "Graphics > PRI" to observe the PRI for the pulses in the sequence. Select different parameters to visualize them on the graph.

To create a scenario with several pulse train sequences

If your test situation requires you to test the receiver's ability to detect different pulse sequences, you can use "Sequences (Collection)" scenario. In this dedicated scenario, you can configure several pulse train sequences and switch between them sequentially.

The following example creates a complex scenario with two pulse train sequences to explain the principle.

1. Select "Repository Tree > Scenario > New".
2. In the "Scenario Wizard", select "Scenario Type > Sequences (Collection)".



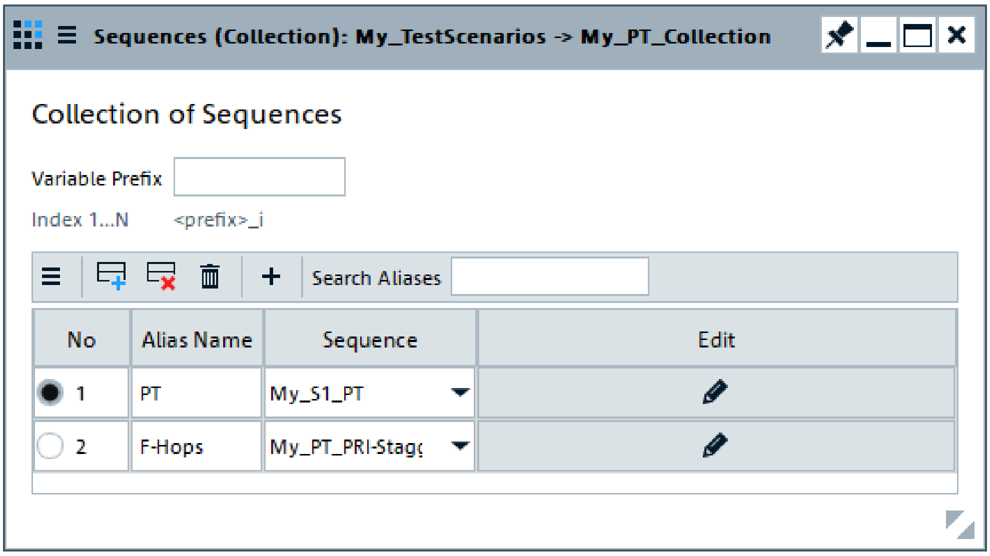
3. Click "Create".
4. Enter a scenario name and a description.
5. In the block diagram, select "Sequences > Edit".

   The "Sequences (Collection)" dialog opens.
6. Use the standard "New" function to insert the first sequence in the list.

   You can select from the list of available sequences or create a new one.

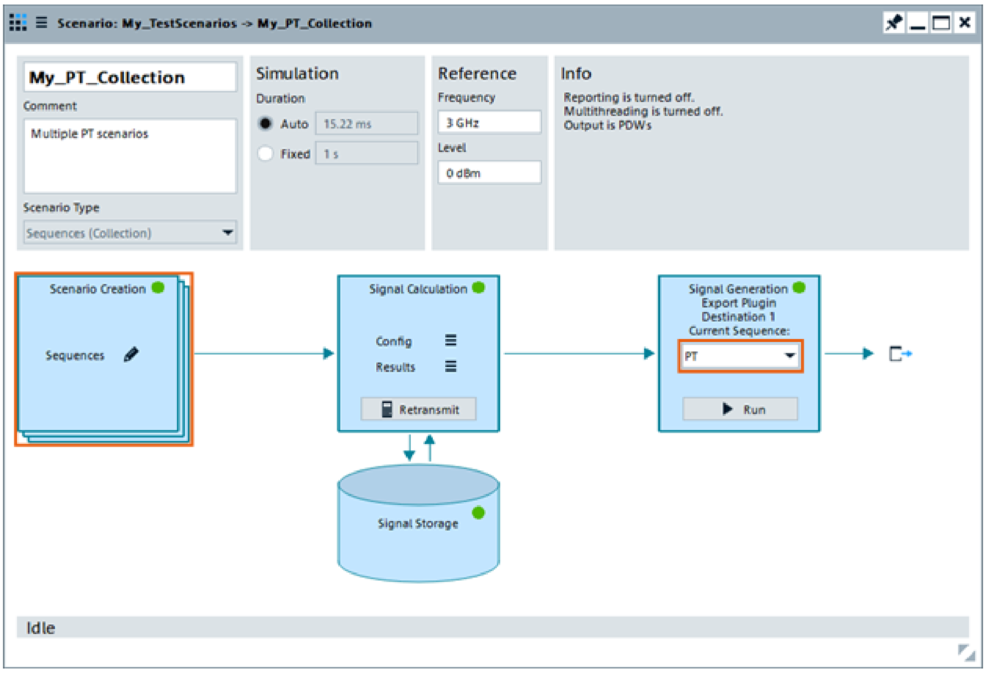
See also "To create a simple sequence" on page 131.

7. In the list of sequences, select a "Sequence". Use the standard "New", "Append/ Remove" and "Up/Down" functions to add items to the list or reorder them.



For description of the provided settings, see Chapter 7.2.5, "Lists with multiple sequences", on page 129.

8. In the "Signal Generation" block, select the name of the sequence, e.g. "PT".



For description of the provided scenario settings, see Chapter 5.2, "Scenario settings", on page 74.

To create a pulse train sequence and scenario automatically with the "Startup Assistant"

Alternatively to the methods described in this section, you can use the "Startup Assistant" to create a pulse train automatically.

1. In the menu bar, select "Help > Wizard".
2. Select "Create a pulse train".
   Click "Next".
3. Follow the instructions, use the default settings and select "Finish".

   For description of the provided settings, see:
   - Chapter 5.2, "Scenario settings", on page 74
   - Chapter 6.2, "Pulse settings", on page 93
   - Chapter 8.1, "IPM profiles settings", on page 142
   - Chapter 20.1, "Marker settings", on page 380

8 Defining and enabling inter-pulse modulation effects

Per default, pulses calculated from the settings of one table row in the sequence description table are identical. To simulate a change of parameters from one pulse to another, like PRI stagger for example, you can define inter-pulse modulation (IPM) effects and assign them to the sequence elements.

Throughout this description, the IPM effects are commonly named parameter variations.

Pulse repetition patterns

The pulse repetition patterns are commonly used to:

- Avoid range ambiguities caused by echoes of far away objects, which fall into the detection period of the next pulse
- Protect the radar signal against jamming
- Deal with clutter (ground, sea, weather)
- Make the radar more difficult to be identified by frequency hopping

IPM effects

In the following, two common IPM techniques, the jittered PRI and staggered PRI, are used to explain the principle:

- *Jittered PRI*
  In general, a jitter is understood as the short-term variations of a significant pulse parameter, deviating from their ideal positions in time in either random or ordered way (see Figure 8-1).
  You can, for example, use the random variation of the rising and falling pulse edge positions to simulate a technically imperfect trigger signal.

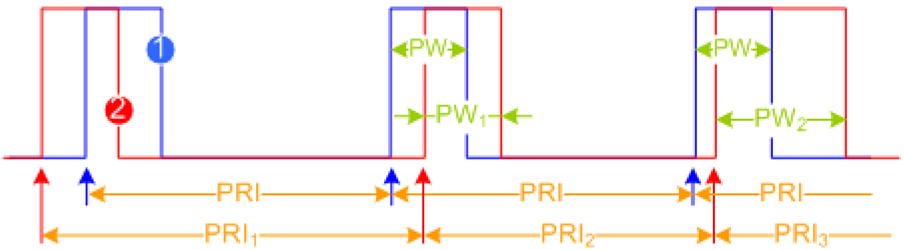
  

  Figure 8-1: Example of Jitter effects

  1 = Ideal pulse with constant pulse repetition interval (PRI) and pulse width (PW)
  2 = Pulse affected by Jitters: Jitter PRI ($PRI_1 \neq PRI_2$) and varying pulse width ($PW_1 \neq PW_2$)

  As illustrated on Figure 8-1, the time between two subsequent pulses varies randomly, on a pulse-to-pulse basis, as does the pulse width.
- *Staggered PRI*
  Staggered pulse trains are commonly used against range ambiguities and blind speeds, see also "Pulse repetition patterns" on page 139.

Staggered PRI is a transmission method based on pulse trains with constant pulse width (PW) and varying PRI. The principle of staggered PRI is illustrated on Figure 8-2

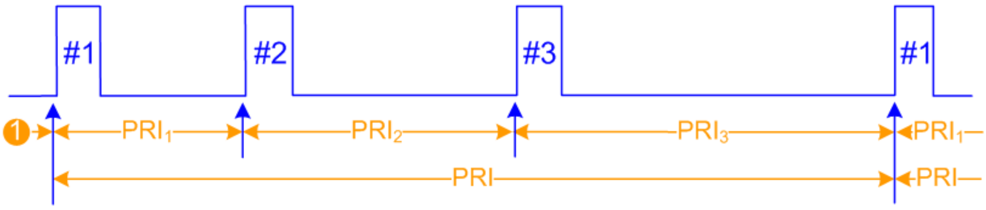


Figure 8-2: Example of an IPM effect: Staggered PRI

1 = Stagger start delay
#1,#2,#3 = Pulse train group composed of three pulses
PRI = Group PRI: Pulse repetition interval of the pulse train group
PRI_1,PRI_2,PRI_3 = Pulses transmitted with different PRI

As illustrated in Figure 8-2, within a pulse train group each successive pulse uses a different PRI; the pulse train group is slewing.

- *Frequency Hopping*
  Frequency hopping and frequency agility is a further anti-jamming method. If frequency hopping is used, the radar signal switches rapidly and periodically between different carrier frequencies.

There are no predefined parameter variations in the software but IPM profiles can be assigned to the parameter listed in Table 8-1.

Table 8-1: Parameters to which IPM profiles can be assigned

Parameter group	Pulse parameter
Level	Offset [dB] Δ Level
MOP	AM Modulation Depth [%]
	AM Frequency [Hz]
	FM Deviation [Hz]
	FM Frequency [Hz]
	Symbol Rate [Hz]
	FSK Deviation [Hz]
	Chirp Deviation [Hz]
Timing	Rise Time [s] Pulse > Time > Rising Edge
	Pulse Width [s] Pulse > Time > Width
	Fall Time [s] Pulse > Time > Falling Edge
	Delay [s]

Parameter group	Pulse parameter
	PRI [s]/PRF [Hz]
Phase	Offset [°]
Δ Freq.	Offset [Hz]

Source profiles

Parameter variations are described as the combination of a source (profile or variable) and the target element (pulse, sequence or waveform), to that the variation is assigned. If you assign more than one variation to an element, these variations are superimposed (see Chapter 8.1.4, "Inter-pulse modulation (IPM) settings", on page 156).

Source profiles are jitter and IPM profiles. You can use the same profiles for more than one variation; the R&S Pulse Sequencer Digital calculates the profile values independent from each other.

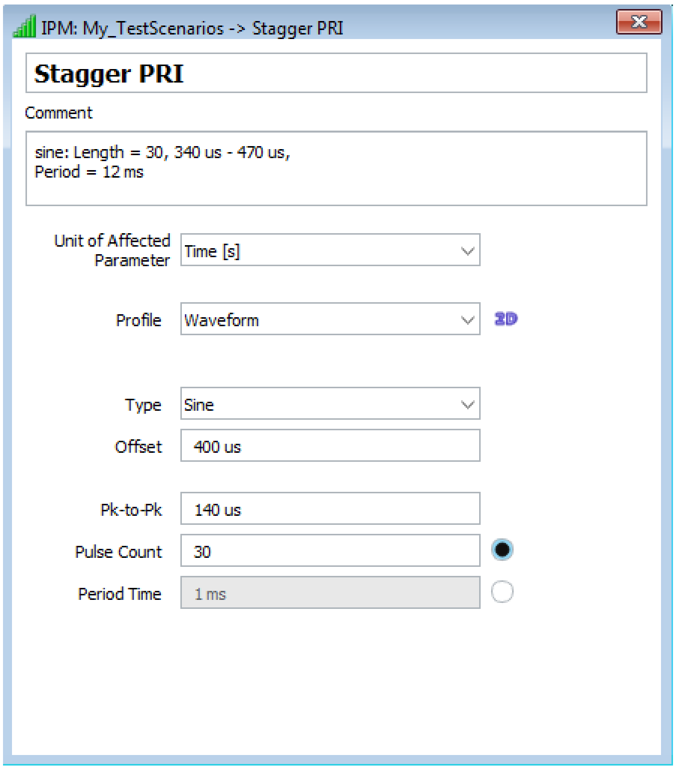
Each jitter and IPM effect affects one of the parameters listed in Table 8-1 and can use the following profiles:

- Steps
- Waveform
- Random List
- Random Steps
- List
- Interpolated Shape
- Random
- Equation
- Plug-in
- Binomial

8.1 IPM profiles settings

Access:

► Select "Repository Tree > IPM > New".



Settings:

8.1.1 Common IPM settings

IPM Profile Name

Enters the name of the profile.

Remote command:
`IPM:CREate` on page 426
`IPM:NAME` on page 427
`IPM:CATalog?` on page 426
`IPM:SELect` on page 427
`IPM:REMove` on page 428

IPM Profile Comment
Adds a description.

Remote command:
`IPM:COMMent` on page 428

Unit of Affected Parameter
Sets the units of the IPM parameters.

Available are: "Time [s]", "Frequency [Hz]","Level [dB]", "Phase [deg]", "Percent [%]".

Remote command:
`IPM:UNIT` on page 478

Profile
Sets the shape of the profile:
- Steps
- Waveform
- Random List
- List
- Interpolated Shape
- Random
- Equation

Further available parameters depend on the selected shape.

Remote command:
`IPM:TYPE` on page 478

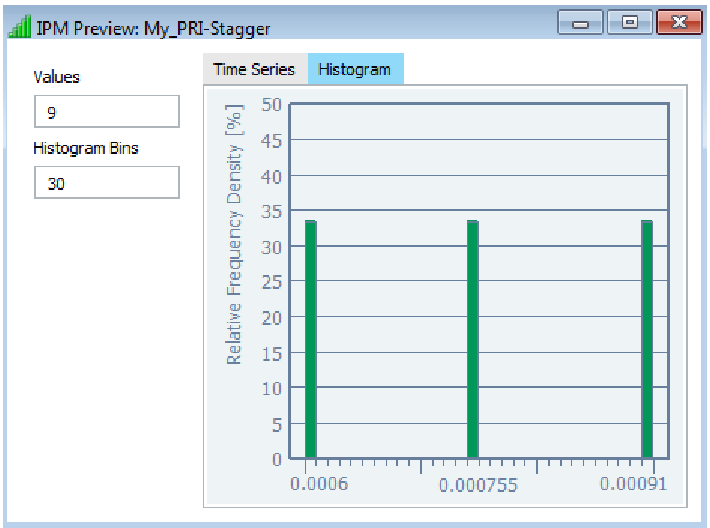
2D
Opens a preview diagram of the selected profile.

Two views are available:
- "Time Series"
  A visualization of the profile variation over time
- "Histogram"
  A statistical representation of the relative frequency density, calculated for the selected number of "Histogram Bins".
  The histogram plots the number of times data points appear in a particular bin.

The displayed characteristics on both views are defined by:

"Values"	Sets the number of values to be displayed.
"Duration"	Sets the displayed duration.

"Histogram Bins"



Sets the number of histogram bins, i.e. the histogram sets used to group the obtained data.
Bins of equal size are used. The bin size is calculated from the value range used on the y-axis in the time series display and the number of bins.

Remote command:
`DIALog:IPMPlot:VIEW` on page 489
`DIALog:IPMPlot:SAMPles` on page 489

8.1.2 IPM profiles settings

The following IPM shapes can be defined:

Steps

This IPM profile follows a staircase shape and creates a sequence of identical values before it moves on the next one.

The profile shape is defined as a function of "Number of Steps", "Start" and "Increment" values; the stop value is calculated automatically.

The parameters "Burst Length" and "Burst Period" are exclusive. They define how often or how long an increment is repeated, i.e. how many identical values are created.

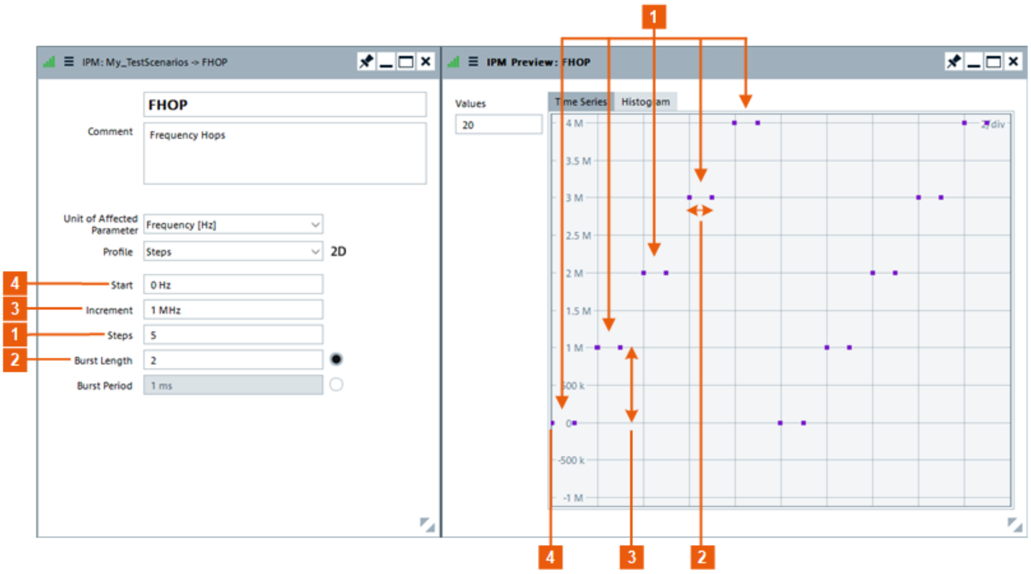


Figure 8-3: IPM with shape "Profile = Steps"

1 = "Number of Steps" = 5
2 = "Burst Length" = 2
3 = "Increment" = 1 MHz
4 = "Start" = 0 Hz

Remote command:
`IPM:STEP:STEPs` on page 486
`IPM:STEP:STARt` on page 485
`IPM:STEP:INCRement` on page 485
`IPM:STEP:BASE` on page 484
`IPM:STEP:BURSt` on page 485
`IPM:STEP:PERiod` on page 485

List
The IPM shape is a sequence of discrete values defined in a list.

The list items are processed sequentially and cyclically. When the list end is reached, the list is processed from the beginning. Each list item is repeated once or several times, or for the defined time duration.

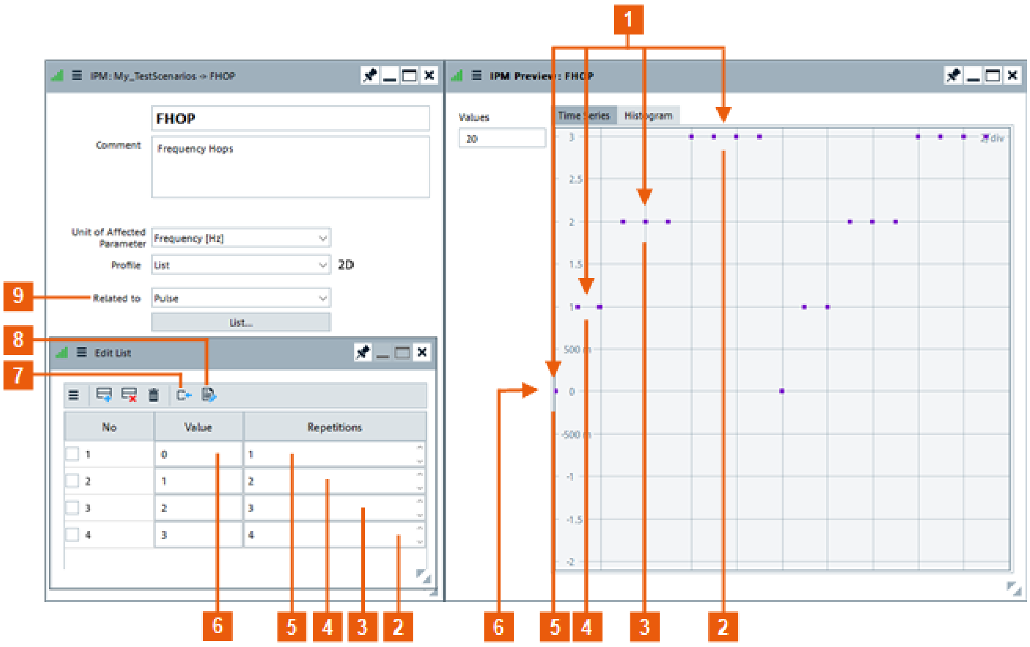


Figure 8-4: IPM with shape "Profile = List"

1 = "List Count" = 4
2, 3, 4, 5 = The increments are repeated different number of times: "Repetitions#1 = 1", "Repetitions#2 = 2", "Repetitions#3 = 3", etc.
6 = "List Start" = 0
7 = "Import Data from File" icon
8 = "Populate list" icon
9 = "Related to = Pulse"

List profiles are defined in the standard list editor in table form, see "Edit List" on page 155.

Use one of the following options to create an IPM list profile:

- To create a list manually, use the standard icons and functions in the context menu (see "Standard function in the context menus" on page 33).
  Enter the list values manually.
- To import a list in ASCII format, select the "Import Data from File" icon.
- To create a list automatically and fill it in with values, select the "Populate list" icon.
  Enter the "Start" and "Increment" values and the number of list elements ("Count") and select "Populate".

"Firing Order" If "Related to = Pulses", defines the sequencing of the PRIs in time. Without the firing order, PRIs defined in lists are processed in their sequential order, where a list element is repeated the number of times defined as "Repetition" (see Figure 8-4).
By using firing order, PRIs defined once can be reused several times and processed in a different order than their sequence order in the list. The "Repetition" field is ignored and the actual number of times a PRI is repeated each time is defined in the firing order definition field.

Supported are the following expressions:
- M – N
- MxL

Where:
- M and N are list elements numbers
- L is number of repetitions.

Example:

With "Firing Order > On", the expression *1x1,2x2,3x3,4x4* results in the sequence 1,2,2,3,3,3,4,4,4,4, that is the same as the example on Figure 8-4.

You can observe the "Time Series" on the 2D view.

Remote command:
IPM:LIST:FIRing:ENABle on page 480
IPM:LIST:FIRing:SEQuence on page 480

Remote command:
IPM:LIST:ITEM:ADD on page 429
IPM:LIST:ITEM:COUNt? on page 430
IPM:LIST:ITEM:SELect on page 431
IPM:LIST:ITEM:VALue on page 479
IPM:LIST:BASE on page 479
IPM:LIST:ITEM:REPetition on page 479
IPM:LIST:ITEM:TIME on page 479
IPM:LIST:ITEM:DELete on page 432

Waveform

Sets an IPM shape that follows a ramp, sine, or triangular waveform function. The waveform amplitude is defined by the "Peak-to-Peak" value. A constant "Offset" can be applied to the entire waveform. The waveform period is set as "Period Time" or as number of pulses ("Pulse Count") it lasts.

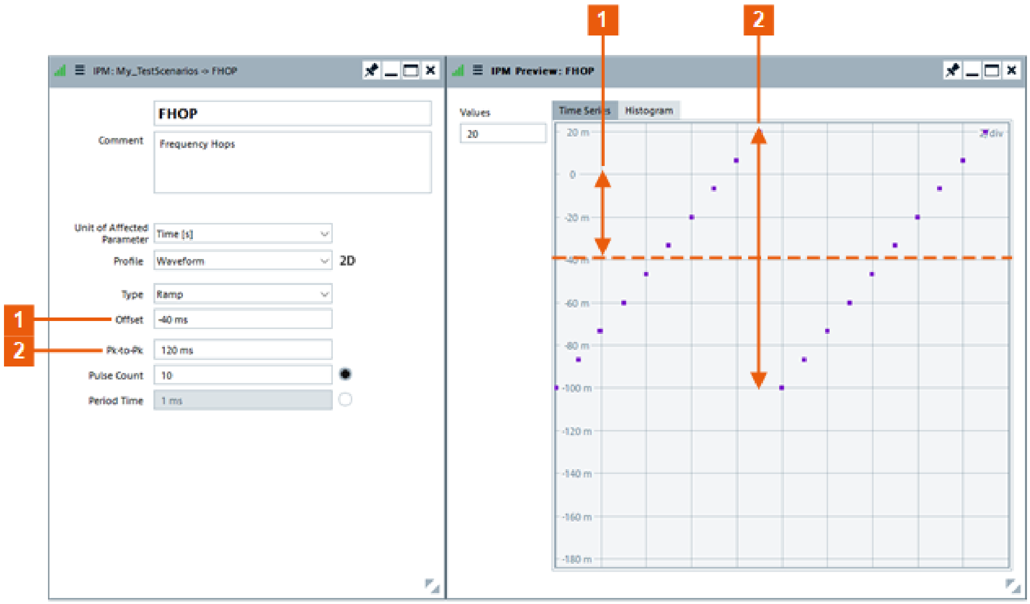


Figure 8-5: IPM with shape "Profile = Waveform" (Type = Ramp)

1 = "Offset" = -40 ms
2 = "Peak-to-Peak" = 120 ms

"Linear Ramp" The linear ramp is a profile with a sawtooth shape.
The generated values monotonically increase from ("Offset"-"Pk-to-Pk"/2) to ("Offset"+"Pk-to-Pk"/2).

"Sine" The sine profile creates values that follow one period of a sine wave.
Enable a "Phase Offset" to change the start phase of the sine wave.

"Triangular" The triangular profile changes a parameter from a minimum to a maximum value and back to the minimum value, following a triangular shape.

Remote command:
`IPM:WAVeform:BASE` on page 487
`IPM:WAVeform:TYPE` on page 487
`IPM:WAVeform:PKPK` on page 488
`IPM:WAVeform:OFFSet` on page 487
`IPM:WAVeform:PHASe` on page 487
`IPM:WAVeform:COUNt` on page 487
`IPM:WAVeform:PERiod` on page 488

Interpolated Shape

The IPM shape is a sequence of values defined in a list. The list items are mapped to several repetitions ("Pulse Count") or equally distributed over a selected period of time ("Period Time").

The transition between the increments can also be linearly interpolated.

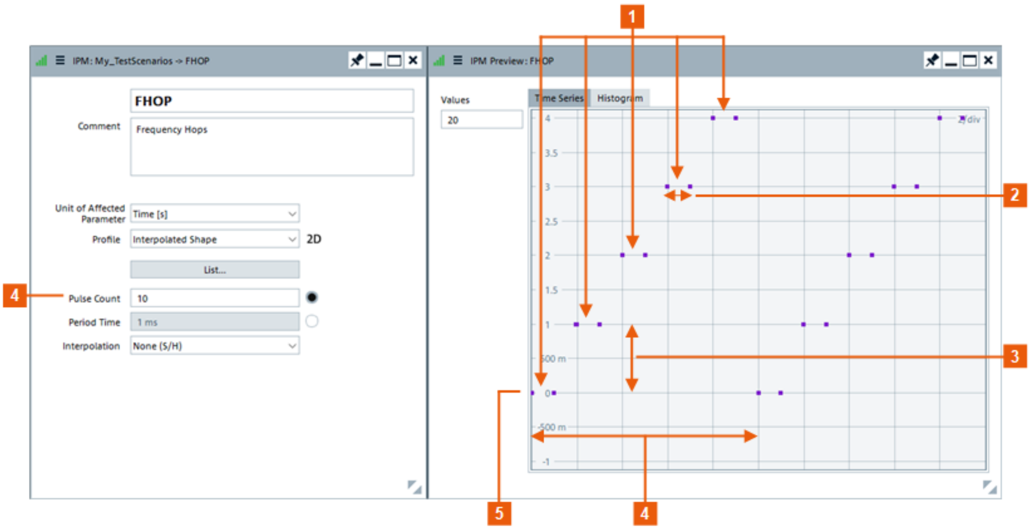


Figure 8-6: IPM with shape "Profile = Interpolated Shape" ("Interpolation = None (S/H)")

1 = "List Count" = 5
2 = Each increment is repeated two times = "Pulse Count"/"List Count"
3 = "List Increment" = 1
4 = "Pulse Count" = 10
5 = "List Start" = 0

"Period Time" Sets the period of time over that the list items are equally distributed.

"Pulse Count" Sets the number of pulses for that the data from the list is used.

Remote command:
`IPM:SHAPe:BASE` on page 484
`IPM:SHAPe:PERiod` on page 484
`IPM:SHAPe:COUNt` on page 484
`IPM:SHAPe:INTerpol` on page 484

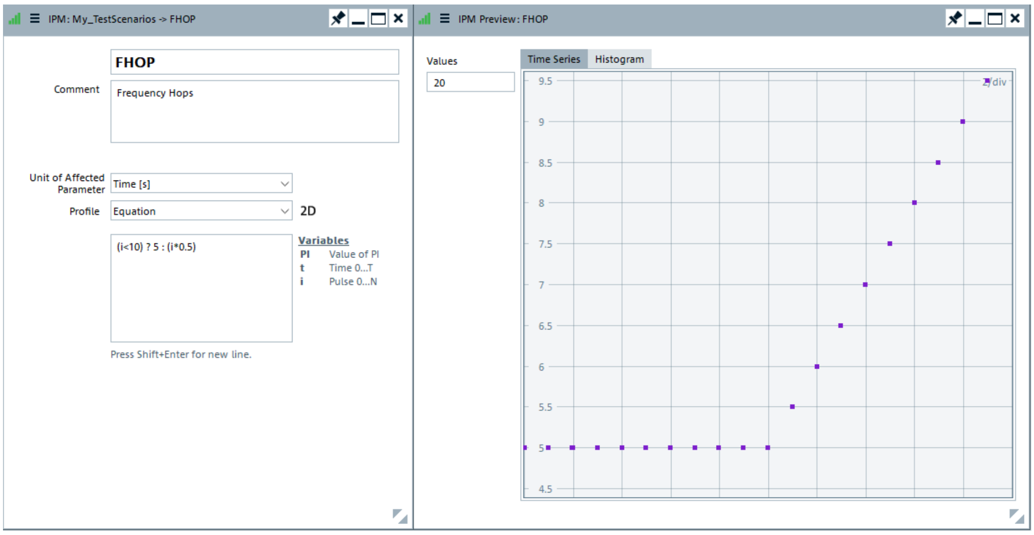
Equation
Defines the IPM shape as a function of:

`t = 0 .. T` is the absolute time.

`i = 0 .. N-1` is a pulse counter, where N is the maximum number of pulses.

Example:

(i<10) ? 5 : (i*0.5)



See Chapter B, "Formula syntax", on page 647.

Remote command:
`IPM:EQUation` on page 478

Random List

The IPM shape is a sequence of discrete values that are randomly selected from a list with user defined values.

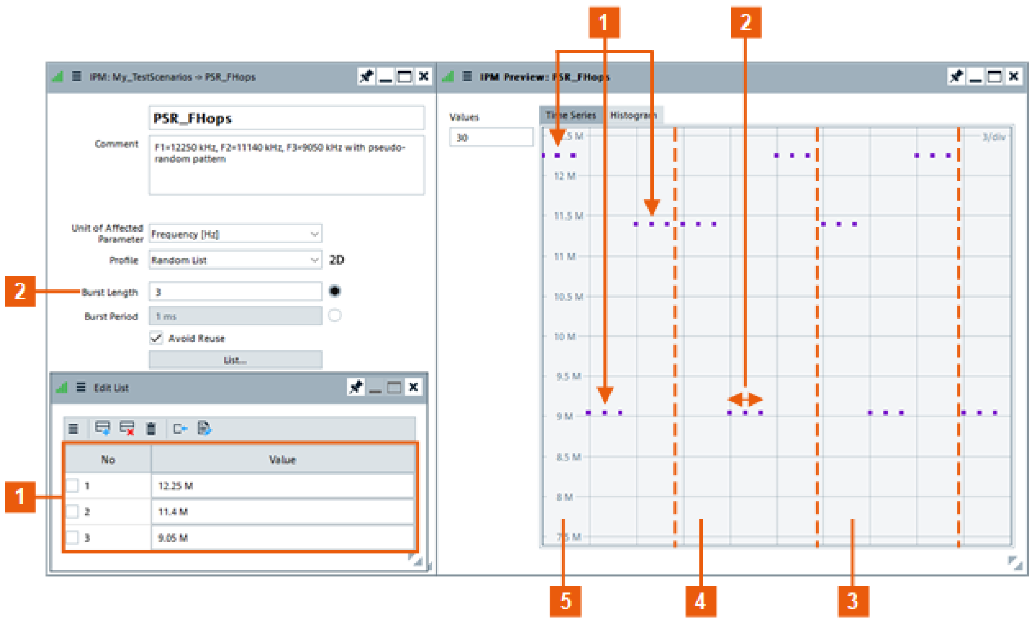


Figure 8-7: IPM with shape "Profile = Random List"

1 = "List Items" = 3
2 = "Burst Length" = 3
3, 4, 5 = Subsequent patterns with values that are selected on a pseudo-random manner from the defined 3 list items

"Burst Length/ Burst Period" The parameters "Burst Length" and "Burst Period" are exclusive. They define how often or how long an increment is repeated, i.e. how many identical values are created.

"Avoid Reuse" Ensures that each value is used only once per burst.

"List" Accesses the standard list editor to define a list in table form, see "Edit List" on page 155

- To create a list manually, use the standard icons and functions in the context menu (see "Standard function in the context menus" on page 33). Enter the list values manually.
- To import a list in ASCII format, select "Import Data from file" icon, navigate to a suitable file, select it and import it.
- To create a list automatically and fill it in with values, select the "Populate List" icon.
  Enter the "Start" and "Increment" values and the number of list elements ("Count") and select "Populate".

Remote command:
`IPM:RLISt:BASE` on page 483
`IPM:RLISt:BURSt` on page 483
`IPM:RLISt:PERiod` on page 483
`IPM:RLISt:REUSe` on page 483

Random Steps

The IPM shape is a sequence of desecrate values and step size that are randomly selected from a user defined value ranges.

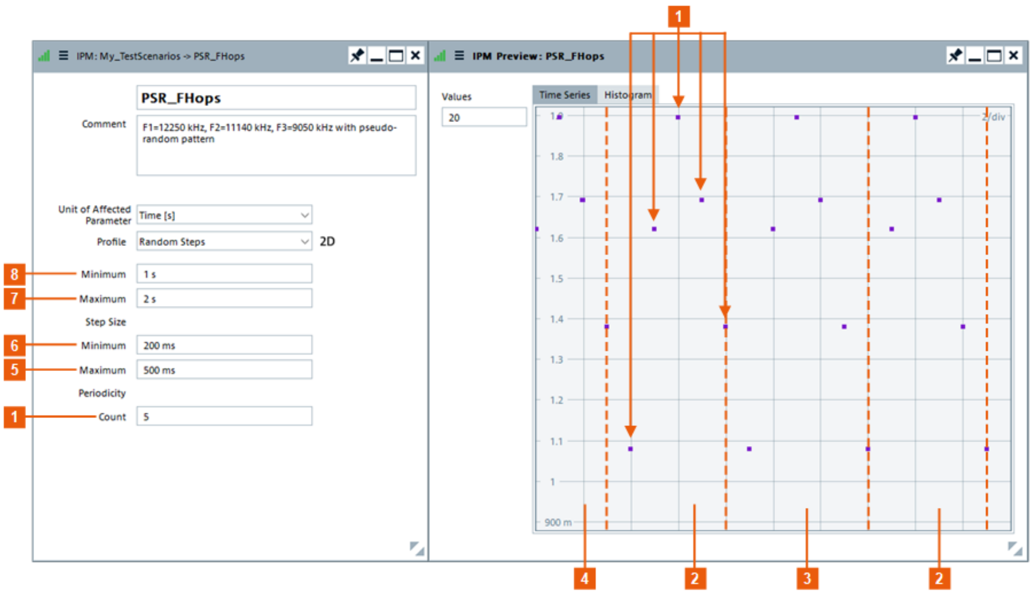


Figure 8-8: IPM with shape "Profile = Random Steps"

1 = "Periodicity > Count = 5", i.e. selected are five random values
2 to 4 = Subsequent patterns
5, 6 = User defined step size
7, 8 = User defined value range

"Periodicity > Count"
Sets the pattern length, i.e. the number of values that are selected on a pseudo-random manner.

Remote command:
`IPM:RSTep:MINimum` on page 486
`IPM:RSTep:MAXimum` on page 486
`IPM:RSTep:STEP:MINimum` on page 486
`IPM:RSTep:STEP:MAXimum` on page 486
`IPM:RSTep:PERiod` on page 486

Random

Random jitter is unlimited in its value range and is caused by thermal noise and similar effects. The IPM shape is a sequence of random values according to the select "Distribution" function.

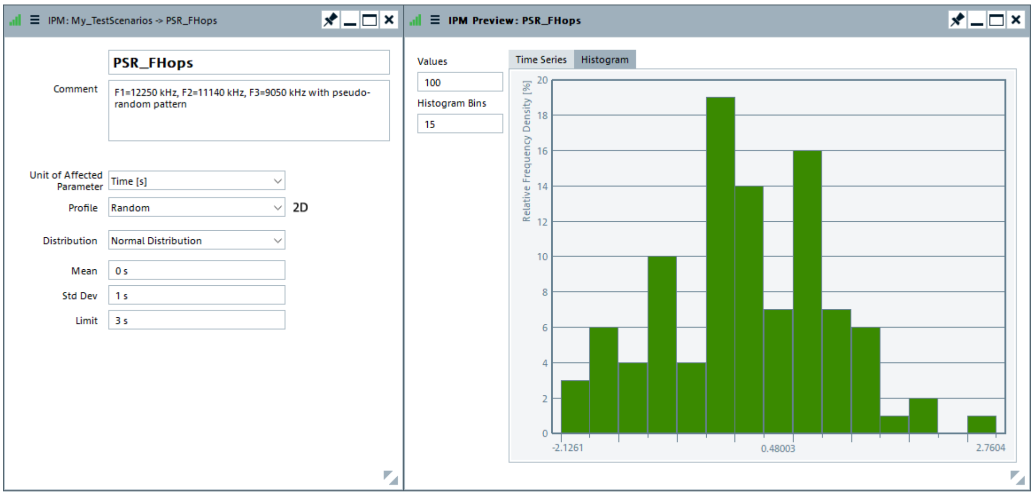


Figure 8-9: IPM with shape "Profile = Random" and "Distribution = Normal Distribution"

"Uniform Distribution"
With this distribution, values occur with the same probability in the range between the minimum and maximum level. The granularity is the value set with the parameter "Step".

"Normal Distribution"
The gauss probability density function or normal distribution is characterized by the static parameters "Mean" (μ), "Standard Deviation" (σ) and "Limit".
The Figure 8-10 illustrates the probability at which values would occur related to the standard deviation if no limit was set.

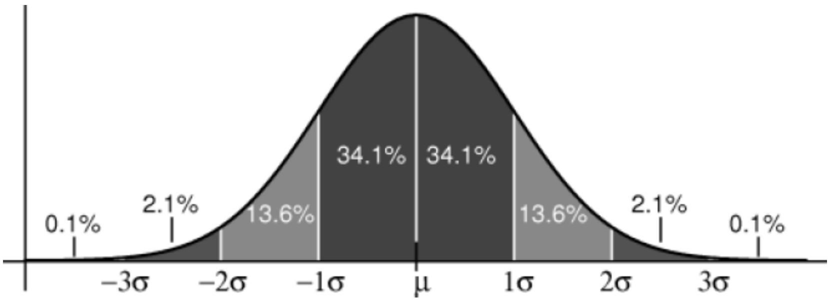
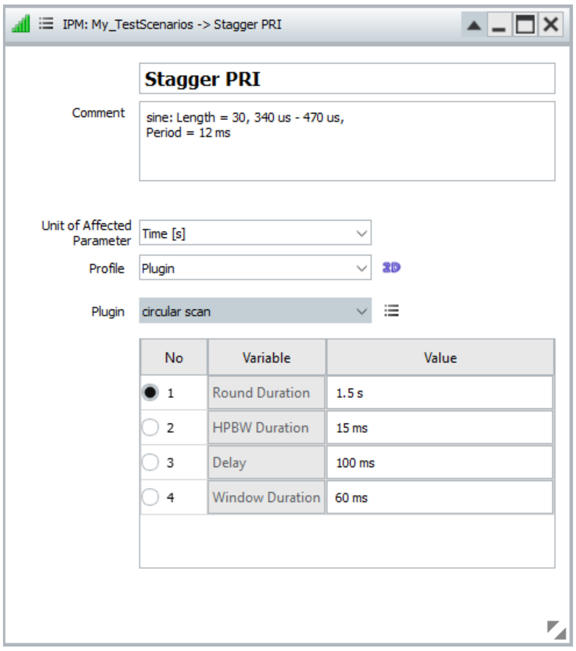


Figure 8-10: Normal distribution

"U Distribution" The U distribution is characterized by a function with lower limit and upper limit that are calculated from selected "Center" and "Range" values.

Remote command:
`IPM:RANDom:DISTribution` on page 481
`IPM:RANDom:UNIForm:MINimum` on page 482
`IPM:RANDom:UNIForm:MAXimum` on page 482
`IPM:RANDom:UNIForm:STEP` on page 482
`IPM:RANDom:NORMal:MEAN` on page 481
`IPM:RANDom:NORMal:STD` on page 481
`IPM:RANDom:NORMal:LIMit` on page 481
`IPM:RANDom:U:CENTer` on page 482
`IPM:RANDom:U:RANGe` on page 482

Plug-in
Custom IPM profiles are used in the same way as the internal IPM profiles.



"Plugin Name" Selects and loads a user-defined IPM profile from a file.
The IPM profile must exist in the "Plug-in" library of the repository.

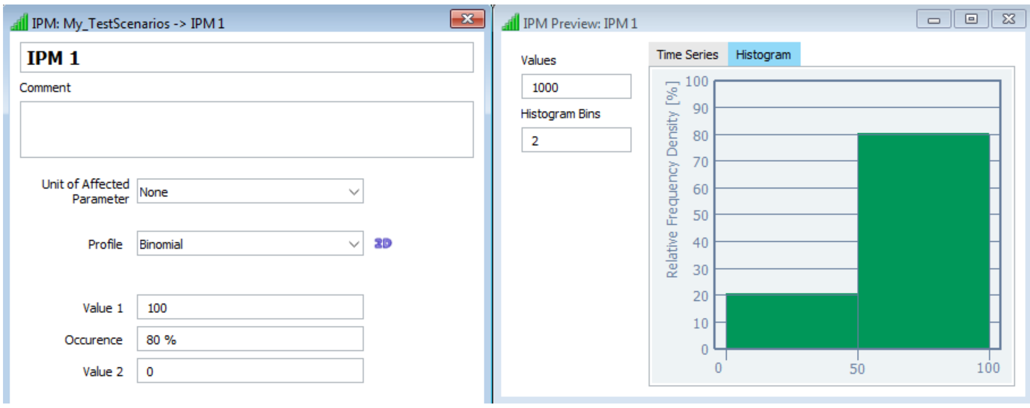
"Plugin Variables"
Sets the values of the variables defined and used in the plug-in.

Use the standard context menu functions to:
- Reset the plug-in variables
- Access the load "Plug-in" settings, see Chapter 19, "Defining complex modulation schemes and IPM profiles", on page 377

Remote command:
IPM:PLUGin:NAME on page 427
IPM:PLUGin:VARiable:CATalog? on page 503
IPM:PLUGin:VARiable:SELect on page 503
IPM:PLUGin:VARiable:VALue on page 504

Binomial

This IPM profile follows a binomial distribution function.



The profile can be used to omit pulses randomly.

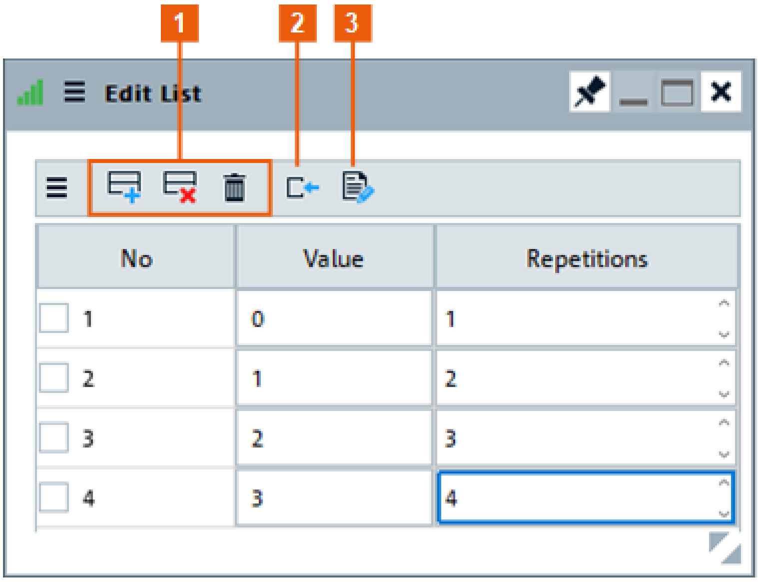
"Value 1, Value 2"
Values of the binomial distribution function.

"Occurrence" Probability of occurrence of value 1 in per cent.
The probability of occurrence of value 2 is calculated as: 100 % - "Occurrence"

Remote command:
IPM:BINomial:VAL1 on page 488
IPM:BINomial:VAL2 on page 488
IPM:BINomial:PVAL1 on page 489

8.1.3 Edit list settings

Edit List
List editor with standard functions.

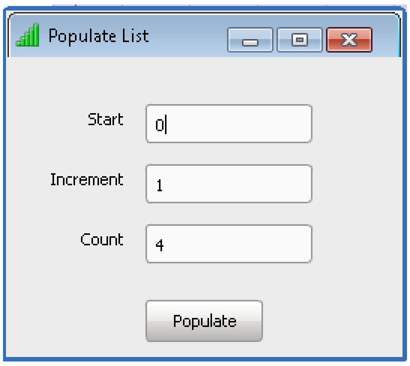


1 = Standard functions for item handling
2 = "Import Data from File" icon
3 = "Populate list" icon

"New/Select" Accesses standard functions for item handling like "Select/Remove All" or "Insert After".
See Table 2-4.

"Import Data from File"
Opens the Windows Explorer to navigate and select an ASCII file with list data.

"Populate List" Accesses the "Populate List" dialog.



To fill in a list with values automatically, set:

- "Start": start value
- "Increment": step with that the values are incremented
- "Count": number of list items.

"Item" Selects an item.

"No" Automatically assigned number to each list item.

"Value" Enters the value of the list item.

"Repetitions" Defines how many times a particular list item is repeated.

Remote command:
`IPM:LIST:LOAD` on page 480
`IPM:LIST:SAVE` on page 480
`IPM:LIST:CLEar` on page 433

8.1.4 Inter-pulse modulation (IPM) settings

Access:

► In the "Sequence" dialog, select a pulse item.

Select "IPM > Static".

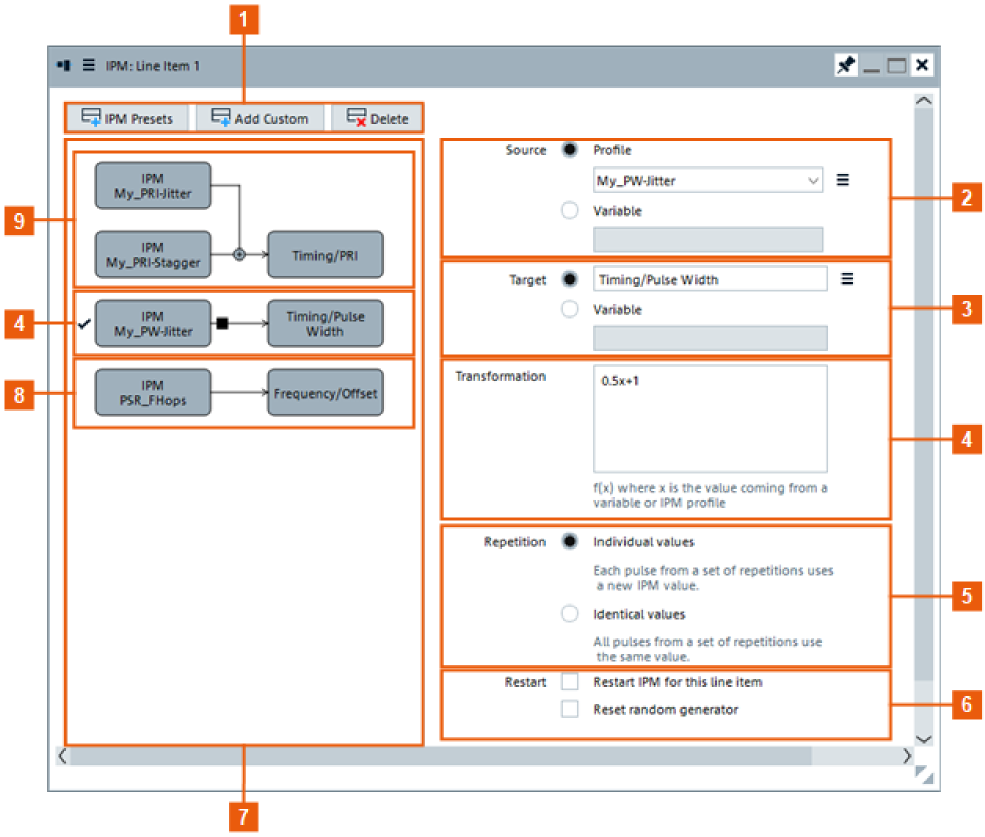


Figure 8-11: IPM: Understanding the displayed information

1 = Standard add/delete functions to create IPM of the selected type or remove the selected IPM from the repository
2 = Variation profile
3 = Defines the pulse parameter or the variable on that the IPM is applied
4 = Enables and defines a transformation
5 = Sets the way repeating pulses are processed
6 = Configures the random generator
7 = Graphical representation of the configured variations and the parameters on that they are applied
8 = Second variation is assigned to a different element ("Target > Frequency > Offset")
9 = Two variations are assigned the same element ("Target > Timing > PRI"); these variations are added

Settings:

Add Custom/Delete
Standard functions to create IPM of the selected type, to disable the selected IPM, or to remove an automatically created one from the repository.

Provided are the following predefined IPM profiles:
- PRI Stagger
- PRI Dwell Switch
- PRI discrete Jitter
- PRI Triangular PFM
- Frequency Agile

Remote command:
`SEQuence:ITEM:IPM:ADD` on page 429
`SEQuence:ITEM:IPM:COUNt?` on page 430
`SEQuence:ITEM:IPM:SELect` on page 431
`SEQuence:ITEM:IPM:DELete` on page 433

IPM graph
Graphical representation of the configured variations and the elements they are applied on.

The elements are interactive. Right-click to open the context menu.

Source
Parameter variations are defined as profiles or variables.

"Profile" Lists the available Jitter and IPM profiles in the repository.
To create a profile or to edit an existing one, select the "New/Select" icon.

"Variable" The variation follows the variable values.
The variable must exist and must be defined. Variable names are case-sensitive.

Variables apply to the following elements:
- **One sequence item** (row in the sequence description table)
- If defined for a **loop**, to all items (pulses, waveforms or nested loops) within the loop.
- In **nested loops**, variables of the parent loop are inherited.

See "To use variables to describe parameter variations" on page 167.

Remote command:
`SEQuence:ITEM:IPM:SOURce:TYPE` on page 476
`SEQuence:ITEM:IPM:SOURce` on page 476
`SEQuence:ITEM:IPM:SOURce:VARiable` on page 476

Target
Profiles can be assigned to an existing pulse parameter or to a variable.

- To select a parameter, use the "New/Select" icon
- To select a variable, enter the variable name (prefix and extension, e.g. "Level_Offset_j"). This variable is available within the current sequence item (row in the sequence description table).
  Variable names are case-sensitive.

Parameter variations can be applied on the pulse parameters listed in Table 8-1.

Remote command:
`SEQuence:ITEM:IPM:TARGet:TYPE` on page 477
`SEQuence:ITEM:IPM:TARGet:PARameter` on page 477
`SEQuence:ITEM:IPM:TARGet:VARiable` on page 477

Transformation f(x)
Transformations are an alternative way to describe the output value of the IPM as a mathematical function of the raw value from the IPM profile.

Example:

0.5*x + 1

The example uses the equation f(x) = a*x+b, where the multiplier a = 0.5 and the offset b = 1.

See:
- "To use variables to describe parameter variations" on page 167
- Chapter B, "Formula syntax", on page 647

Remote command:
`SEQuence:ITEM:IPM:EQUation` on page 477

Repetition
Defines the way the variations are applied on repeating pulses:

"Individual value"
A new value is calculated per pulse.

"Identical value"
The same values are used for all repetition of a particular item in the Sequence > Sequence Description Table.
New values are selected each time the same IPM profile is applied to a different item.
Use the "Restart" options to reset an IPM profile.

Remote command:
`SEQuence:ITEM:IPM:MODE` on page 477

Restart
If the IPM profile uses a random generator, you can:
- Restart the IPM per sequence line item, i.e.:
  – Resets incremented values
  – Starts processing of list from the first item
  – Restarts plug-ins
- Reset the start seed of random generator.

Remote command:
`SEQuence:ITEM:IPM:RESTart` on page 478
`SEQuence:ITEM:IPM:RANDom:RESet` on page 478

8.2 How to create IPM profiles and vary pulse parameters

See:

- "To create an IPM with the "Startup Assistant"" on page 160
- "To access the IPM profile dialog" on page 160
- "To create a simple staggered PRI IPM profile" on page 160
- "To assign the staggered PRI profile to the PRI parameter of a pulse" on page 162
- "To visualize the impact of the IPM profiles" on page 164
- "To create and assign a PRI Jitter" on page 165
- "To vary pulse parameters simultaneously" on page 166
- "To create and apply a frequency agile (frequency hopping) pattern" on page 164
- "To use variables to describe parameter variations" on page 167

To create an IPM with the "Startup Assistant"

1. In the menu bar, select "Help > Wizard".
2. Select "Create a new inter-pulse modualtion > On".
3. Select "Next".
4. Select the "Unit" and the "Profile" type.
   See Unit of Affected Parameter and Profile.

To access the IPM profile dialog

Perform one of the following:

1. Select "Repository Tree > IPM > New".
2. Open the "Repository Tree > Sequence > Sequence Description > IPM > IPM/Static > IPM" dialog and:
   a) On the right of "Source Profile" parameter, open the context menu
   b) Select "New IPM".

   A new empty IPM profile is created and added to the project tree.
   New profiles are named `IPM <n>` where n is a number starting at one.
   You can add information to describe and identify the profile, like a name or a comment.

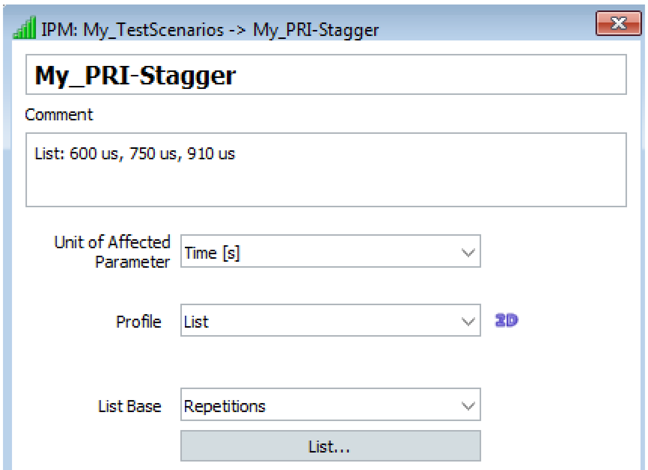
To create a simple staggered PRI IPM profile

In this example, we create an IPM profile that contains three time values, defined in a list form. We assign this IPM profile to the PRI parameter of a pulse, see "To assign the staggered PRI profile to the PRI parameter of a pulse" on page 162.

Generated is a pulse train containing three pulses, as, for example, the pulse train on Figure 8-2.

1. Open an IPM profile dialog.
   See "To access the IPM profile dialog" on page 160.

2. Select "Unit >Time [s]" and "Profile > List"



See also Chapter 8.1.2, "IPM profiles settings", on page 144.

3. Select "List...".
Create a list with three items.

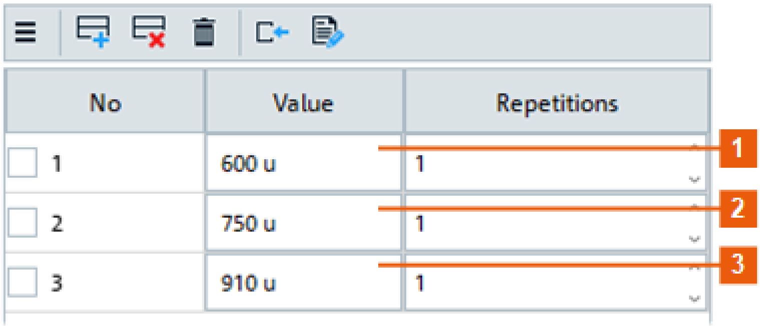
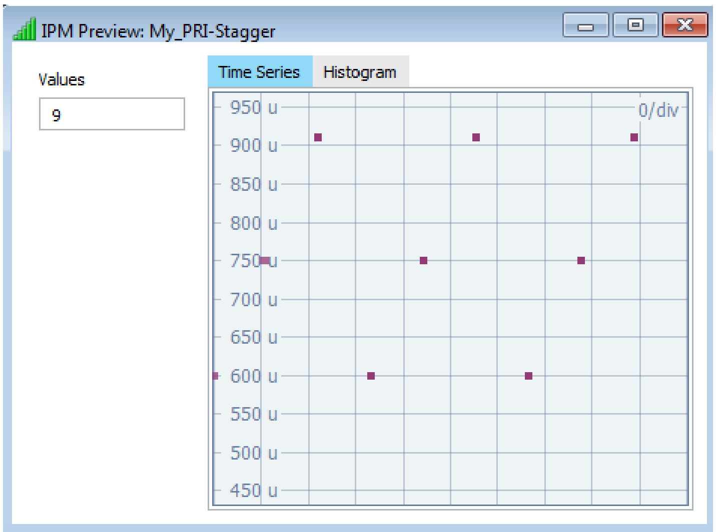


Figure 8-12: Example of a simple staggered PRI IPM profile

1 = PRI_1 = 600 us
2 = PRI_2 = 750 us
3 = PRI_3 = 910 us

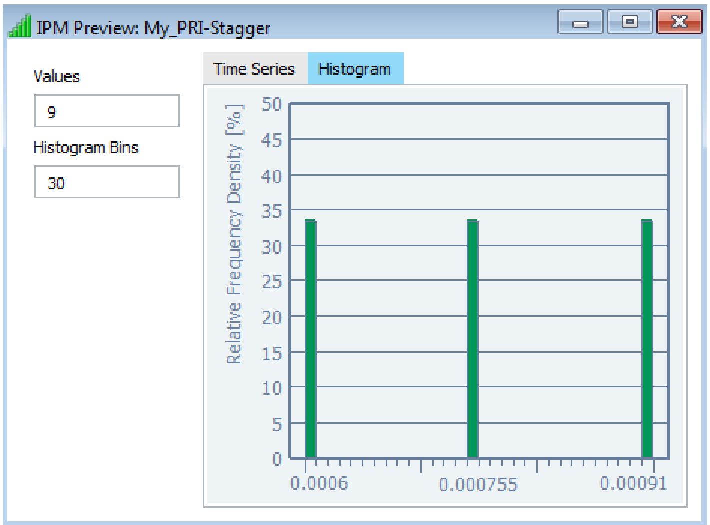
The value of each item is set to the required PRI; items are processed once.

4. In the "IPM" dialog, select "2D > Time Series" to visualize the configured IPM profile in the time domain.



Created is a simple profile with three values only.

5. Select "Histogram" to retrieve statistical information on the IPM profile.

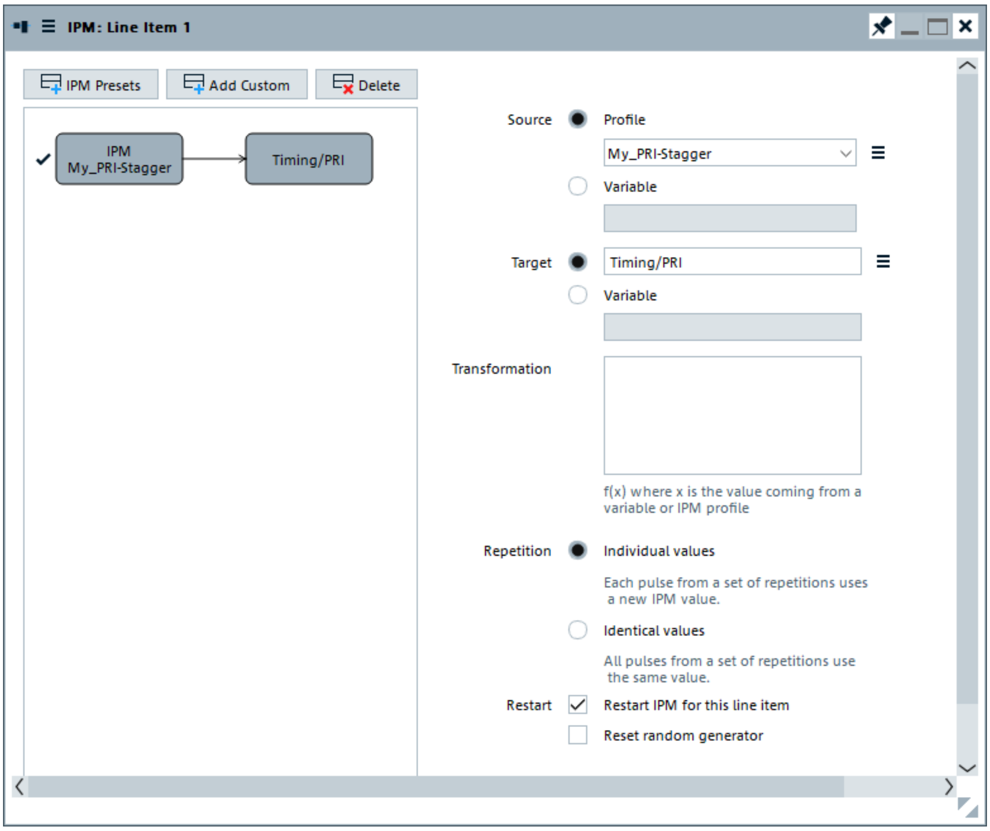


In practice, a staggered PRI profile follows more complex structure and comprises of a larger number of pulses, see for example Figure 8-4.

To assign the staggered PRI profile to the PRI parameter of a pulse

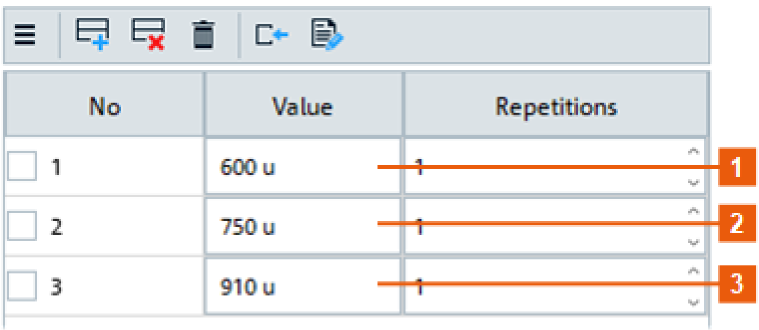
1. Open the pulse sequence to which the created profile is applied, e.g. "My_PT_PRI-Stagger".
   See also "To create a simple sequence" on page 131.
2. In the "Sequence" dialog, select a pulse item.
   Select "IPM > Static".
   Per default, there are no predefined parameter variations in the "IPM" dialog.
3. To assign the created IPM profile:

a) Select "Add Custom".
New empty profile "IPM None" is created.
b) Select "Source > Profile > My_PRI_Stagger".
c) On the right of "Target" parameter, open the context menu.
Select "Timing > PRI".



For description of the provided settings, see Chapter 8.1.4, "Inter-pulse modulation (IPM) settings", on page 156.

4. In the "Sequence" dialog:
a) Set "Repetition Count = 12"
b) Set "Delay = 230 us" to enable a stagger start delay, see Figure 8-2



The "Sequence" dialog confirms, that a parameter variation varying the PRI is applied ("IPM = IPM" and "PRI = var").

To visualize the impact of the IPM profiles

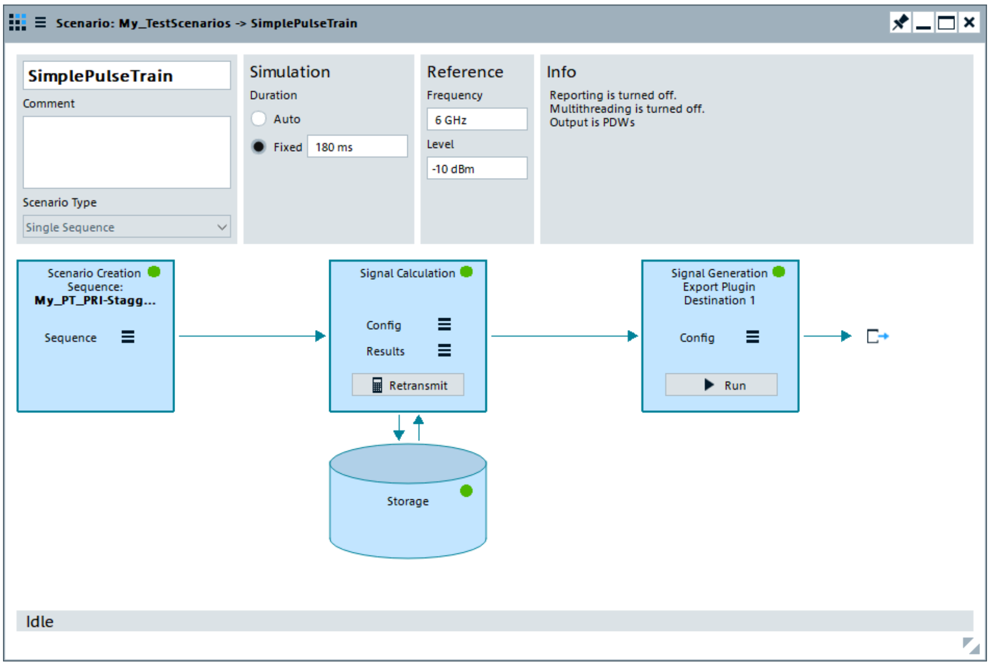
We assume, that a PRI stagger has been configured and applied to a pulse train, as described in "To assign the staggered PRI profile to the PRI parameter of a pulse" on page 162.

We also assume, that a "Pulse Train" or a "Pulse Train (Collection)" scenario is configured, a destination is selected, and the sequence is assigned to this scenario (see Chapter 5.3, "How to select and create a test scenario", on page 90).

To visualize the resulting sequence:

1. In the "Scenario" dialog, select "Signal Calculation > Calculate".

   The software calculates the signal. Green LEDs indicate that the processing is completed.

2. Select the sequence in "Signal Calculation > Results > View".

   The "Signal Preview" confirms the configured pulse sequence with enabled PRI stagger.
   For detailed description of the provided settings, see Chapter 15, "Visualizing and analyzing signals", on page 284 .

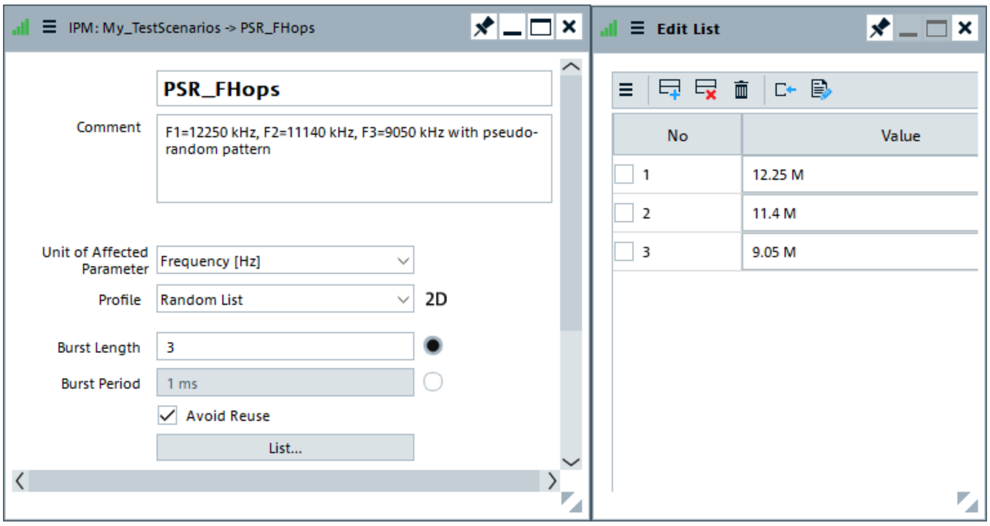
To create and apply a frequency agile (frequency hopping) pattern

1. In the "Sequence" dialog, select "IPM" to open the "IPM" dialog.

2. Select "Frequency Agile".

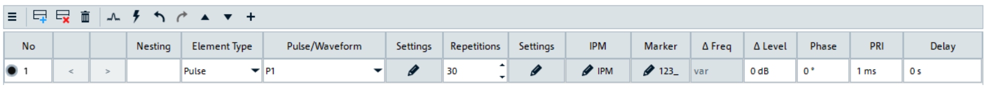
   A new IPM profile is created, automatically named and assigned to the "Frequency/ Offset" parameter.

3. Configure the parameters, for example, as shown on the following figure.



4. Use the 2D view to display the configured pattern.
   See also Figure 8-7.
5. Observe the "Sequence" dialog.

   The sequence configuration settings confirm the configuration.



6. Use the "Signal Preview" dialog to visualize the generated signal.

   See also:
   - "To visualize the impact of the IPM profiles" on page 164

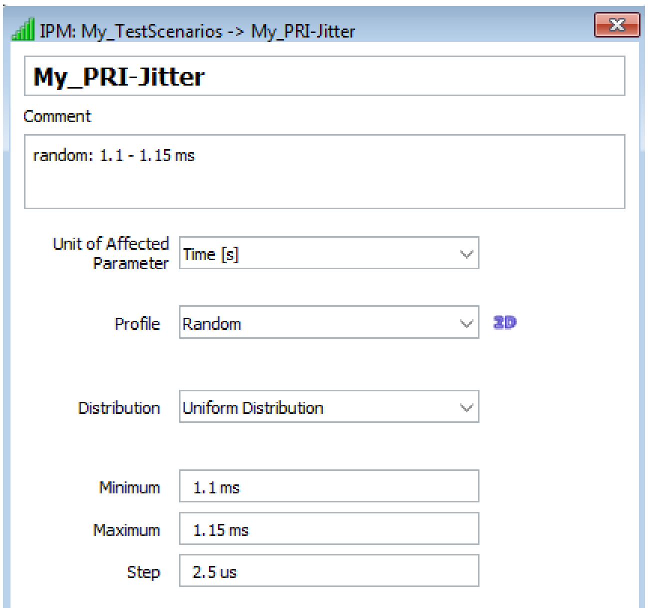
   The signal will hop the three frequencies according to a pseudo-random pattern. Each hop frequency is used three times.

To create and assign a PRI Jitter

There is a dedicated function for creating PRI Jitters, the "PRI Discrete Jitter" in the "IPM" dialog (see also "To create and apply a frequency agile (frequency hopping) pattern" on page 164).

In this example, however, we create manually a randomly varying IPM profile with 20 different values. Within 20 ms, the PRI varies in the range 1.1 ms to 1.15 ms.

1. Create an IPM Profile that uses "Profile > Random" and, for example, a "Distribution > Uniform Distribution".
   See also "To access the IPM profile dialog" on page 160.



2. Select "2D" to visualize the function.

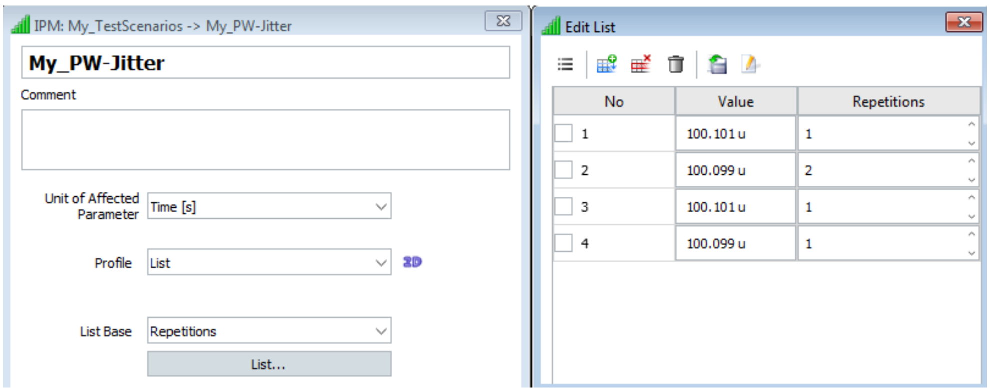
3. Open the required sequence. Assign the IPM profile to a pulse train.
   See "To create and apply a frequency agile (frequency hopping) pattern" on page 164.

To vary pulse parameters simultaneously

We assume, that a PRI jitter has been configured and applied to a pulse train, as described in "To create and assign a PRI Jitter" on page 165.

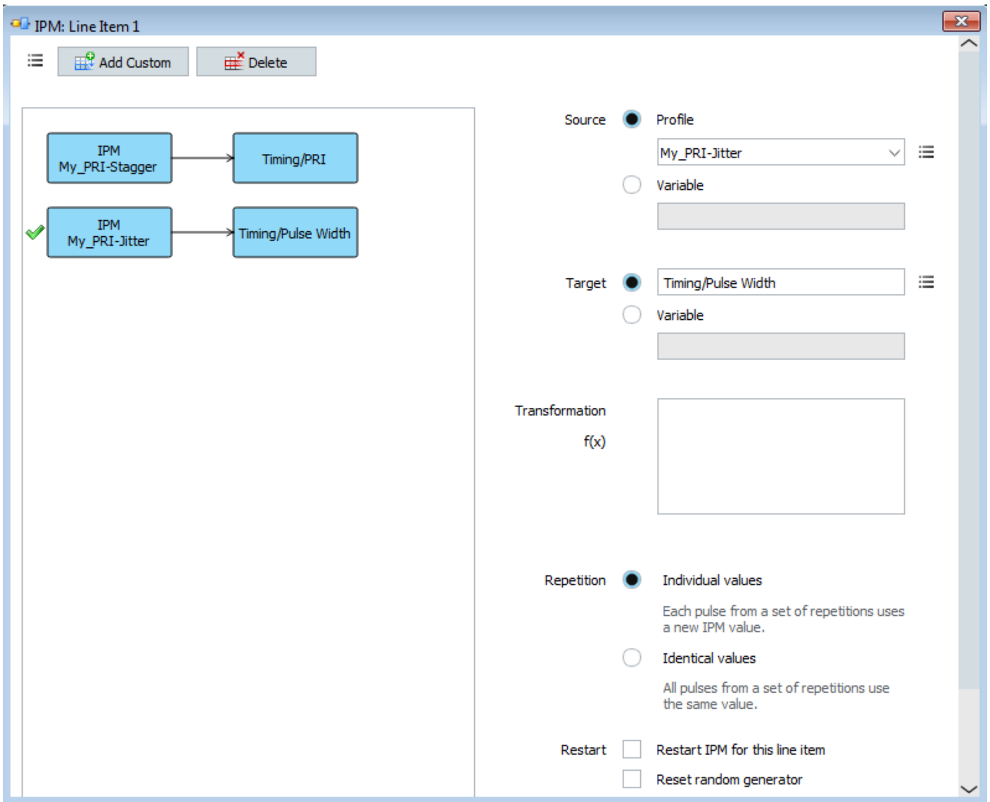
To apply a second variation, for example a PW jitter, perform the following:

1. Create an IPM Profile, for example as shown on the following figure.



   Configured is a pulse width (PW) pattern with five values.

2. Open the required sequence.
   Assign the IPM profile to the same pulse train as the PRI Jitter.



Tip: You can vary one pulse parameter with more than one profile, see Figure 8-11.

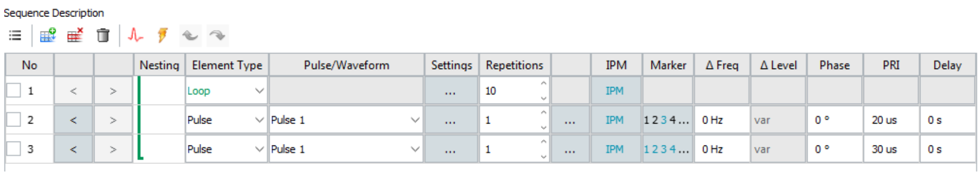
Configured is a pulse train, consisting of 20 pulses, each with different PRI value. The PW pattern is applied on five consequent pulses and is repeated four times. In this example, the PRI varies in the range 1.1 ms to 1.15 ms; the PW between 100.101 us and 100.099 us.

To use variables to describe parameter variations

In this example, we create a sequence of two pulses P1 and P2 that are included in a loop. We use an IPM step profile that creates ten values within the range -10 dB to 0 dB.

We assign this IPM profile to the loop and use it to create and vary a variable. This variable in turn is the source for the variations of the two pulses, included in the loop.

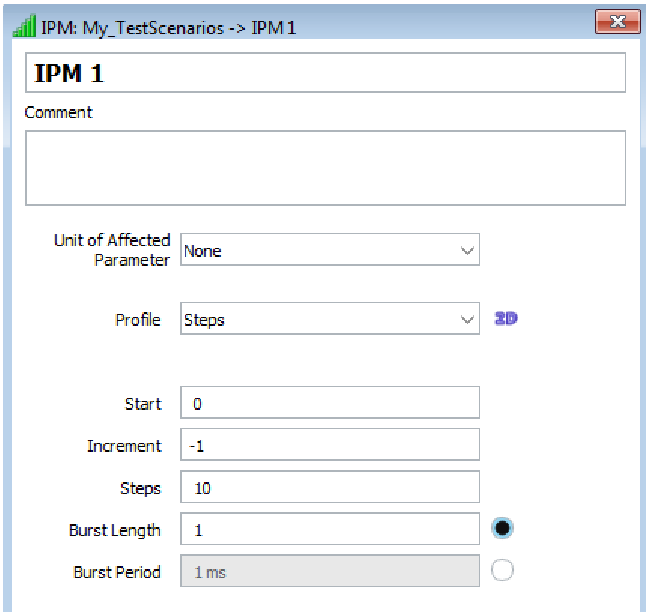
1. Create two unmodulated pulses with pulse width of 10 us.
2. Create a sequence with the following settings:



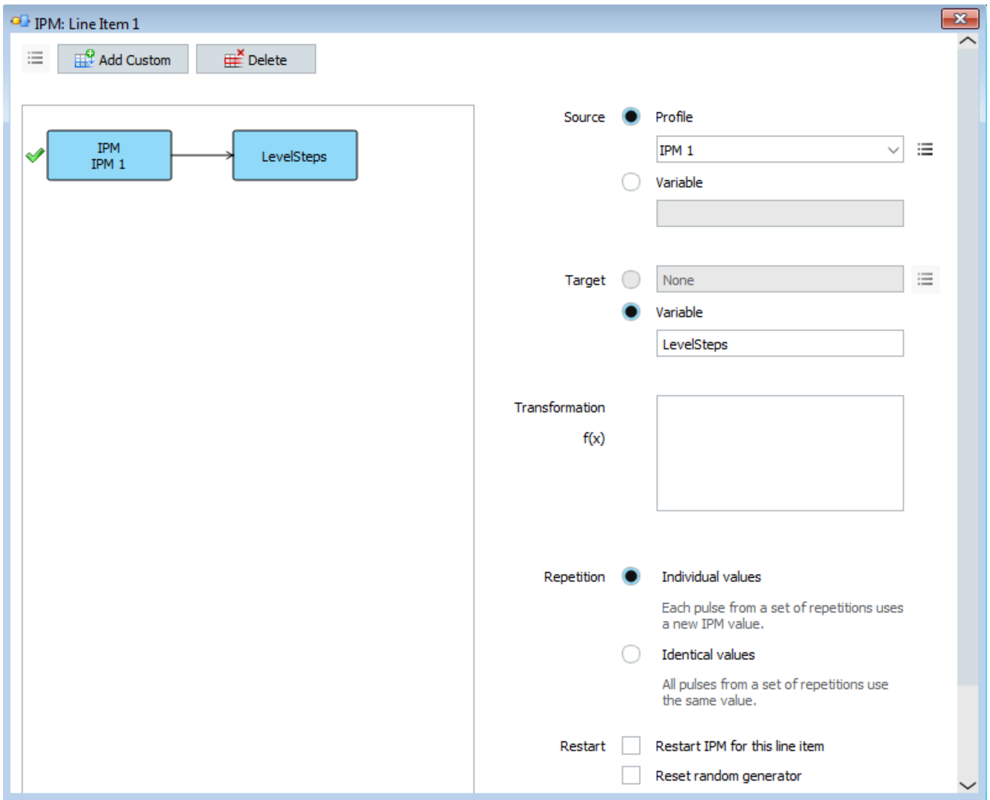
Sequence Description

No			Nesting	Element Type	Pulse/Waveform	Settings	Repetitions		IPM	Marker	Δ Freq	Δ Level	Phase	PRI	Delay
1	<	>		Loop		...	10		IPM						
2	<	>		Pulse	Pulse 1	...	1	...	IPM	1 2 3 4 ...	0 Hz	var	0 °	20 us	0 s
3	<	>		Pulse	Pulse 1	...	1	...	IPM	1 2 3 4 ...	0 Hz	var	0 °	30 us	0 s

3. Create a step profile with the following settings:

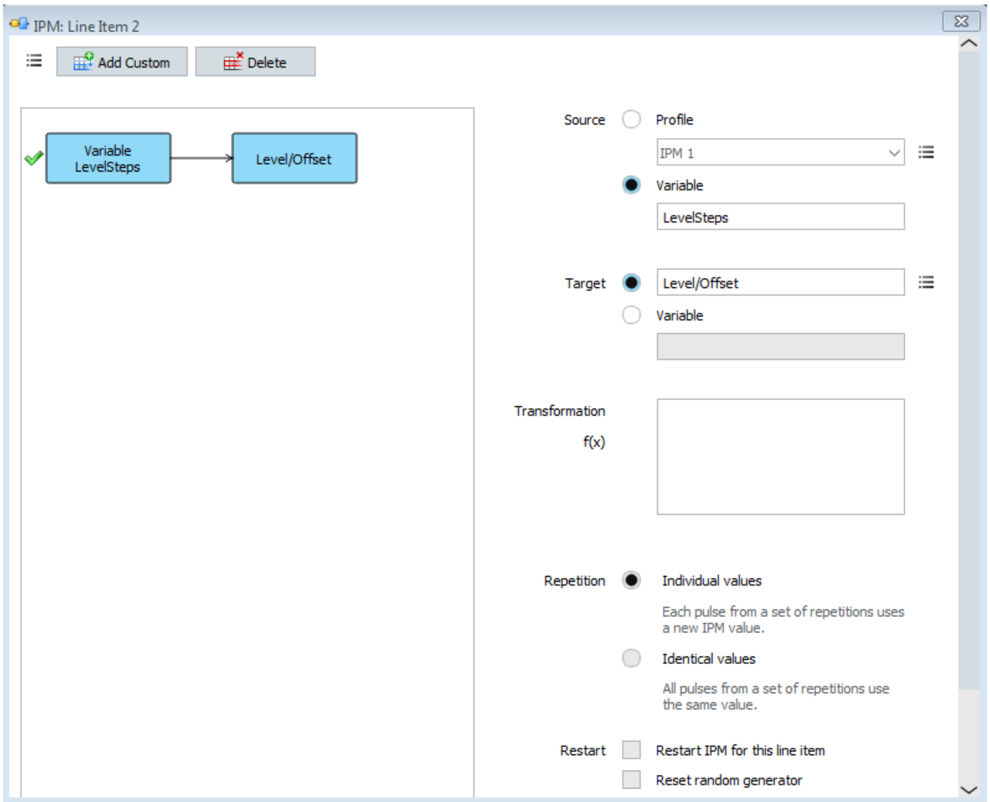


4. Assign the IPM profile to the loop item.
   a) Select "Target > Variable".
   b) Set the variable name.



5. For both pulses:
   a) Assign the created variable as a source of the IPM variations.

b) Set the "Target > Level > Offset".



6. Only for "Pulse 2", type into "Transformation f(x)" *x -10*.
7. Assign the sequence to a scenario.
   Calculate the waveform.
   Open the "Signal Preview" dialog.

   The display confirms that the level of "Pulse 1" decreases from 0 dB with step of 1 dB.
   The level of "Pulse 2" follows the same pattern but the level is 10 dB lower than the "Pulse 1" level.

9 Defining antenna patterns and antenna scans

This section informs you about the antenna pattern types and antenna scan types provided by R&S Pulse Sequencer Digital. It also helps you select and configure a subset of antenna characteristics as a basis for the emitter configuration and its different operating modes.

9.1 Overview of the main antenna parameters

Transmitting antennas are elements that radiate energy. Energy emitted in a particular direction is referred to as a *beam* or *lobe*. The direction in which the most energy is radiated is the *main beam* or *main lobe* (see Figure 9-1). Energy radiated energy in other directions (i.e. not in the main lobe) forms the side lobes and the back lobe.

The beamwidth is a key parameter. Beamwidth is usually defined as an angle. The angle is measured at the transmitting antenna (centered on the beam-axis) as shown in Figure 9-1. The value (e.g. 5°) indicates the angle at which the power of the beam is 50% (3 dB) lower than the peak power on the beam-axis. This angle is referred to as the half-power beamwidth (HPBW).

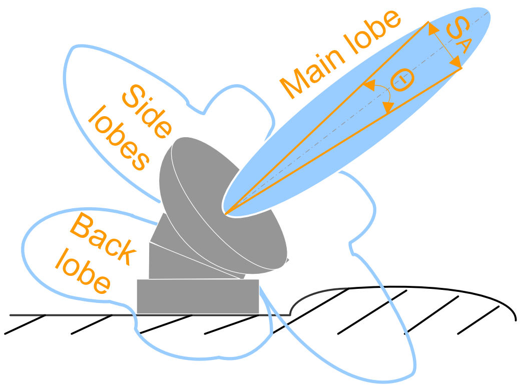


Figure 9-1: Main characteristics of an antenna

Main lobe/bream	= The region around the direction with maximum radiation
Side lobes	= Energy radiated in undesired direction
S_A	= Angular resolution
Θ (theta)	= Antenna beamwidth or the -3 dB angle, defined by the half-power (HPBW)

Antenna pattern

Antennas of different type and with different purpose use wider or more directed beams, for example pencil beams have a typical HPBW of about 2 deg.

The R&S Pulse Sequencer Digital provides a wide range of commonly used antenna patterns, such as parabolic, cosecant squared, or planar phased array antennas. Moreover you can:

- Adjust the antenna pattern settings to simulate, for example, a directive antenna that concentrates the radiated energy in a particular direction
- Simulate a back lobe pattern
- Create a custom antenna pattern based on the required antenna characteristics (see "To create a custom antenna pattern based on the required antenna characteristics" on page 211)
- Import your custom antenna pattern
(see "To import a user-defined antenna pattern from file in one of the predefined file formats" on page 212 and "To import an antenna pattern file in custom file format" on page 215)
- Design your phased array antenna
- Visualize the antenna pattern on a 2D and a 3D preview, see Figure 9-12.

Antenna scan

Antenna scans describe the movement of the antenna beam in three-dimensional space. A conical scan, for example, is a movement where the beam points in the direction of the target and is rotated in a circle. The half-power points of the beam draw a circle.

The R&S Pulse Sequencer Digital provides a wide range of commonly used antenna scans, such as a raster scan, helical scan, or conical scan. You can also:

- Adjust the antenna scan settings
- Visualize the configured antenna scan on a 3D preview
- Design your antenna scan
- Observe the signal received by a receiver for a particular antenna pattern and antenna scan configuration
See:
  – "Received signal amplitude" on page 259
  – "To visualize the signal received by a static receiver" on page 234

Antenna polarization and polarization loss

The polarization of an antenna describes the orientation of the electric wave radiated by the antenna. Polarization can be constant, or it can rotate with each wave cycle.

The Table 9-1 gives an overview and a short description of the supported polarization types.

Table 9-1: Polarization types

Polarization	Type	Short description
Vertical (V)	Linear polarization	The electrical field is vertical to the propagation direction (y-axis).
Horizontal (H)		The electrical field is horizontal to the propagation direction (y-axis).

Polarization	Type	Short description
Circuit right (CR)	Circular polarization	The electrical field rotates clockwise (CW) relative to the propagation direction.
Circuit left (CL)		The electrical field rotates counterclockwise (CCW) relative to the propagation direction.
Slant right (SR)	Linear polarization	The electrical filed is rotated at 45° from the vertical case.
Slant left (SL)		The electrical filed is rotated at -45° from the vertical case

The theoretical maximum received power is achieved if the transmit and the receive antenna are co-polarized and aligned. That is, if the antennas at the emitter and the receiver both use the same polarization and point to each other. If the antennas use different polarizations, the received power decreases. The polarization loss is given by the ratio of the received power to the maximum power. Theoretically, if the transmit and the receive antennas are cross-polarized, the polarization loss is infinite. In practice, however, this value is not reached. In this implementation, the polarization loss value used if the antennas are cross-polarized is a user-definable value.

Table 9-2 lists the polarization loss for any combination of transmit and receive antenna polarizations.

Table 9-2: Polarization loss values depending on the different antenna combinations

Polarization of the transmit antenna	Polarization of the receive antenna	Polarization loss [dB]
Vertical (V)	Vertical (V) (co-polarized antennas)	0
	Horizontal (H) (cross-polarized antennas)	-∞ (As set with the parameter Maximum Attenuation for Antenna Cross Polarization)
	Slant right (SR) or Slant left (SL)	- 3
	Circuit right (CR) or Circuit left (CL)	- 3
Horizontal (H)	Horizontal (H) (co-polarized antennas)	0
	Slant right (SR) or Slant left (SL)	- 3
	Circuit right (CR) or Circuit left (CL)	- 3
Circuit right (CR)	Circuit right (CR) (co-polarized antennas)	0

Polarization of the transmit antenna	Polarization of the receive antenna	Polarization loss [dB]
	Circuit left (CL) (cross-polarized antennas)	-∞ (As set with the parameter Maximum Attenuation for Antenna Cross Polarization)
Circuit right (CR) or Circuit left (CL)	Slant right (SR) or Slant left (SL)	- 3

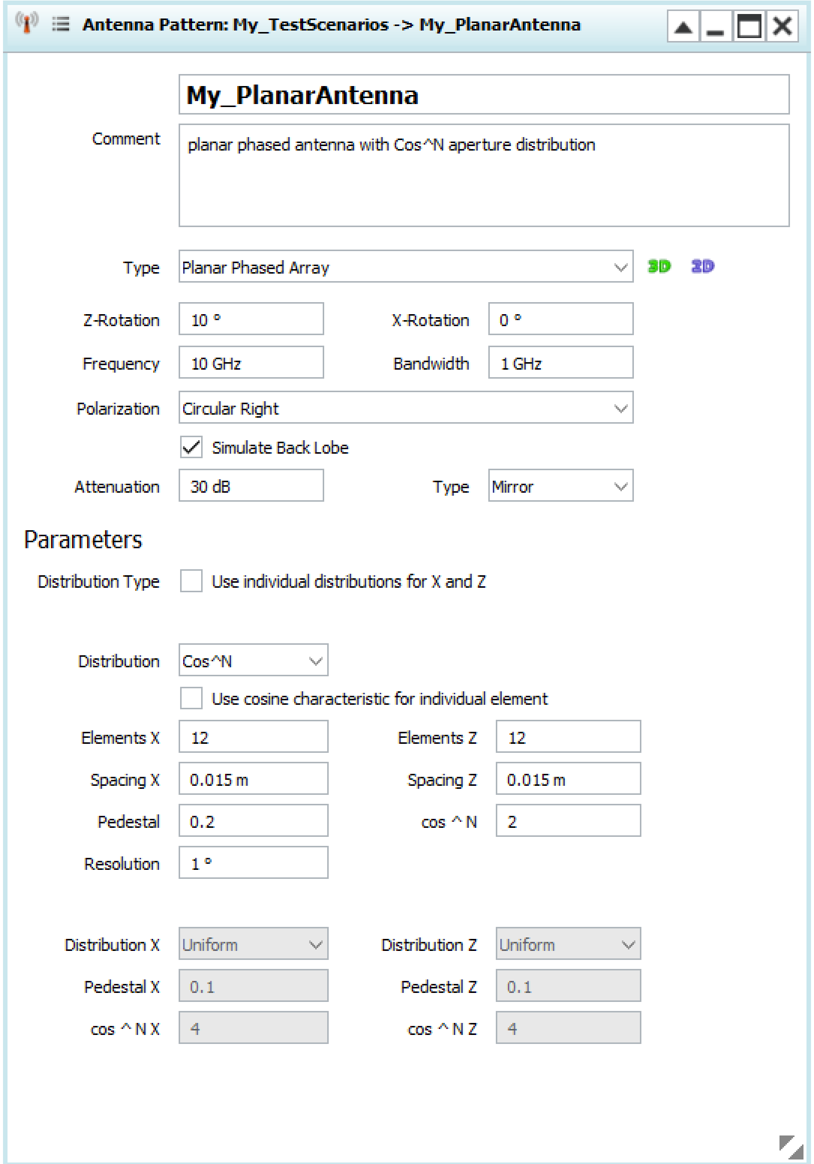
See:

- "Polarization" on page 176
- "Polarization" on page 178
- "To observe the effect of polarization" on page 215

9.2 Antenna pattern settings

Access:

► Select "Repository Tree > Antenna Pattern > New".



Settings

Antenna patterns are characterized by the following common and specific settings:

Antenna Pattern Name

Enter the name of the antenna pattern.

Remote command:
`ANTenna:NAME` on page 427
`ANTenna:CATalog?` on page 425
`ANTenna:CREate` on page 426
`ANTenna:SELect` on page 426
`ANTenna:REMove` on page 428

Comment

Enter a short description.

Remote command:
`ANTenna:COMMent` on page 428

Type

Selects the antenna pattern. Use the "2D" and "3D" diagrams to visualize the polar antenna gain patterns.

Further provided antenna pattern settings depend on the selected antenna model, see:
- Dipole Antenna Settings
- Parabolic Antenna Settings
- Gaussian Antenna Settings
- Sin(x)/x Antenna Settings
- Pyramidal Horn Antenna Settings

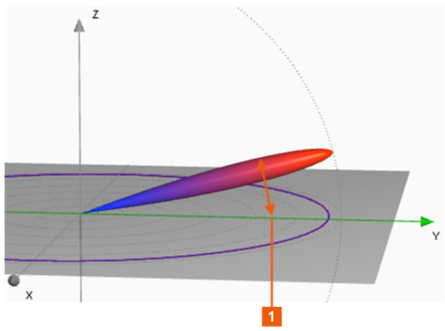
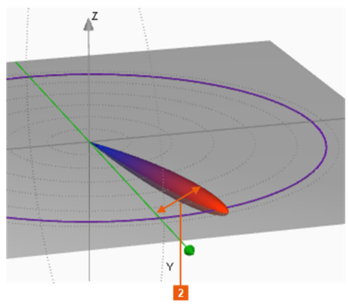
- Cosecant Squared Antenna Settings
- Planar Phased Array Antenna Settings
- Imported antenna settings
- Custom Antenna Settings
- Custom Phased Array Settings
- Cardioid Antenna Settings

Remote command:
`ANTenna:MODel:TYPE` on page 440

Z-Rotation, X-Rotation

Rotates the antenna pattern around the selected axis.

Table 9-3: Effect of antenna pattern rotation

"X-Rotation = 15°"	"Z-Rotation = 15°"
	
1) rotation around the x-axis Positive values rotate the antenna boresight towards the positive z-axis.	2) rotation around the z-axis Positive values rotate the antenna boresight counterclockwise towards the x-axis.

This parameter is useful for example:
- To simulate a radar boresight displacement
- To align a user-defined antenna pattern to the boresight;
  (see "To import a user-defined antenna pattern from file in one of the predefined file formats" on page 212).

Remote command:
`ANTenna:MODel:ROTation:X` on page 449
`ANTenna:MODel:ROTation:Z` on page 449

Frequency, Bandwidth

Sets the frequency and bandwidth the antenna pattern is calculated for.

See also:
- Frequency
- Emitter Modes

Remote command:
`ANTenna:MODel:FREQuency` on page 447
`ANTenna:MODel:BANDwidth` on page 447

Polarization

Sets the antenna polarization.

The polarization of an antenna describes the orientation of the electric wave radiated by the antenna. Polarization can be constant, or it can rotate with each wave cycle.

The selected polarization is indicated on the 3D diagram.

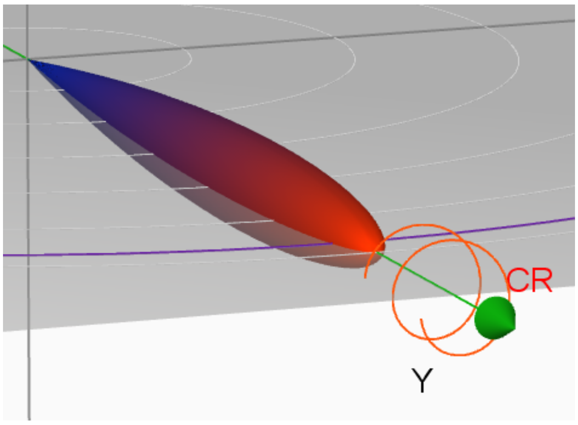


Figure 9-2: Indication of the selected polarization on the 3D diagram

See also "Antenna polarization and polarization loss" on page 171.

"Vertical (V)"
Linear polarization.
The electrical field is vertical to the propagation direction (y-axis).

"Horizontal (H)"
Linear polarization.
The electrical field is horizontal to the propagation direction (y-axis).

"Circuit Right (CR)"
Circular polarization.
The electrical field rotates clockwise (CW) relative to the propagation direction.

"Circuit Left (CL)"
Circular polarization.
The electrical field rotates counterclockwise (CCW) relative to the propagation direction.

"Slant Right (SR)"
Linear polarization.
The electrical filed is rotated at 45° from the vertical case.

"Slant Left (SL)"
Linear polarization.
The electrical filed is rotated at -45° from the vertical case.

Remote command:
`ANTenna:MODel:POLarization` on page 440

Simulate Back Lobe, Attenuation, Type
Enables the simulation of a back lobe and sets its power level and shape.

The parameter "Type" defines the shape of the back lobe pattern:
- "Mirror": the back lobe pattern is the mirrored antenna pattern, attenuated with the selected value, see Figure 9-13.

- "Omnidirectional": the back lobe pattern is a hemisphere, attenuated with the selected value.

Remote command:
`ANTenna:MODel:BACKlobe:ENABle` on page 444
`ANTenna:MODel:BACKlobe:TYPE` on page 444
`ANTenna:MODel:BACKlobe:ATTenuation` on page 444

Resolution
This parameter is only available when you use R&S Pulse Sequencer Digital in CPU mode. Either select CPU mode on initial start-up, or change to CPU mode by selecting "Configure > Settings... > Graphics > GPU Support" and deselecting GPU support. Changing modes requires you to restart the application.

Sets a custom resolution for the antenna pattern simulation.

Use this parameter to reduce interpolation errors when simulating antennas with very narrow HPBWs and side lobes.

Remote command:
`ANTenna:MODel:SINC:RESolution` on page 448

2D and 3D diagrams
You can visualize the antenna characteristics as 2D and 3D interactive diagrams.

2D ← 2D and 3D diagrams
This diagram is a plot of the radiated energy, measured at different angles and at a constant distance from the antenna. The 2D diagram indicates the calculated HPBW and the 3 dB line ("Show 3dB Line > On").

The 2D diagram display the antenna pattern as a XY (azimuth cut) or YZ (elevation cut) graph with polar or with Cartesian coordinates.
- The polar diagram shows the radiation in all possible directions, i.e. the main, side and back lobes, see Figure 9-12.
- The Cartesian diagram is useful to determine the main lobe to back lobe and main lobe to side lobes ratio of highly directive antennas.

Remote command:
`PLOT:POLar:TYPE` on page 449
`PLOT:POLar:CUT` on page 450
`PLOT:POLar:LOG:MIN` on page 450

3D ← 2D and 3D diagrams
This diagram displays the radiated energy as a 3D spatial radiation model.

Minimum Level ← 2D and 3D diagrams
Defines the minimum displayed side lobes level.

Polarization ← 2D and 3D diagrams
Shows the co-polarized and the cross-polarized antenna pattern.

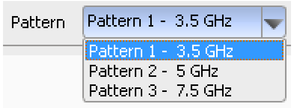
The co-polarized and cross-polarized antennas are selected automatically for the current antenna, see Table 9-2. The cross-polarization is calculated at a polarization loss, as selected with the parameter Maximum Attenuation for Antenna Cross Polarization.

See "To observe the effect of polarization" on page 215.

This function cannot be used for user-defined antenna patterns imported as a file.

Pattern ← 2D and 3D diagrams

A file describing custom antenna pattern can contain more than one antenna patterns. If a custom antenna pattern is loaded, this parameter selects the antenna pattern number that is displayed.



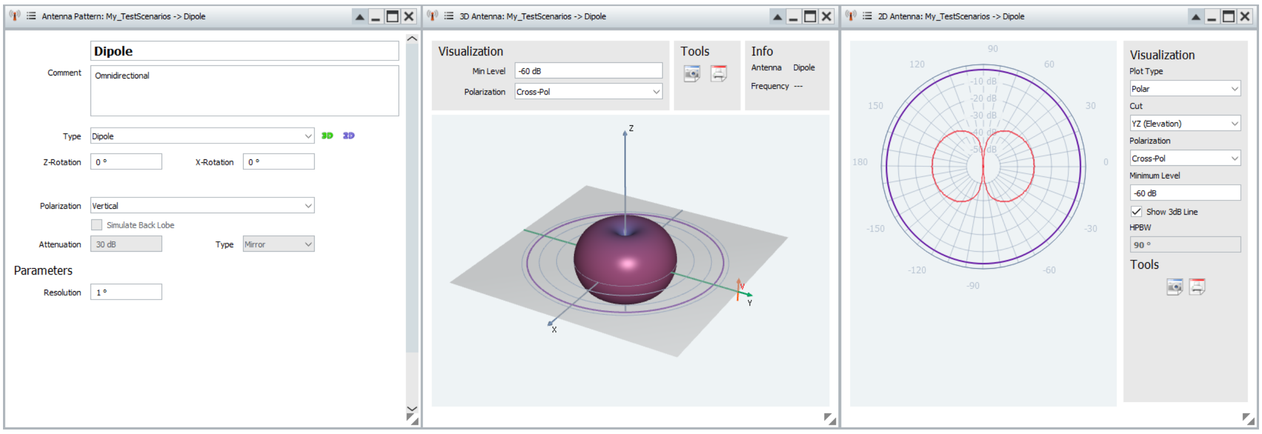
The parameter also indicates the frequency for that the antenna pattern is designed.

Electronic Steering > Z-/X-Rotation ← 2D and 3D diagrams

For "Antenna Model > Custom Phased Array", turns the antenna diagram around the selected axis.

Dipole Antenna Settings

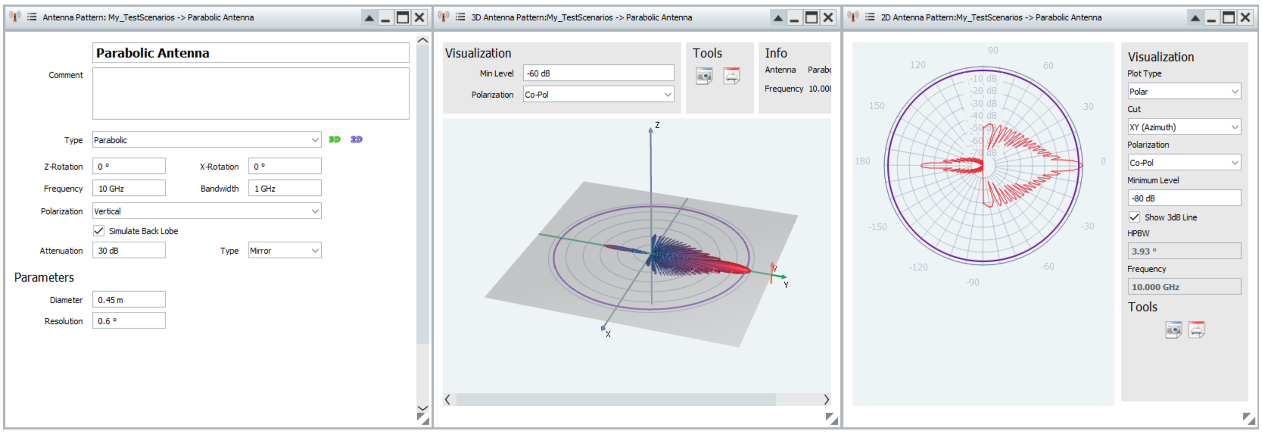
A dipole antenna is a simple antenna with a radiation pattern, shaped like a toroid (doughnut) symmetrical about the axis of the dipole.



Parabolic Antenna Settings

A parabolic antenna has a curved surface with the cross-sectional shape of a parabola with a user defined "Diameter".

Typically, a parabolic antenna radiates the power in a narrow main lobe along the antenna y-axis.



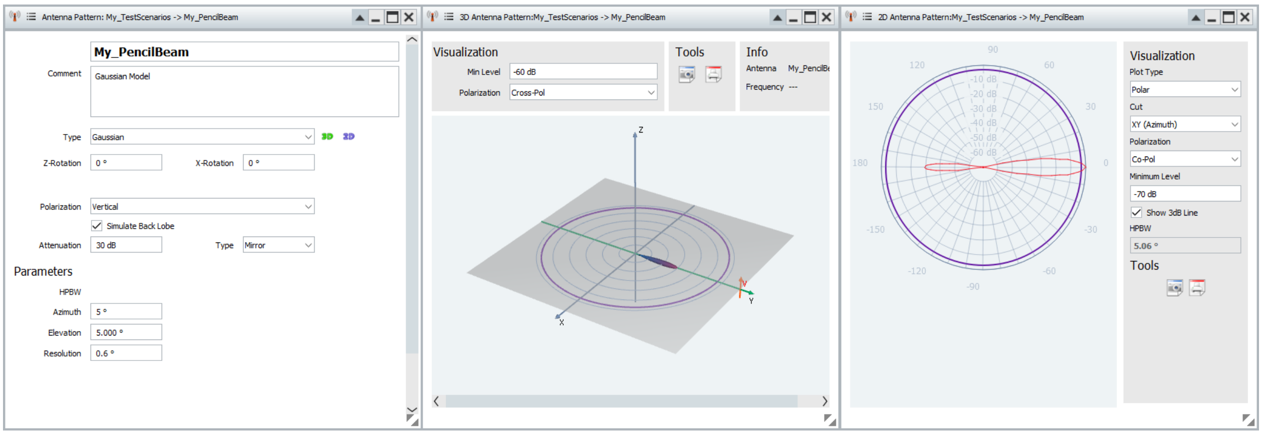
Remote command:

`ANTenna:MODel:PARabolic:DIAMeter` on page 449

Gaussian Antenna Settings

An antenna with radiation pattern that follows the Gaussian distribution.

The main parameter is the HPBW that specifies the angular width within which the antenna is most sensitive. The HPBW can be defined for azimuth and elevation separately.



Remote command:

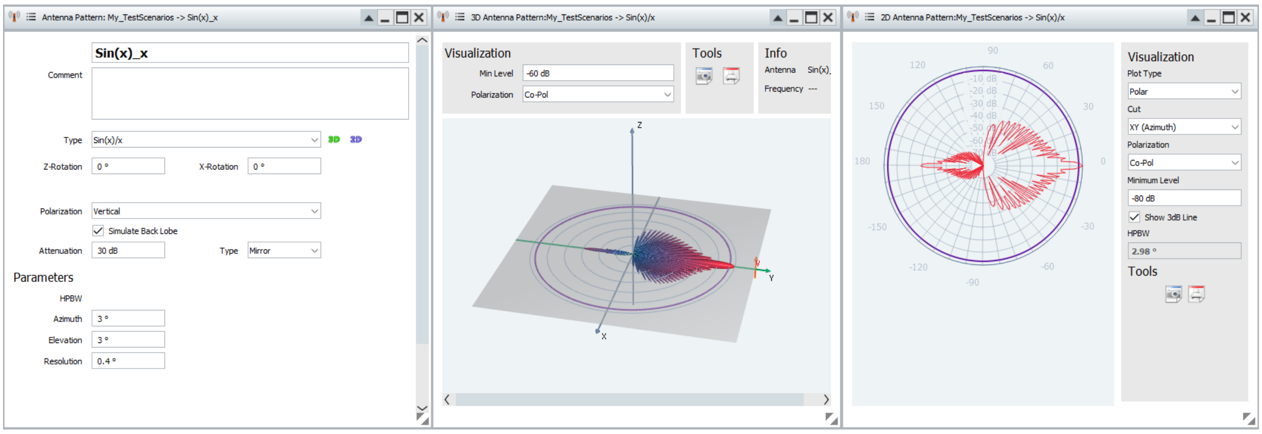
`ANTenna:MODel:GAUSsian:HPBW:AZIMuth` on page 448

`ANTenna:MODel:GAUSsian:HPBW:ELEVation` on page 448

Sin(x)/x Antenna Settings

An antenna with radiation pattern that follows the Sin(x)/x distribution.

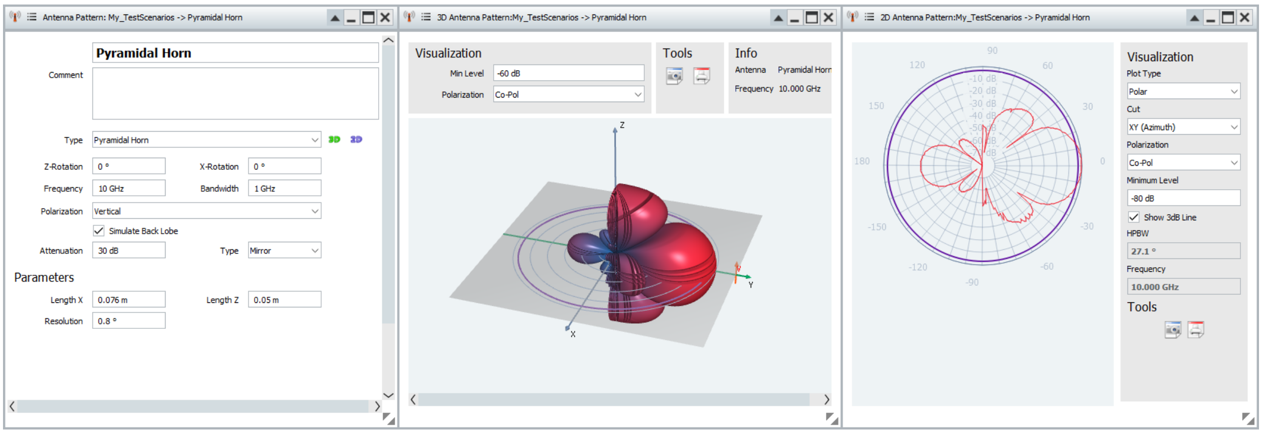
The antenna is characterized by the HPBW. The HPBW can be defined for azimuth and elevation separately.



Remote command:
ANTenna:MODel:SINC:HPBW:AZIMuth on page 448
ANTenna:MODel:SINC:HPBW:ELEVation on page 448

Pyramidal Horn Antenna Settings

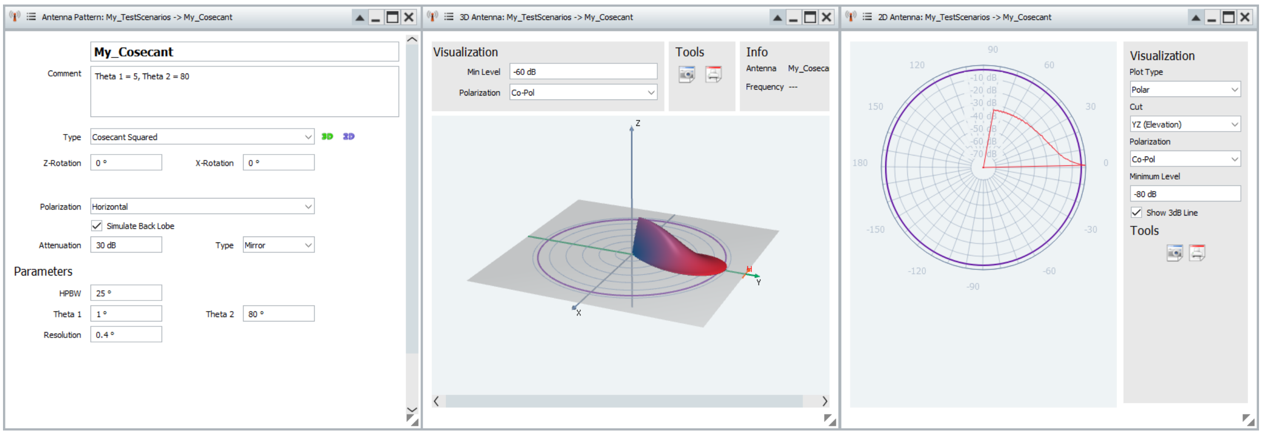
A pyramidal horn antenna has a horn shape in the form of a four-sided pyramid and a rectangular cross section. The lengths of the rectangular sides "Length X, Length Z" are configurable.



Remote command:
ANTenna:MODel:HORN:LX on page 448
ANTenna:MODel:HORN:LZ on page 448

Cosecant Squared Antenna Settings

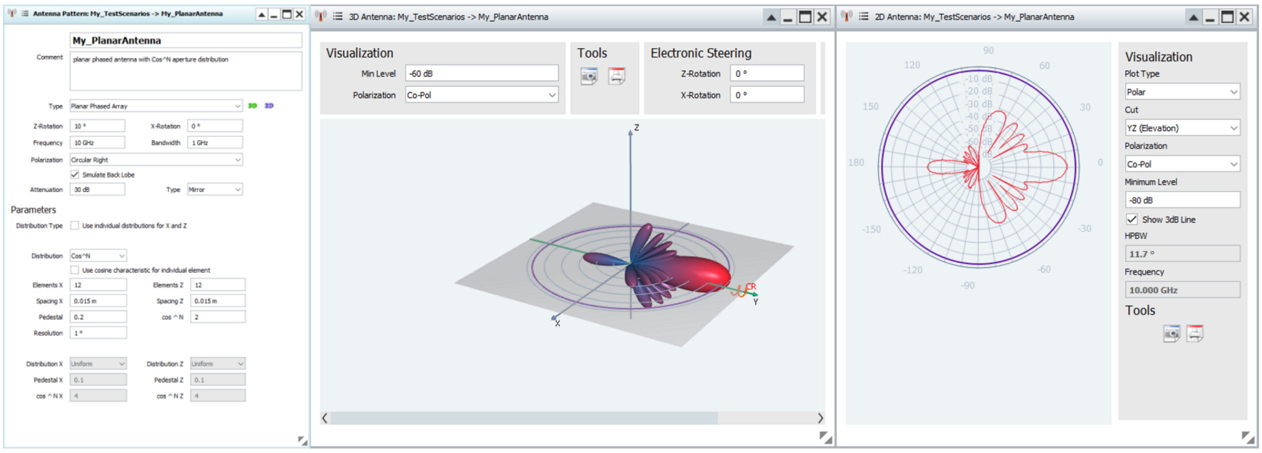
Cosecant squared antennas are designed for air-surveillance radar sets.



Remote command:
ANTenna:MODel:COSecant:HPBW on page 447
ANTenna:MODel:COSecant:T1 on page 444
ANTenna:MODel:COSecant:T2 on page 444

Planar Phased Array Antenna Settings

A planar phased array antenna consists of several antenna elements.



The characteristics of the individual antenna elements influence the transition between the main/side lobes and the back lobes of the phased array antenna, see Table 9-4.

See also "To create an antenna pattern" on page 207.

"Distribution Type"

Enables using the individual distribution function for X and Z direction.

Remote command:
ANTenna:MODel:ARRay:DIStribution:TYPE on page 441

"Distribution" Determines the antenna pattern as a function of the variables *x* and *z*, the pedestal level *P* and the coefficient *N*:

Where:
- P = 0 to 1
  As set with the parameter "Pedestal".
- N = 2 to 10
  As set with the parameter "$\cos^N$"
- x = -1 to +1
  z = -1 to + 1

To simplify the description, only the *f(x)* functions are listed. The *f(z)* values are calculated in the same manner.

The distribution functions are calculated as follows:
- "Uniform"
  $f(x) = 1$
- "Parabolic"
  $f(x) = 1 - (1 - P)*x^2$
- "Cosine"
  $f(x) = P + (1 - P)*cos(0.5*x*\pi)$
- "$\cos^2$"
  $f(x) = P + (1 - P)*0.5*(1+cos(x*\pi))$
- "$\cos^N$"
  If N > 2, $f(x) = P + (1 - P)*(cos(0.5*x*\pi))^N$
  For N = 2, $f(x) = 1$
- "Triangular"
  If N > 2, $f(x) = P + (1 - P)*(1 - abs (x))$
  (the *abs* function returns the absolute value of the variable)
  For N = 2, $f(x) = 1$
- "Hamming"
  $f(x) = P+(1-P)*(0.54-0.46*cos((1.0+x)*\pi))$
- "Hann"
  $f(x) = P+(1-P)*(0.5-0.5*cos((1.0+x)*\pi))$

Remote command:
ANTenna:MODel:ARRay:DIStribution on page 441

"Use cosine characteristic for individual antenna element"
Per default, the individual antenna elements use omnidirectional characteristic. If the parameter "Use cosine characteristic for individual antenna element" is enabled, the cosine characteristic is used.

Remote command:
ANTenna:MODel:ARRay:ELEMent:COSine on page 442

"Elements, Spacing"
Number of radiating elements in X/Z direction and spacing between the elements.

Remote command:
ANTenna:MODel:ARRay:NX on page 442
ANTenna:MODel:ARRay:NZ on page 442
ANTenna:MODel:ARRay:XDIStance on page 444
ANTenna:MODel:ARRay:ZDIStance on page 444

"Pedestal" Sets the pedestal level P.

Remote command:
ANTenna:MODel:ARRay:PEDestal on page 443

"cos^N" Sets the value of the coefficient N in the cos^N distribution.

Remote command:
ANTenna:MODel:ARRay:COSN on page 440

"Distribution X"/"Distribution Z"
Requires "Distribution Type" enabled.
Sets the individual distribution function for X and Z direction.

Remote command:
ANTenna:MODel:ARRay:DISTribution:X on page 442
ANTenna:MODel:ARRay:DISTribution:Z on page 442

"Pedestal X"/"Pedestal Z"
Requires "Distribution Type" enabled.
Sets the individual pedestal level for X and Z direction.

Remote command:
ANTenna:MODel:ARRay:PEDestal:X on page 443
ANTenna:MODel:ARRay:PEDestal:Z on page 443

"cos^N X"/"cos^N Z"
Requires "Distribution Type" enabled.
Sets the individual value of the coefficient N in the cos^N distribution for X and Z direction.

Remote command:
ANTenna:MODel:ARRay:COSN:X on page 440
ANTenna:MODel:ARRay:COSN:Z on page 440

Remote command:
ANTenna:MODel:ARRay:DISTribution on page 441
ANTenna:MODel:ARRay:ELEMent:COSine on page 442
ANTenna:MODel:ARRay:NX on page 442
ANTenna:MODel:ARRay:NZ on page 442
ANTenna:MODel:ARRay:XDIStance on page 444
ANTenna:MODel:ARRay:ZDIStance on page 444
ANTenna:MODel:ARRay:PEDestal on page 443
ANTenna:MODel:ARRay:COSN on page 440

Custom Antenna Settings

You can define an antenna pattern based on the required antenna characteristics.

See Figure 9-15.

"HPBW XY, HPBW YZ"
HPBW specifies the angular width within which the antenna is most sensitive.

"Side lobe level"
Attenuation applied on the first pair of side lobes compared to the power level of main lobe

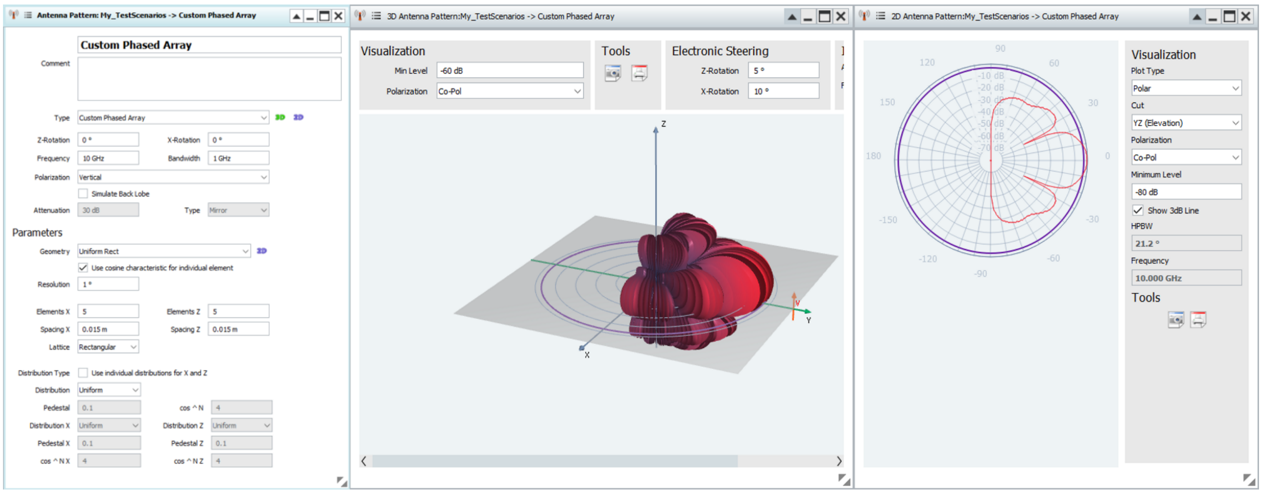
"Rolloff factor" Step size, used to calculate the attenuation of the subsequent side lobes.

"Side lobe scale"
Scale factor, used to calculate the HPBW of the side lobes, i.e. how much smaller or wider the side lobes are compared to the main lobe.

Remote command:
`ANTenna:MODel:CUSTom:HPBW:XY` on page 446
`ANTenna:MODel:CUSTom:HPBW:YZ` on page 446
`ANTenna:MODel:CUSTom:SLSTart` on page 446
`ANTenna:MODel:CUSTom:SLRolloff` on page 446
`ANTenna:MODel:CUSTom:SLSCale` on page 447

Custom Phased Array Settings

You can define a phased array antenna with user-definable geometry, number of elements and spacing between them.



"Geometry" An antenna array is a set of two or more antennas, called antenna elements.

The geometry describes how the single antenna elements of the array are arranged:
- "Uniform Linear" - in a row
- "Uniform Rectangular" - in a rectangular grid
- "Circular Planar" - in a circle
- "Uniform Hex" - as a hexagon

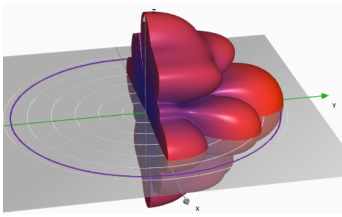
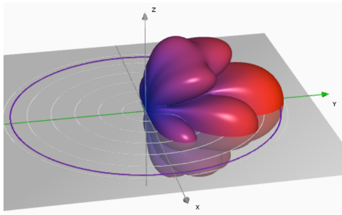
Remote command:
`ANTenna:MODel:CARRay:GEOMetry` on page 445

"Use cosine characteristic for individual antenna element"

Per default, the individual antenna elements use omnidirectional characteristic. If the parameter is enabled, the cosine characteristic is used.

The characteristics of the individual antenna elements influence the transition between the main/side lobes and the back lobes of the phased array antenna, see Table 9-4.

Table 9-4: Effect of the parameter "Use cosine characteristic for individual antenna element"

Off (default)	On
	

Remote command:
`ANTenna:MODel:CARRay:ELEMent:COSine` on page 442

"2D"

Illustrates the array geometry.

Click an antenna element to activate or deactivate it.

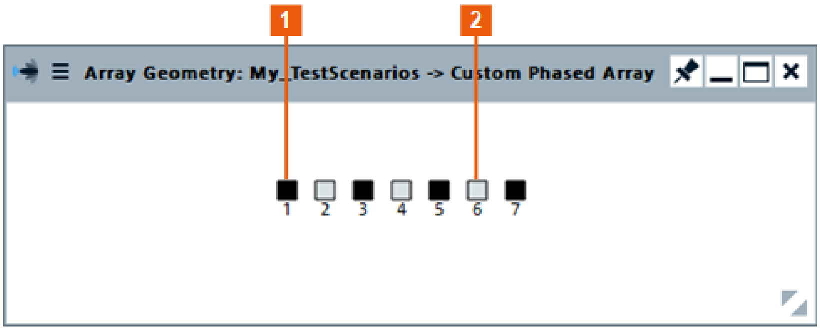


Figure 9-3: Example: Linear array antenna with seven elements

1 = Active element
2 = Deactivated element

Remote command:
`ANTenna:MODel:CARRay:ELEMent` on page 445

"Uniform rectangular antenna array"

A rectangular array is a planar antenna array with X number of columns and Z number of rows. The antenna elements can be placed on rectangularly or triangularly lattice.
Per default, the number of elements in the rows and in the columns is the same and the elements are uniformly distributed. You can change the spacing and use unequal number of elements.

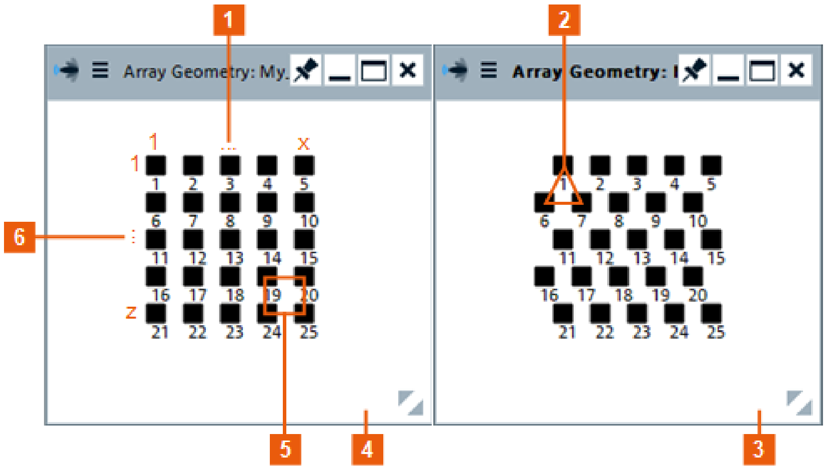


Figure 9-4: Example: Rectangle array antenna with different grids

1, 6 = Number of antenna elements
2, 3 = Triangular lattice
4, 5 = Rectangular lattice

Remote command:
`ANTenna:MODel:CARRay:RECTangular:NX` on page 442
`ANTenna:MODel:CARRay:RECTangular:NZ` on page 442
`ANTenna:MODel:CARRay:RECTangular:XDIStance` on page 444
`ANTenna:MODel:CARRay:RECTangular:ZDIStance` on page 444
`ANTenna:MODel:CARRay:RECTangular:LATTice` on page 445

"Uniform Linear"

A linear array is a planar array where the antenna elements are placed in a row, see Figure 9-3.
You can change the number of antenna elements and the distance between them. The elements are uniformly distributed.

Remote command:
`ANTenna:MODel:CARRay:LINear:N` on page 442
`ANTenna:MODel:CARRay:LINear:DISTance` on page 443

"Uniform hexagonal antenna array"

An array with a hexagonal shape, where each of the six sides has the selected number of antenna elements.

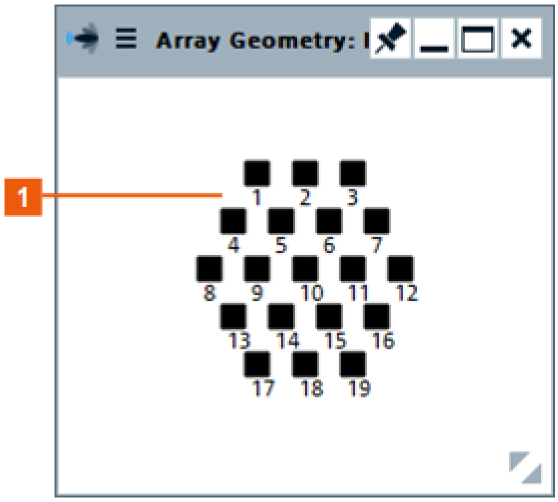


Figure 9-5: Example: Hexagonal array antenna with three elements per side

1 = Elements per side

Remote command:
`ANTenna:MODel:CARRay:HEXagonal:N` on page 442
`ANTenna:MODel:CARRay:HEXagonal:DISTance` on page 443

"Planar Circular"

An array with a shape similar to a circle, selectable radius size, and lattice.

Remote command:
`ANTenna:MODel:CARRay:CIRCular:RADius` on page 445
`ANTenna:MODel:CARRay:CIRCular:DISTance` on page 443
`ANTenna:MODel:CARRay:CIRCular:LATTice` on page 445

"Distribution Type"

Enables using the individual distribution function for X and Z direction.

Remote command:
`ANTenna:MODel:CARRay:DISTribution:TYPE` on page 441

"Distribution" Determines the antenna pattern as a function of the variables *x* and *z*, the pedestal level *P* and the coefficient *N*:

Where:
- P = 0 to 1
  As set with the parameter "Pedestal".
- N = 2 to 10
  As set with the parameter "cos^N"
- x = -1 to +1
  z = -1 to + 1

To simplify the description, only the *f(x)* functions are listed. The *f(z)* values are calculated in the same manner.

The distribution functions are calculated as follows:
- "Uniform"
  $f(x) = 1$
- "Parabolic"
  $f(x) = 1 - (1 - P)*x^2$
- "Cosine"
  $f(x) = P + (1 - P)*cos(0.5*x*\pi)$
- "cos^2"
  $f(x) = P + (1 - P)*0.5*(1+cos(x*\pi))$
- "cos^N"
  If N > 2, $f(x) = P + (1 - P)*(cos(0.5*x*\pi))^N$
  For N = 2, $f(x) = 1$
- "Triangular"
  If N > 2, $f(x) = P + (1 - P)*(1 - abs(x))$
  (the *abs* function returns the absolute value of the variable)
  For N = 2, $f(x) = 1$
- "Hamming"
  $f(x) = P+(1-P)*(0.54-0.46*cos((1.0+x)*\pi))$
- "Hann"
  $f(x) = P+(1-P)*(0.5-0.5*cos((1.0+x)*\pi))$

Remote command:
`ANTenna:MODel:CARRay:DIStribution` on page 441

"Pedestal" Sets the pedestal level P.

Remote command:
`ANTenna:MODel:CARRay:PEDestal` on page 443

"cos^N" Sets the value of the coefficient N in the cos^N distribution.

Remote command:
`ANTenna:MODel:CARRay:COSN` on page 440

"Distribution X"/"Distribution Z"
Requires "Distribution Type" enabled.
Sets the individual distribution function for X and Z direction.

Remote command:
`ANTenna:MODel:CARRay:DIStribution:X` on page 442
`ANTenna:MODel:CARRay:DIStribution:Z` on page 442

"Pedestal X"/"Pedestal Z"

Requires "Distribution Type" enabled.
Sets the individual pedestal level for X and Z direction.

Remote command:
`ANTenna:MODel:CARRay:PEDestal:X` on page 443
`ANTenna:MODel:CARRay:PEDestal:Z` on page 443

"cos^N X"/"cos^N Z"

Requires "Distribution Type" enabled.
Sets the individual value of the coefficient N in the $\cos^N$ distribution for X and Z direction.

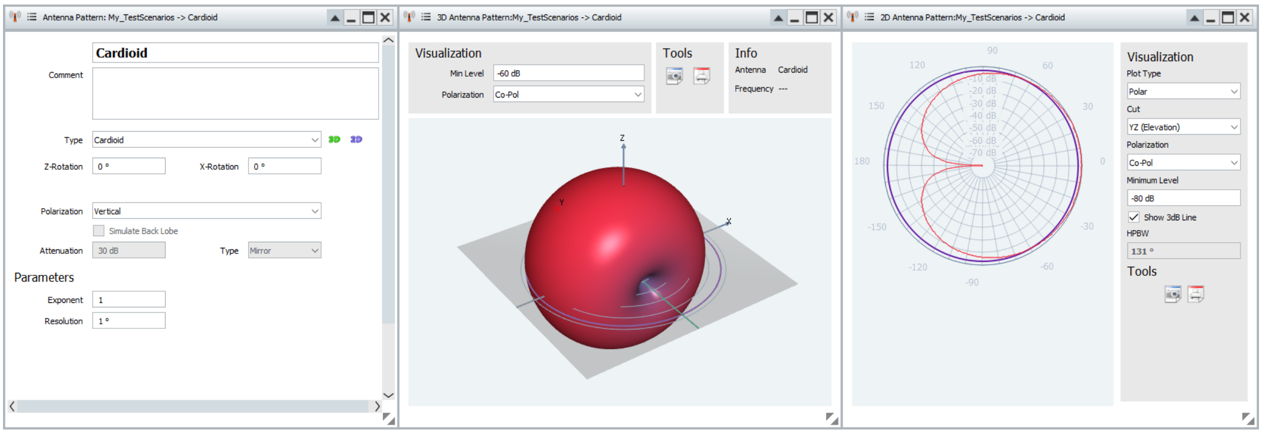
Remote command:
`ANTenna:MODel:CARRay:COSN:X` on page 441
`ANTenna:MODel:CARRay:COSN:Z` on page 441

Cardioid Antenna Settings

A cardioid antenna is an antenna with a heart-like radiation pattern.

The cardioid equation expressed in polar coordinates is calculated as follows:

$r(\varphi) = [1 + \cos(\varphi)]^{exp}$, where:
- φ is the rotation angle
- *exp* is the exponent.



"Exponent"

Use values greater than 1 to narrow the antenna beam.

Remote command:
`ANTenna:MODel:CARDoid:EXPonent` on page 449

Imported antenna settings

You can define and load you custom antenna patterns.

See:
- "To import a user-defined antenna pattern from file in one of the predefined file formats" on page 212.

Import Filter ← Imported antenna settings

Selects the import filter used to parse the antenna pattern file.

"Internal" Parses antenna patterns in the following file formats:
- FEKO Far Field (`*.ffe`) files
- `*.ant_pat` files in the Rohde & Schwarz proprietary format
- Antenna Magus `*.tsv` files
- ANSYS HFSS `*.ffd` files
- Antenna patterns in `*.csv` file format.
  The Import wizard assists you to import `*.csv` files.

"Plugin Name" Any selection other than "Internal" selects a user-defined antenna pattern import filter (plug-in). The import filter must exist in the "Plug-in" library.
See Chapter C.5, "Custom antenna pattern import functions", on page 662.
Antenna patterns in custom file format are files with extension `*.txt`.

Load/Purge ← Imported antenna settings
Standard functions for file management.

Remote command:
`ANTenna:MODel:USER:LOAD` on page 449
`ANTenna:MODel:USER:CLEar` on page 433

Number of patterns and points ← Imported antenna settings
Indicates the number of patterns and points as retrieved from the load file.

Import wizard ← Imported antenna settings
When a `*.csv` file is loaded, the "Import Wizard" opens and tries to determine:
- The used column separator
- The decimal delimiter
- The number of the first line with meaningful data, where the numbering start with 1 and empty lines are counted
- The data content of each column.

You can change the values afterwards.

For example, the file on Figure 9-6 contains a line that explains the parameters in each column.

If your file does not contain this information or if the automatically selected column separators do not match the information in the file, change the values. Always check if the parameters in the "Import Wizard" dialog match the particular file format and content.

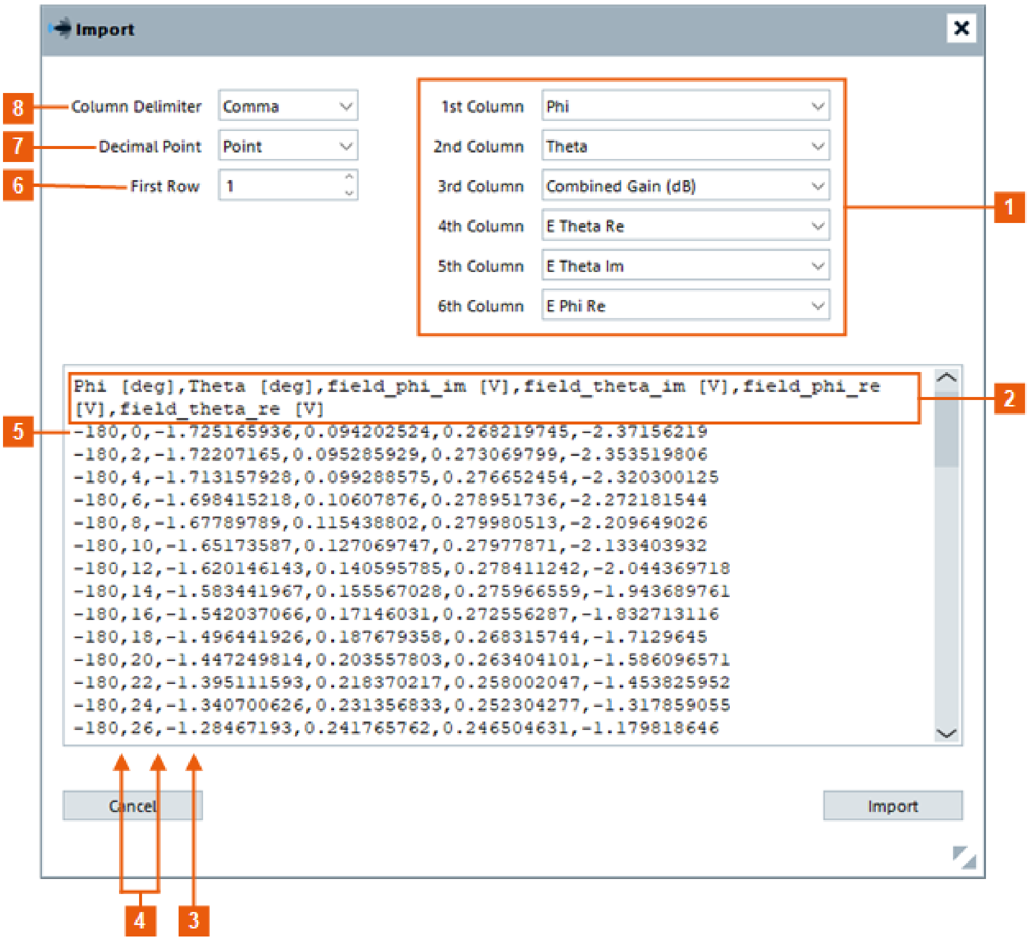


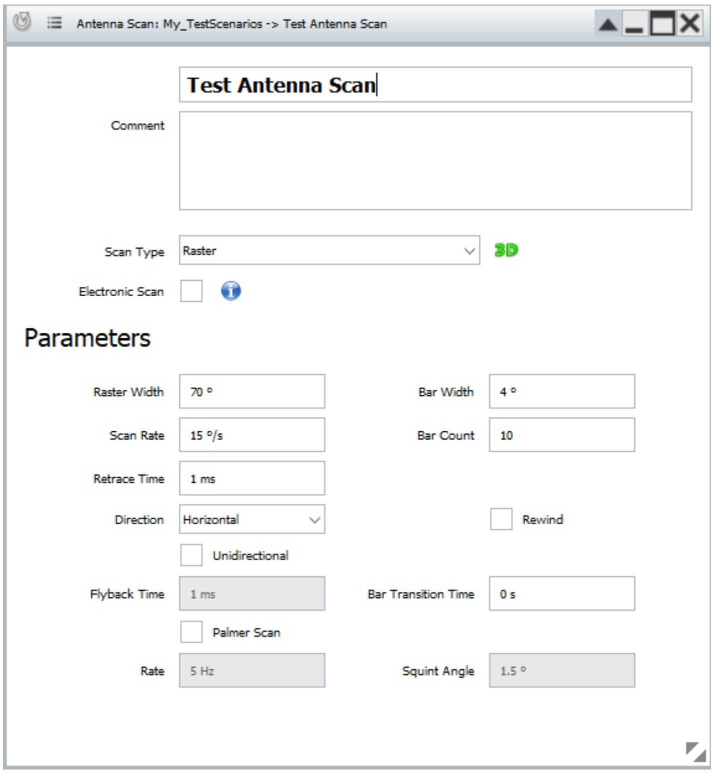
Figure 9-6: Import Wizard: understanding the displayed information

1, 2 = Parameter per column and column header information
3, 7 = Decimal separator: point or comma
4, 8 = Column separator: comma, semicolon, withe space
5, 6 = First row

9.3 Antenna scans settings

Access:

► Select "Repository Tree > Antenna Scan > New".



Settings:

Antenna Scan Name

Enter the name of the antenna scan.

Remote command:
SCAN:NAME on page 427
SCAN:CATalog? on page 426
SCAN:SELect on page 427
SCAN:CREate on page 426
SCAN:REMove on page 428

Comment

Enter a short description.

Remote command:
SCAN:COMMent on page 428

Scan Type

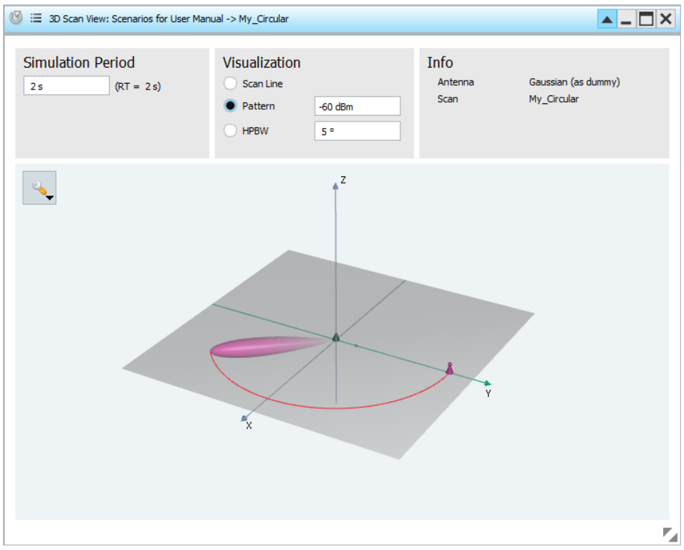
Defines the scan type.

Use the "3D Scan View" diagram to visualize the selected scan.

Remote command:
SCAN:TYPE on page 458

3D Scan View

To access the "3D Scan View" open any "Antenna Scan" or "Emitter Type" and click the "Display 3D antenna scan" button.



This window contains control and information elements at the top of the window. In the top left corner of the pattern visualization area there is a "Transparency settings" icon that enables you to change the way the scan is displayed. This icon is also available in the "3D Single Emitter Preview" window.

Electronic Scan
Defines how the scan works.

If this setting is activated and an emitter uses this scan together with a phased array antenna, electronic steering is applied.

In a normal scan, the antenna is mounted on a mechanical motor that performs the scan. In electronic scan mode, the antenna is fixed and the scan is performed electronically. The antenna does not move. The pattern is aimed towards the desired location by applying phase offsets to the individual antenna elements.

Remote command:
`SCAN:STEering` on page 464

3D Scan View
Visualizes the antenna scan in a 3D preview.

See the example on Figure 9-14.

Simulation Period ← 3D Scan View
Sets the time it takes the animation to complete a scan.

The "Real Time" is also displayed.

Visualization ← 3D Scan View
Defines the way the scan is represented:
- "Scan Line = On": the scan is represented by a line; the antenna pattern is disregarded.
- "Pattern Visualization > Pattern": the visualization uses a Gaussian pattern.
  If you have opened this dialog form the "Emitter" dialog or the 2D map, the visualization corresponds to the current selected antenna pattern.
- "Pattern Visualization > HPBW": the antenna is represented by its HPBW.
- "Minimum Displayed Level": the minimum displayed side lobes level.

Info
Summary information.

Circular Scan
A circular antenna turns in a circle and scans 360°.

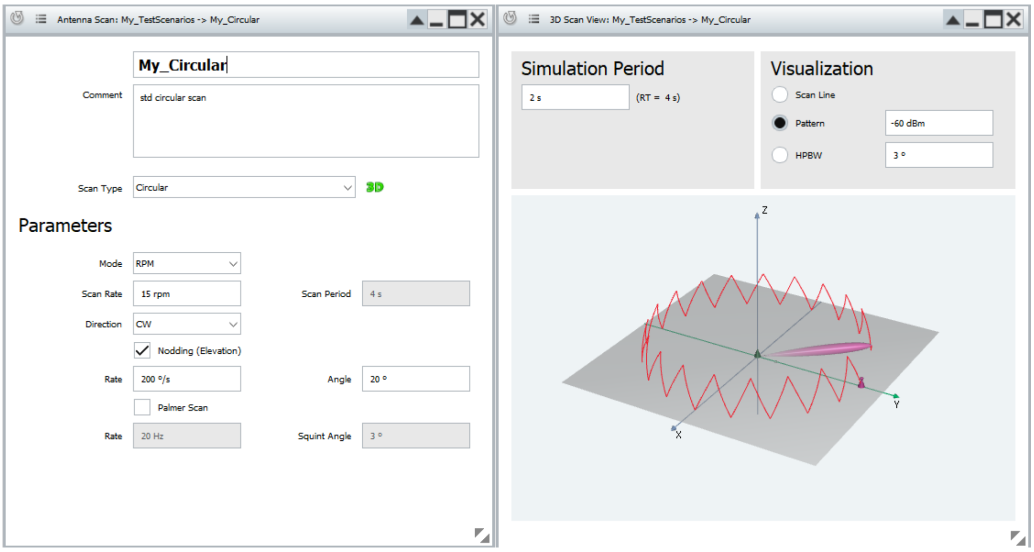


Figure 9-7: Circular scan with enabled nodding

"Mode" Sets if the scan turning speed is set as a scans rate or as a period. The value "Scan Rate" and "Scan Period" are interdependent and updated automatically whenever one of the two is changed.

"Scan Rate" Sets speed the antenna is turning with.

"Scan Period" Sets the time it takes for the antenna to turn once.

"Direction" Sets the direction, clockwise ("CW") or counterclockwise ("CCW"), the antenna is turning in.

"Nodding (Elevation)" If enabled, superimposes a horizontal nodding on the circular scan. The height of the nods and their rise time are defined with the parameters "Angle" and "Rate", see Figure 9-7.

"Palmer Scan" If enabled, superimposes a conical scan on the circular scan, see Figure 9-8.

"Rate" Enabled if "Palmer Scan > On".
Defines how fast the palmer scan is turning.

"Squint Angle"
Enabled if "Palmer Scan > On".
Sets the radius of the scanned circle.

Remote command:
`SCAN:CIRCular:MODE` on page 453
`SCAN:CIRCular:RPM` on page 454
`SCAN:CIRCular:PERiod` on page 453
`SCAN:CIRCular:ROTation` on page 453
`SCAN:CIRCular:NODDing` on page 459
`SCAN:CIRCular:NRATe` on page 459
`SCAN:CIRCular:NELevation` on page 459
`SCAN:CIRCular:PALMer` on page 459
`SCAN:CIRCular:PRATe` on page 460
`SCAN:CIRCular:PSQuint` on page 460

Sector Scan

A sector antenna scans only a sector with defined "Sector Width".

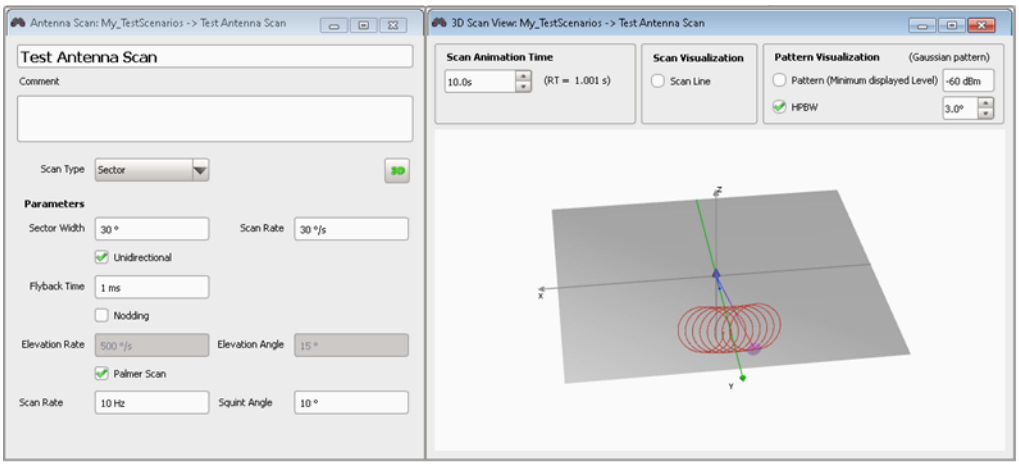


Figure 9-8: Sector scan with enabled Palmer scan

"Sector Width, deg"
: Sets the width of the scanned sector.

"Scan Rate, deg/s"
: Defines how fast the antenna is turning.

"Unidirectional"
: The antenna uses a unidirectional scan mode, i.e. the antenna turns in one direction only and "jumps" back within the selected "Flyback Time".

"Nodding"
: If enabled, superimposes a horizontal nodding on the circular scan. The height of the nods and their rise time are defined with the parameters "Elevation Angle" and "Elevation Rate", see Figure 9-7.

"Palmer Scan"
: If enabled, superimposes a conical scan on the circular scan, see Figure 9-8.

"Rate"
: Enabled if "Palmer Scan > On".
Defines how fast the palmer scan is turning.

"Squint Angle"
: Enabled if "Palmer Scan > On".
Sets the radius of the scanned circle.

Remote command:
`SCAN:SECTor:WIDTh` on page 457
`SCAN:SECTor:RATE` on page 454
`SCAN:SECTor:UNIDirection` on page 457
`SCAN:SECTor:FLYBack` on page 456
`SCAN:SECTor:NODDing` on page 459
`SCAN:SECTor:NELevation` on page 459
`SCAN:SECTor:NRATe` on page 459
`SCAN:SECTor:PALMer` on page 459
`SCAN:SECTor:PRATe` on page 460
`SCAN:SECTor:PSQuint` on page 460

Raster Scan

The raster antenna scans a sector with defined "Sector Width" but performs several scans with different elevations.

See Figure 9-14.

"Raster Width, deg"
Sets the width of the scanned sector.
Scanned is an angle of ±"Raster Width"/2 degrees, starting from the Y axis.

"Scan Rate, deg/s"
Defines how fast the antenna is turning.

"Bar Count, Bar Width"
Sets the number of bars and the distance between two consecutive scanned bars (sectors).

"Retrace Time" Defines how fast the antenna returns to its initial orientation after completing the last bar.

"Direction" Switches between a horizontal and a vertical scanning direction.

"Rewind" Continuous scan in that upon reaching the end of the last bar, the antenna scans backwards.

"Unidirectional" The antenna uses a unidirectional scan mode. The scan is not a continuous but the antenna turns in one direction only (counterclockwise, CCW) and "jumps" to the next bar within the selected "Flyback Time".

"Palmer Scan" If enabled, superimposes a conical scan on the circular scan, see Figure 9-8.

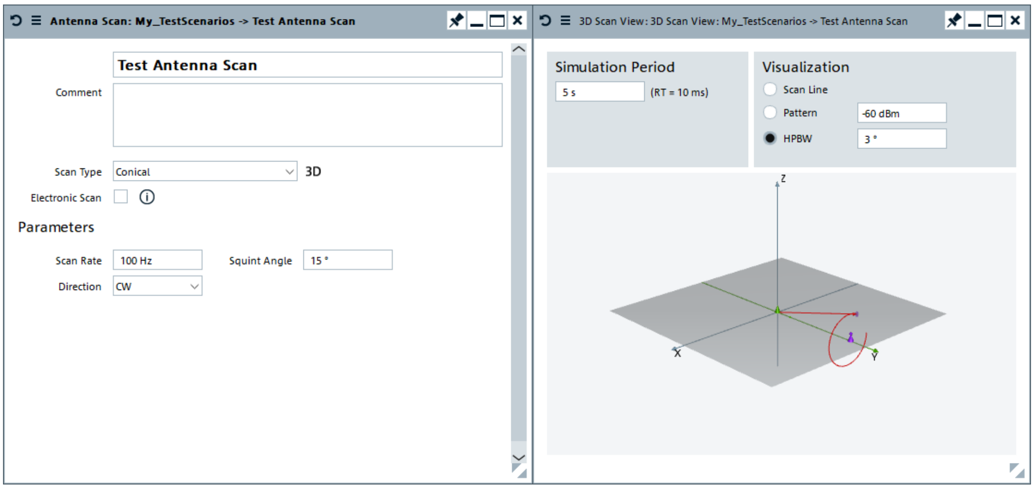
"Rate" Enabled if "Palmer Scan > On".
Defines how fast the palmer scan is turning.

"Squint Angle"
Enabled if "Palmer Scan > On".
Sets the radius of the scanned circle.

Remote command:
`SCAN:RASTer:WIDTh` on page 457
`SCAN:RASTer:RATE` on page 454
`SCAN:RASTer:BARWidth` on page 456
`SCAN:RASTer:BARS` on page 455
`SCAN:RASTer:RETRace` on page 457
`SCAN:RASTer:DIRection` on page 457
`SCAN:RASTer:REWind` on page 456
`SCAN:RASTer:UNIDirection` on page 457
`SCAN:RASTer:FLYBack` on page 456
`SCAN:RASTer:PALMer` on page 459
`SCAN:RASTer:PRATe` on page 460
`SCAN:RASTer:PSQuint` on page 460

Conical Scan

A conical antenna scans a conus with defined "Squint Angle" (radius); the radar beam is rotated around the y-axis (i.e. boresight).



"Scan Rate, Hz"
Defines how fast the antenna is turning.

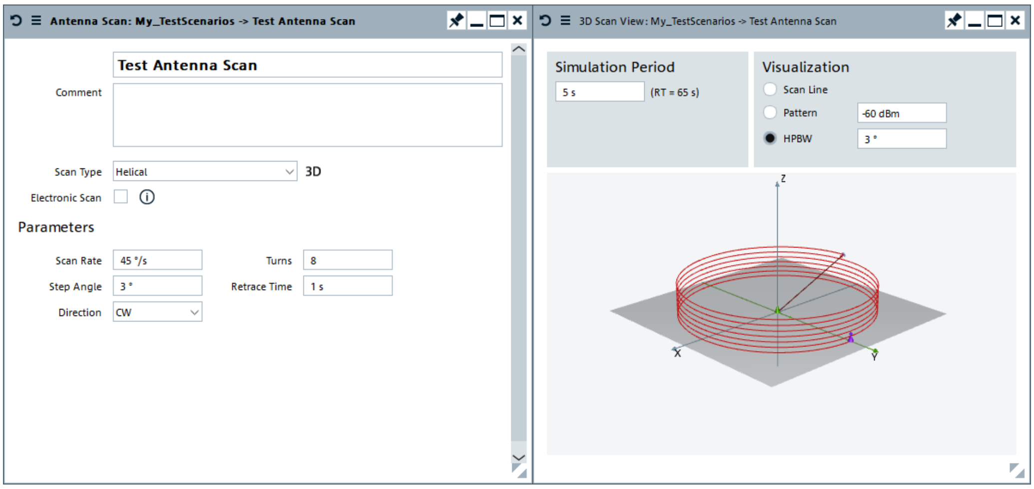
"Squint Angle, deg"
Sets the radius of the scanned circle.

"Direction"
Sets the direction, clockwise ("CW") or counterclockwise ("CCW"), the antenna is turning in.

Remote command:
`SCAN:CONical:RATE` on page 454
`SCAN:CONical:ROTation` on page 453
`SCAN:CONical:SQUint` on page 455

Helical Scan

A helical antenna scan turns in a circle (scans 360°) and performs several scans with different elevations.



"Scan Rate, deg/s"
Defines how fast the antenna is turning.

"Step Angle, deg"
Sets the step width with that the antenna azimuth changes.

"Turns" Sets the number of 360° turns the antenna performs.

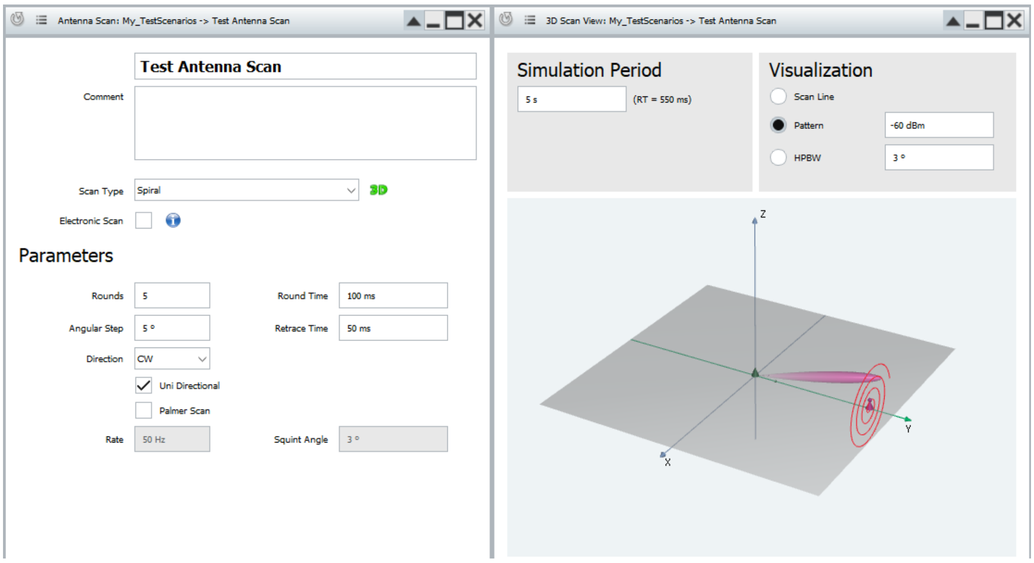
"Retrace Time" Defines how fast the antenna returns to its initial orientation after completing the last turn.

"Direction" Sets the direction, clockwise ("CW") or counterclockwise ("CCW"), the antenna is turning in.

Remote command:
`SCAN:HELical:RPM` on page 454
`SCAN:HELical:ELEVation:STEP` on page 454
`SCAN:HELical:TURNs` on page 454
`SCAN:HELical:RETRace` on page 457
`SCAN:HELical:ROTation` on page 453

Spiral Scan

A spiral antenna scan turns in a circle (scans 360°) and performs several scans with different radii.



"Rounds" Sets the number of rounds the antenna performs.

"Angular Step" Sets the distance in degrees with that the scan radius is increased.

"Round Time" Defines how fast the antenna is turning.

"Retrace Time" Define how fast the antenna returns to its initial orientation after completing the last turn.

"Direction" Sets the direction, clockwise ("CW") or counterclockwise ("CCW"), the antenna is turning in.

"Uni Directional"

Sets the scan behavior when the scan reaches the last round:

- "Uni Directional = On" - the scan is not continuous.
  It turns for the specified number of rounds and then takes the retrace time to return to the origin of the spiral scan.
- "Uni Directional = Off" - the scan is continuous; it turns forth and backwards.
  When the specified number of rounds is reached, the scan direction turns backwards back to the origin of the spiral scan.

"Palmer Scan" If enabled, superimposes a conical scan on the circular scan, see Figure 9-8.

"Rate", "Squint Angle" Enabled if "Palmer Scan > On".
Defines how fast the palmer scan is turning.

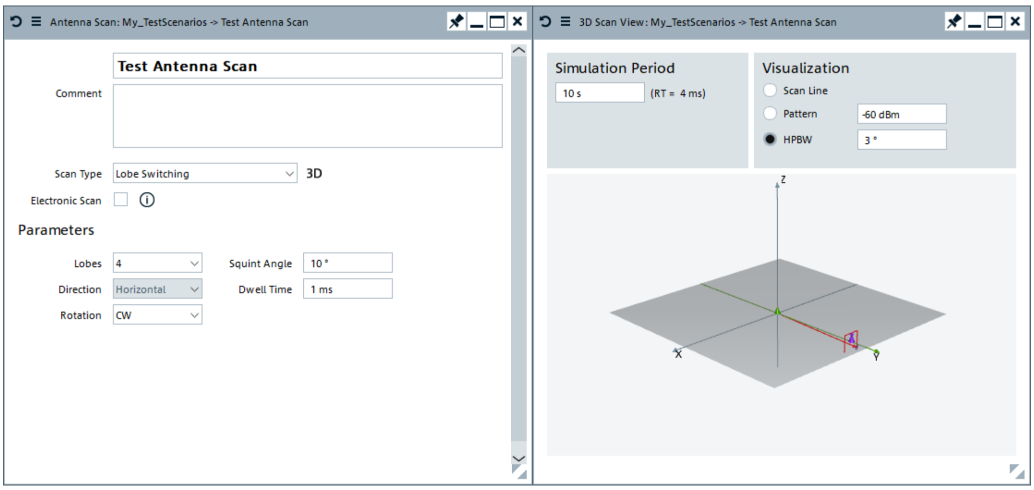
"Squint Angle"

Enabled if "Palmer Scan > On".
Sets the radius of the scanned circle.

Remote command:
`SCAN:SPIRal:ROUNds` on page 458
`SCAN:SPIRal:STEP` on page 458
`SCAN:SPIRal:RTIMe` on page 458
`SCAN:SPIRal:RETRace` on page 457
`SCAN:SPIRal:ROTation` on page 453
`SCAN:SPIRal:UNIDirection` on page 457
`SCAN:SPIRal:PALMer` on page 459
`SCAN:SPIRal:PRATe` on page 460
`SCAN:SPIRal:PSQuint` on page 460

Lobe Switching Scan

A lobe switching antenna uses two or four slightly separated antenna elements.



"Lobes" Set the number of lobes.

"Direction" Sets the switching direction, i.e. horizontal or vertical.

"Squint Angle" Sets the angle that the beam axis is offset from the tracking axis (i.e. the y-axis) of the antenna.

"Dwell Time" Sets the antenna motion, i.e. how fast the switches between the lobes.

"Direction" Sets the direction, clockwise ("CW") or counterclockwise ("CCW"), the antenna is turning in.

Remote command:
`SCAN:LSW:LOBes` on page 455
`SCAN:LSW:DIRection` on page 455
`SCAN:LSW:SQUint` on page 455
`SCAN:LSW:DWELl` on page 455
`SCAN:LSW:ROTation` on page 453

Sine Scan

A sine antenna scan draws a sine wave.

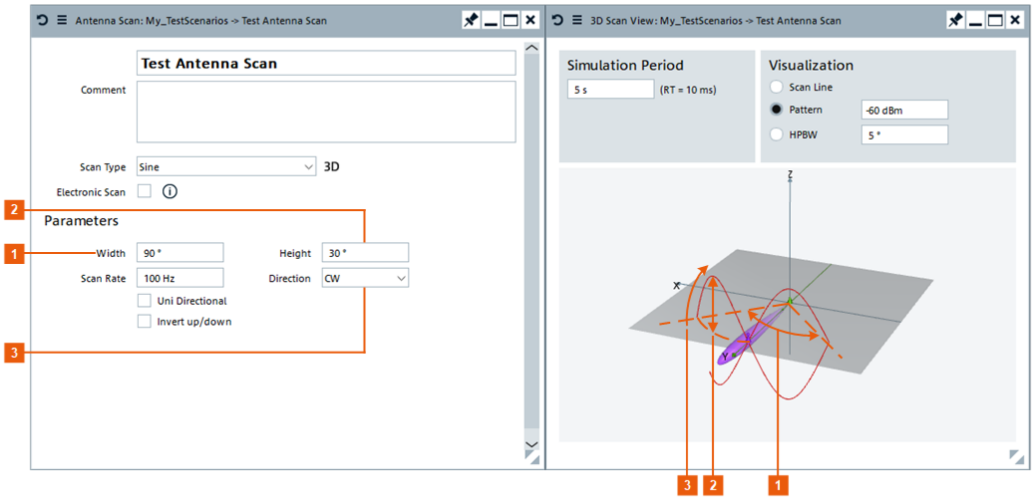


Figure 9-9: Sine scan: understanding the displayed information

1 = Width is the sector width, i.e. the angle on the XY plane between the origin and the end of the scan
2 = Height is the sine amplitude
3 = Seen from the receiver, the scan turns clockwise ("Direction = CW") and draws the upper sine wave first ("Invert up/down scan = Off")

"Width, deg"
Sets the width of the scanned sector. i.e. the angle on the XY plane between the origin and the end of the scan.

"Height, deg"
Sets the amplitude of the sine wave.

"Scan Rate, deg/s"
Defines how fast the antenna is turning.

"Direction" Sets the antenna turning direction, clockwise ("CW") or counterclockwise ("CCW").

"Uni Directional"
Sets the scan behavior when the scan reaches the last point:

- "Uni Directional = On": the scan is not continuous.
  It draws 1 sine wave and then jumps back to the origin of the scan.
- "Uni Directional = Off": the scan is continuous; it draws two symmetrical sine waves
  When the scan reaches the sine end position, it returns back to the origin drawing a sine wave that is the mirrored version of the sine scan.

"Invert up/down scan"
Sets the sine scan order and defines whether the upper or the down (mirrored) sine scan is used first.

Remote command:
`SCAN:SIN:WIDTh` on page 463
`SCAN:SIN:HEIGht` on page 463
`SCAN:SIN:RATE` on page 454
`SCAN:SIN:ROTation` on page 453
`SCAN:SIN:UNIDirection` on page 457
`SCAN:SIN:INVersion` on page 464

Custom Scan

Describes the antenna scan as a sequence of discrete scan positions.

Each position is defined by an elevation angle and an azimuth angle.

The trajectory between two subsequent positions is interpolated.

Figure 9-10 shows a simple custom scan defined using 4 beam positions.

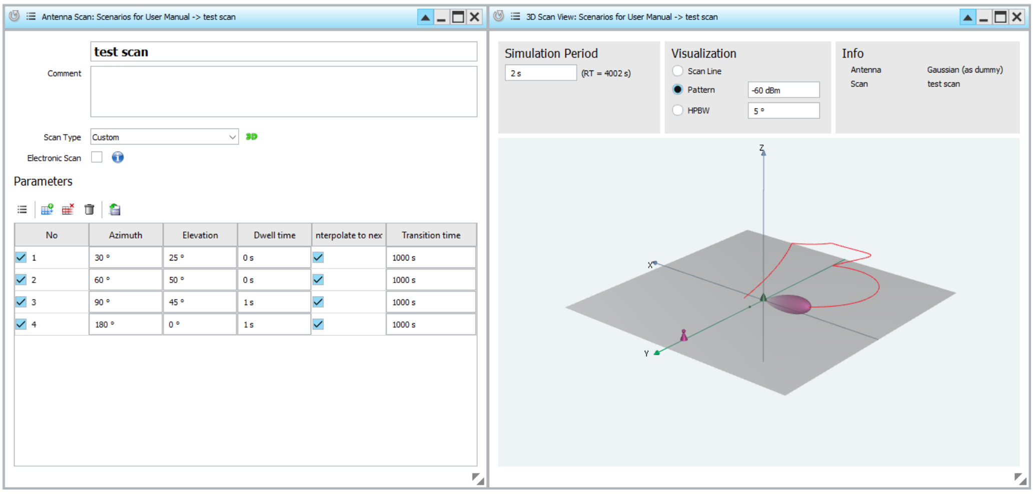


Figure 9-10: Custom scan defined by four positions

"New, Select, Append, Remove, Clear Items"
Standard functions for item handling, see Table 2-4.

"Import data from file"

Opens a dialog that enables you to import a custom scan definition from a CSV file (see Figure 9-10).

The dialog previews the data in the file and enables you to ignore the headers (if present) and map the columns in the file to the columns in the "Parameters" table.

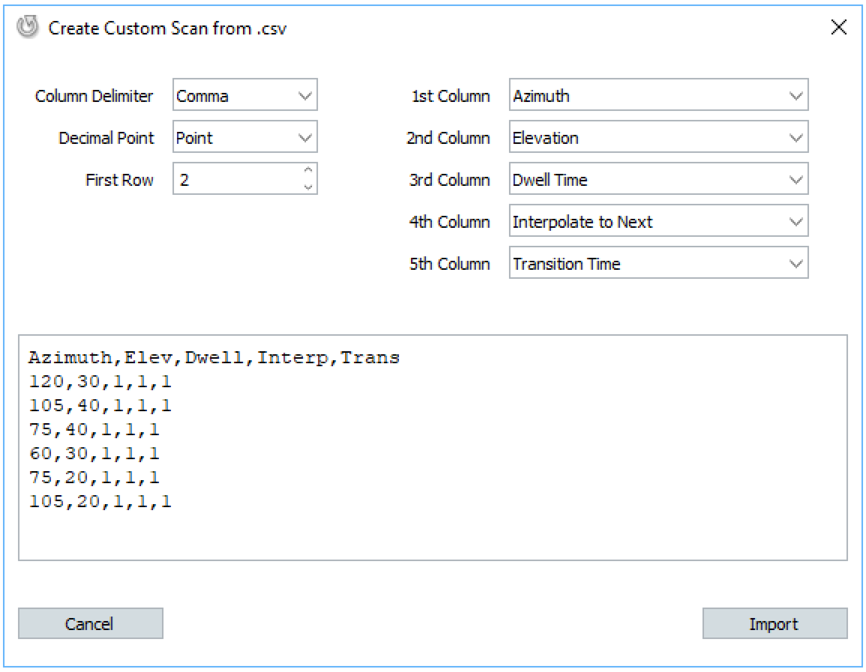


Figure 9-11: Importing custom scan parameters

Positions table

Each row of the table contains parameters that define a position and the behavior of the custom scan. The scan moves sequentially down the table. The scan direction depends on the "Azimuth" (value and sign) and "Elevation" (value and sign) parameters. The scan behavior depends on the other parameters.

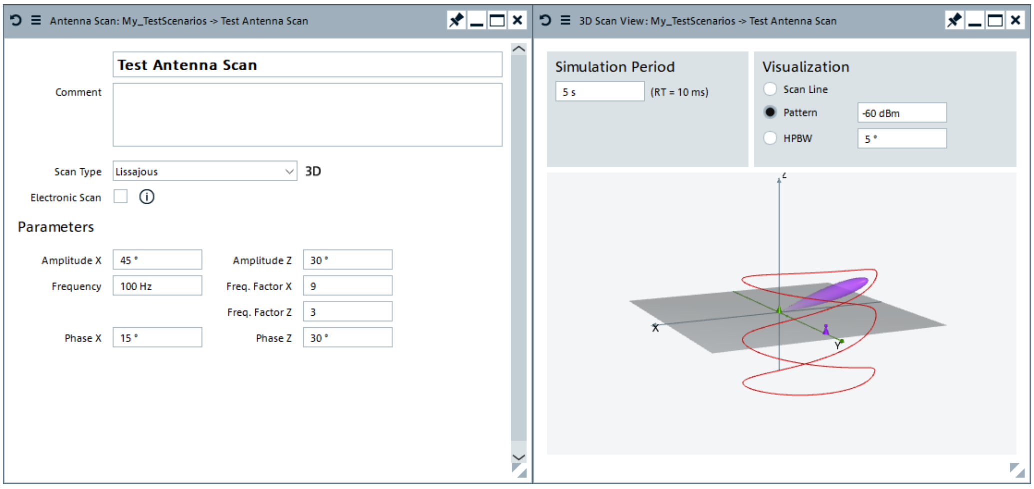
Each row consists of the following parameters:
- "Azimuth" (angle offset from the y-axis)
- "Elevation" (angle offset from horizontal)
- "Dwell time" defines how long the beam remains at a particular position.
- "Interpolate to next" defines whether the beam jumps or moves smoothly from the current position to the next position.
  – Selected = smooth movement
  – Deselected = jump
- "Transition time" defines the time over which the beam movement is interpolated.

Note: When the scan repeats, the beam position jumps from the last position to the first position.
If a jump is the correct behavior, do not select "Interpolate to next" for the last position (e.g. like the flyback at the end of a helical scan).

If the correct behavior is a smooth transition (e.g. the custom scan defines a closed shape), select "Interpolate to next" for the last position so that the scan repeats without a jump.

Remote command:
`SCAN:CUSTom:ENTRy:ADD` on page 429
`SCAN:CUSTom:ENTRy:INSert` on page 432
`SCAN:CUSTom:ENTRy:COUNt?` on page 430
`SCAN:CUSTom:ENTRy:SELect` on page 431
`SCAN:CUSTom:ENTRy:DELete` on page 432
`SCAN:CUSTom:ENTRy:CLEar` on page 433
`SCAN:CUSTom:IMPort:FILE` on page 460
`SCAN:CUSTom:IMPort:EXEC` on page 460
`SCAN:CUSTom:ENTRy:AZIMuth` on page 460
`SCAN:CUSTom:ENTRy:ELEVation` on page 461
`SCAN:CUSTom:ENTRy:DWELl` on page 461
`SCAN:CUSTom:ENTRy:JUMPtype` on page 461
`SCAN:CUSTom:ENTRy:TRANstime` on page 462

Lissajous
The antenna scan follows a Lissajous curve.



Lissajous curves are mathematically described as follows:

- $x(t) = A*cos(\omega_x*t - \sigma_x)$
- $z(t) = B*cos(\omega_z*t - \sigma_z)$

"Amplitude X, Amplitude Z"
Magnitudes A and B of two harmonic vibrations.

"Frequency" Base frequency

"Freq X : Freq Z"
The two angular frequencies ω_x and ω_z are defined by their ratios and calculated as follows:
ω_x = "Freq X"*"Frequency"
ω_z = "Freq Z"*"Frequency"
The ratio "Freq X : Freq Z" determines the number of lobes in the curve.

"Phase X, Phase Z"
Phases σ_x and σ_z of the two harmonic vibrations.

Remote command:
`SCAN:LISSajous:AMPX` on page 462
`SCAN:LISSajous:AMPZ` on page 462
`SCAN:LISSajous:FREQ` on page 462
`SCAN:LISSajous:XFACtor` on page 463
`SCAN:LISSajous:ZFACtor` on page 463
`SCAN:LISSajous:PHIX` on page 463
`SCAN:LISSajous:PHIZ` on page 463

9.4 How to create a library with antenna patterns and scans

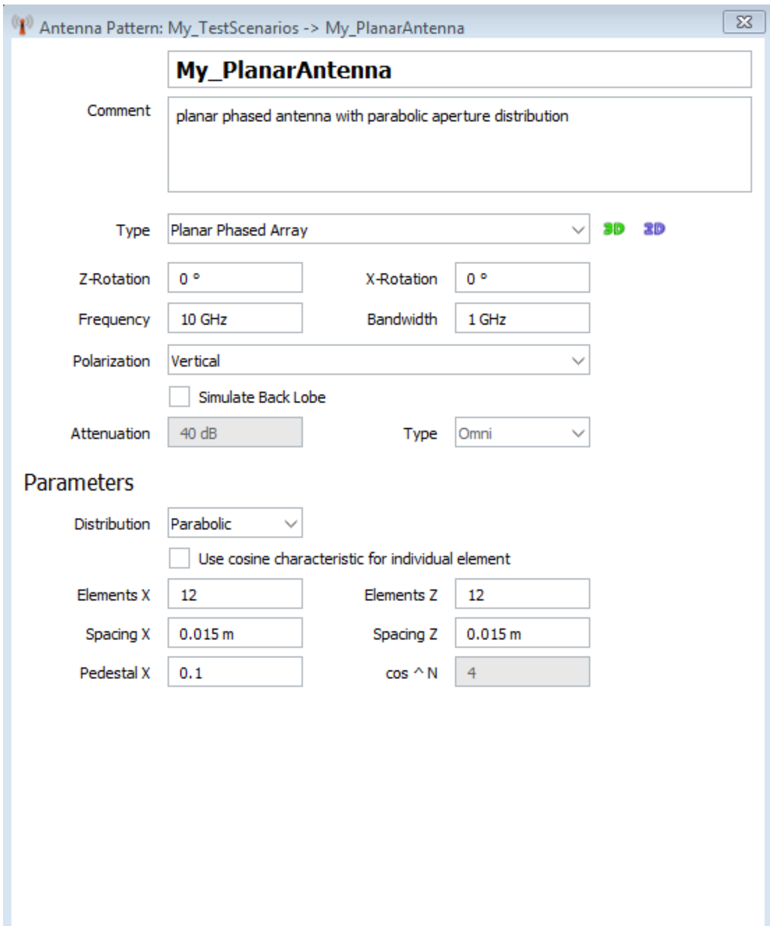
See:

- "To create an antenna pattern" on page 207

- "To create an antenna scan" on page 210
- "To create a custom antenna pattern based on the required antenna characteristics" on page 211
- "To import a user-defined antenna pattern from file in one of the predefined file formats" on page 212
- "To import an antenna pattern file in custom file format" on page 215
- "To observe the effect of polarization" on page 215

To create an antenna pattern

1. Select "Repository Tree > Antenna Pattern > New" .
2. Select "Antenna Model", e.g. "Planar Phased Array".



3. Adjust further settings, like "Rotation", "Frequency" or "Aperture Distribution".
4. Select "2D" to display the antenna pattern in a graph in polar or with Cartesian coordinates.

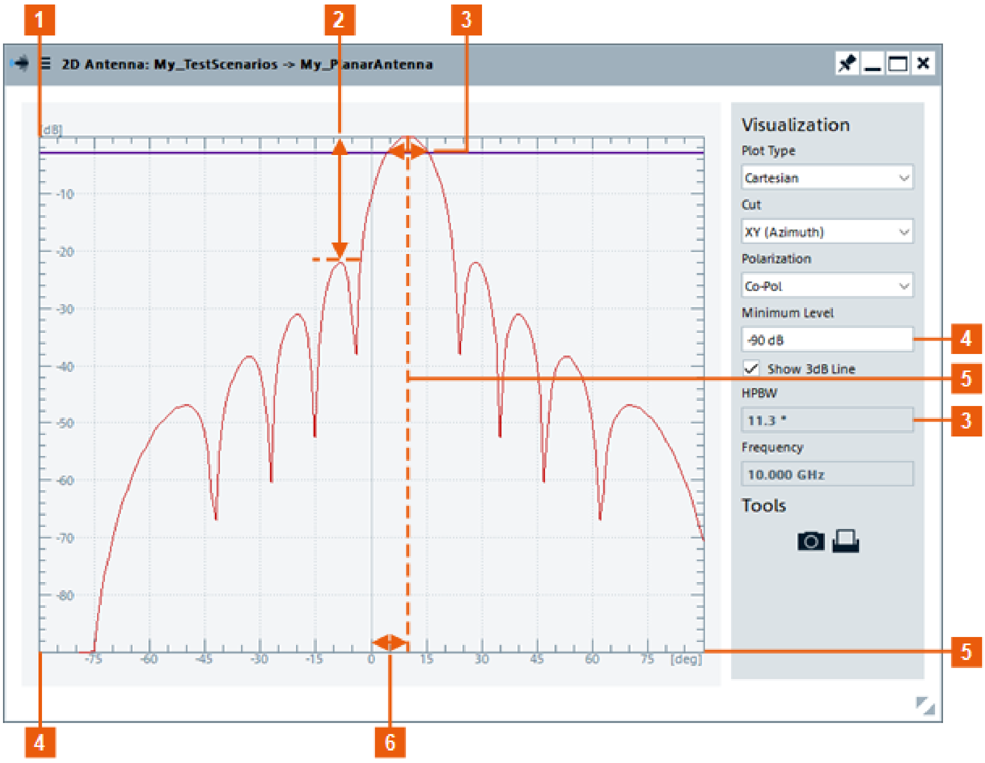


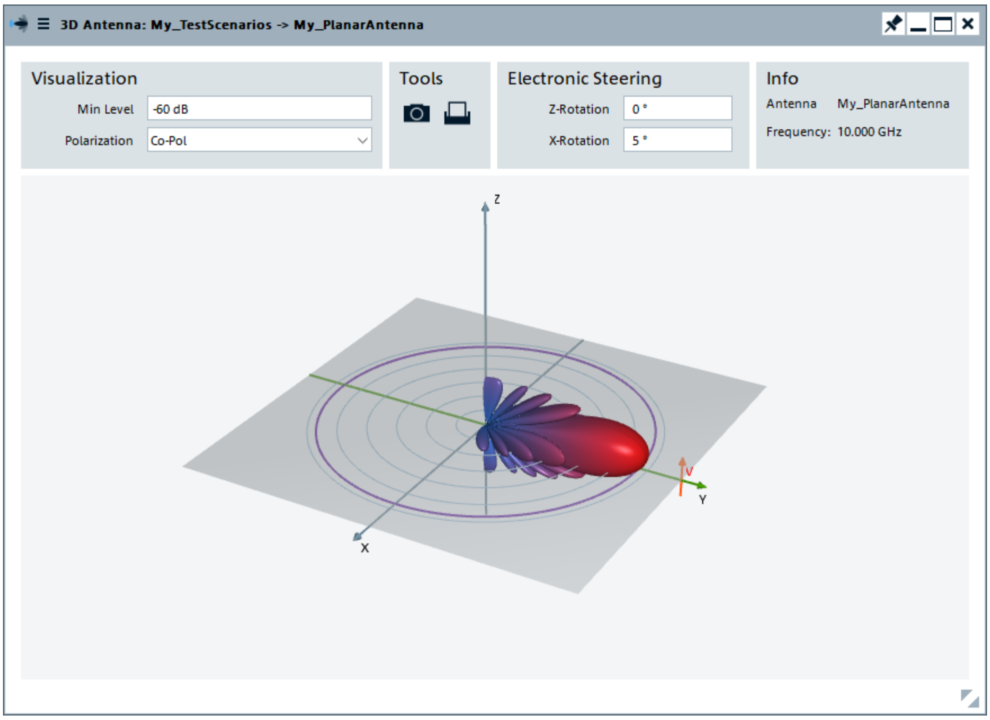
Figure 9-12: 2D antenna pattern: understanding the displayed information

1, 5 = Antenna pattern diagram, as a function of angle (deg) and antenna gain (dB)
2 = Side lobe level
3 = HPBW, i.e. the beamwidth at half power (3 dB)
4 = Antenna gain minimum (-90 dB)
5 = Beam axis
6 = "Z-Rotation = 10°"

5. Change the number of used elements ,e.g. "Elements X = 20". Compare the 2D diagram.

   More antenna elements result in a smaller beamwidth and side lobes with lower level.

6. Select "Antenna Pattern > 3D" to display the 3D antenna pattern.



The 3D view is interactive. Use the mouse cursor to turn it around any of the axes and the mouse wheel to zoom in the view.

7. Select "Antenna Pattern > Simulate Back Lobe > On" and "Attenuation = 0 dB".
8. In the 2D view, select "Polar" coordinates.

   The display confirms the antenna pattern with a back lobe, where the back lobe is simulated as the mirrored pattern of the main lobe ("Type > Mirror").
   See Figure 9-13.
9. Select "Antenna Pattern > Back Lobe Attenuation = 40 dB".

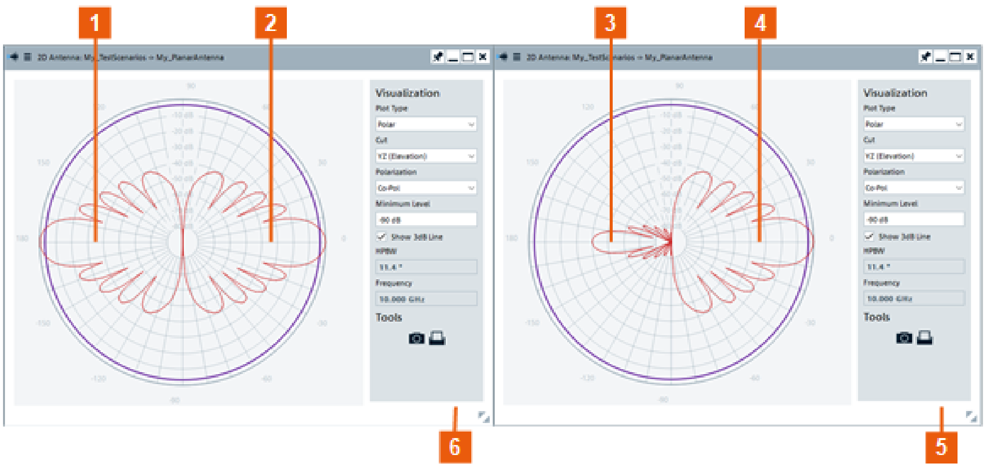


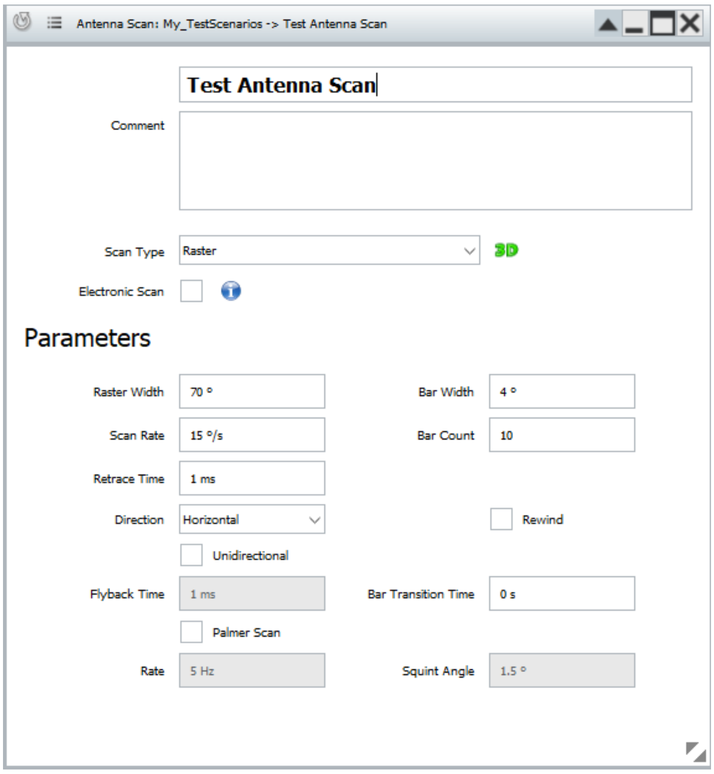
Figure 9-13: Simulate back lobe: understanding the displayed information

1 = Unattenuated back lobe, i.e. the mirrored pattern of the main lobe
2, 4 = Main lobe
3 = Attenuated back lobe
5 = "Attenuation = 40 dB"
6 = "Attenuation = 0 dB"

For description of the provided settings, see Chapter 9.2, "Antenna pattern settings", on page 174.

To create an antenna scan

1. Select "Repository Tree > Antenna Scans > New".
2. Select "Scan Type", e.g. "Raster Scan".



3. Adjust further settings, like "Raster Width" and "Bar Width".
4. Select "3D" to display the 3D antenna pattern.
5. Use the mouse cursor to turn it around any of the axes and the mouse wheel to zoom in the view.

   The antenna scan confirms the settings.

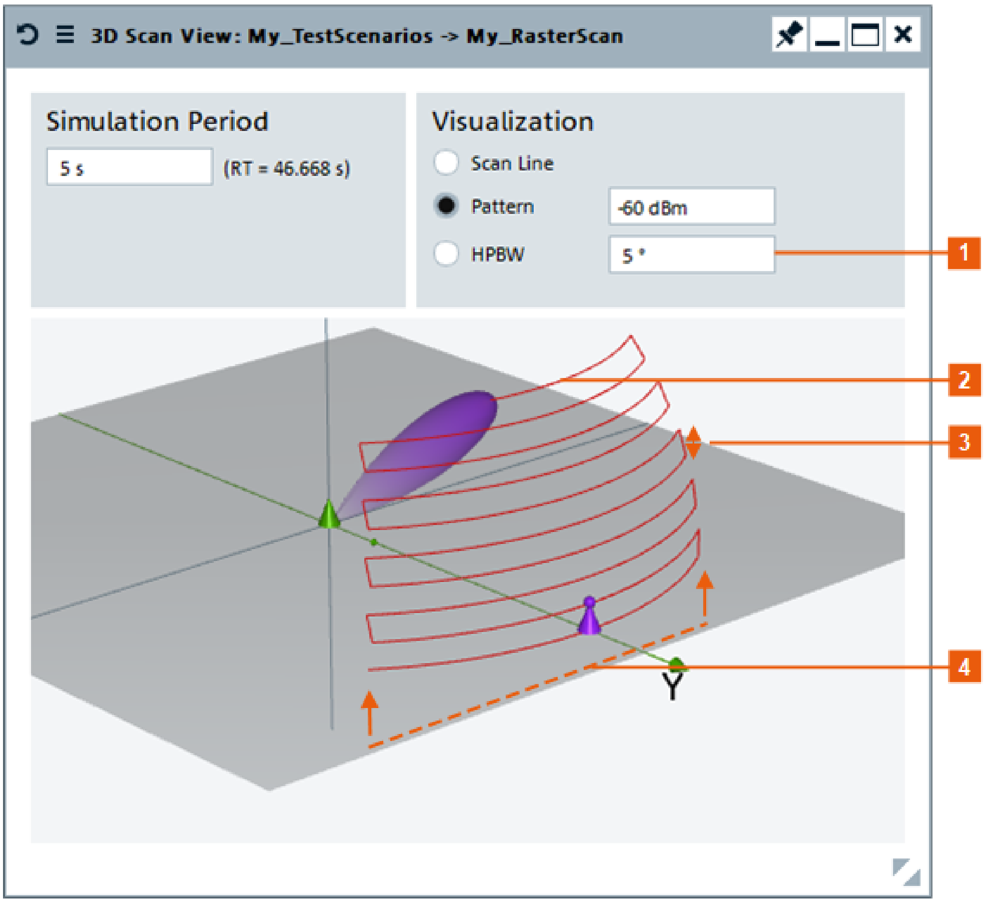


Figure 9-14: Horizontal raster scan: understanding the displayed information

1 = Antenna pattern visualized as "HPBW = 5°"
2 = Tenth scan as defined with "Bar Count"
3 = "Bar Width = 4°"
4 = "Raster Width"

For description of the provided settings, see Chapter 9.3, "Antenna scans settings", on page 193.

To create a custom antenna pattern based on the required antenna characteristics

1. Create an "Antenna Pattern".
   See "To create an antenna pattern" on page 207.
2. Select "Antenna Model > Custom".
3. Adjust the settings, e.g. set the "HPBW XY = 10°", "HPBW YZ = 3°", etc.
4. Observe the antenna pattern on the 2D and 3D views.

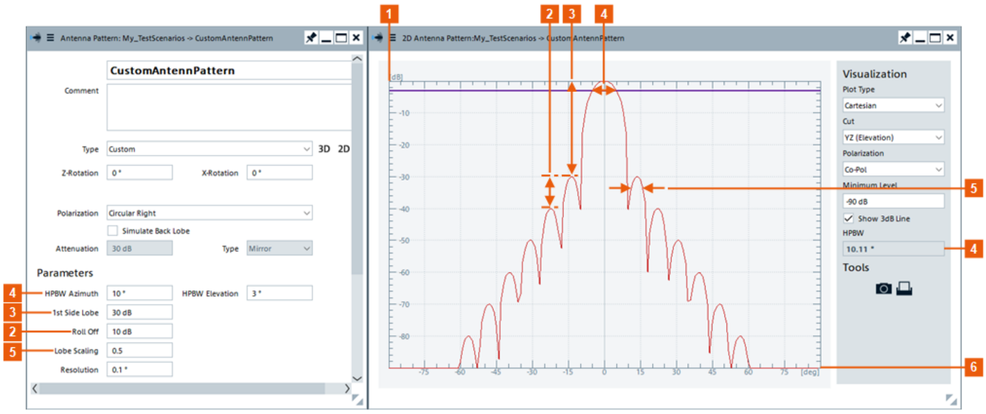


Figure 9-15: Custom antenna pattern: understanding the displayed information

1, 6 = Antenna pattern diagram, as a function of angle (deg) and antenna gain (dB)
2 = Roll-off factor or step size, used to calculate the attenuation of the subsequent side lobes
3 = Side lobe level, i.e. the attenuation of the first pair of side lobes
4 = "HPBW XY = 10°", i.e. the main beamwidth at half power (3 dB)
5 = Scale factor to calculate the HPBW of the side lobes, i.e. how much smaller or wider the side lobes are, compared to the main lobe

To import a user-defined antenna pattern from file in one of the predefined file formats

You can import custom antenna pattern files of different file types.

1. Create an "Antenna Pattern".
2. Select "Antenna Model > Import from File".
3. Select "Import Filter > internal".
4. Select "Load". Navigate to the folder with your custom antenna patterns. Load a `*.ffe`, a `*.tsv`, a `*.ffd`, a `*.csv`, or a `*.ant_pat` file.
5. Open the 2D view to visualize the antenna pattern.

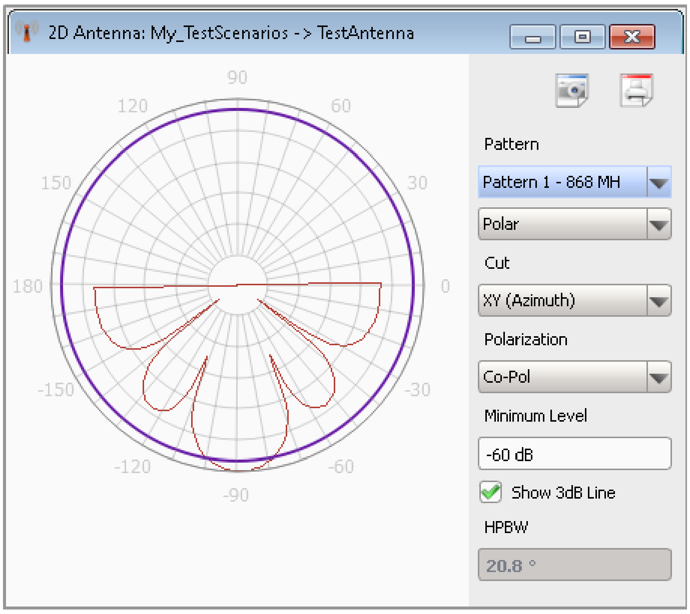
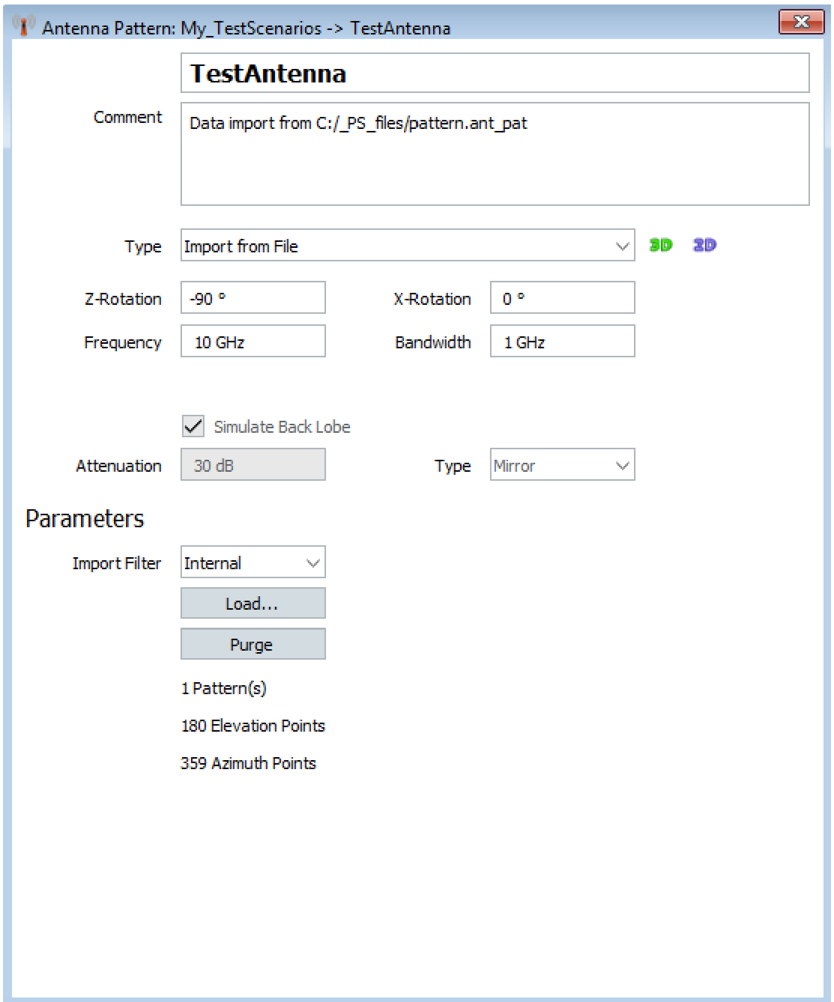


Figure 9-16: 2D view: Example of a user-defined antenna pattern

In this example, the antenna pattern is directed to -90°.

6. Select "Antenna Pattern > Z-Rotation = 90°".


Antenna Pattern: My_TestScenarios -> TestAntenna
TestAntenna
Comment
Data import from C:/_PS_files/pattern.ant_pat
Type
Import from File
3D
2D
Z-Rotation
-90 °
X-Rotation
0 °
Frequency
10 GHz
Bandwidth
1 GHz
Simulate Back Lobe
Attenuation
30 dB
Type
Mirror
Parameters
Import Filter
Internal
Load...
Purge
1 Pattern(s)
180 Elevation Points
359 Azimuth Points

7. On the 2D and 3D views, observe the effect of this parameter.

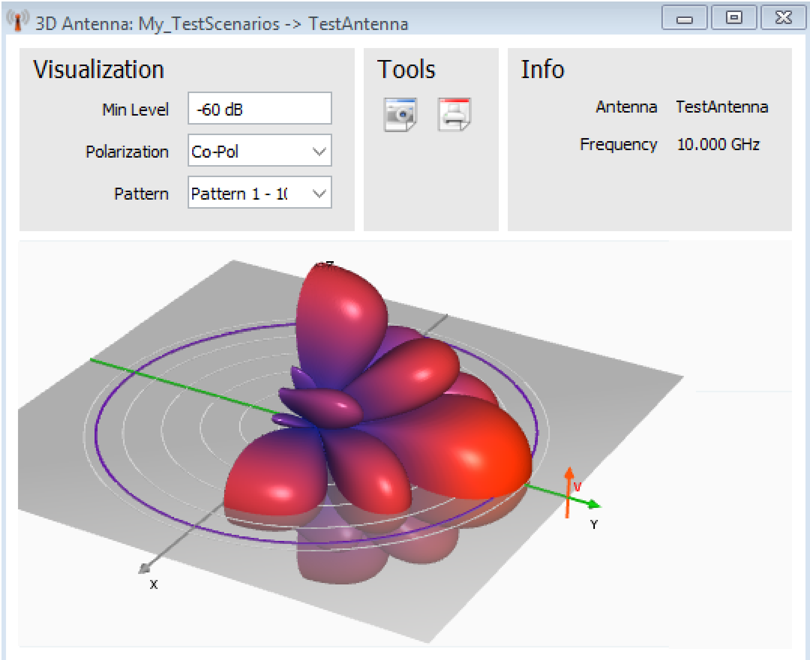


Figure 9-17: 3D view: Example of a user-defined antenna pattern

The antenna pattern is directed on the y-axis.

To import an antenna pattern file in custom file format

Importing antenna patterns in custom file format require that you create a suitable import filter first. That is, that you define and import the file format as a plug-in.

1. Import the plug-in in the repository, for example `My_AntPatternFileFormat`. See "To import a plugin" on page 377.
2. Create an antenna pattern. Select "Model > Import from File".
3. Select "Import Filter > My_AntPatternFileFormat".
4. Select "Load". Navigate to the folder with your custom antenna pattern (`*.txt` file). Select it.
5. Open the 2D and 3D view to visualize the antenna pattern.

   If the file contains more than one antenna patterns, select the pattern to be displayed.

To observe the effect of polarization

1. Create an antenna with a Gaussian beam pattern and a vertical polarization.
2. Open the 3D display. Observe the pattern for "Polarization > Co-Pol" and "Minimum Level = -90 dB".
3. Select "Polarization > Cross-Pol".

The automatically selected cross-polarized antenna is an antenna with *horizontal (H)* polarization.

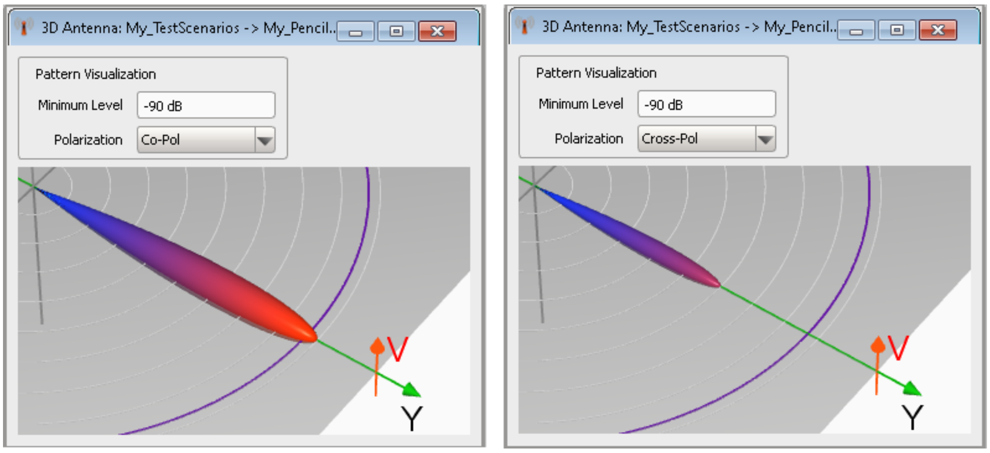


Figure 9-18: Effect of cross-polarization

The left figure shows the Gaussian antenna pattern. The right figure shows the resulting antenna pattern. This pattern has the same shape but the maximum power that is 30 dB lower than the co-polarized case. That is, the applied *polarization loss is -30 dB*.

4. Create a 2D scenario with one emitter and a receiver.
5. Assign the Gaussian beam pattern to the emitter and the receiver.
   To simplify the explanation, this example does not use an antenna scan.

   Note: If the receiver uses an isotropic omnidirectional antenna, the antenna polarization is unused.
6. Align the receiver and the emitter: for the emitter, set "Altitude > Point to Receiver > On" and "Roll = 0°".
7. Open the "3D View with Receiver". Observe the maximum normalized power level at the receiver.

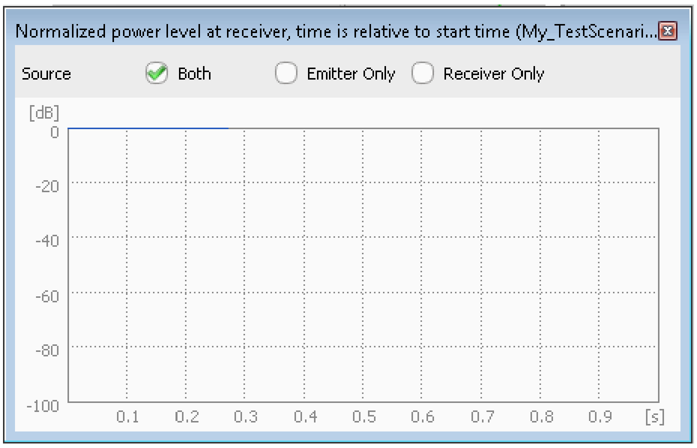
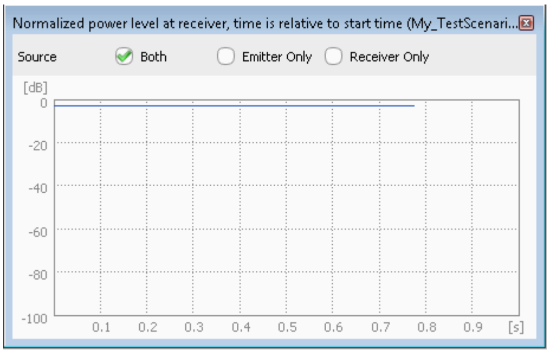
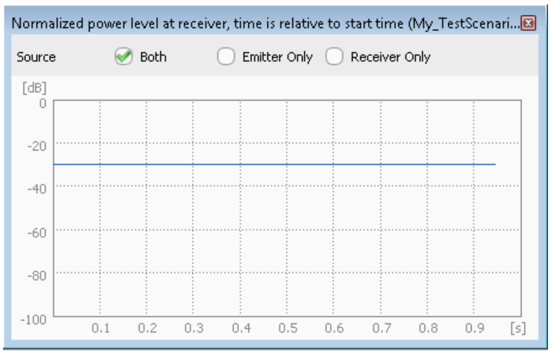


Figure 9-19: Normalized power level at the receiver, if the antennas are co-polarized (Roll = 0°)

The maximum power level is 0 dB.

8. Change the antenna roll angle. Compare the maximum normalized power level at the receiver.

Normalized power level at the receiver for "Roll = 45°"	**Normalized power level at the receiver for "Roll = 90°"**
	
The maximum power level is 3 dB less than the co-polarized case.	The antennas are cross-polarized. The maximum power level is 30 dB less than the co-polarized case.

See also:
- Table 9-2
- Chapter 16.1, "How to create scenarios with receiver and TX items", on page 290

10 Emulating emitters

In the context of this software, a simple emitter describes a pulse sequence whereas a complex emitter emulates a radar system. The emitters are characterized by parameters like operating mode, beam, waveform, antenna pattern, and antenna scan.

The available emitter settings depend on the complexity of the simulated system:

- Simple emitters are described only by a sequence of pulses. Antenna beam patterns and antenna scans are not required
- Complex emitters are described by the combination of an operating mode, antenna pattern, antenna scan and a sequence.

Operational mode

As in the real-word where radar systems can run in different operational modes, in the R&S Pulse Sequencer Digital an emitter can also work in more than one operational mode and switch between them. The operational mode is a description of the radar purpose, such as scanning, searching or tracking. For example, an airport surveillance radar can switch between different scan types to observe its airspace.

Each mode can have individual antenna and signal configuration. Multiple beams are possible within a mode. Moreover, the R&S Pulse Sequencer Digital can emulate emitters that change their mode over time.

Related settings:

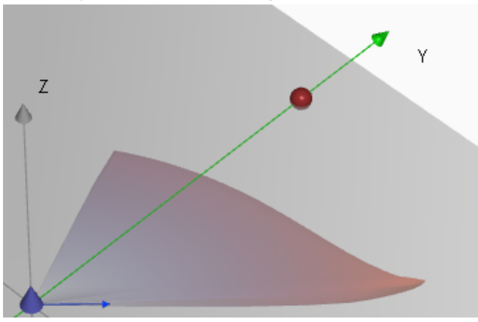
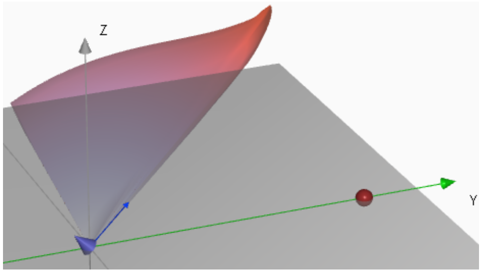
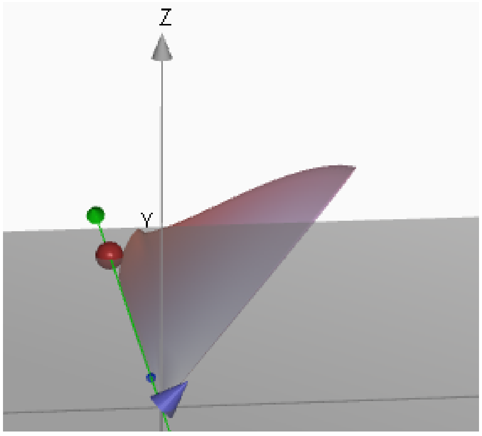
- "Emitter Modes" on page 221
- "Behaviour" on page 312
- Chapter 10.3.3, "Mode editor settings", on page 230

Emitter attitude

The attitude is a description of the emitter orientation. Emitters can be rotated in three dimensions: yaw, pitch and roll.

The examples in Table 10-1 show an emitter with a cosecant antenna pattern in a single emitter scenario. In direction finding scenarios and scenarios with multiple emitters, the effect is the same.

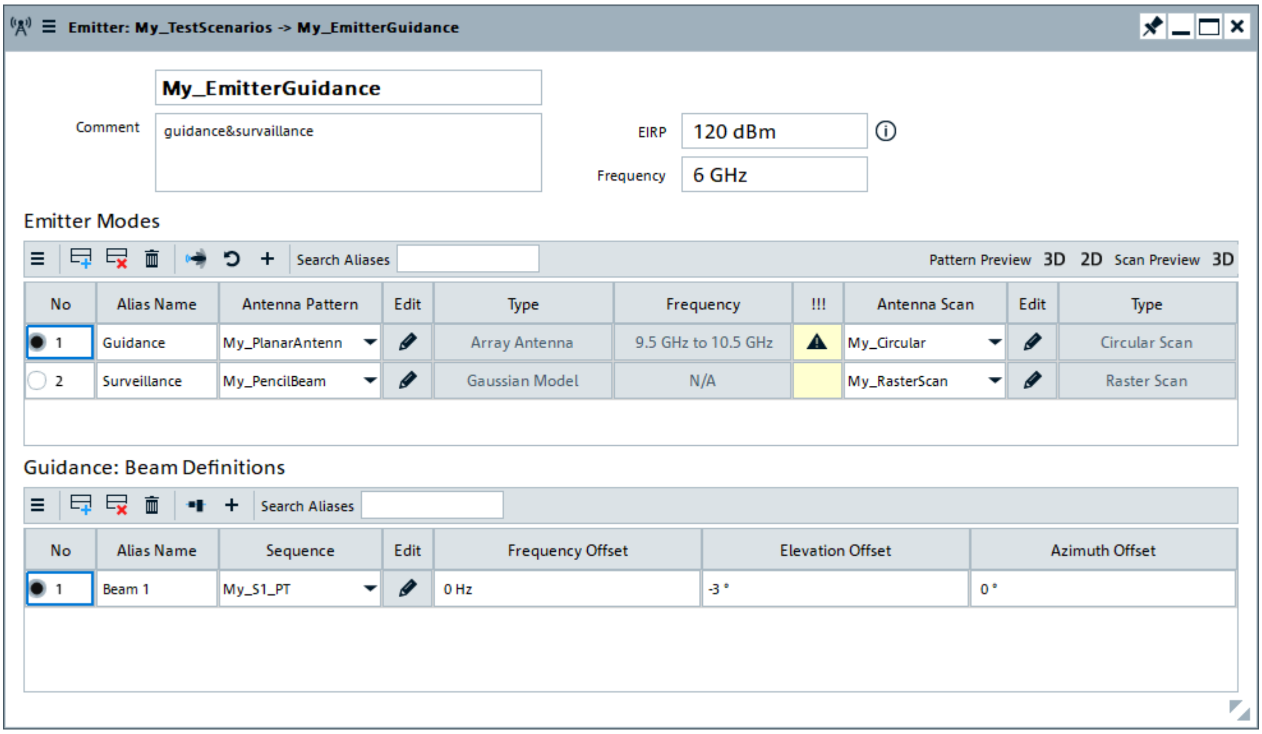
Table 10-1: Impact of the yaw, pitch and roll angles on the emitter attitude

Attitude	Rotation around axis	Rotation
Yaw	Z-axis	In the x/y plane, to the left/right  Yaw = 45° ("Pitch = Roll = 0°")
Pitch	X-axis	In the y/z plane, up/down  Pitch = 45° ("Yaw = Roll = 0°")
Roll	Y-axis	In the x/z plane  Roll = 45° ("Yaw = Pitch = 0°")

10.1 Emitter settings

Access:

► Select "Repository Tree > Emitter > New".



An emitter is described as a combination of an operating mode, an antenna pattern, an antenna scan, and a sequence. Complex emitters support several beams per operating mode.

Enter the settings either in the table or in the dialog fields.

Settings:

Emitter Name

Enter the emitter's name.

Remote command:

`EMITter:NAME` on page 427
`EMITter:CREate` on page 426
`EMITter:SELect` on page 426

EMITter:CATalog? on page 425
EMITter:REMove on page 428

Comment
Enter a short description.

Remote command:
EMITter:COMMent on page 428

EIRP
Sets the equivalent isotopically radiated power (EIRP) of the emitter.

This parameter is used, if the emitter is configured in a "Scenario Type > Localized Emitter".

Example:

For emitter with:
- Transmit power = 1 MW, i.e. 60 dBW
- Antenna gain = 20 dBi

Set "EIRP = 80 dBW".

Remote command:
EMITter:EIRP on page 469

Frequency
Sets the emitter's operating frequency.

If the selected antenna is calculated for a particular frequency, set the emitter's "Frequency" to a value within the suggested frequency range, see "Emitter Modes" on page 221.

Remote command:
EMITter:FREQuency on page 469

Emitter Modes
Defines one or more modes that the emitter works in (see "Operational mode" on page 218).

Use the standard functions in the context menu or in the table to create or rename modes.

Copy a selected line in the table and append it on the existing table via "+".

Search for a defined emitter mode via "Search Alias".

Each emitter mode is defined by the combination of an antenna pattern, antenna scan and at least one beam with assigned pulse sequence.

R&S Pulse Sequencer Digital can emulate radar systems that change their mode over time, see "Behaviour" on page 312.

Use the 3D view to visualize the selected antenna scan and pattern, see:
- "2D and 3D diagrams" on page 178
- "3D Scan View" on page 195

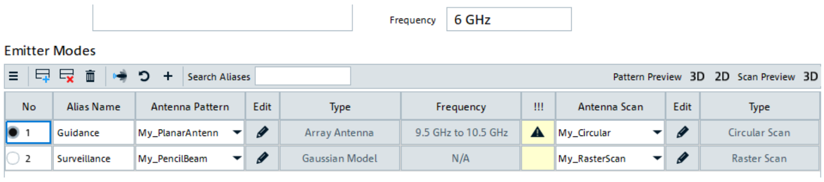
"Antenna Pattern"

Selects an existing antenna pattern or accesses a dialog for creating an antenna pattern.

See "To create an antenna pattern" on page 207.

The antenna pattern and type are displayed.

A warning symbol indicates that the current antennas design frequency does not match the emitter's Frequency.



Set the emitter's Frequency to a value within the suggested frequency range.

"Antenna Scan"

Selects an existing antenna scan or accesses a dialog for creating an antenna scan.

See "To create an antenna scan" on page 210

Remote command:

`EMITter:MODE:NAME` on page 427
`EMITter:MODE:ADD` on page 429
`EMITter:MODE:COUNt?` on page 430
`EMITter:MODE:SELect` on page 431
`EMITter:MODE:CLEar` on page 433
`EMITter:MODE:DELete` on page 432
`EMITter:MODE:ANTenna` on page 469
`EMITter:MODE:ANTenna:CLEar` on page 433
`EMITter:MODE:SCAN` on page 470
`EMITter:MODE:SCAN:CLEar` on page 433

ID

Sets the identifier for the selected emitter mode.

Remote command:

`EMITter:MODE:ID` on page 471

Emitter Beams Definition

In each operating mode, the emitter uses one or more beams (e.g. with different pulse sequences), while using the same antenna and scan type. Emitters use multiple beams for exact position detection, especially when the distance to the objects and beamwidth increase.

Use the standard functions in the table to create or rename beams.

Search for a defined beam via "Search Alias".

Assign a pulse sequence to each beam.

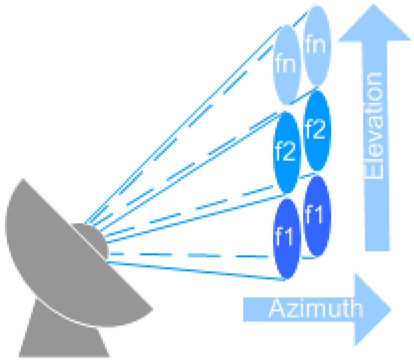


Figure 10-1: Stacked beams (simplified representation)

"Sequence" Selects an existing sequence or accesses a dialog for creating a sequence.
See Chapter 7.3, "How to create sequences and use the control elements", on page 131

"Frequency Offset"
Offsets the beam frequency by the selected value, see Figure 10-1.

"Beam Position Offset"
Offsets the beam in terms of an "Elevation" and an "Azimuth", see Figure 10-1.

To *offset the complete emitter's antenna*, use one of the "Elevation" parameters, depending on your current scenario:

- "Single Emitter" scenario: Elevation
(See "To configure an emitter in a scenario with static receiver" on page 234)
- "Localized Emitters" scenario: Elevation

Remote command:
EMITter:MODE:BEAM:ADD on page 429
EMITter:MODE:BEAM:COUNt? on page 430
EMITter:MODE:BEAM:NAME on page 427
EMITter:MODE:BEAM:SELect on page 431
EMITter:MODE:BEAM:STATe on page 470
EMITter:MODE:BEAM:SEQuence on page 470
EMITter:MODE:BEAM:OFFSet:AZIMuth on page 469
EMITter:MODE:BEAM:OFFSet:ELEVation on page 470
EMITter:MODE:BEAM:OFFSet:FREQuency on page 470
EMITter:MODE:BEAM:CLEar on page 433
EMITter:MODE:BEAM:DELete on page 432

10.2 3D single emitter preview settings

Access:

1. Open a "Single Emitter" or an "Emitter (Collection)" scenario.
2. In the block diagram, select "3D".

   The dialog displays:
   - A 3D view of the emitter antenna with its pattern and scan
   - A live plot of the normalized signal power level at the receiver

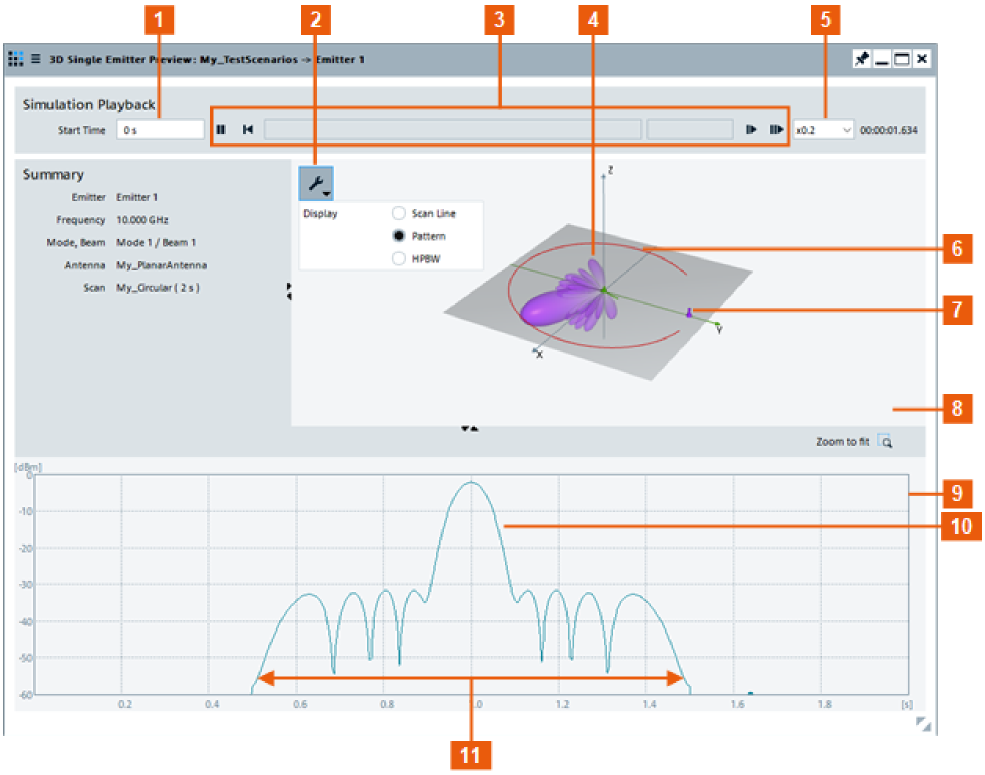


Figure 10-2: 3D Emitter Preview: Understanding the displayed information

1 = Preview time span (selected is one antenna turn)
2 = "Display settings" icon; defines how the scan is visualized
3 = Time bar; use the slider to set the "Start Time"
4 = Antenna pattern diagram of the emitter; antenna pattern without back lobes
5 = Play speed and time counter
6 = Circular antenna scan
7 = Receiver with an isotropic antenna pattern, no antenna scan and at a fixed location
8 = Interactive 3D view
9 = Live plot of the normalized signal power level at the receiver
10 = Antenna pattern without back lobes
11 = Antenna pattern (main and side lobes), compare with the pattern on Figure 9-12

Available settings:

Simulation Playback

Controls the timing of the simulation.

For example, with these settings you can analyze when/where/how often the beams overlap while simulating:

- A static scanning RX and a static scanning TX with different scan cycles.
- A moving RX and static TX
- A moving TX with different modes and a static RX

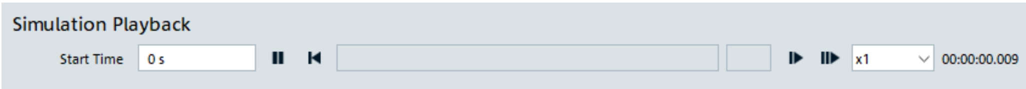


Figure 10-3: Simulation Playback

In addition to setting the overall duration, you can also observe a particular time-segment of the simulation by starting and stopping at particular times.

Note: The duration of a simulation depends on the scenario. An emitter with a long trajectory and low speed causes a long duration. A short trajectory and/or high speed cause a short duration.
The duration provides the maximum value for the "Stop Time" and "Preview Interval" settings.

"Start Time" Set to *0 s* or *minimum*, to play the simulation from the beginning. Typically, set this value to begin the simulation at a particular point along the trajectory.

Buttons Control the simulation using the following buttons:

- Pause/Play - use this toggle to pause the simulation and resume playing.
- Restart - use this button to restart the simulation at the configured "Start Time".
- Play slower/Play faster - decrease/increase the "Speed-up factor"

Display settings

Defines the way that the scan is represented:

"Scan Line" The scan is visualized by a line; the antenna pattern is disregarded.

"Pattern" The visualization corresponds to the current selected antenna pattern.

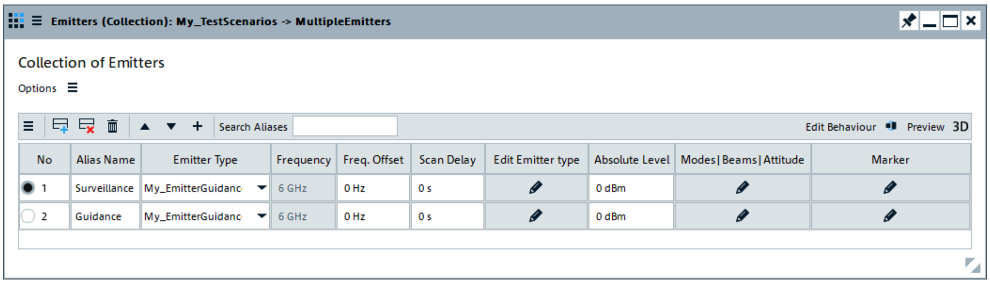
10.3 Emitters (Collection) settings

Access:

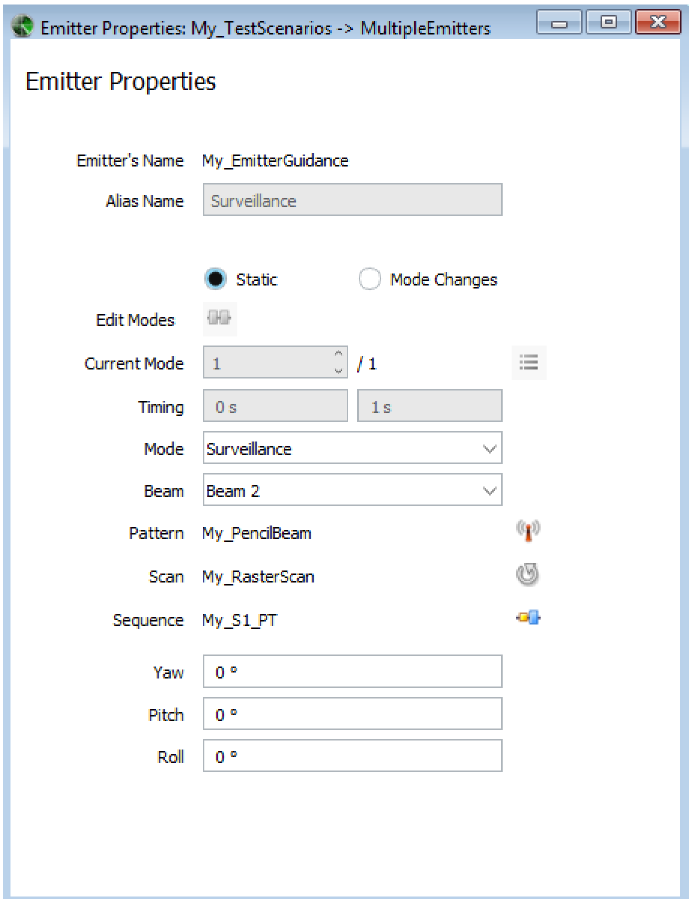
1. Open an "Emitters (Collection)" scenario.

2. In the block diagram, select "Emitters > Edit".

   The "Emitters (Collection)" scenario comprises several emitters. They display in list form.



3. To observe and configure the properties of the individual emitters, select "Edit Emitter type". To configure attitude and behavior, select "Modes|Beams|Attitude".



   The display resembles the emitters properties as they are configured in the "Emitter" dialogs, see Chapter 10.1, "Emitter settings", on page 220.

10.3.1 Collection of emitters

Settings:

Search Aliases

Allows a fast search of the configured emitters; useful in scenarios with many emitters.

Filters and displays emitters in table rows according to the entered search string.

Options

Joint configuration of these values for all emitters at once.

- "Reset Freq. Offset"
- "Reset Scan Delay"
- "Reset Abs. Level"

Insert, Append, Prepend, Remove, Clear All, Move Up/Down, Copy and append

Standard functions, available as context menu and as icons.

See also "To configure complex scenarios with several emitters" on page 235.

Remote command:

`SCENario:CEMit:ADD` on page 429
`SCENario:CEMit:SELect` on page 431
`SCENario:CEMit:DELete` on page 433
`SCENario:CEMit:CLEar` on page 434

Preview 3D

Opens the "3D Scan Pair View" for the selected "Emitter".

See Chapter 16.3, "3D scan pair view settings", on page 302.

Edit Modes

Accesses the "Mode Editor" dialog.

This dialog allows graphical configuration for defining the duration that an emitter is working in the different modes.

Works like the Mode Changes settings but rather than defining the start and end time of a mode, you drag the start and end time on the time scale.

See:
- "Behaviour" on page 312
- Chapter 10.3.3, "Mode editor settings", on page 230

List of emitters
Displays an overview of the configured emitters in table form.

No ← List of emitters
Subsequent number.

Selects the emitter for that the "Preview 3D" and "Edit Modes" apply.

Remote command:
`SCENario:CEMit:SELect` on page 431

Alias Name ← List of emitters
Enters an alias name.

Remote command:
`SCENario:CEMit:ALIas` on page 548

Emitter Type ← List of emitters
Selects an emitter.

Remote command:
`SCENario:CEMit:EMITter` on page 549

Frequency ← List of emitters
Indicates the Frequency value of the selected emitter.

Frequency offset ← List of emitters
Sets a frequency offset for the selected emitter.

Use this feature to configure the same emitter transmitting on different frequency.

Remote command:
`SCENario:CEMit:FQOFfset` on page 555

Scan delay ← List of emitters
Adds a time delay for the selected emitter and thus shifts the scans in time compared to each other.

Use this feature to simulate more realistic situation in that the emitters do not all point to the receiver at the beginning of the simulation.

Remote command:
`SCENario:CEMit:SCNDelay` on page 555

Edit Emitter Type ← List of emitters
Accesses the Emitter settings dialog.

Absolute Level ← List of emitters
Sets the absolute level of the selected emitter.

Remote command:
`SCENario:CEMit:LVABs` on page 556

Modes|Beams|Attitude ← List of emitters
Accesses the Emitter properties dialog that shows an overview information of the properties of the selected emitter.

For emitters that can work in several modes, the dialog allows configuration of the mode changes. Some other settings are read-only.

Markers
Accesses the "Emitter Marker Config" dialog.

See "Emitter Marker" on page 315.

10.3.2 Emitter properties

Each emitter is described with the following parameters:

Emitter's Name, Alias Name
Indicates the emitter name and its alias name, as selected in the Collection of emitters table.

Edit Modes
Accesses the "Mode Editor".

The mode editor allows graphical configuration for defining the duration that an emitter is working in the different modes.

Works like the Mode Changes settings but rather than defining the start and end time of a mode, you drag the start and end time on the time scale.

See:
- "Behaviour" on page 312
- Chapter 10.3.3, "Mode editor settings", on page 230

Static, Mode Changes
Enables emulation of emitters that change their mode over time.

See "Behaviour" on page 312.

"Mode"	Selects the mode that the emitter is working in.
"Beam"	Sets the number of the currently used beam.

Remote command:
SCENario:CEMit:EMITter:MODE on page 550
SCENario:CEMit:EMITter:MODE:BEAM on page 550

Antenna Pattern, Scan, Sequence
Displays the current selected antenna pattern, antenna scan, and sequence.

To change any of them, select the corresponding icon.

Pitch, Yaw, Roll
Offsets the antenna in terms of a "Pitch", a "Yaw" and a "Roll", see "Emitter attitude" on page 218.

Remote command:
SCENario:CEMit:DIRection:PITCh on page 549
SCENario:CEMit:DIRection:YAW on page 548
SCENario:CEMit:DIRection:ROLL on page 549
SCENario:CEMit:EMITter:MODE:TRACkrec on page 551

10.3.3 Mode editor settings

Access:

1. Open an "Emitter (Collection)" scenario.
2. In the block diagram, select "Emitters > Edit".
3. Select "Edit Modes".

   You access the "Mode Editor" dialog for configuration of the mode changes of *all emitters* available in the scenario.

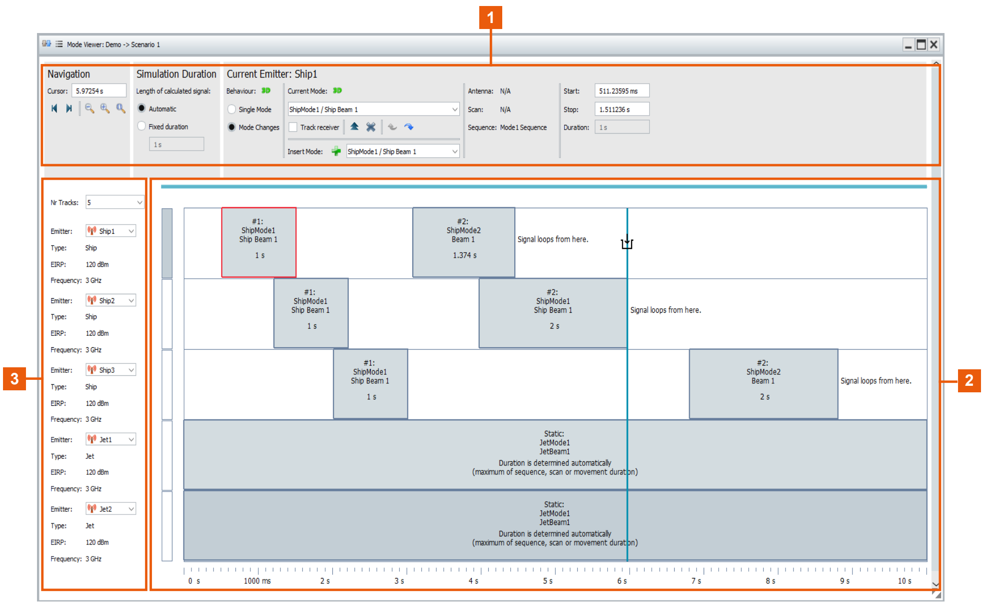


Figure 10-4: Mode Editor in Scenario Type = Localized Emitters

1 = Emitter and visualization settings
2 = Visualization area
3 = Configuration of the visualization area: select the number of displayed tracks ("Nr Tracks") and which emitter is displayed in the respective track

The "Mode Editor" is a graphical editor for joint configuration and defining the duration emitters are working in the different modes.
Works like the Mode Changes settings for each of the emitters. Rather than defining the start and end time of a mode, you drag the start and end time on the time scale.

Settings:

Navigation
Standard functions: Move start/end into view and zooming.

"Cursor" indicates and sets the current cursor position.

Simulation Duration
Resembles the settings in "Duration" on page 349. Included here for your convenience.

Current Emitter
You can configure mode changes only for emitters with several modes, see "Emitter Modes" on page 221.

Behavior ← Current Emitter
Select the mode for the emitter.

"Single Mode" Constant behavior over time.

"Mode Changes" Change of behavior over time.

Current Mode ← Current Emitter
Sets the mode of the selected emitter.

Track receiver ← Current Emitter
If activated, modes track and follow the receiver with their antenna scan.

Editing Emitter Sequences ← Current Emitter

Edit emitter sequences directly in the visualization area.

Extract selected mode
Removes the selected mode and its respective timeslot. Subsequent modes are shifted to close the timeslot of the removed mode.

Lift selected mode
Removes the selected mode. Timeslot of the removed mode is left as gap in the sequence.

Undo/Redo
Standard functions to reverse or repeat operation.

Insert Mode ← Current Emitter

Inserts an emitter with the selected mode/beam at the cursor position.

Antenna/Scan/Sequence ← Current Emitter

Displays the mode and beam configuration of the selected emitter.

Start/Stop ← Current Emitter

Sets the beginning and the end time of the selected mode.

To define the beginning and end time, use one of the following:

- Enter the time value in the corresponding field
- Drag the beginning and end borders of the bar representing the selected mode on the graph

Duration ← Current Emitter

Indicates the duration of the selected mode. The value is calculated automatically from the selected "Start" and "Stop" values.

Remote command:
`SCENario:DF:MCHG:STATe` on page 557
`SCENario:DF:MCHG:SELect` on page 431
`SCENario:DF:MCHG:ADD` on page 429
`SCENario:DF:MCHG:CLEar` on page 434
`SCENario:DF:MCHG:DELete` on page 433
`SCENario:DF:MCHG:COUNt?` on page 430
`SCENario:DF:MCHG:STARt` on page 557
`SCENario:DF:MCHG:STOP` on page 557

Graph

Comprises one or more bars, one per emitter available in the scenario.

To configure a particular emitter, select it. A blue bar next to the name indicates the currently selected one.

To edit a mode, select it. Currently selected modes are highlighted in red. You can set the start and end time of a mode by dragging it on the time axis or by setting the "Start" and "Stop" time.

10.4 How to create and configure emitters

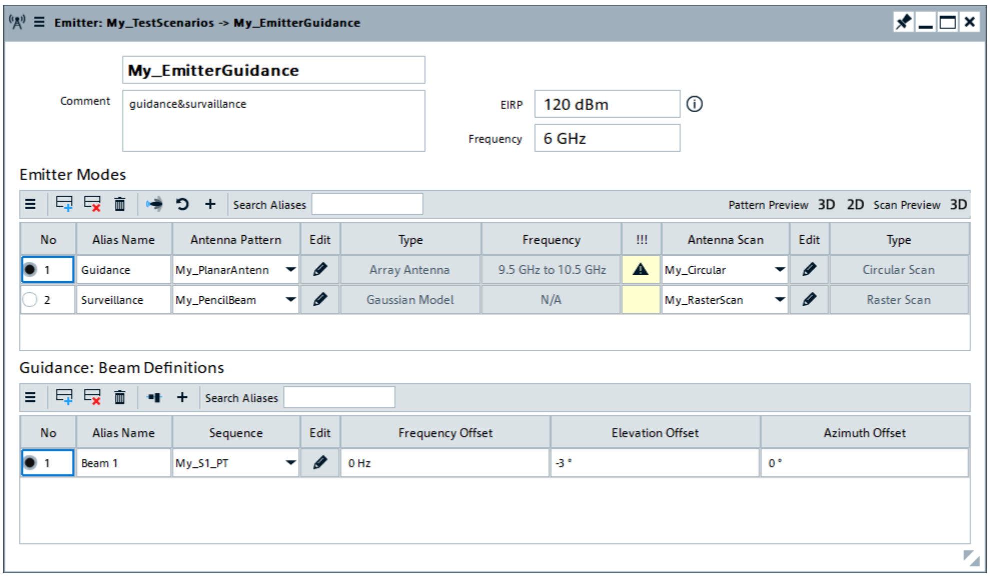
See:

- "To create and configure a new emitter" on page 233
- "To configure an emitter in a scenario with static receiver" on page 234
- "To visualize the signal received by a static receiver" on page 234
- "To configure complex scenarios with several emitters" on page 235

To create and configure a new emitter

You can clone or copy an existing emitter or create an emitter:

1. Select "Repository Tree > Emitter > New".
2. Enter a name and a comment.
3. Select "Mode 1".
   To rename the emitter, double-click the alias name. Enter a name for the emitter mode, e.g. "Guidance".
4. Select "Ant. Pattern" and select an antenna pattern from the antenna library.
5. Select "Antenna Scan" and select an antenna scan from the scan library.
6. To configure the "Beam":
   a) Select the beam via "No".
   b) Select "Sequence" and select a sequence from the library.
7. If necessary, add a second beam.



For description of the provided settings, see Chapter 10.1, "Emitter settings", on page 220.

To configure an emitter in a scenario with static receiver

1. Open a "Single Emitter" scenario.

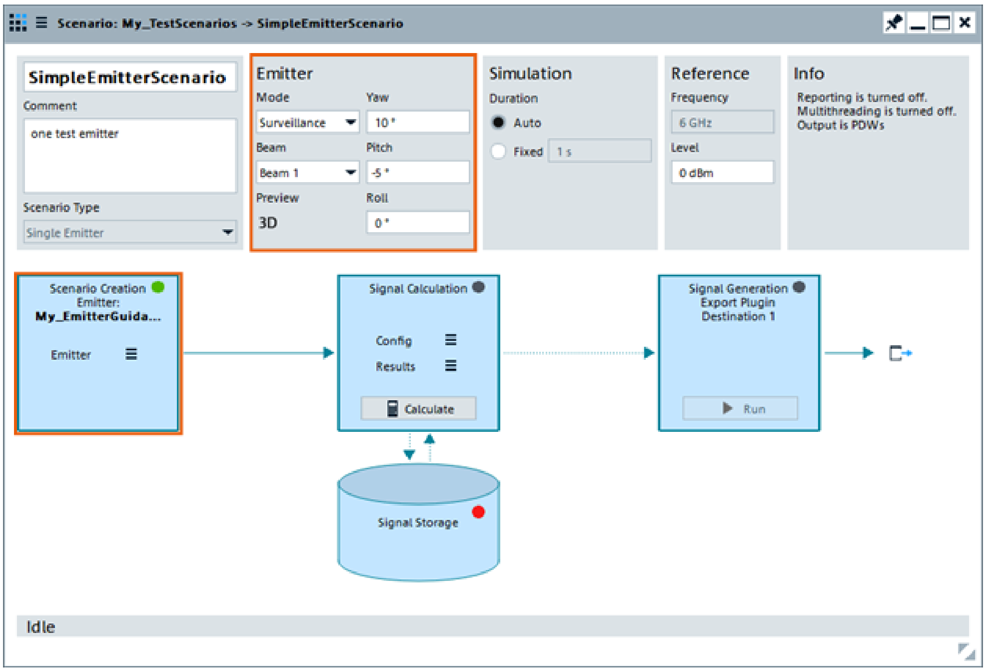


Figure 10-5: Single emitter scenario settings

2. In the block diagram, select "Scenario Creation > Emitter".
   In the context menu, select an emitter from the list of available emitters.
3. Configure the emitter settings, for example, select the current "Emitter Mode" and "Emitter Beam".

   For a description of the possible settings, see Chapter 10.3, "Emitters (Collection) settings", on page 225.

To visualize the signal received by a static receiver

1. Open a scenario with single emitter, e.g. "SimpleEmitterScenario".
2. In the block diagram, select "Emitter" settings > "Preview 3D".

   The dialog displays the signal as "seen" by a static receiver. By default, the receiver is located in such a way that the emitter scan is oriented with 0 deg "Pitch" and "Yaw" at it.
3. To "move" the receiver on the x-axis and on the z-axis, use "Emitter" settings > "Pitch" and "Yaw" (see Figure 10-5). The position on the y-axis (the distance to the emitter) is a fixed value.

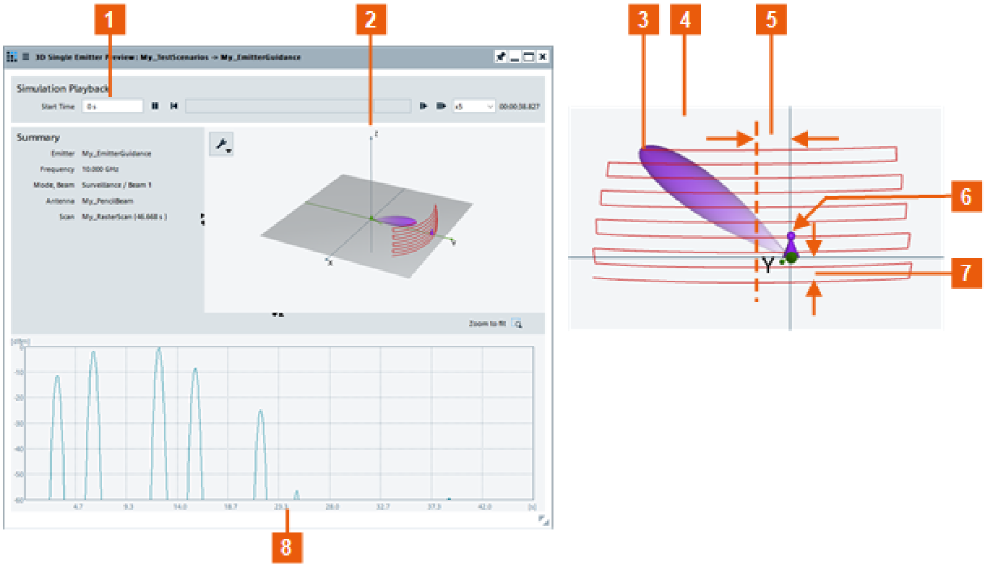


Figure 10-6: Signal at the receiver (XZ view): understanding the displayed information

1 = Selection of preview time span (selected is one antenna turn). See "Simulation Playback" on page 225.
2, 4 = 4 is the XZ representation of the 2 3D view; 2 uses "Scan Line" = "Pattern"; 4 uses "Scan Line" = "On"
3 = Current position of the scanning beam
5 = "Yaw = 10 deg" turns the antenna horizontally, i.e. the receiver is not at the central scan axis but rotated in clockwise (CW) direction on the x-axis
6 = Receiver's position, determined by the selected "Yaw" and "Pitch"; the position on the y-axis (the distance to the emitter) is fixed
7 = "Pitch = -5 deg" turns the antenna vertically, i.e. "moves" the receiver on the z-axis
8 = Amplitude changes in the received signal over time

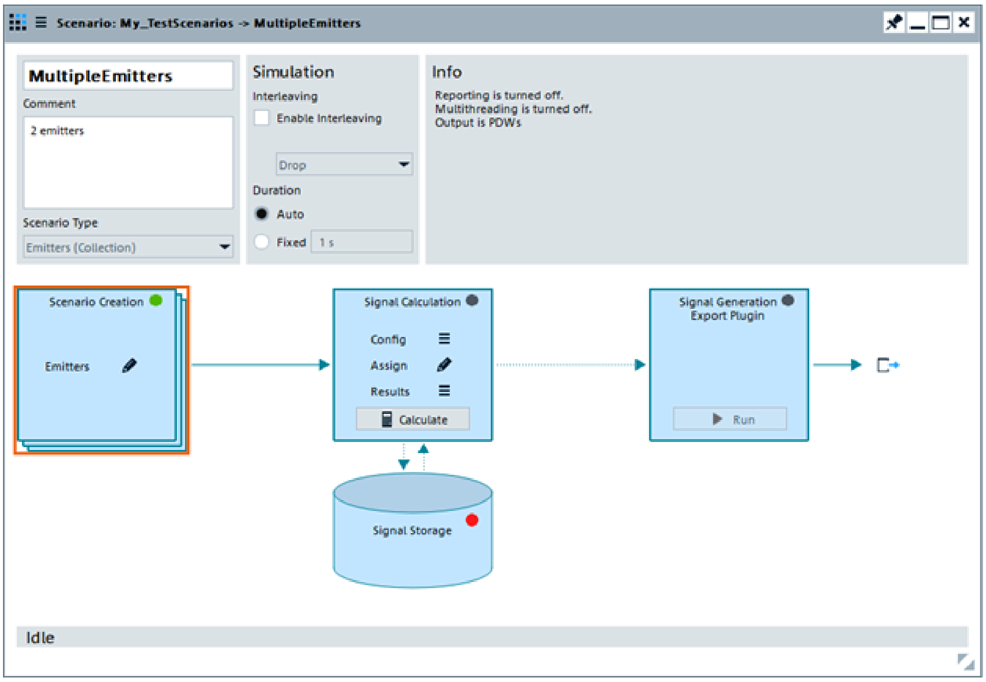
4. Use the mouse to change the orientation of the interactive 3D diagram and to zoom on it.
   For a description of the provided settings, see Chapter 10.2, "3D single emitter preview settings", on page 224.

To configure complex scenarios with several emitters

If your test situation requires testing the receiver's ability to detect the signal from different static emitters, you can use the "Emitter (Collection)" scenario. In this dedicated scenario, you can configure several emitters and switch between them sequentially. You can also configure the receiver and change its position in the scanning beam of the emitters.

The following example creates a complex scenario with several emitters and a receiver to explain the principle.

1. Open an "Emitter (Collection)" scenario.



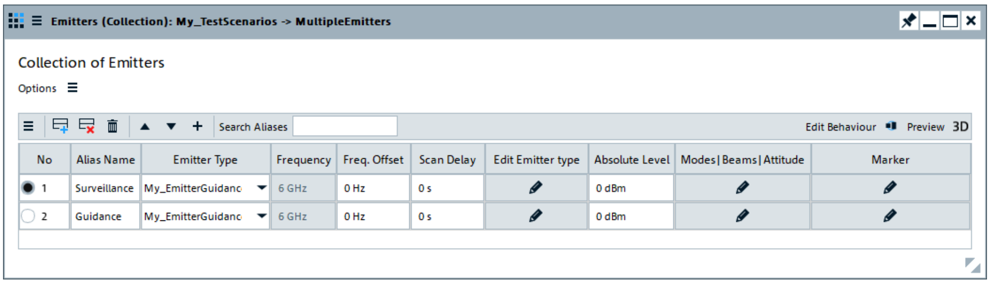
2. In the block diagram, select "Scenario Creation > Emitters".

   The "Emitters (Collection)" dialog opens.

3. Use the standard "Append" function to insert the first emitter in the list.

   You can select from the list of available emitters or create an emitter.

4. Configure the emitter settings, for example, enter an alias name.



5. In the "Emitters (Collection)" dialog, select the emitter that is transmitting, e.g. "Surveillance" in the "No" column.

6. From the controls above the table, select the "Preview 3D" view to visualize the signal received by a static receiver.
   See Figure 10-6.

   For a description of the provided settings, see:
   - Chapter 10.3, "Emitters (Collection) settings", on page 225
   - Chapter 10.2, "3D single emitter preview settings", on page 224

To create a single emitter scenario automatically with the "Startup Assistant"

Alternatively to the methods described in this section, you can use the "Startup Assistant" to create a single emitter scenario.

1. In the menu bar, select "Help > Wizard".
2. Select "Create an emitter".
   Click "Next".
3. Follow the instructions, use the default settings and select "Finish".
   For detailed step-by-step description, see Chapter 2.8.5, "Using the wizard to create a complex scenario", on page 39.

   For information on the provided settings, see:
   - Chapter 10.1, "Emitter settings", on page 220
   - Chapter 9.2, "Antenna pattern settings", on page 174
   - Chapter 9.3, "Antenna scans settings", on page 193
   - Chapter 6.2.3, "Modulation on pulse (MOP) settings", on page 96

11 Combining the signals of multiple emitters

In a real-world scenario, radar receivers have to detect and process several simultaneous signals form different emitters. Such scenarios are referred as multi-emitter scenarios. In multi-emitter scenarios, the signal at the radar receiver is a combination of the signals of several emitters.

About the interleaving

Option:R&S PULSE-K39

In addition to processing the emitters sequentially, you can also combine them into a single output file.

During the interleaving process in "Drop" mode, overlapping pulses within the same interleaving group and with lower priority are dropped. The priority of the individual emitters is configurable as is the definition of several groups and the assignment of the emitters to them. In "Merge" mode, priorities are not set and overlapping pulses are interleaved without dropping.

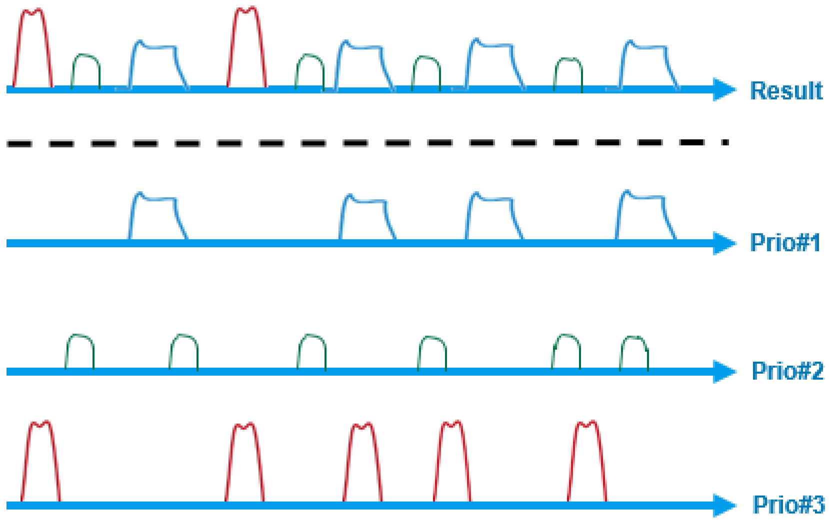


Figure 11-1: Principle of the priority-based interleaving algorithm (applied per interleaving group)

Prio#1 = Pulses with the highest priority (i.e. the priority value is 0 or smaller than the value for the other emitters)
Prio#3 = Pulses with the lowest priority (i.e. the highest priority value within the emitters)

Interleaving signals and combining them into single output signal is function provided in the following scenario types:

- Emitters (collection)
- Localized emitters
- Direction finding
- PDW list (collection)

This section described the settings in the first three scenario types. For description of the settings in the last one, see Chapter 18.6, "PDW list (Collection) interleaving settings", on page 370.

11.1 Emitters (Collection) and localized emitters interleaving settings

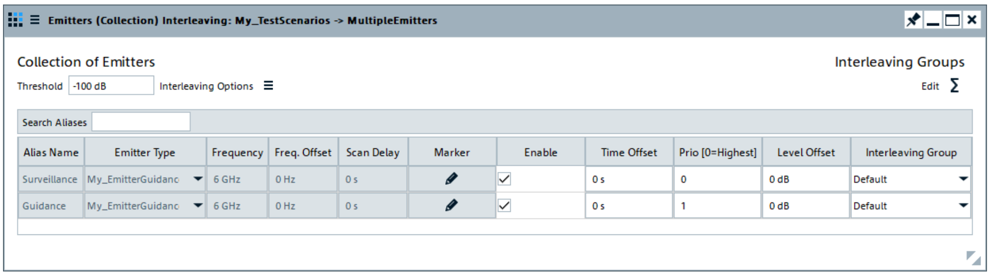
Option:R&S PULSE-K39

Access:

1. In a "Scenario Type = Emitters (Collection)/Localized Emitters/Direction Finding", select "Emitter > Interleaving > On".
2. Select "Interleaving > Config".

   Dialog displays the available emitters as configured in the dialogs:
   - Emitters (Collection)
   - "Localized Emitters" > Properties of TX items (emitters and platforms)
   - "Direction Finding" > Properties of TX items (emitters and platforms).

   The interleaving related settings, like "Time Offset" and "Priority", are also displayed.



Settings:

Threshold
Enters a level threshold to limit the dynamic range of the signal. Pulses at levels below this threshold are omitted.

Resembles the value set with the parameter "Scenario > Signal Generation > Config > Waveform Generation > Output > Threshold".

Remote command:
`SCENario:CEMit:THReshold` on page 563
`SCENario:DF:THReshold` on page 564
`SCENario:LOCalized:THReshold` on page 564
`SCENario:OUTPut:THReshold` on page 564

Search Aliases
Allows fast search of the configured emitters; useful in scenarios with many emitters.

Filters and displays emitters in table rows according to the entered search string.

Interleaving Options
Joint configuration of all emitters at once.

- "Enable All"/"Disable All"
- "Reset Time offset"/"Reset Level offset"
- "Reset Prio"

Edit
Opens the "Interleaving Groups" dialog.

See "Edit Interleaving Groups" on page 242.

Alias Name
Displays the alias name.

Remote command:
`SCENario:CEMit:ALIas` on page 548

Emitter type
Displays the emitter name as selected with Emitter Type.

Remote command:
`SCENario:CEMit:EMITter` on page 549

Frequency
Indicates the Frequency value of the selected emitter, as selected in the Emitter dialog.

Remote command:
SCENario:CEMit:FREQuency? on page 555

Freq. Offset
Displays the frequency offset of the emitter as selected with Frequency offset.

Remote command:
SCENario:CEMit:FQOFfset on page 555

Scan delay
Displays the scan delay for the emitter as selected with Scan delay.

Remote command:
SCENario:CEMit:SCNDelay on page 555

Marker
Opens the "Emitter Marker Config" dialog.

See Emitter Marker.

Enable
Selects the emitter for interleaving.

If disabled, the signal of the emitter is not included in the interleaved output signal.

Remote command:
SCENario:CEMit:ENABle on page 559
SCENario:DF:ENABle on page 559
SCENario:LOCalized:ENABle on page 559

Time Offset
Available if "Enable" = On.

Shifts the signals relative to each other to prevent high drop out rates due to pulse overlap.

Use this function, for example, to decrease the number of dropped pulses.

Remote command:
SCENario:CEMit:LDELay on page 560
SCENario:DF:LDELay on page 560
SCENario:LOCalized:LDELay on page 560

Priority
Available if "Enable" = On.

Sets the priority of the selected emitter. Higher value means lower priority. If pulses are overlapping, the pulses from the emitter with lower priority are discarded.

See Figure 11-1.

Remote command:
SCENario:CEMit:PRIority on page 560
SCENario:DF:PRIority on page 560
SCENario:LOCalized:PRIority on page 560

Level Offset
Available if "Enable" = On.

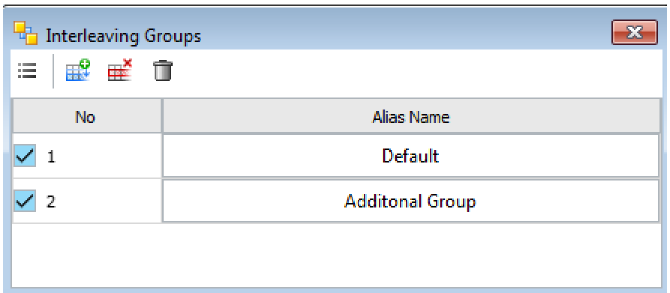
Adds a relative level offset between the signals.

Remote command:
`SCENario:CEMit:LEVel` on page 560
`SCENario:DF:LEVel` on page 560
`SCENario:LOCalized:LEVel` on page 560

Interleaving Group
Assigns the emitter to one of the available interleaving groups, defined in the Edit Interleaving Groups dialog.

Remote command:
`SCENario:CEMit:GROup` on page 561
`SCENario:DF:GROup` on page 561
`SCENario:LOCalized:GROup` on page 561

Edit Interleaving Groups
Select the icon to access the "Interleaving Groups" dialog.



Interleaving groups allow you to combine the emitter signals into different output signals.

Use the standard functions (icons) to add a new group, rearrange the groups, delete the existing one or all groups.

Select, Insert, Append/Prepend, Remove, Clear ← Edit Interleaving Groups
Standard functions for items handling, see "Standard function in the context menus" on page 33.

Select a row, for example, to delete it or to insert a row before it.

Remote command:
`SCENario:CEMit:GROup:CATalog?` on page 561
`SCENario:CEMit:GROup:SELect` on page 430
`SCENario:CEMit:GROup:ADD` on page 429
`SCENario:CEMit:GROup:DELete` on page 432
`SCENario:CEMit:GROup:CLEar` on page 433
`SCENario:DF:GROup:CATalog?` on page 561
`SCENario:DF:GROup:SELect` on page 430
`SCENario:DF:GROup:ADD` on page 429
`SCENario:DF:GROup:DELete` on page 432

SCENario:DF:GROup:CLEar on page 433
SCENario:LOCalized:GROup:CATalog? on page 561
SCENario:LOCalized:GROup:SELect on page 430
SCENario:LOCalized:GROup:ADD on page 429
SCENario:LOCalized:GROup:DELete on page 432
SCENario:LOCalized:GROup:CLEar on page 433

No. ← Edit Interleaving Groups
Select a row to perform any row-based actions.

Interleaving groups are indicated by consecutive number; also used for indication in remote control.

Remote command:
SCENario:CEMit:GROup:COUNt? on page 430
SCENario:CEMit:GROup:SELect on page 430
SCENario:DF:GROup:COUNt? on page 430
SCENario:DF:GROup:SELect on page 430
SCENario:LOCalized:GROup:COUNt? on page 430
SCENario:LOCalized:GROup:SELect on page 430

Alias Name ← Edit Interleaving Groups
Enter an alias name for the interleaving group.

Remote command:
SCENario:CEMit:GROup:ALIas on page 561
SCENario:DF:GROup:ALIas on page 561
SCENario:LOCalized:GROup:ALIas on page 561

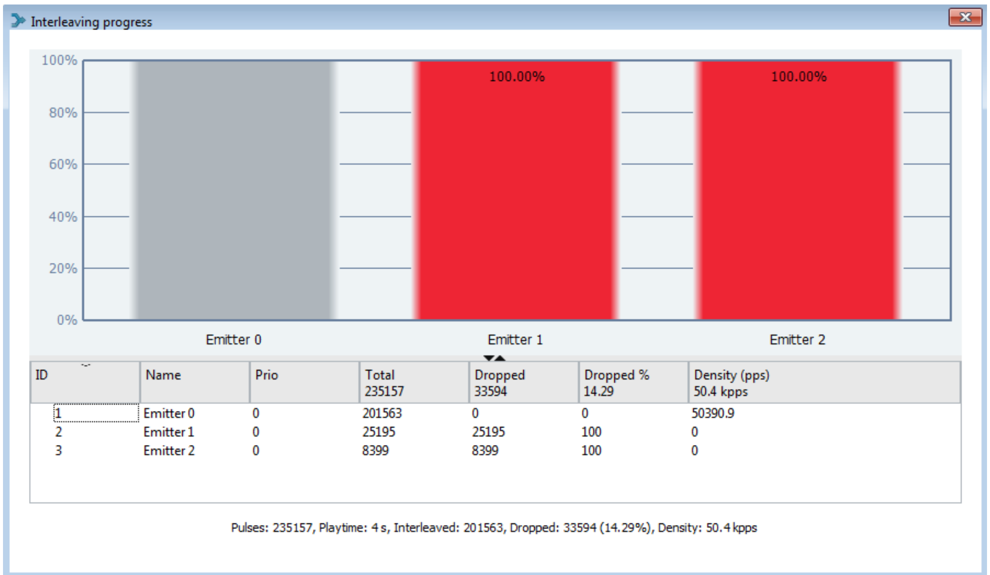
11.2 Emitters interleaving progress statistics

Option:R&S PULSE-K39

Access:

1. In a "Scenario Type = Emitters (Collection)", select "Emitter > Interleaving > On".
2. Select "Signal Control > Prepare".
3. Select "Signal Control > Interleave".

   The "Interleaving Progress" display opens automatically during the signal calculation, if pulses lists are merged. It indicates the percentage of pulses being processed and dropped for each of the emitters.

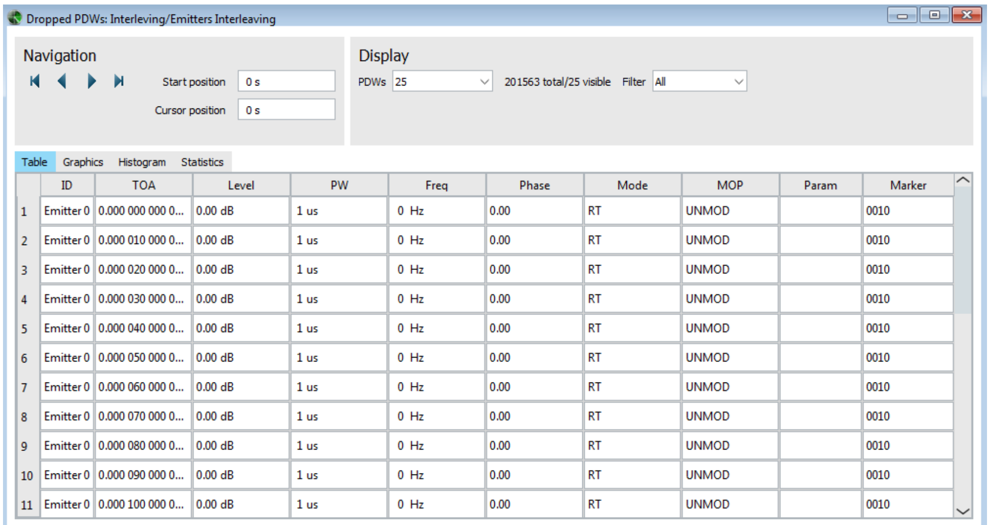


11.3 Signal preview with interleaved and dropping statistics

Option:R&S PULSE-K39

Access:

1. In a "Scenario Type = Emitters (Collection)", select "Emitter > Interleaving > On".
2. Select "Signal Control > Prepare".
3. Select "Signal Control > Interleave".
4. Select "Interleaving > Results > Interleaved PDW/Dropped PDWs".



The dialog is similar to the "Signal Preview" dialog.

This section describes the settings dedicated to the "Interleaved/Dropped" dialogs. For description of all other settings, see Chapter 15.1, "Signal preview settings", on page 284.

Settings:

Display > Filter
If interleaving is used, the calculated output signal contains pulses originating for multiple PDW lists or emitters. By default, all pulses are displayed. Use the "Filter" parameter to narrow down the displayed information to the pulses of one of the emitters/PDW lists.

The filter does not change the zoom level or the scaling on both axis.

Table
Provides summary information on the number of processes pulses and the content of the output file.

Graphics
Visualize the variation of the parameters over time.

For details, see "Graphics" on page 287.

Graphics > Highlight
Highlights pulses belonging to the selected emitter/PDW list.

Histogram
Resembles the information displayed during the signal processing, see Chapter 18.7, "PDW interleaving progress statistics", on page 373.

Use the filtering option to narrow down to the displayed information according to one of the predefined criteria.

Statistic
The drop out statistic is the table form representation of the same information as the Histogram.

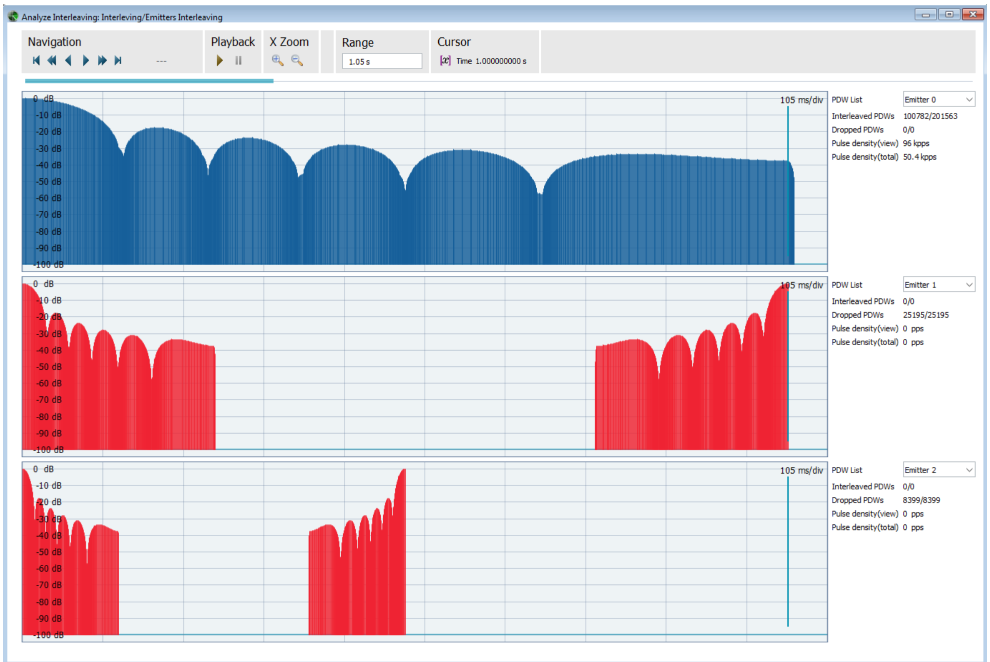
11.4 Analyze interleaved signals

Option:R&S PULSE-K39

Access:

1. In a "Scenario Type = Emitters (Collection)", select "Emitter > Interleaving > On".
2. Select "Simulation > Interleaving".
3. Select "Signal Calculation > Calculate".
4. Select "Interleaving > Results > View > Analyze".

   The "Interleaving progess" dialog indicates the pulses that are and are not included in the resulting output file. Pulses displayed in **red are dropped out**.

   The provided settings are a subset of the settings available in the "Signal Preview" displays and they have the same effect.

5. To increase the number of interleaved pulses, add a *delay to one of the emitters*.
   For example, zoom in the beginning of the signal until you can observe the individual pulses and estimated the required delay.
   Set, for example, "Interleaving > Config > Emitter#1 > Time Offset = 5 us".
   Create the output signal in the same manner. The signal is not recalculated; only the new interleaving settings applied.
   Observe the resulting output signal.

   Depending on the settings, the output file includes more or even all pulses originating from all emitters.

12 Creating platforms

This section describes the use of platforms in the R&S Pulse Sequencer Digital. It covers their characteristics and explains the provided settings and views.

12.1 About platforms

A platform is a group of up to 8 emitters assigned to a single vehicle.

A platform is characterized by its icon, list of emitters, position and trajectory.

The emitters on a platform can be individually configured.

One or more platforms can be included in a scenario. Within a scenario, each platform can be individually enabled or disabled.

12.2 Platform settings

This section describes how platforms can be created and configured.

Access:

► To create a platform, do one of the following:

a) In the repository tree, select "Platform Types" > context menu > "New".
b) Select an existing platform > context menu > "Clone".

A new platform is added to the repository. The "Platform" dialog opens.

R&S Pulse Sequencer Digital generates new platform names automatically.

You can edit the platform name.

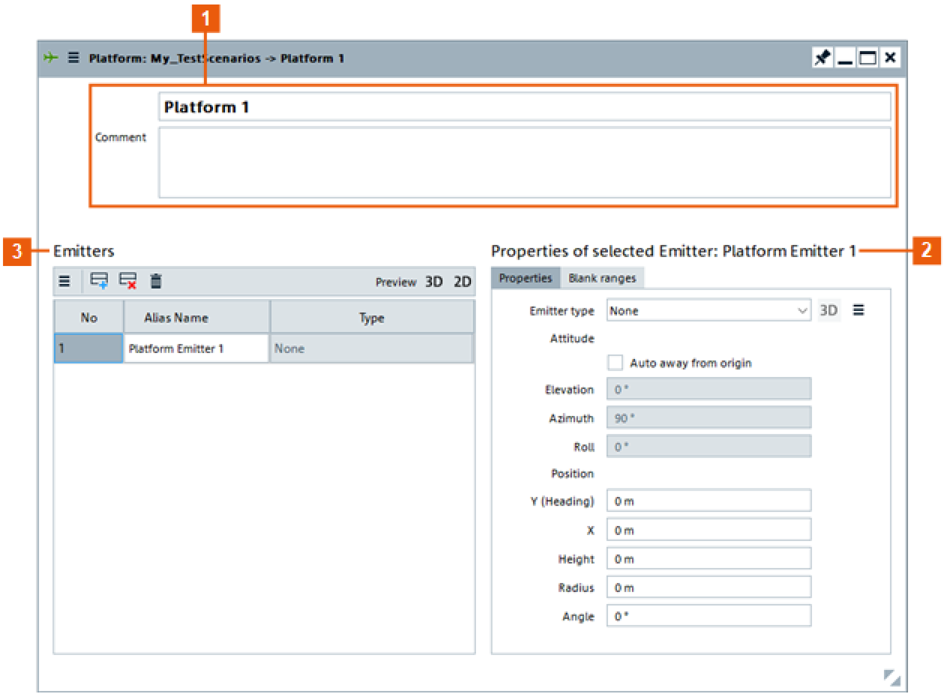


Figure 12-1: New platform in the Platform dialog

1 = Platform information
2 = Properties of selected emitter
3 = Emitter list

Settings

12.2.1 Platform information

Name

The name of the new platform is created automatically.

The format of the new name depends on how it was created.

- **New platform**
  The automatically generated name consists of the word *Platform* and a number - e.g. Platform 1. The number is incremented for every new platform.
- **Cloned platform**
  The name of a cloned platform consists of the name of the parent and a numerical index. For example:
  – Platform 1
  – Platform 1.1 = first clone of *Platform 1*
  – Platform 1.2 = second clone of *Platform 1*

– Platform 1.2.1 = first clone of *Platform 1.2*

To change the name, select the text and edit it.

Note: R&S Pulse Sequencer Digital does not support duplicate names.
If you enter a duplicate name, a popup error message appears.

Remote command:
`PLATform:NAME` on page 427
`PLATform:CREate` on page 426
`PLATform:SELect` on page 426
`PLATform:CATalog?` on page 426
`PLATform:REMove` on page 428

Comment
Use this optional field to provide additional information about the platform.

Remote command:
`PLATform:COMMent` on page 428

ID
Platform identifier.

Remote command:
`PLATform:ID` on page 579

12.2.2 Emitter list

Enables you to add, remove and edit the emitters of the platform.

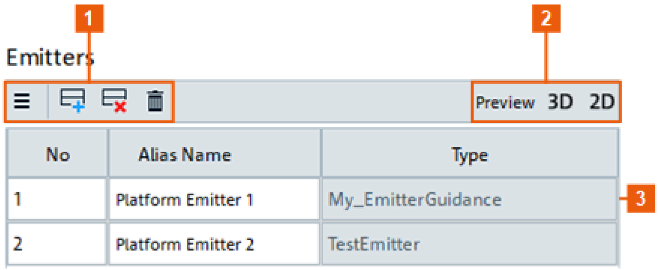


Figure 12-2: Emitter list example

1 = Icons for standard functions
2 = Icons for 2D or 3D previews of the platform
3 = List of emitters: by number ("No") and "Alias Name"

Icons for standard functions
Enable you to access properties, add or delete an emitter (see also Table 2-4).

List of emitters
A platform must have at least one emitter, so one unconfigured emitter is added to a new platform.

You can configure up to 8 emitters on a platform.

Select an emitter to view/edit its properties in Properties of selected emitter.

Remote command:
`PLATform:EMITter:SELect` on page 426
`PLATform:EMITter:ALIas` on page 579

Icon for 2D preview
Icon that accesses the 2D preview of the platform.

Figure 12-3 shows an example with two emitters.

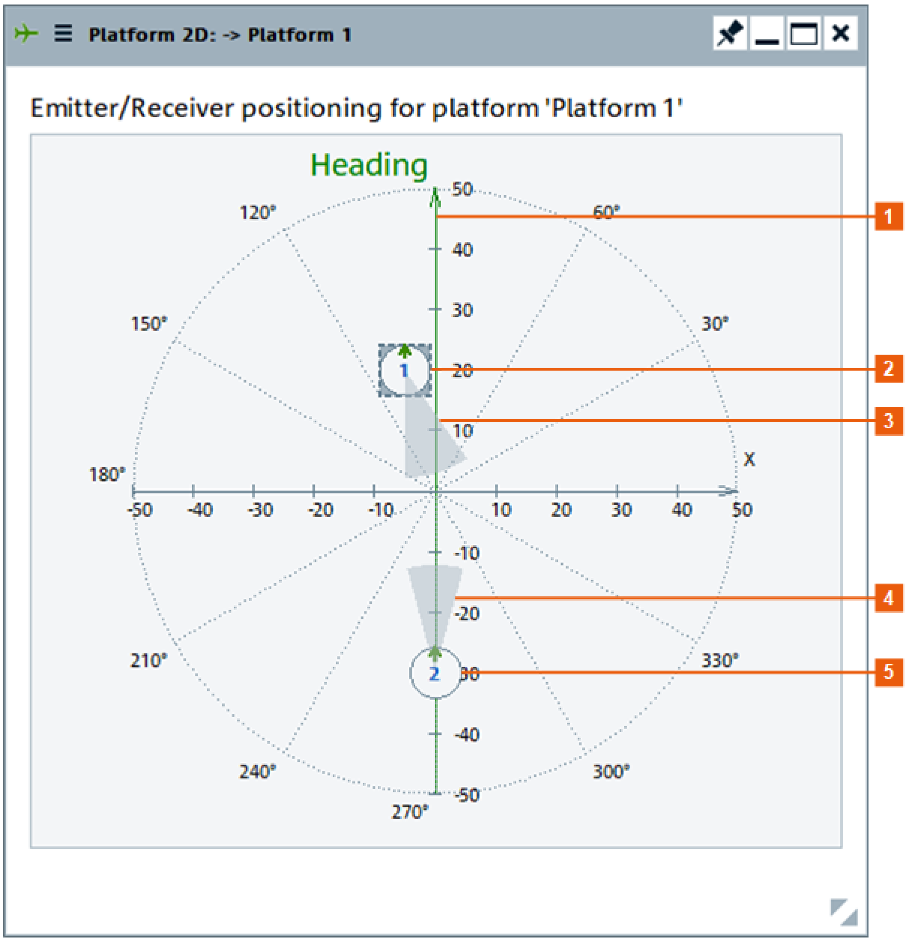


Figure 12-3: Platform 2D view example

1 = Heading of platform
2 = Emitter-1
3 = Blank range of emitter-1
4 = Blank range of emitter-2
5 = Emitter-2

This example shows you:
- A platform that represents two radars aboard a ship.
- "Heading" indicates the platform's direction of travel.
- Emitter-1 is mounted off-axis and has a blanking range towards the rear of the platform.
- Emitter-2 is mounted on-axis and has a blanking range towards the front of the platform.
- The icon for emitter-1 has a frame. This frame indicates that this emitter is selected in the emitter list.

Icon for 3D preview
Icons that access the 2D or 3D preview of the platform.

Figure 12-4 shows an example with the same two emitters as Figure 12-3.

The "2D" view shows the precise horizontal placement of the emitters. The "3D" view shows the placement of the emitters, both horizontally and vertically, relative to the platform's origin.

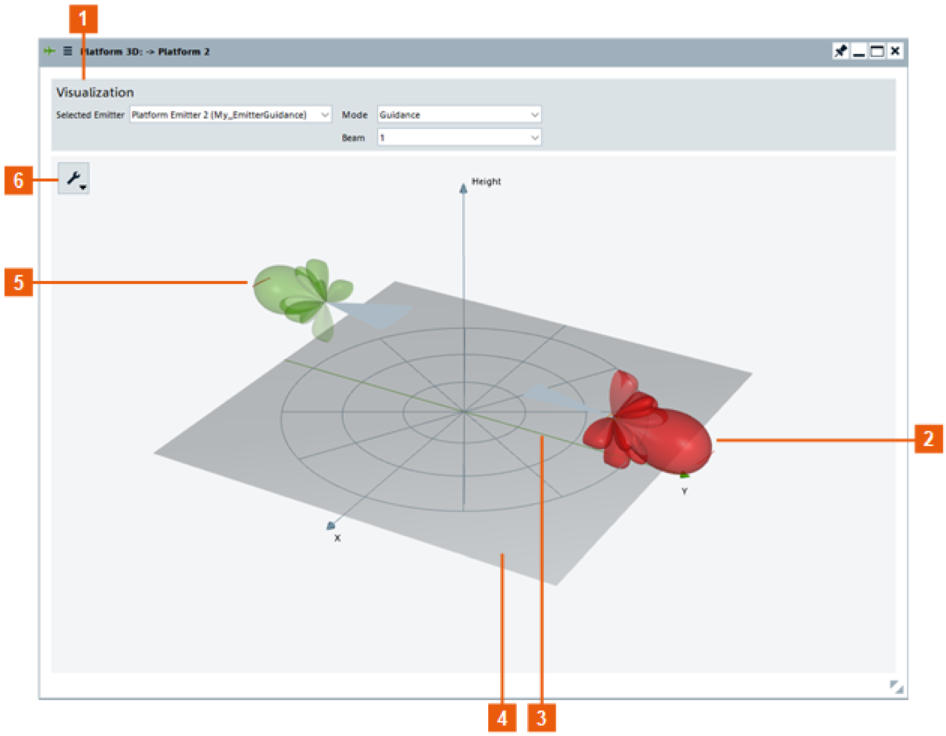


Figure 12-4: Platform 3D view example

1 = Visualization settings
2 = Antenna pattern and blank range of emitter 1
3 = Heading of platform

4 = Ground plain
5 = Antenna pattern and blank range of emitter 2
6 = Tools

Settings and tools:

Selected Emitter ← Icon for 3D preview
Sets the emitter for which the "Mode" and "Beam" settings apply.

The drop-down list contains all emitters of the selected platform.

The selected emitter is colored red in the 3D preview.

Mode ← Selected Emitter ← Icon for 3D preview
Defines which mode of the selected emitter is visualized in the 3D preview.

The drop-down list contains all modes of the selected emitter.

Beam ← Selected Emitter ← Icon for 3D preview
Defines which beam of the selected emitter is visualized in the 3D preview.

The drop-down list contains all beams of the selected emitter.

Opacity ← Icon for 3D preview
Use this function to make the antenna patterns more or less transparent.

12.2.3 Properties of selected emitter

This tab contains the parameters of the emitter selected in the "Emitter" list of the "Platform" dialog.

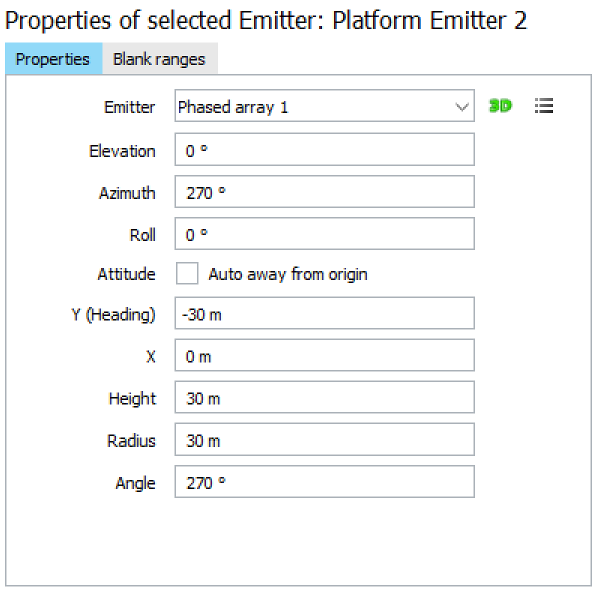


Figure 12-5: Example of emitter properties configuration

Settings

Emitter

Selects the emitter assigned to the item selected in the "Emitter" list of the "Platform" dialog.

Remote command:
`PLATform:EMITter` on page 579

Elevation

Sets the elevation of the emitter item's pointing direction, relative to the platform's azimuth plane.

Values can be:

- Positive - pointing above the azimuth plane (e.g. air traffic control radar)
- Negative - pointing below the azimuth plane (e.g. aircraft landing radar)

This setting "Attitude" on page 254 overrides the "Elevation" parameter.

Remote command:
`PLATform:EMITter:ELEVation` on page 582

Azimuth

Sets the azimuth of the emitter item's pointing direction, relative to the platform's heading.

For static emitters, this parameter sets the azimuth of the beam.

For circular emitters, this parameter sets the beam direction at the start of the scan.

For sector scans, this parameter sets center of the scanned sector.

The setting "Attitude" on page 254 overrides the "Azimuth" parameter.

Remote command:
`PLATform:EMITter:AZIMuth` on page 580

Roll

Sets the roll of the emitter item's pointing direction, relative to the platform's up direction.

Remote command:
`PLATform:EMITter:ROLL` on page 582

Attitude
Selecting the "Auto away from origin" checkbox, sets the "Azimuth" and "Elevation" automatically.

- "Elevation" - automatically set to 0°
- "Azimuth" - automatically configured so that the beam axis is radial to the platform origin.

Remote command:
`PLATform:EMITter:DIRection:AWAY` on page 581

Y (Heading)
The position of an emitter on a platform is defined by three parameters:
- "Y (Heading)"
- "X" (see "X" on page 254)
- "Height" (see "Height" on page 254)

The "Y (Heading)" parameter defines the position of the emitter (in the direction of movement) relative to the platform's origin.

Note: The XY position can also be defined via the "Radius" from the origin and the "Angle" relative to the X-axis.

Remote command:
`PLATform:EMITter:Y` on page 583

X
The position of an emitter on a platform is defined by three parameters:
- "Y (Heading)" (see "Y (Heading)" on page 254)
- "X"
- "Height" (see "Height" on page 254)

The "X" parameter defines the position of the emitter (at right-angles to the direction of movement) relative to the platform's origin.

Note: The XY position can also be defined via the "Radius" from the origin and the "Angle" relative to the X-axis.

Remote command:
`PLATform:EMITter:X` on page 583

Height
The position of an emitter on a platform is defined by three parameters:
- "Y (Heading)" (see "Y (Heading)" on page 254)
- "X" (see "X" on page 254)
- "Height"

The "Height" parameter defines the position of the emitter relative to the XY plane.

Note: You can use the "Height" parameter with the *XY* positioning method or the *angle and radius* method.

Remote command:
`PLATform:EMITter:HEIGht` on page 582

Radius

The position of an emitter on a platform is defined by three parameters:

- "Angle" (see "Angle" on page 255)
- "Radius"
- "Height" (see "Height" on page 254)

The "Radius" parameter defines the position of the emitter on the XY plane, relative to the platform's origin.

Note: The XY position can also be defined via the "X" and "Y (Heading)" parameters.

Remote command:
PLATform:EMITter:RADius on page 582

Angle

The position of an emitter on a platform is defined by three parameters:

- "Angle"
- "Radius" (see "Radius" on page 255)
- "Height" (see "Height" on page 254)

The "Angle" parameter defines the position of the emitter on the XY plane, relative to the platform's X-axis.

Note: The XY position can also be defined via the "X" and "Y (Heading)" parameters.

Remote command:
PLATform:EMITter:ANGLe on page 579

12.2.4 Blank ranges of selected emitter

This tab displays the sectors in which the selected emitter mutes its signal output.

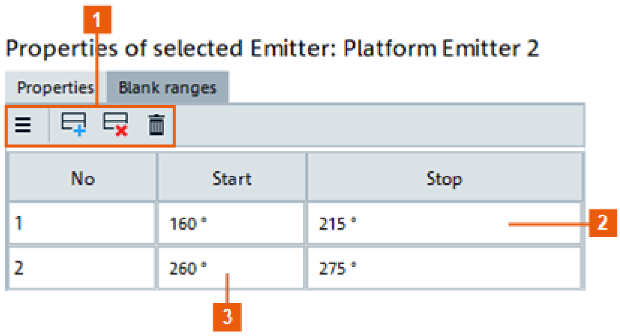


Figure 12-6: Example of blank ranges configuration

1 = Icons that access standard functions
2 = Blank range stop angle
3 = Blank range start angle

On real platforms, blank ranges represent:

- Sectors in which the radar does not transmit.
  For example, to avoid saturating other antennas with the high-power signal from the emitter or to avoid wave-scatter from metal structures such as masts.
- Sectors in which the propagation of the transmitted radar signal is blocked.

For example, due to obstructions such as masts and superstructure.

Settings

No
Number of the individual blank range entry.

Remote command:
`PLATform:EMITter:BLANkranges:SELect` on page 430

Start
Set the angle for the beginning of the selected blank range.

Remote command:
`PLATform:EMITter:BLANkranges:STARt` on page 580

Stop
Set the angle for the end of the selected blank range.

Remote command:
`PLATform:EMITter:BLANkranges:STOP` on page 581

Set of ranges
You can enter a set of blank ranges for the selected emitter, using a single SCPI command.

The following SCPI example, adds a set of three blank ranges to emitter number 3.

The comma-separated value list (90,120,160,200,350,10) contains three pairs of start/stop values.

- start=90, stop=120
- start=160, stop=200
- start=350, stop=10

This approach is more efficient than using several blank range start/stop commands.

Remote command:
`PLATform:EMITter:BLANkranges` on page 580

12.3 Working with platforms

The platforms you create appear in the repository, as shown in Figure 12-7.

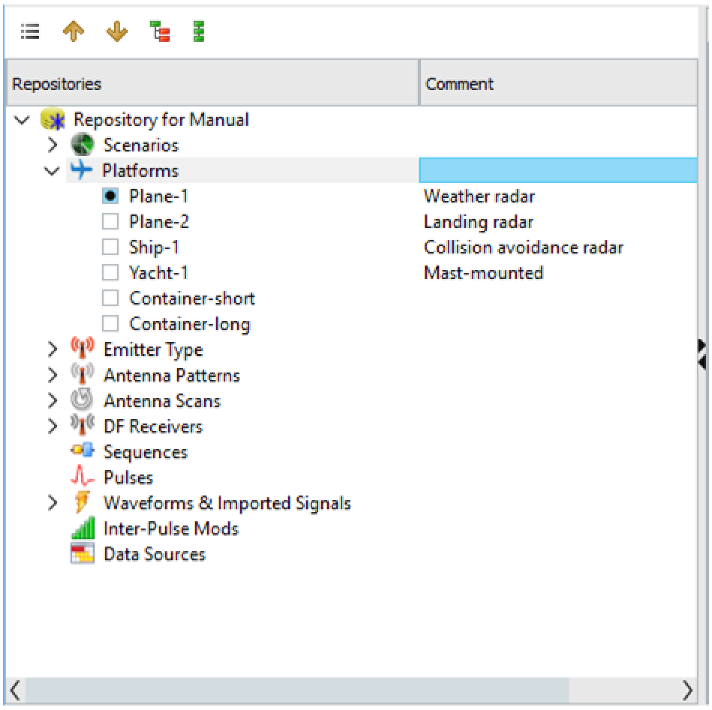


Figure 12-7: Examples of platforms in a repository

To create a platform

1. To create a platform, do one of the following:
   a) In the repository tree, select "Platform Types > New".
   b) Select an existing platform > context menu > "Clone".

   A new platform is added to the repository. The "Platform" dialog opens.
2. Configure the platform as described in Chapter 12.2, "Platform settings", on page 247.

To edit a platform

1. In the repository tree, select a platform.

   The "Platform" dialog opens.
2. Configure the platform as described in Chapter 12.2, "Platform settings", on page 247.

To delete a platform

1. In the repository tree, select a platform.
2. In the context menu, select "Delete".

   A dialog opens to confirm the action.
3. Click "Yes" to delete the platform.

   The selected platform is removed from the repository.

To add a platform to a scenario

1. In the repository tree, select a scenario.
2. In the scenario dialog, click "Scenario Creation > Map".
3. To add a platform to a scenario, do one of the following:
   a) Drag&drop the platform onto the 2D map.
   b) Drag&drop the platform into the "TX Items" list.
   c) In the 2D map: In the context menu, select "Add Platform".

To edit a platform in a scenario

1. To open the platform dialog to edit the platform, do one of the following:
   a) Double-click the platform in the "TX Items" list.
   b) Double-click the platform in the 2D map.

   The "TX Items Properties" dialog opens.
2. Edit the platform parameters.

To remove a platform from a scenario

1. In the repository tree, select a scenario.
2. In the scenario dialog, click "Scenario Creation > Map".
3. To remove a platform from a scenario, do one of the following:
   a) In the "TX Items" list: Select the platform > context menu > "Remove".
   b) In the 2D map: Select the platform > context menu > "Remove".

   The selected entry is removed from the scenario.

13 Emulating receivers

The available receiver settings depend on the complexity of the simulated system:

- Receivers in the "Single Emitter" and "Emitters (Collection)" scenario types.
  These receivers have *an omnidirectional antenna with no antenna scan*.
  These scenarios only simulate the effect of emitter antennas and scans. Attenuation due to distance and free space propagation is not considered.
- Receivers in a 2D multi-emitters receiver scenario have *one receive antenna*.
  They are described by an antenna beam pattern, an antenna scan, and an attitude information. The distance between the emitters and the receiver is configurable.
  This receiver complexity is used in the "Localized Emitters" scenario.
- Direction finding (DF) receivers can have *up to 20 antenna elements*.
  Each antenna element is described by the combination of an antenna pattern, antenna position and attitude. The receiver itself is also described with its position and attitude information.
  This receiver complexity is used in the "Direction Finding" scenario.
  Option:R&S PULSE-K39 required for direction finding.

Table 13-1: Overview of the receiver types and provided settings

<table>
<tr><th>Scenario type</th><th>"Single Emitter"
"Emitters (Collection)"</th><th>"Localized Emitters"</th><th>"Direction Finding"</th></tr>
<tr><td>Separate repository elements</td><td>-</td><td>-</td><td>Yes</td></tr>
<tr><td>Related settings</td><td>-</td><td>Chapter 13.1, "Receiver properties in localized emitters scenario", on page 261</td><td>Chapter 13.2, "Receiver settings", on page 264</td></tr>
<tr><td>Scenario</td><td>Emitter-receiver</td><td>2D multi-emitter-receiver</td><td>2D multi-antenna receiver</td></tr>
<tr><td>2D map (gaming area)</td><td>-</td><td colspan="2">Yes</td></tr>
<tr><td>Emitter-receiver distance</td><td>Fixed</td><td colspan="2">Configurable
(Emitters placed on the 2D map)</td></tr>
<tr><td>Receiver attitude</td><td>-</td><td colspan="2">Configurable
(Pitch, yaw, roll, and height)</td></tr>
<tr><td>Number of antenna (elements)</td><td>1</td><td>1</td><td>≤ 20</td></tr>
<tr><td>Antenna pattern</td><td>Isotropic</td><td colspan="2">Configurable</td></tr>
<tr><td>Antenna scan</td><td>None (steady)</td><td>Configurable</td><td>-</td></tr>
<tr><td>Antenna attitude</td><td>-</td><td>-</td><td>Configurable
(Pitch, yaw, and roll per antenna element)</td></tr>
</table>

Received signal amplitude

One of the typical receiver characteristics is the minimum detective signal (MDS). The MDS is a measure of the receiver sensitivity and describes the minimum received signal amplitude S_{min} that the receiver is able to detect. A typical receiver achieves an S_{min} within the range of -80 dBm and -110 dBm.

In any type of emitter-receiver scenarios, the R&S Pulse Sequencer Digital calculates the received signal and displays the time variation of the received normalized power.

The principle is illustrated on Figure 13-1.

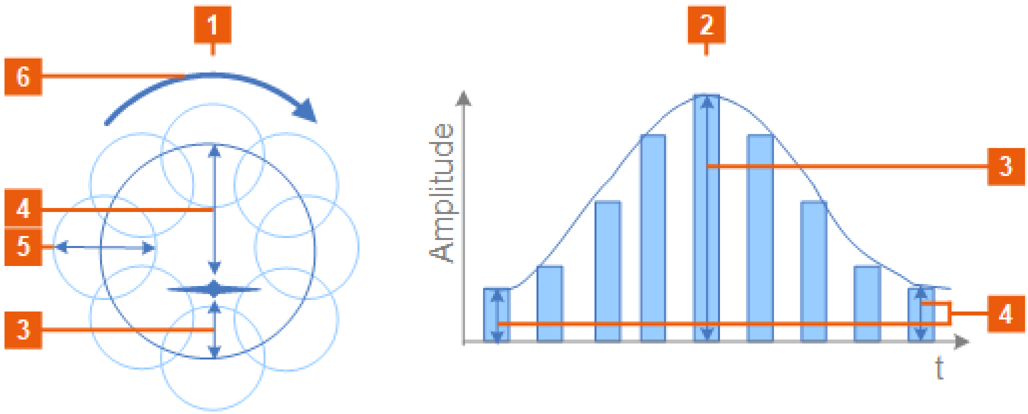


Figure 13-1: Simplified representation of a conical scan and the received signal

1 = Conical scan (boresight perspective)
2 = Amplitude variation of the received signal
3, 4 = Target-to-beam distance determines the amplitude of the received signal
5 = Antenna beam represented as its HPBW
6 = Direction of the rotating scan

The signal received by a static receiver is a sinusoidal waveform. The amplitude of the wave is proportional to the distance between the target and the beam axis. Figure 13-2 shows how this simple case of an emitter-receiver scenario is visualized in the software.

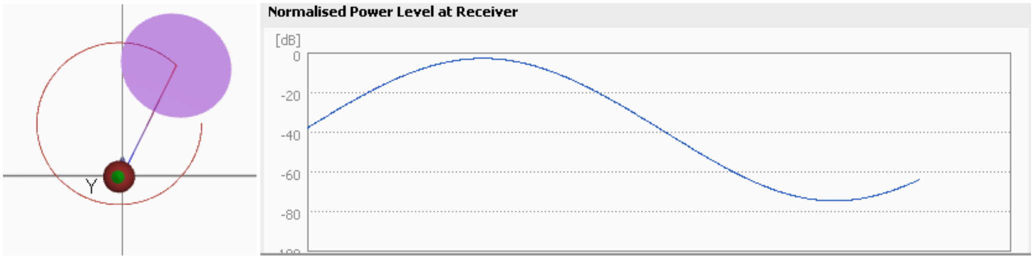


Figure 13-2: Normalized power level at receiver (simple emitter-receiver scenario)

For example, see "To visualize the signal received by a static receiver" on page 234.

For step-by-step instructions, see:

- "How to create a direction finding scenario" on page 291

For description of the provided settings, see:

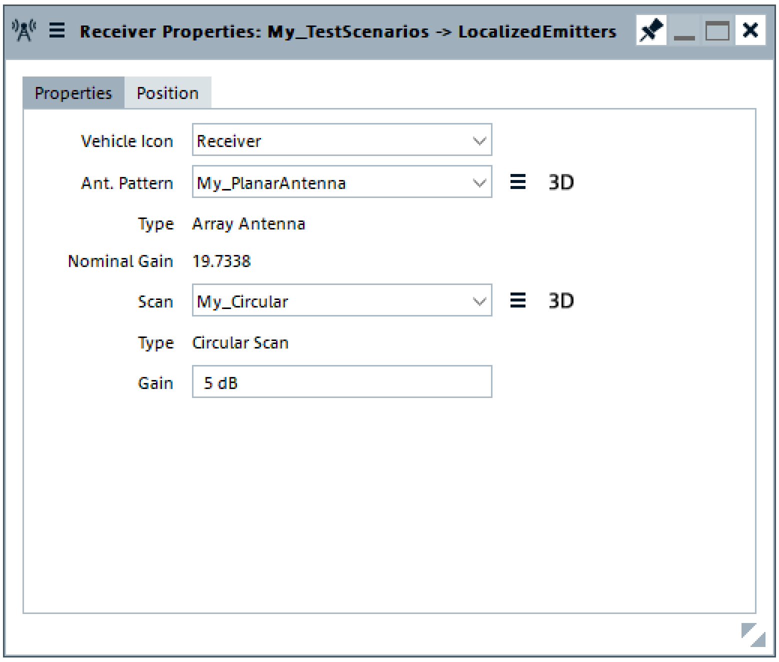
- Chapter 13.1, "Receiver properties in localized emitters scenario", on page 261
- Chapter 13.2, "Receiver settings", on page 264

- Chapter 13.3, "DF system configuration settings", on page 268

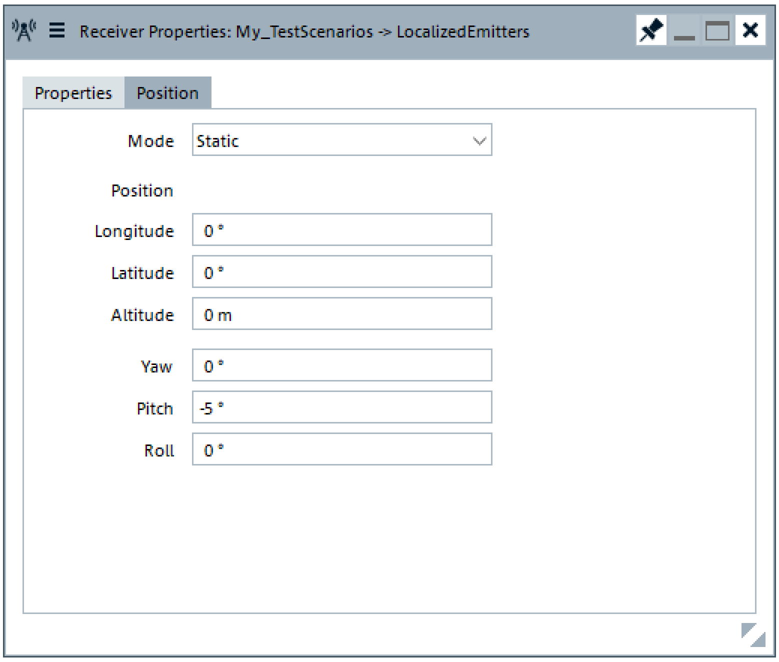
13.1 Receiver properties in localized emitters scenario

Access:

1. In the 2D map of a "Localized Emitters" scenario, select "Receiver Properties > Properties".



2. Select "Receiver Properties > Position".



Settings:

Vehicle Icon

Selects the icon for the receiver on the 2D map.

The icons are merely indication; the function and the properties of the receiver do not depend on the icon.

Remote command:
`SCENario:LOCalized:RECeiver:MOVement:VEHicle` on page 574

Antenna Pattern/Scan, Type, , 3D

Selects an antenna pattern and antenna scan.

3D opens a plot that visualizes the antenna pattern and scan.

Remote command:
`SCENario:LOCalized:RECeiver:ANTenna` on page 553
`SCENario:LOCalized:RECeiver:SCAN` on page 553

Nominal Gain

Displays the nominal gain of the antenna pattern.

Gain

Sets the antenna Gain.

Remote command:
`SCENario:LOCalized:RECeiver:GAIN` on page 554

Position > Latitude, Longitude, Altitude, Yaw, Pitch, Roll
The receiver's position is defined as latitude/longitude coordinates and attitude parameters. The available settings depend on the selected "Mode".

The position receiver settings in "Localized Emitters" and "Direction Finding" scenarios are identical.

For description, see Chapter 13.4, "Receiver position settings in localized emitters and direction finding scenarios", on page 270.

Remote command:
`SCENario:DF:RECeiver:LATitude` on page 552
`SCENario:DF:RECeiver:LONGitude` on page 552
`SCENario:DF:RECeiver:HEIGht` on page 553
`SCENario:LOCalized:RECeiver:LATitude` on page 552
`SCENario:LOCalized:RECeiver:LONGitude` on page 552
`SCENario:LOCalized:RECeiver:HEIGht` on page 553

13.2 Receiver settings

Access:

► Select "Repository Tree > DF Receiver Types > New".

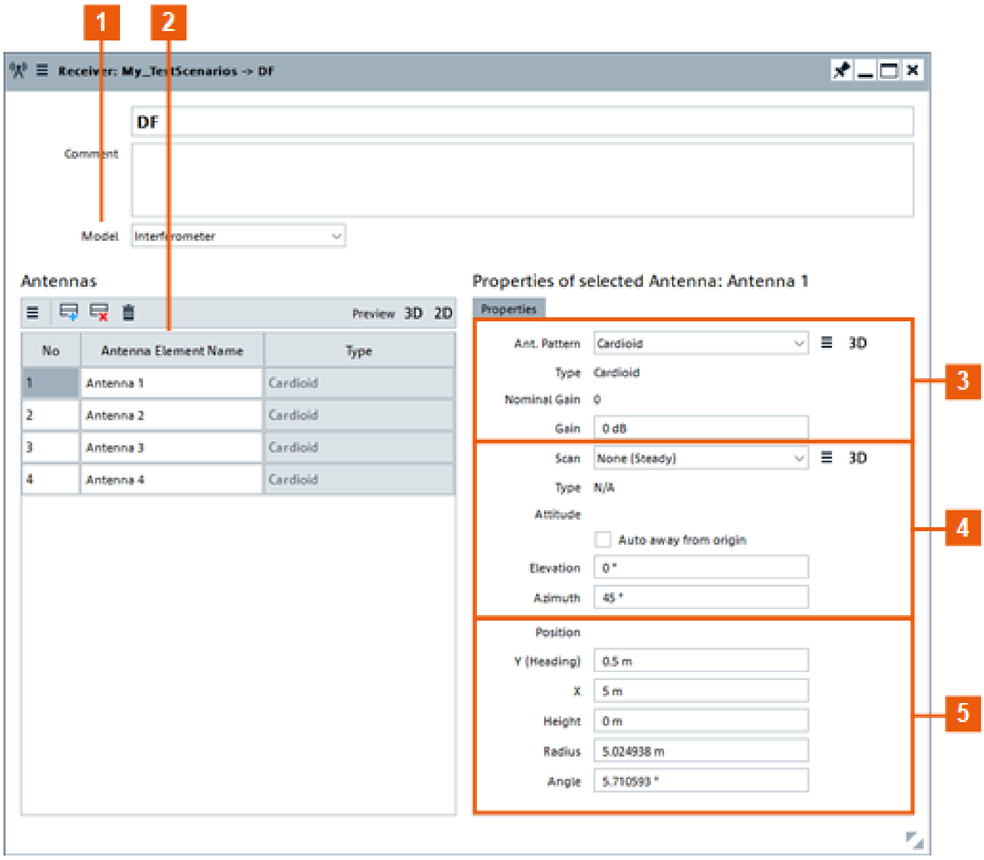


Figure 13-3: Receiver: understanding the displayed information

1 = Receiver model
2 = Up to 20 individual antenna elements
3 = Antenna pattern and gain per antenna element
4 = Altitude and orientation per antenna element
5 = Location and position information per antenna element

Direction finding (DF) receivers can have up to 20 antenna elements, each described with an antenna pattern and position.
See also "How to create a direction finding scenario" on page 291.

Settings:

Receiver Name

Enter the receiver's name.

Remote command:
`RECeiver:NAME` on page 427
`RECeiver:CREate` on page 426
`RECeiver:SELect` on page 426
`RECeiver:CATalog?` on page 426
`RECeiver:REMove` on page 428

Comment

Enter a short description.

Remote command:
`RECeiver:COMMent` on page 428

Model

Defines the model of the DF receiver.

"Interferometer"
: Interferometer direction finding receivers work phase coherent and evaluate the **phase difference** between the single antenna ports. They are usually **small**, with a diameter in the range of a wavelength. In this mode, the R&S Pulse Sequencer Digital calculates the relative phase difference between the incoming signal and the antenna positions. The differences in the time of arrival (TOA) of the incoming signals are assumed to be equal for all antenna to prevent small rounding errors in sample granularity.

"TDOA" (Time difference of arrival)
: TDOA direction finders use the **absolute time of arrival** of a signal to determine the direction. They do not evaluate phase information, because they are usually **big-sized**. In TDOA receiver, the antenna ports are placed far from one another so that the time delay is large enough and suitable for analysis.
Examples of TDOA receivers are the antennas mounted at the two wings of an airplane or at the outer borders of a ground station. Because of the distance between the antennas, there is a difference in the time the incoming signal arrives at the antennas. In this mode, the R&S Pulse Sequencer Digital calculates the absolute TOA of the incoming signal for each antenna. The phase difference between the single antenna ports is calculated from their absolute distance to the emitter.

"Combined (Interferometer/TDOA)"
: The combined model calculates the relative phases between the antenna ports and calculates the individual TOAs for each antenna port.

Remote command:
RECeiver:MODel on page 523

Antenna Configuration

DF receiver can have up to 10 antenna elements.

"Positioning Preview 2D"

Displays the distribution of the antenna elements in a 2-dimensional chart (XY plane).

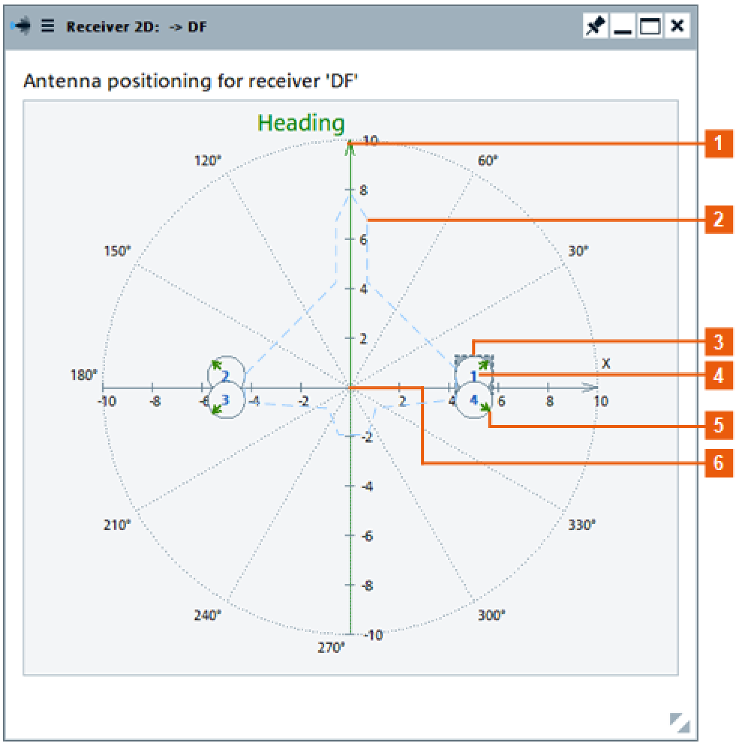


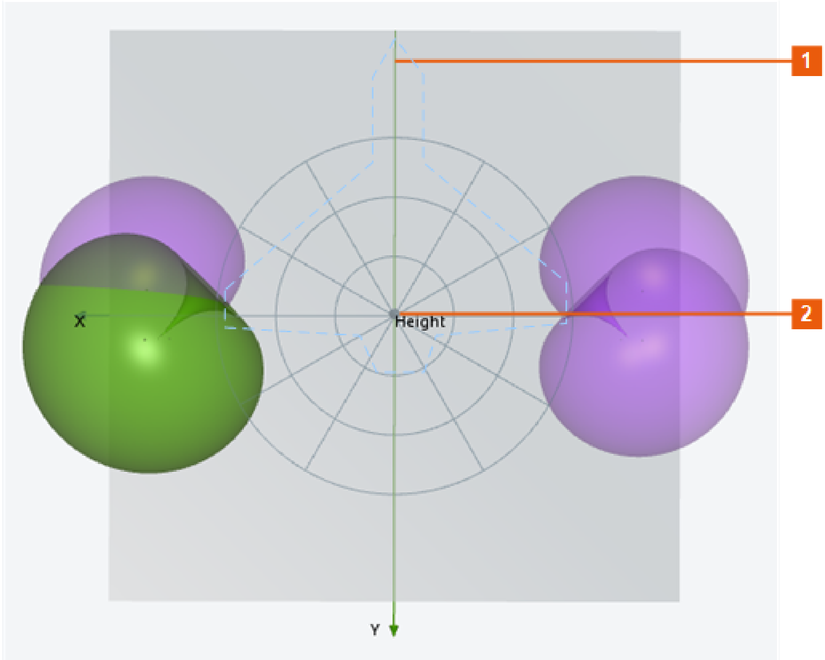
Figure 13-4: Antenna Positioning: understanding the displayed information

1 = Heading (y axis); indicated with a colored line
2 = Airplane graphic is not displayed in the software (shown here for better understanding)
3 = Selected antenna element
4 = Antenna element consequent number, as listed in the "List of antenna elements" table
5 = Antenna element orientation
6 = Receiver origin

See also "How to create a direction finding scenario" on page 291.

"Positioning Preview 3D"

Displays the distribution of the antenna elements in a 3-dimensional chart (XY plane).



1 = Heading (y axis); indicated with a colored line
2 = Receiver origin

"List of antenna elements"

Use the standard functions in the context menu and the icons to add, reorder, or remove antenna elements.
New elements are named automatically but you can change the alias name.

Remote command:
RECeiver:ANTenna:ADD on page 429
RECeiver:ANTenna:SELect on page 431
RECeiver:ANTenna:ALIas on page 523
RECeiver:ANTenna:DELete on page 432
RECeiver:ANTenna:CLEar on page 433

Properties of selected Antenna: Antenna x
Describes the selected antenna element.

Open the "2D/3D Positioning Preview" view to display the configuration, see Figure 13-4.

Properties ← Properties of selected Antenna: Antenna x
Comprises the antenna and receiver orientation-related settings:

"Antenna Pattern, Antenna Scan, Type"
Selects an existing antenna pattern or accesses a dialog for creating a new one.
The antenna pattern, scan and type are displayed.
See also "To create an antenna pattern" on page 207.

Remote command:
`RECeiver:ANTenna:PATTern` on page 524
`RECeiver:ANTenna:SCAN` on page 525

"Nominal Gain" Displays the nominal gain of the antenna pattern.

"Gain" Sets the antenna Gain.

Remote command:
`RECeiver:ANTenna:GAIN` on page 525

"Pointing Direction"
Turns the antenna beam axis in the elevation and the azimuth.
"Auto away form origin" sets the azimuth automatically, so that the beam axis is radial to the receiver origin.

Remote command:
`RECeiver:ANTenna:DIRection:AWAY` on page 525
`RECeiver:ANTenna:DIRection:AZIMuth` on page 525
`RECeiver:ANTenna:DIRection:ELEVation` on page 526

Position ← Properties of selected Antenna: Antenna x
Antenna elements are spaced relative to the receiver origin.

Their 3D positions are defined as:
- "X" and "Y (heading)" values relative to the receiver origin.
- "Angle" (azimuth plane) and "Radius" (distance to the receiver origin on the XY plane), where "Angle = 0°" is on the X-axis.
- "Height" offset on the Z-axis relative to the receiver origin.

Example:
"X = 0 m" and "Y = -0.14 m" corresponds to "Radius = 0.14 m" and "Angle = 270 °".

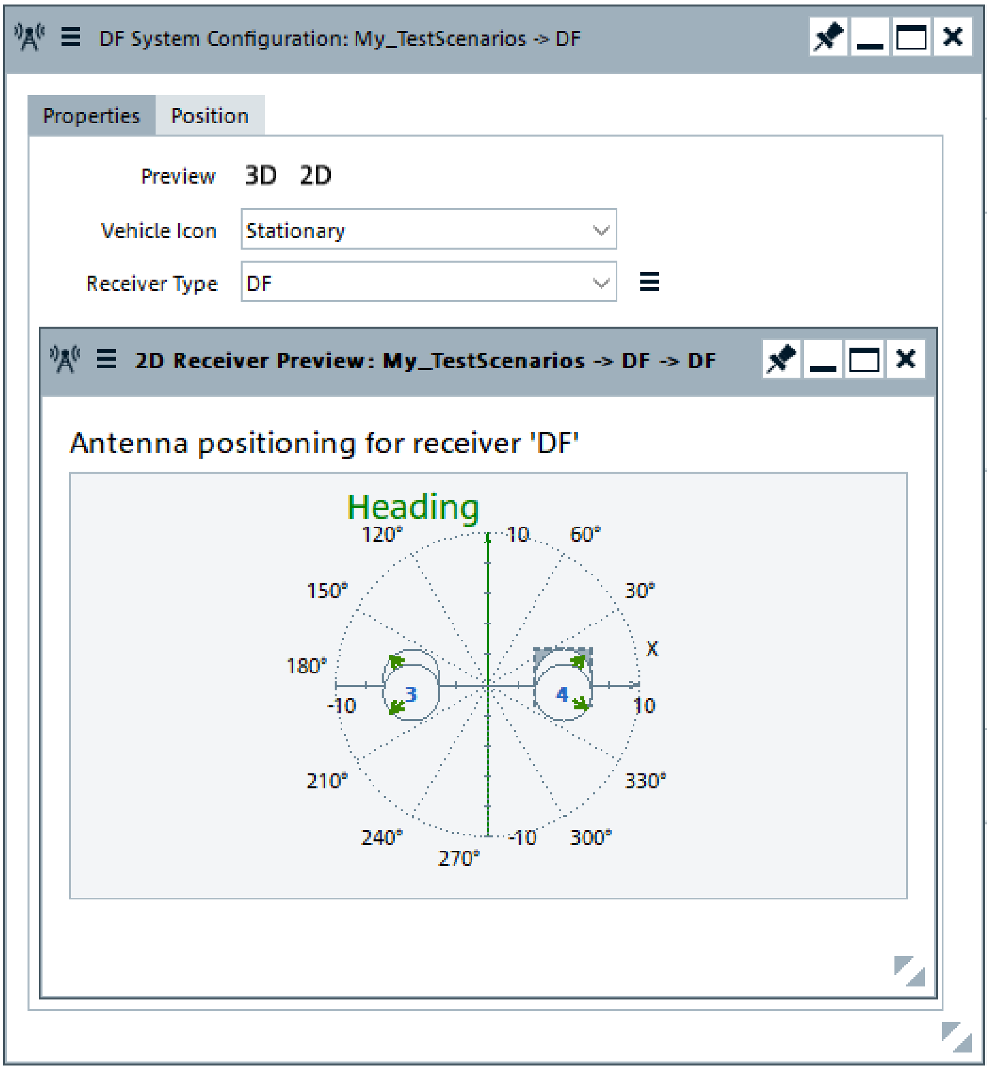
Remote command:
`RECeiver:ANTenna:POSition:X` on page 524
`RECeiver:ANTenna:POSition:Y` on page 524
`RECeiver:ANTenna:POSition:RADius` on page 524
`RECeiver:ANTenna:POSition:ANGLe` on page 524
`RECeiver:ANTenna:POSition:HEIGht` on page 524

13.3 DF system configuration settings

Option:R&S PULSE-K39

Access:

1. Open a "Direction Finding" scenario.
2. In the block diagram, select "Map > 2D".
3. On the 2D map, select "Receiver > Properties".
4. Select "2D/3D" to display the "Antenna Positioning".



See also "How to create a direction finding scenario" on page 291.

Settings:

Properties
Each receiver is described with the following parameters:

Vehicle Icon ← Properties
Selects the icon for the receiver on the 2D map.

The icons are merely indication; the function and the properties of the receiver do not depend on the icon.

Remote command:
`SCENario:DF:RECeiver:MOVement:VEHicle` on page 574

Receiver ← Properties
Selects an existing receiver or accesses a dialog for creating a new one.

Remote command:
`SCENario:DF:RECeiver` on page 556

Position > Latitude, Longitude, Height
The receiver's position is defined as latitude/longitude coordinates and attitude parameters. The available settings depend on the selected "Mode".

The position receiver settings in "Localized Emitters" and "Direction Finding" scenarios are identical.

For description, see Chapter 13.4, "Receiver position settings in localized emitters and direction finding scenarios", on page 270.

Antenna Positioning
Displays the distribution of the antenna elements on a 2D map (XY plane).

The graph resembles the one in the "Receiver" dialog, see Figure 13-4.

Use the split window icon to detach the diagram and open it in a separate window.

13.4 Receiver position settings in localized emitters and direction finding scenarios

Option:R&S PULSE-K39 required for direction finding.

As with the emitters, you can also define receivers that move along a trajectory with a defined shape. The receiver configuration is similar to the configuration of the moving emitters.

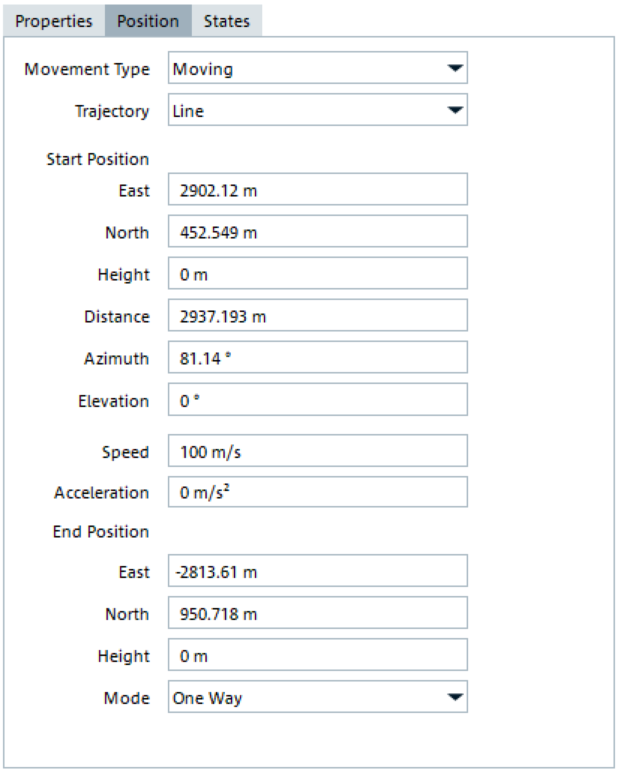
For introduction to the topic and details on the waypoint files, see:
- Chapter 16.5, "Moving emitters and platforms", on page 323
- Chapter A.3, "Movement files", on page 638.

Access:

1. In the 2D map of a "Localized Emitters" or "Direction Finding" scenario, select "Receiver Properties > Properties".

2. Select "Position".

   The available settings depend on the selected "Mode".



   For example of possible further configurations, see Chapter 16.5, "Moving emitters and platforms", on page 323.

Settings:

Mode
Sets if the receiver is static or moving.

Option:R&S PULSE-K39 is required for moving receivers.

"Static" The receiver is static; its coordinates are fixed during the whole scenario.
See Start Position > Latitude, Longitude, Height and Attitude > Pitch, Yaw, Roll.

"Moving" The receiver is moving from defined start position and follows the selected trajectory.
See Trajectory.

Remote command:
`SCENario:LOCalized:LOCation:REC:PMODe` on page 569

Trajectory
Defines the trajectory shape.

"Line" The receiver is moving on a straight line, starting from its current position (Start Position > Latitude, Longitude, Height) and ending at the selected "End Position".

"Arc" The receiver moves on a part of a circle, where the center of the circle is defined with the "Center Position".

"Waypoint" Receiver with varying coordinates, defined in waypoints files.
You find a subset of example files in installation package but you can also create and load your own files.

"Trace" The receiver follows a user-defined route. You can define the route in the map view using the "Trace" function. R&S Pulse Sequencer Digital generates waypoints from the trace, automatically.

Remote command:
`SCENario:LOCalized:RECeiver:MOVement:TYPE` on page 569
`SCENario:DF:RECeiver:MOVement:TYPE` on page 569

Start Position > Latitude, Longitude, Height
Sets the receiver's position.

Remote command:
SCENario:DF:RECeiver:LATitude on page 552
SCENario:DF:RECeiver:LONGitude on page 552
SCENario:LOCalized:RECeiver:HEIGht on page 553

Attitude > Pitch, Yaw, Roll
For static receivers, defines the orientation of the antenna as "Pitch", "Yaw", "Roll".

See Table 10-1.

Remote command:
SCENario:LOCalized:RECeiver:DIRection:YAW on page 548
SCENario:LOCalized:RECeiver:DIRection:PITCh on page 549
SCENario:LOCalized:RECeiver:DIRection:ROLL on page 549
SCENario:DF:RECeiver:DIRection:YAW on page 548
SCENario:DF:RECeiver:DIRection:PITCh on page 549
SCENario:DF:RECeiver:DIRection:ROLL on page 549

Line trajectory definition
A line is defined with:

Speed ← Line trajectory definition
Sets the speed of the moving receiver.

Remote command:
SCENario:LOCalized:RECeiver:MOVement:SPEed on page 572
SCENario:DF:RECeiver:MOVement:SPEed on page 572

Acceleration ← Line trajectory definition
Sets the acceleration of the moving receiver.

Remote command:
SCENario:LOCalized:RECeiver:MOVement:ACCeleration on page 572
SCENario:DF:RECeiver:MOVement:ACCeleration on page 572

End Position ← Line trajectory definition
Sets the "East/North" coordinates and the "Height" of the receiver at the end of the movement.

Remote command:
SCENario:LOCalized:RECeiver:MOVement:EAST on page 572
SCENario:LOCalized:RECeiver:MOVement:NORTh on page 573
SCENario:LOCalized:RECeiver:MOVement:HEIGht on page 573
SCENario:DF:RECeiver:MOVement:EAST on page 572
SCENario:DF:RECeiver:MOVement:NORTh on page 572
SCENario:DF:RECeiver:MOVement:HEIGht on page 573

Mode ← Line trajectory definition
Defines the behavior of the moving object when the end of the trajectory is reached.

"Cyclic" The trajectory file is repeated cyclically. Once the "End Position" is reached, file movement starts again from the beginning.

Using this mode is recommended in the following cases:
- A circle trajectory
- A trajectory in which the start and the end positions are close to each other.

"Round Trip" The emitter moves back and forth on the defined trajectory.

"One Way" The movement is executed once.
When the "End Position" is reached, this position is assumed to be a static one.

Remote command:
`SCENario:DF:RECeiver:MOVement:RMODe` on page 576
`SCENario:LOCalized:RECeiver:MOVement:RMODe` on page 576

Arc trajectory definition
An arc is defined with:

Speed ← Arc trajectory definition
Sets the speed of the moving receiver.

Remote command:
`SCENario:LOCalized:RECeiver:MOVement:SPEed` on page 572
`SCENario:DF:RECeiver:MOVement:SPEed` on page 572

Angle ← Arc trajectory definition
Sets the arc angle and thus defines the arc length.

Remote command:
`SCENario:LOCalized:RECeiver:MOVement:ANGLe` on page 572
`SCENario:DF:RECeiver:MOVement:ANGLe` on page 572

Center Position ← Arc trajectory definition
Sets the "East/North" coordinates of the center of the circle on that the receiver is moving.

Remote command:
`SCENario:LOCalized:RECeiver:MOVement:EAST` on page 572
`SCENario:LOCalized:RECeiver:MOVement:NORTh` on page 573
`SCENario:DF:RECeiver:MOVement:EAST` on page 572
`SCENario:DF:RECeiver:MOVement:NORTh` on page 572

Mode ← Arc trajectory definition
Defines the behavior of the moving object when the end of the trajectory is reached.

"Cyclic" The trajectory file is repeated cyclically. Once the "End Position" is reached, file movement starts again from the beginning.

Using this mode is recommended in the following cases:
- A circle trajectory
- A trajectory in which the start and the end positions are close to each other.

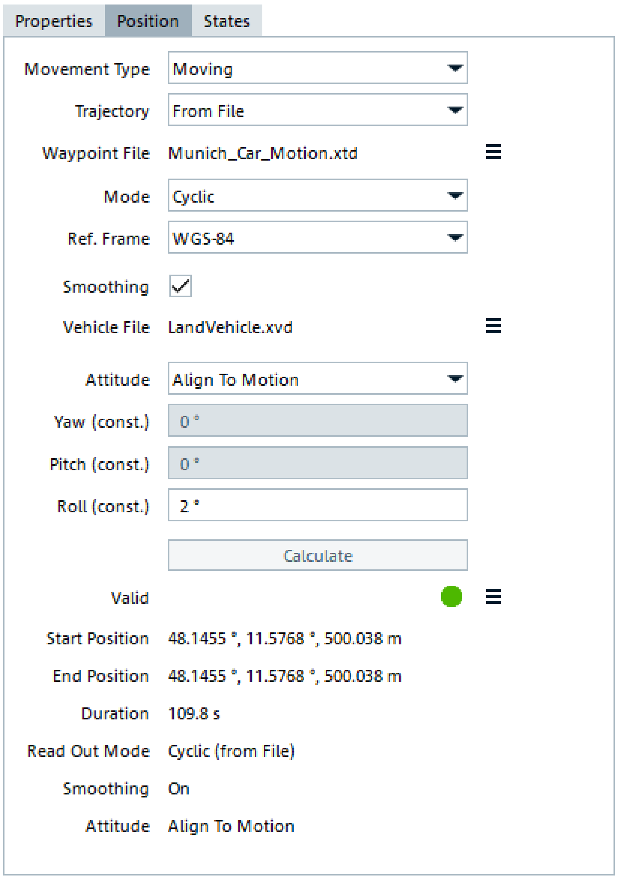
"Round Trip" The emitter moves back and forth on the defined trajectory.

"One Way" The movement is executed once.
When the "End Position" is reached, this position is assumed to be a static one.

Remote command:
`SCENario:DF:RECeiver:MOVement:RMODe` on page 576
`SCENario:LOCalized:RECeiver:MOVement:RMODe` on page 576

Waypoints trajectory definition
A waypoint file is description of a movement with possibly attitude coordinates that can have different forms, like, for example, a sequence of positions or vector arc movement. A waypoint file must have the extension `*.txt`, `*.kml` or `*.xtd`. For description, see Chapter A.3, "Movement files", on page 638.

You find a subset of example files in installation package but you can also create and load custom-specific files. Moreover, waypoint smoothing can be activated if vehicle description files are used.



If waypoint files are used, the receiver position and trajectory are retrieved from the file content; the traditional position coordinates parameters are disabled.

Waypoint File ← Waypoints trajectory definition
Indicates the name of the currently used waypoint file.

Load, Delete ← Waypoints trajectory definition

Opens the Windows explorer so that you can navigate to and load waypoint file.

Discards the selected waypoint file.

Remote command:
`SCENario:DF:RECeiver:MOVement:WAYPoint` on page 573
`SCENario:DF:RECeiver:MOVement:WAYPoint:CLEar` on page 573
`SCENario:LOCalized:RECeiver:MOVement:WAYPoint` on page 573
`SCENario:LOCalized:RECeiver:MOVement:WAYPoint:CLEar` on page 574

Read Out Mode ← Waypoints trajectory definition

Defines the way that the waypoint file is processed.

"Cyclic" The waypoint file is processed cyclically. Once the last waypoint is reached, file processing starts again from the beginning.

Using this mode is recommended if the waypoint file describes one of the following:
- A circle trajectory
- A trajectory in which the start and the end positions are close to each other.

"Round Trip" By reaching the end of the file, the file is processed backwards.

"One Way" The file is processed once.
By reaching the end of the file, the last described position is assumed to be a static one.

Remote command:
`SCENario:DF:RECeiver:MOVement:RMODe` on page 576
`SCENario:LOCalized:RECeiver:MOVement:RMODe` on page 576

Reference Frame ← Waypoints trajectory definition

Select the reference frame used to define the emitters coordinates.

The transformation between the reference frames is performed automatically.

The following applies:
- $X_{WGS84} = (1 - 0.008*10^{-6})*X_{PZ\ 90} - 0.2041*10^{-7}*Y_{PZ\ 90} + 0.1716*10^{-7}*Z_{PZ\ 90} - 0.013$
- $Y_{WGS84} = (1 - 0.008*10^{-6})*Y_{PZ\ 90} - 0.2041*10^{-7}*X_{PZ\ 90} + 0.1115*10^{-7}*Z_{PZ\ 90} + 0.106$
- $Z_{WGS84} = (1 - 0.008*10^{-6})*Z_{PZ\ 90} - 0.1716*10^{-7}*X_{PZ\ 90} - 0.1115*10^{-7}*Y_{PZ\ 90} + 0.022$

Both reference frames are ECEF frames with a set of associated parameters.

"WGS-84" The world geodetic system WGS-84 is the reference frame used by GPS.

"PZ 90.11 (GLONASS)"
Parametry Zemli PZ (parameters of the Earth) is a reference frame, used, for example, by GLONASS.

Remote command:
`SCENario:DF:RECeiver:MOVement:RFRame` on page 576
`SCENario:LOCalized:RECeiver:MOVement:RFRame` on page 576

Smoothing > State ← Waypoints trajectory definition
The discrete positions (waypoints) defined in the waypoints file can cause abrupt changes in the movement direction.

The R&S Pulse Sequencer Digital provides an internal interpolating algorithm that smooths the movement or the trajectory. This algorithm evaluates the dedicated vehicle description (`*.xvd`) file, retrieves the velocity vector and the `<proximity>` value, and inserts waypoints to smooth the trajectory. The resulting movement is more realistic.

To use the algorithm, enable "Smoothing", select "Vehicle File > Load" to load a `*.xvd` file and select "Calculate". See Chapter A.4, "Vehicle description files (Used for smoothening)", on page 645.

Remote command:
`SCENario:DF:RECeiver:MOVement:SMOothening` on page 576
`SCENario:LOCalized:RECeiver:MOVement:SMOothening` on page 576

Vehicle File ← Waypoints trajectory definition
Indicates the name of the currently used vehicle description (`*.xvd`) file.

Load, Delete ← Waypoints trajectory definition
Opens the Windows explorer so that you can navigate to and load vehicle description file.

Discards the selected vehicle description file.

Remote command:
`SCENario:DF:RECeiver:MOVement:VFILe` on page 575
`SCENario:DF:RECeiver:MOVement:VFILe:CLEar` on page 575
`SCENario:LOCalized:RECeiver:MOVement:VFILe` on page 575
`SCENario:LOCalized:RECeiver:MOVement:VFILe:CLEar` on page 575

Attitude Behavior ← Waypoints trajectory definition
Defines how the attitude information is defined.

"From Waypoint File"
The attitude parameters are extracted from the selected waypoint file. Further settings are not required.
This setting forces the attitude parameters to motion direction even if the waypoint has attitude information, like, for example, in a `*.xtd` file with `<property waypointformat="position_attitude">`.
For specific emitters like ships or land vehicles, it is realistic to set the yaw and pitch to vehicle's motion direction. This is because the usual body axes angles of a land vehicle are in the direction of the velocity vector.
For other emitters, however, like landing plane, this parameter is not useful. As an example, the nose of the plane is in an upward direction at the time when the plane is moving downwards.

"Align to Motion"
Enables a constant rate of change of the roll.

"Constant" Emitter's attitude is set as the combination of the "Yaw/Heading", "Pitch/Elevation", "Roll/Bank" values.
The resulting attitude is a constant value.

Remote command:
`SCENario:DF:RECeiver:MOVement:ATTitude` on page 574
`SCENario:LOCalized:RECeiver:MOVement:ATTitude` on page 574

Yaw/Heading, Pitch/Elevation, (Start) Roll/Bank ← Waypoints trajectory definition
Sets the angles of rotation in the corresponding direction, i.e. the rotation around the respective yaw, pitch and roll axes. "Yaw/Heading, Pitch/Elevation, Roll/Bank" are defined relative to the local horizon.

Remote command:
`SCENario:DF:RECeiver:MOVement:YAW` on page 575
`SCENario:DF:RECeiver:MOVement:PITCh` on page 575
`SCENario:DF:RECeiver:MOVement:ROLL` on page 575
`SCENario:LOCalized:RECeiver:MOVement:YAW` on page 575
`SCENario:LOCalized:RECeiver:MOVement:PITCh` on page 575
`SCENario:LOCalized:RECeiver:MOVement:ROLL` on page 575

Calculate ← Waypoints trajectory definition
Loads the selected waypoint and vehicle description file.

The "Movement Data" display indicates a summary of the major parameters:
- Start/End position
- Duration of the movement
- Read-out mode as defined in the file
- Smoothing state
- Attitude behavior

The status LED indicates the following states:
- **Red**: the waypoint file is not selected, the file is not imported and the movement not calculated or the file is erroneous.
- **Green**: Movement is calculated.

Remote command:
`SCENario:DF:RECeiver:MOVement:IMPort` on page 576
`SCENario:LOCalized:RECeiver:MOVement:IMPort` on page 577

Clear ← Waypoints trajectory definition
Discards the waypoint and vehicle description file.

Remote command:
`SCENario:DF:RECeiver:MOVement:CLEar` on page 577
`SCENario:LOCalized:RECeiver:MOVement:CLEar` on page 577

Trace trajectory definition
The "Trace" trajectory definition is similar to the "Waypoint" trajectory definition.

The differences are:

- The "Waypoint" trajectory is defined by a file.
  The waypoint file contains all needed coordinates so no additional settings are required.

- The "Trace" trajectory is defined by a set of user-defined "Trace Points".
  No coordinate information is available, so additional settings are required. As with a "Line" or "Arc" trajectory, the start position must be specified. Also, the position of each trace point must be individually configured.

A trace is defined with:

Start Position > Latitude, Longitude ← Trace trajectory definition
Sets the receiver's starting position.

Remote command:
`SCENario:DF:RECeiver:LATitude` on page 552
`SCENario:DF:RECeiver:LONGitude` on page 552

Trace Points ← Trace trajectory definition
Selects a specific trace point.

Remote command:
`SCENario:LOCalized:RECeiver:MOVement:PSTep:SELect` on page 571
`SCENario:DF:RECeiver:MOVement:PSTep:SELect` on page 571

Speed ← Trace trajectory definition
Sets the speed of the moving receiver at the selected trace point.

Remote command:
`SCENario:LOCalized:RECeiver:MOVement:SPEed` on page 572
`SCENario:DF:RECeiver:MOVement:SPEed` on page 572

East, North, Height ← Trace trajectory definition
Sets the "East/North" coordinates and the "Height" of the receiver at the selected trace point.

Remote command:
`SCENario:LOCalized:RECeiver:MOVement:EAST` on page 572
`SCENario:LOCalized:RECeiver:MOVement:NORTh` on page 573
`SCENario:LOCalized:RECeiver:MOVement:HEIGht` on page 573
`SCENario:DF:RECeiver:MOVement:EAST` on page 572
`SCENario:DF:RECeiver:MOVement:NORTh` on page 572
`SCENario:DF:RECeiver:MOVement:HEIGht` on page 573

Mode ← Trace trajectory definition
Defines the behavior of the moving object when the end of the trajectory is reached.

"Cyclic"
The trajectory file is repeated cyclically. Once the "End Position" is reached, file movement starts again from the beginning.
Using this mode is recommended in the following cases:
- A circle trajectory
- A trajectory in which the start and the end positions are close to each other.

"Round Trip"
The emitter moves back and forth on the defined trajectory.

"One Way" The movement is executed once.
When the "End Position" is reached, this position is assumed to be a static one.

Remote command:
SCENario:DF:RECeiver:MOVement:RMODe on page 576
SCENario:LOCalized:RECeiver:MOVement:RMODe on page 576

14 Working with imported signals

In addition to creating pulse sequences in R&S Pulse Sequencer Digital, you can also create sequences composed of imported signals.

If the data format is in one of the supported import formats listed below, you can import the pulses into the pulse library.

Supported file types

- Custom text-based PDW lists in ASCII or coma-separated file format (`*.txt` or `*.csv`), see Chapter 18.2, "PDW import mechanism", on page 362. You can use the imported signals to retrieve information on the reference signal level and to change it.
- AMMOS IF (`*.aif`), AMMOS PDW (`*.pdw` or `*.ppdw`) and AMREC recording (`*.dat`) files
  Rohde & Schwarz proprietary data formats used with Rohde & Schwarz monitoring equipment. These files typically contain multiple data streams.
  The R&S Pulse Sequencer Digital extracts IF (I/Q) or PDW data from these files.

Related settings

Import Wizard
The data import wizard converts custom I/Q data into a Rohde & Schwarz waveform file.

The dialog and the provided settings differ depending on the selected file format but the differences are self-explanatory.

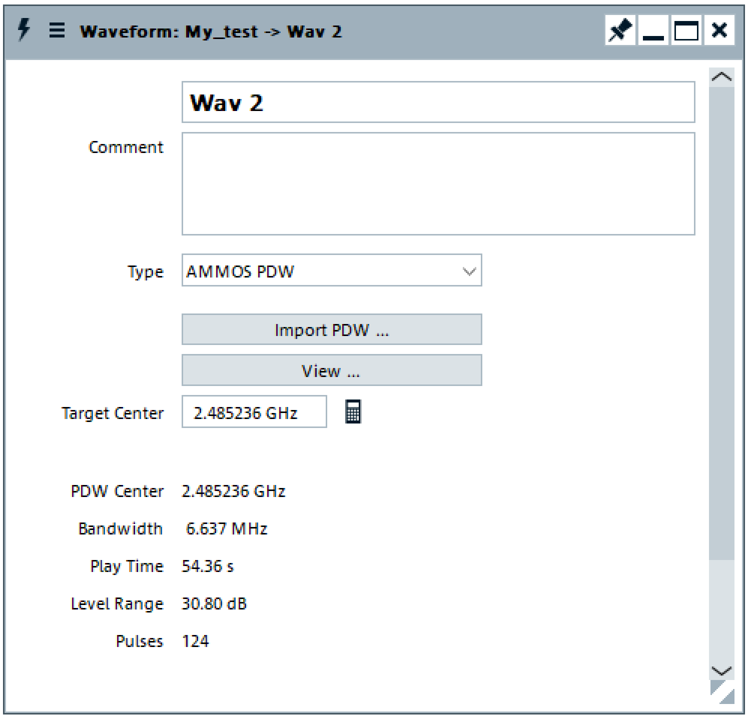
AMMOS IF and AMMOS PDW Import
AMMOS IF, AMREC recording files and AMMOS PDW are Rohde & Schwarz proprietary data formats used with Rohde & Schwarz monitoring equipment. These files typically contain multiple data streams.

The R&S Pulse Sequencer Digital extracts IF (I/Q) or PDW data from these files.

The import process and wizard are described in:
- "Import Wizard" on page 281
- Chapter 18.4, "PDW data import wizard settings", on page 365.

The following is a description of the dedicated settings.

Access: "Imported Signals > Type = AMMOS PDW".



"Import PDW" Accesses a file import wizard, see "Import Wizard" on page 281.

"View"

Access dialogs which visualize the waveform and displays more useful information on it.

"Target Center"

Sets the center frequency of the generator.
When a PDW list file is imported, the value is set to the automatically calculated center frequency of the PDW file. The value resembles the value indicated as "PDW Center".
The frequency offsets of the pulses are recalculated. The bandwidth increases.

"Calculate" The "Calculate" icon resets the "Target Center" field to the value displayed as "PDW Center".

"PDW Center, Bandwidth, Play Time, Level Range, Pulses"

If the PDW list file is imported in the repository, the dialog displays information on the waveform:

- "PDW Center": *Calculated center frequency*
  The indicated value is the center frequency that the software uses during the import process. The pulses are calculated relatively to this value.
  The "PDW Center" is calculated as the middle frequency between the min and the max frequency values included in the PDW file. Chirp frequency deviations are considered.
- "Bandwidth": Calculated bandwidth
- "Play Time": Duration
- "Level Range": Calculated level range
- "Pulses": Number of pulses in the waveform.

Input File ← AMMOS IF and AMMOS PDW Import

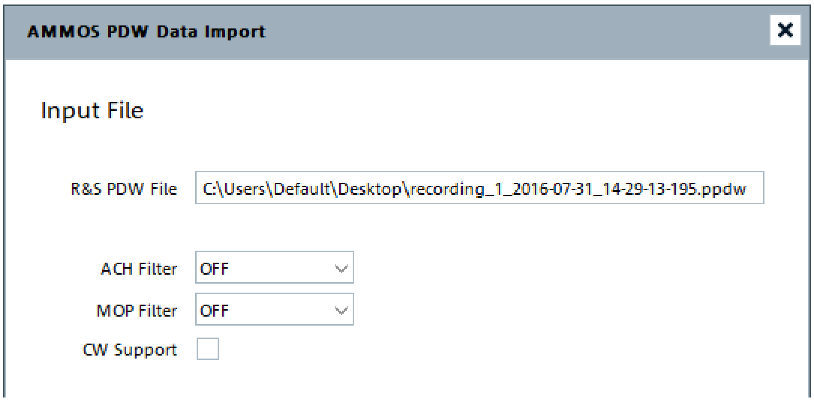


Figure 14-1: Import Wizard for Type = AMMOS PDW

"R&S PDW File"
Selects the source file.

"ACH Filter" Sets the analysis channel filter that defines which of the four analysis channels within the bandwidth is used.
If ACH filter is not used ("ACH Filter = Off"), the import mechanism discards all pulses that overlap with the current one. This behavior cannot be changed. To ensure processing of one pulse per time, apply an ACH filter.

"MOP Filter" Filters out pulses based on the used modulation.

"CW Support" Enables import and replay of CW signals.

Level Attenuation ← AMMOS IF and AMMOS PDW Import

Graphical representation of the retrieved information, together with statistical information on the processed data.

Displayed is the time variation of the following parameters:

- Level Attenuation
- PRI
- PW
- Frequency

See also Chapter 15.1, "Signal preview settings", on page 284.

15 Visualizing and analyzing signals

Signals generated with the software or imported into the repository can be visualized with the built-in "Signal Preview" function. The R&S Pulse Sequencer Digital reads a generated signal, evaluates it and displays the I and Q data, the spectrum, and the constellation diagram of a signal section or of the entire signal. The "Signal Preview" is only available if **a signal was calculated (created) successfully**.

Moreover, with R&S Pulse Sequencer Digital you can visualize the content of signals in Rohde & Schwarz format created by external software. Double-clicking the signal name in your file explorer automatically opens the "Signal Preview" dialog. Furthermore, you can select the R&S Pulse Sequencer Digital as the default program for opening such files.

15.1 Signal preview settings

R&S PULSE-K39.

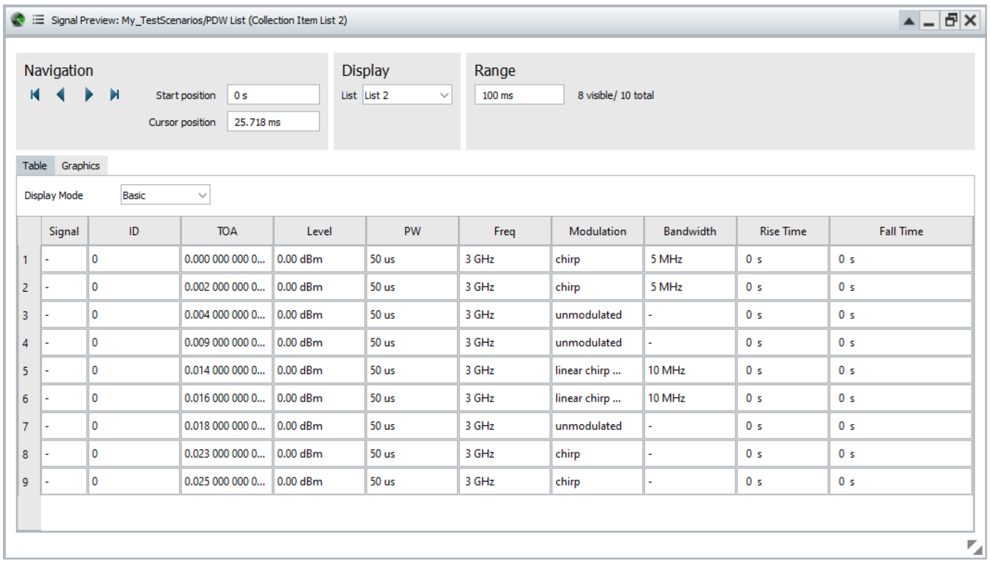
Per default, the R&S Pulse Sequencer Digital evaluates the settings and creates simulated signals out of them.

However, if extending sequencing functionality is used, the output is in Rohde & Schwarz proprietary file format.

Files generated in this way cannot be evaluated as I and Q data, as this evaluation is done with the waveforms. Instead, a dedicated "Signal Preview" dialog displays the main pulse characteristics in a table form.

Access:

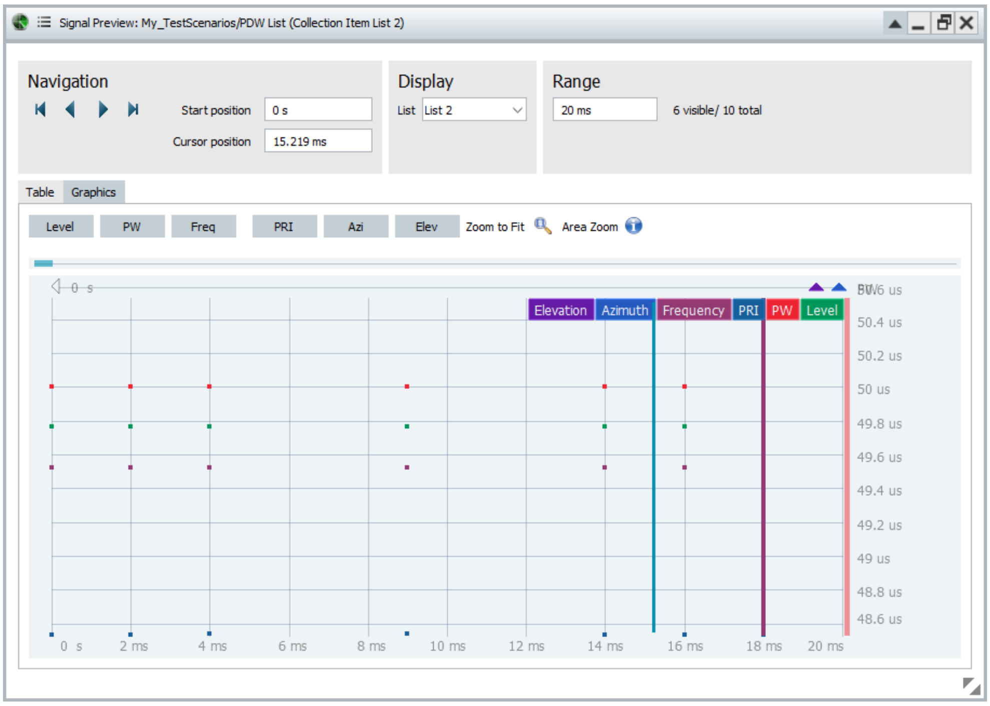
1. In the "Scenario" dialog, depending on the scenario type, select "Signal Generation > Config > [Target] >Destinations".
2. Select "Signal Calculation > Calculate".
3. Select "Signal Calculation > Results > View" and select the required file.
4. In the "Signal Preview" dialog, select "Table".



	Signal	ID	TOA	Level	PW	Freq	Modulation	Bandwidth	Rise Time	Fall Time
1	-	0	0.000 000 000 0...	0.00 dBm	50 us	3 GHz	chirp	5 MHz	0 s	0 s
2	-	0	0.002 000 000 0...	0.00 dBm	50 us	3 GHz	chirp	5 MHz	0 s	0 s
3	-	0	0.004 000 000 0...	0.00 dBm	50 us	3 GHz	unmodulated	-	0 s	0 s
4	-	0	0.009 000 000 0...	0.00 dBm	50 us	3 GHz	unmodulated	-	0 s	0 s
5	-	0	0.014 000 000 0...	0.00 dBm	50 us	3 GHz	linear chirp ...	10 MHz	0 s	0 s
6	-	0	0.016 000 000 0...	0.00 dBm	50 us	3 GHz	linear chirp ...	10 MHz	0 s	0 s
7	-	0	0.018 000 000 0...	0.00 dBm	50 us	3 GHz	unmodulated	-	0 s	0 s
8	-	0	0.023 000 000 0...	0.00 dBm	50 us	3 GHz	chirp	-	0 s	0 s
9	-	0	0.025 000 000 0...	0.00 dBm	50 us	3 GHz	chirp	-	0 s	0 s

5. Select, for example, "Graphics > Frequency".
6. Select "Zoom to Fit".

   The "Level", "PW", "Frequency", "PRI", "Azi" and "Elev" graphs visualize the variation of the corresponding parameter over time.



7. You can observe the variation of one or several PDW parameters simultaneously. Select a parameter to include it on the display.

   The display is color-coded.
8. Click the parameter name to change the units on the y-axis.

The color bar on the right of the display indicates the current PDW parameter.

Settings:

Navigation
Steps forward or backward to the first or last page.

Start Position
Sets the start time of the current page.

Cursor Position
Sets the positon of the cursor, expressed in seconds.

Display
Selects a list to display if more than one list is used in the calculation.

Range
Selects the time range of the current page.

Selecting a lower value reduces the number of pulses visible. The "visible/total" display next to the field updates accordingly.

You can also change this value by holding the cursor over the graph and using the mouse scroll wheel.

Table
Lists the file content in table form.

Select "Display Mode" > "Basic" / "Advanced" to show/hide some pulse parameters in the table. Parameters requiring "Advanced" mode are indicated below.

Each row represents one pulse. The sequence in this example consists of 9999 pulses, each described with:

"Signal"	Displays the emitter ID (emitter source of the pulse). The emitter ID is available in collection and map-based scenarios. This information is useful for interleaved signals where the output file contains the pulses of multiple emitters. In DF scenarios, displays emitter ID (source of the pulse) and Rx antenna ID (the antenna that received the pulse). The last entry in this column, "EOF", indicates the file duration.
"ID"	Assigned ID number for the emitter.

"TOA" Time of arrival
The value reflects the selected PRI/PRF, incl. Delay.

"Level" Pulse level, as configured with the parameterLevel, incl. IPM variations.

"PW" Pulse width, as configured with the parameter Width, incl. IPM variations.

"Freq." Frequency as set with the parameter Δ Freq., incl. IPM variations.
In this example, a frequency hopping IPM profile varies the frequency in the range 0 Hz to 5 MHz.

"Modulation" Indicates the modulation type.

"Bandwidth" Indicates the pulse bandwidth, as configured in "Pulse Shape Settings" on page 95.

"Rise Time" Indicates the pulse rise time, as configured in "Pulse Shape Settings" on page 95.

"Fall Time" Indicates the pulse fall time, as configured in "Pulse Shape Settings" on page 95.

"Marker [4321]"
Available in "Display Mode > Advanced".
Indicates if the corresponding marker is active.

"Azimuth" Available in "Display Mode > Advanced".
Indicates the azimuth of the pulse.

"Elevation" Available in "Display Mode > Advanced".
Indicates the elevation of the pulse.

"Rx Ant. Azimuth"
Indicates the azimuth of the Rx antenna.

"Rx Ant. Elevation" Available in "Display Mode > Advanced".
Indicates the elevation of the Rx antenna.

Graphics
Visualize the variation of the corresponding parameter over time.

Available parameters ← Graphics
The difference in TOA between two consecutive pulses is used to calculate the pulse repetition interval (PRI).

You can observe the variation of one or more PDW parameters simultaneously. The display is color-coded.

The color bar on the right of the display indicates the current PDW parameter. Click the PDW parameter name on the right corner of the graph to change the units on the y-axis.

Figure 15-1: Signal Preview: understanding the displayed information

1 = PDW parameters displayed on the graph
2 = Current PDW parameter, indicated by the label on the y-axis and the color bar. To change it, click the colored label of the PDW parameter of interest
3 = Zoom/shift slider scales and repositions the PDW parameters on the display
4 = Auto-scale arrows indicated that the PDW parameter is zoomed in, that is not all values are visible. Click the arrow to zoom out and display all values

The value ranges for both axis are selected automatically. To change the zoom:

- On the x-axis, select the Zoom to Fit or turn the mouse wheel.
- On the y-axis, click the labels and move the "Zoom/shift slider".
  To zoom out, click the auto-scale arrow (see Figure 15-1).

Zoom to Fit ← Graphics

Scales the graphs automatically to fit into the current time range.

Turn the mouse wheel to zoom in/out in the display.

Zoom Area ← Graphics

Provides information on defining a zoom area.

Hold the cursor over the blue icon to see the Tooltip.

On the graph, hold [CTRL] and the left mouse button, and drag the mouse to define the zoom area.

16 Creating complex 2D scenarios with receiver and TX items

The dedicated scenario types for complex 2D emitter-receiver scenarios are:

- "Localized Emitters"
- "Direction Finding" (Option:R&S PULSE-K39)

Complex 2D emitter-receiver scenarios can include the following elements:

- **"Receiver"**
  Simple or complex, depending on the scenario.
  There is exactly one receiver per gaming area (2D map).
  The receiver is described by one ore more antenna elements, static position and attitude information.
- **"Emitters"**
  Simple or complex, depending on their definition.
  Emitters are described by the combination of an operating mode, antenna pattern, antenna scan and a sequence, static position and attitude information.
- **"Background emitters"**
  Background emitters are a source of a noise-like pulsed background signal, broadcasted with equal power in all directions. They are added to the scenario as sequences of pulses.
- **"Platforms"**
  A platform is a group of up to 8 emitters associated with a single icon.
  Platforms are differentiated from other emitters by a configurable colored background.
  Each emitter on the platform can be individually configured and controlled.

Complex 2D emitter-receiver scenarios support the following georeferenced map formats:

- GeoTIF

For step-by-step instructions, see:

- Chapter 16.1, "How to create scenarios with receiver and TX items", on page 290

For a description of the related settings, see:

- Chapter 16.2, "2D map settings", on page 298
- Chapter 16.3, "3D scan pair view settings", on page 302
- Chapter 16.4.1, "Available TX items", on page 308
- Chapter 16.5, "Moving emitters and platforms", on page 323
- Chapter 16.7, "Background emitters properties", on page 335

16.1 How to create scenarios with receiver and TX items

Depending on the test situation, select one of the following scenario types:

- If your test situation requires testing the receiver's ability to detect the sum signal from different static emitters, you can use a **"Localized Emitter"** scenario.
  In this dedicated scenario, you configure the signal of one or more emitters that are received by a receiver with defined characteristics. You can also configure the receiver and change its position in the scanning beam of the emitters.
- If your test situation requires testing a multichannel receiver and its ability to detect the origin of a signal, you can use a **"Direction Finding"** scenario.
  Option:R&S PULSE-K39 required for direction finding.
  In this dedicated scenario, you configure the signal of one or more emitters that is received by a receiver with up to 10 antenna elements. The receiver evaluates the receiver signal and estimates the origin of the emitter, based on received power level and time delays.

See:

- "General workflow for creating complex 2D scenarios" on page 290
- "How to create a direction finding scenario" on page 291
- "How to create the scenario and configure the receiver" on page 292
- "How to configure moving emitter and receiver" on page 294
- Chapter 16.8, "Platforms with multiple emitters", on page 337
- Chapter 16.10, "Using georeferenced maps", on page 342

General workflow for creating complex 2D scenarios

To explain the principle, the provided example creates a complex "Direction Finding" scenario. One emitter is placed on the 2D map at a given distance to a multichannel receiver. The receiver and the emitter characteristics are configured.

To create "Localized Emitter" scenarios, follow the same workflow.

For example, drag and drop further elements, such as emitters, platforms and background signals to the 2D map and configure their settings.

Follow the following general steps:

1. Open a suitable scenario.
2. In the block diagram, select "Map".

   The "2D" dialog displays a 2D view of the receiver and currently configured emitters, together with their main characteristics.
3. Add a georeferenced background map to the "2D" dialog, if necessary.
4. Drag and drop items from the repository onto the 2D map (or relevant list).
   - Emitters (drop on map or "TX Items" list)
   - Platforms (drop on map or "TX Items" list)
5. Use the mouse to reposition (drag and drop) items on the map.

6. Use the context menu of any of the elements on the 2D map to:
   a) Access and configure the "Properties".
      For example, define the elements position more precisely. Or change the attitude (orientation) of the antenna.
   b) Display the 3D antenna patterns and scans.
      For example, select "Normalized power level at Receiver > Source > Emitter Only > On" to display only the signal at the receiver.
      The calculation assumes a receiver with an omnidirectional antenna pattern.
   c) To add/remove an emitter or platform.
   d) To select an icon.
   e) To configure a trajectory (movement path).
7. Use the mouse wheel to change the scale of the 2D diagram, i.e. to change the distance between the receiver and the emitters.

   For a description of the provided settings, see:
   - Chapter 16.2, "2D map settings", on page 298
   - Chapter 16.3, "3D scan pair view settings", on page 302
   - Chapter 16.4.1, "Available TX items", on page 308
   - Chapter 16.7, "Background emitters properties", on page 335

How to create a direction finding scenario

This example shows you how to create and configure a scenario for testing of a four-channel receiver. A four-channel receiver can be located for example on the wings of an airplane, see Figure 16-1.

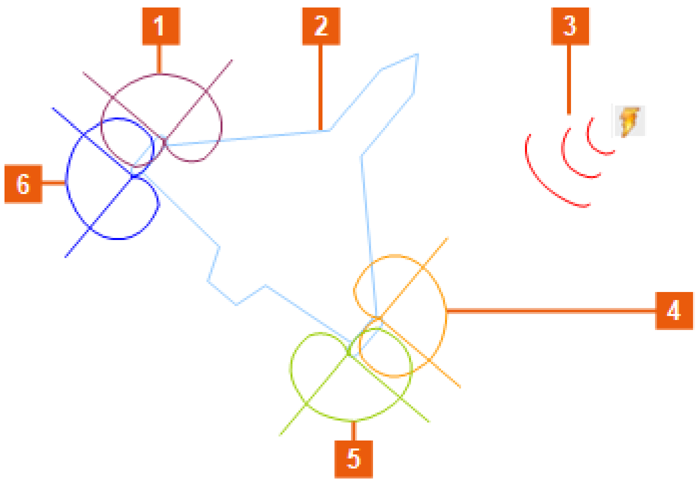


Figure 16-1: Direction finding: four channel receiver

1, 4, 5, 6 = Four sectors (i.e. the four antennas of a multichannel receiver)
2 = Airplane with a direction finding receiver
3 = Emitter

Each of the four antennas receives a signal with a different power level. In this example, the antenna at the first sector (at the front on the right side) receives the strongest signal. The signals are also received with different time delays. The signal at the third sector, for example, has the longest time delay. Multichannel direction finding receivers measure the received power levels and time delays and estimates the origin of the emitter.

The illustrations on Figure 16-2 show the principle of amplitude and time delay-based direction finding.

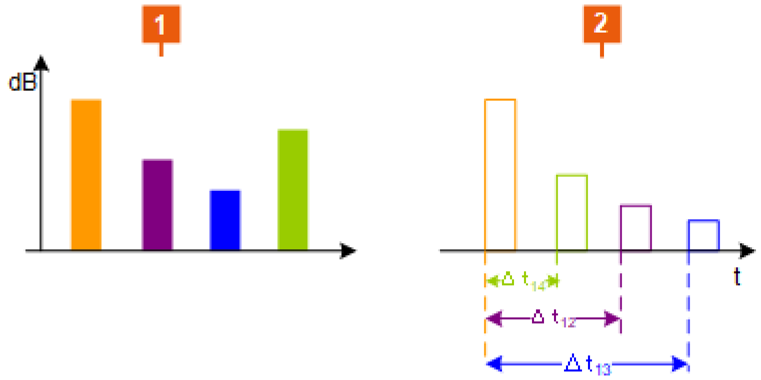


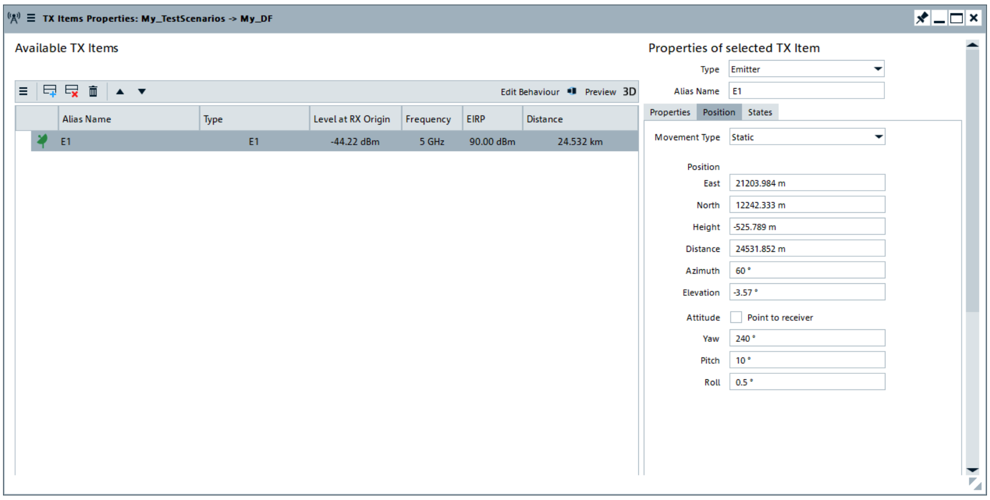
Figure 16-2: Principle of the amplitude and time delay based direction finding

1 = Power levels received by each of the four antennas at the receiver
2 = Relative time delays between the received

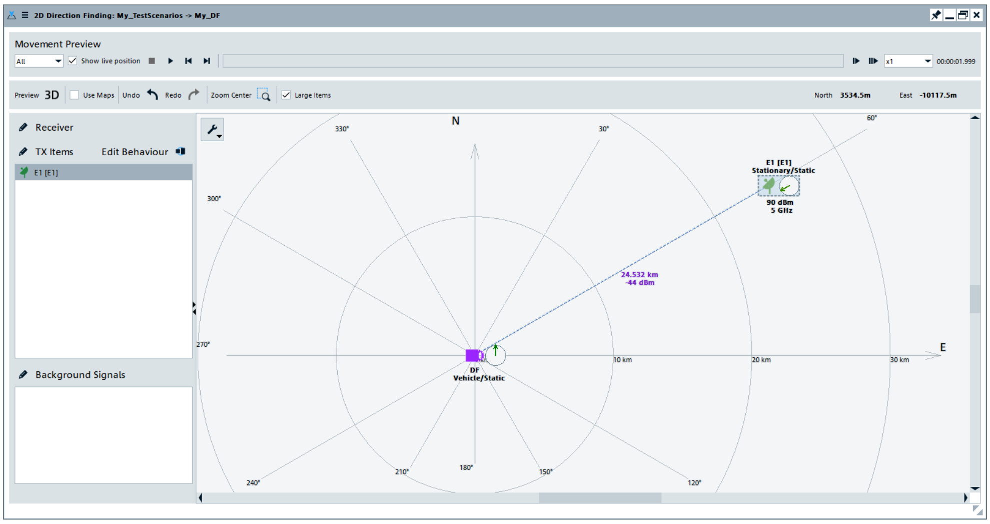
How to create the scenario and configure the receiver

1. Create a direction finding scenario.
2. Create a receiver.

   For an example of suitable receiver settings, see Chapter 13.2, "Receiver settings", on page 264 and:
   - Figure 13-3
   - Figure 13-4.
3. In the scenario dialog, select "Map > 2D".
4. Create an emitter.
   For example, use a raster scan and an antenna with a narrow (pencil) antenna beam.
   Enable a "Pitch = 10°" because the receiver is 10000 higher than the emitter.

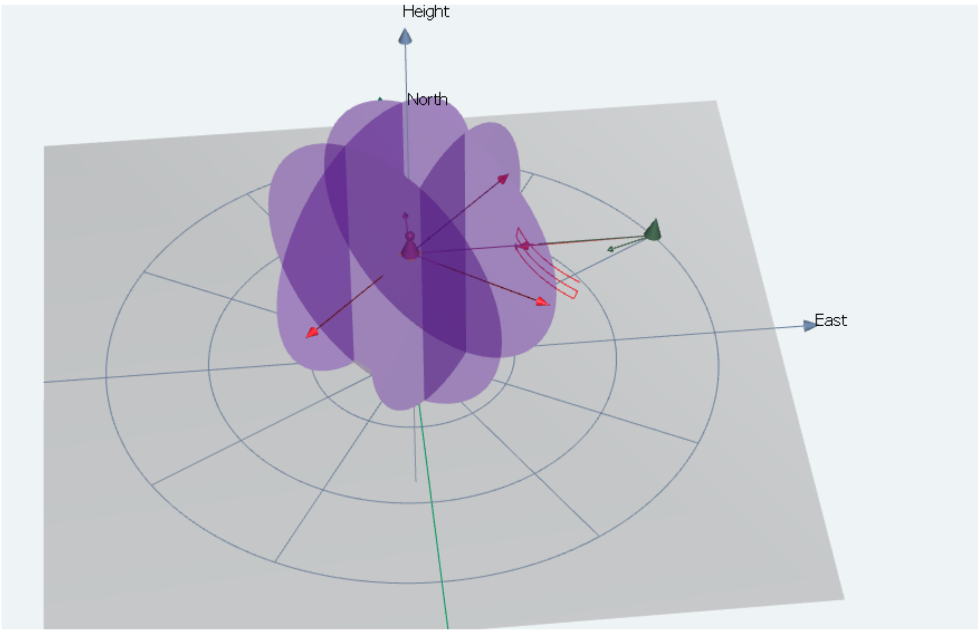


5. Drag and drop the emitter onto the 2D map.



6. Select "Receiver > context menu > Properties".
   Set "Height = 10000 m".
7. Select "Available TX Items > Preview 3D".
   Observe the display. Zoom in; turn on the 3D display for a better overview.

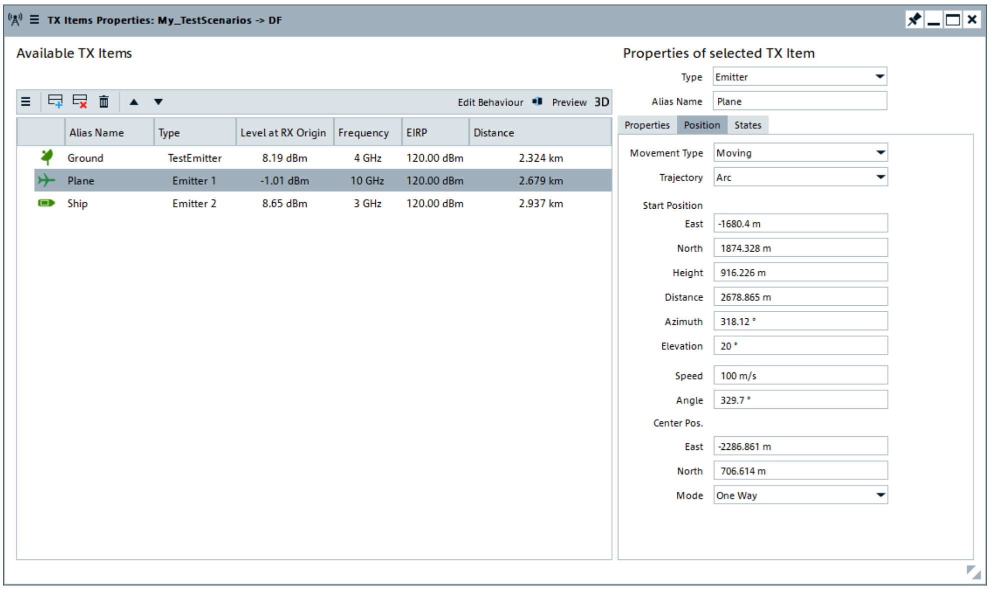
Change the antenna selected with the parameter "Receiver Antenna".



The 3D view displays the four antennas of the multichannel receiver. The receiver and the emitter are at different heights.
The power level of the received signal changes depending on the current "Receiver Antenna".
In this example, the signal from the antenna placed on the right side at the front of the airplane ("Antenna = front right") receives the strongest signal (see Figure 16-2).

How to configure moving emitter and receiver

1. On the 2D map, select "Properties > TX Items".
2. In the list of "Available TX Items", select an emitter.
3. Select "Properties of the Selected TX item > Position".
4. Select "Mode > Moving".



5. Configure the "Trajectory" and further settings.
   See Chapter 16.9, "Creating trajectories on a 2D map", on page 341.
6. To observe the movement on the 2D map, select "Movement Preview > Play".

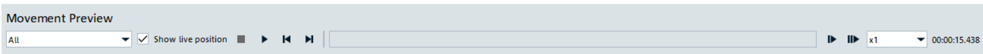
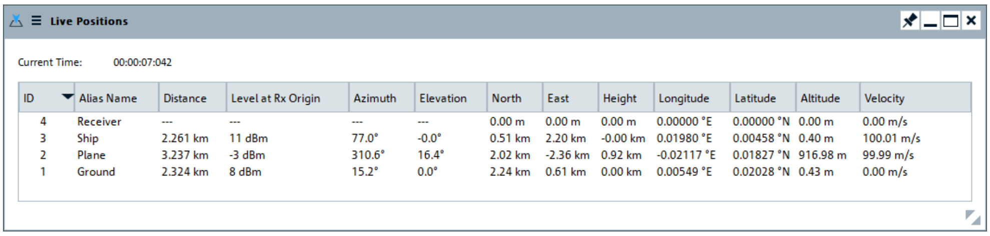


Figure 16-3: 2D map > Movement Preview

If "Show live position" is activated, an overview dialog opens and displays a live preview of important position parameters per TX item and RX item. You can close the "2D Live Preview" at any time. The movement is not interrupted.



The "Movement Preview" dialog displays the duration of the moving scenario. With "Scenario > Signal Calculation > Config > Config > Signal Calculation Settings > Duration > Simulation Duration = Auto", the duration is calculated automatically so that the movement of all objects is completed.

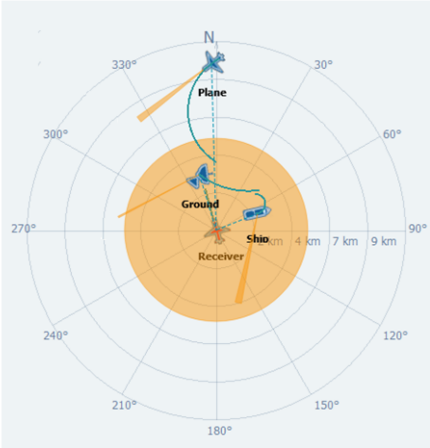
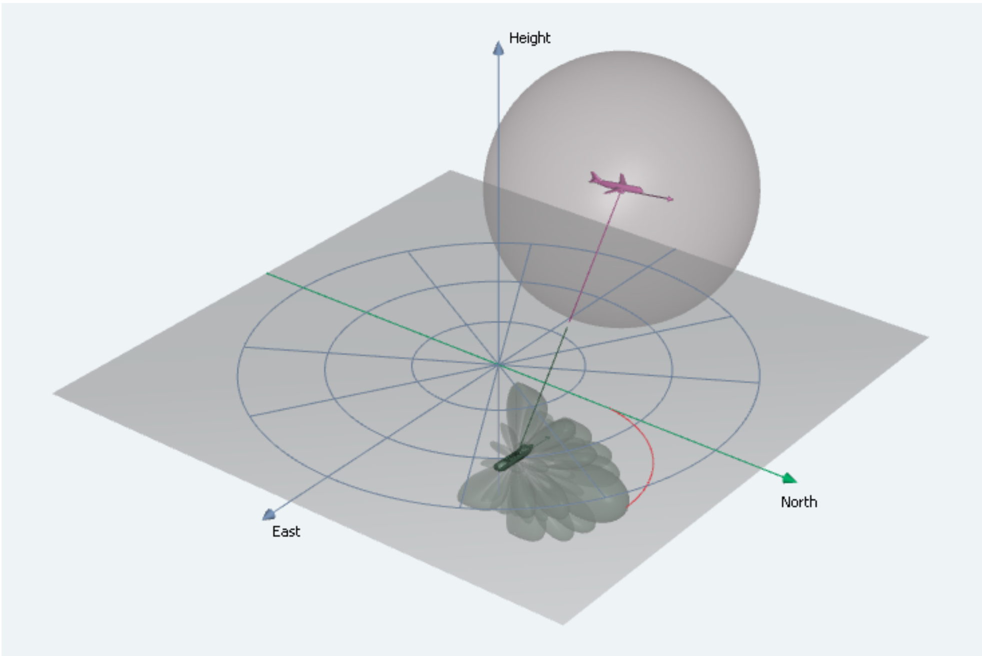


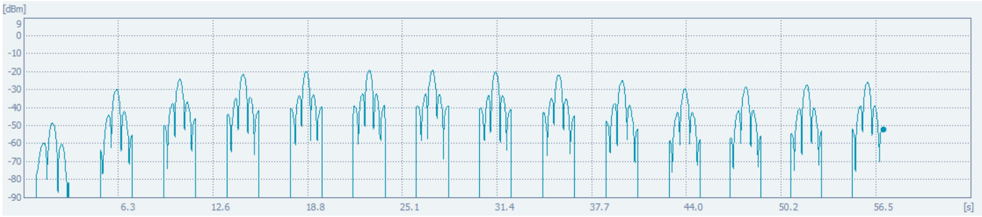
Figure 16-4: 2D map showing antenna types

Beam = Indicates a directional antenna of any kind. The sector direction corresponds to the actual scan direction.
Circle = Represents an omni-directional (isotropic) antenna.

7. To observe the signal variation at the receiver, select an "Emitter > context menu > 3D View with Receiver".



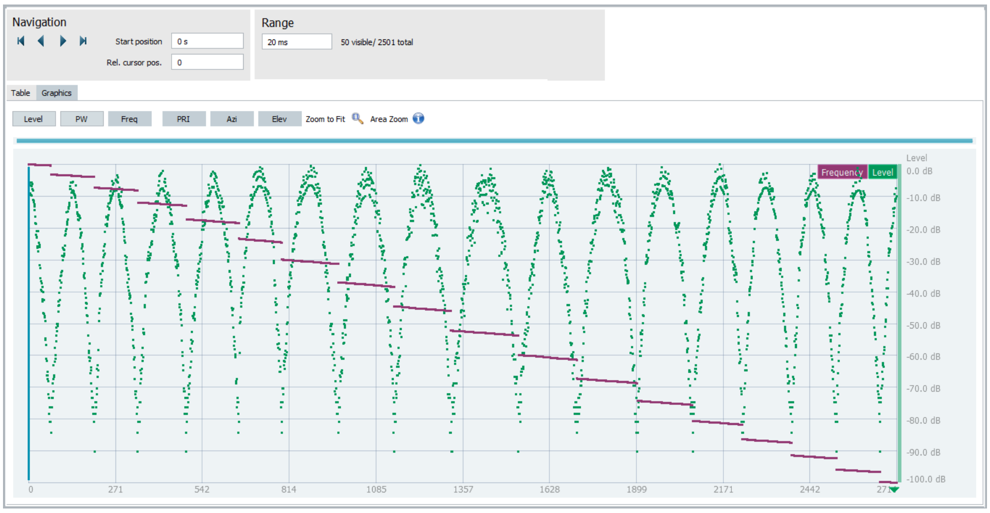
The power level at the receiver varies due to the receiver's antenna pattern and scan. The free-space attenuation over the dynamically changing distance between the emitter and the receiver also affects the level.



When interpreting the results, consider the type of receive antenna and the emitter's trajectory and altitude. A low received power level does not always correspond to a long distance between emitter and receiver. A receive antenna with high directivity and gain only provides a high signal power if the emitter is in the main beam. If the emitter overflies the receiver, you can expect large sudden changes in the signal level. If the receive antenna is omni-directional, you can expect a smooth rise and fall in level as the emitter passes by.

8. Calculate the signal. Observe the generated signals:
   a) In the scenario dialog, select "Signal Calculation > Config > Config > Signal Calculations Settings > Output" and modify as required.
   b) Select "Signal Generation > Assign" and select the destination.
   c) Select "Signal Calculation > Calculate".
   d) Select, for example "Signal Generation > Results > View > Ground".
   e) In the "Signal Preview" dialog, select "Frequency".

   In this example, the receiver and the emitter (Ground) are moving objects, where the receiver approaches the emitter and later departs from it.



   The "Frequency" and "Level" displays confirm the expected frequency variation caused by the Doppler effect. The Doppler is a positive value during the approaching phase and a negative one during the time the receiver is flying away from the emitter.

16.2 2D map settings

Access:

▶ Perform one of the following:
- In a "Scenario Type = Localized Emitters", select "Map".
- In a "Scenario Type = Direction Finding", select "Map".

The name of the dialog changes depending on the scenario but the settings and the workflow are similar.
The 2D diagram displays the configuration on a map with North/East coordinates.

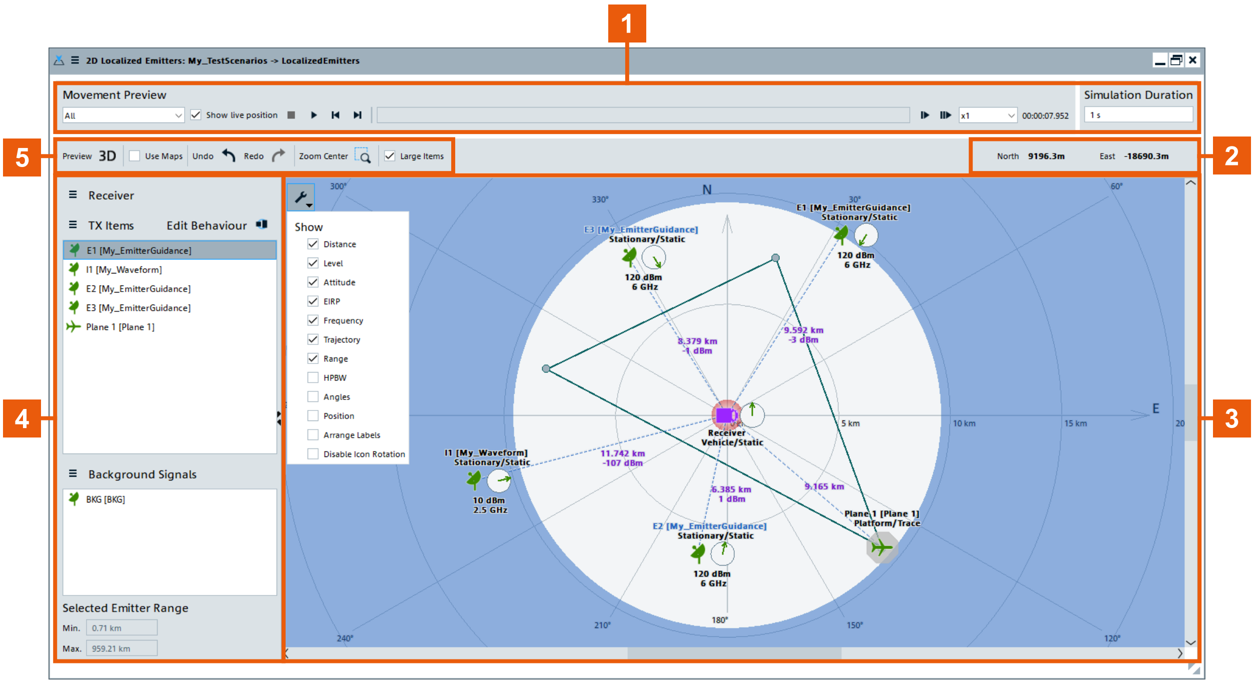


Figure 16-5: 2D view: understanding the displayed information

1 = Simulation tools and information
2 = Cursor coordinates
3 = Visualization area
4 = Receiver/TX Items
5 = Display controls

See also Chapter 16.5, "Moving emitters and platforms", on page 323.

Settings:

Movement Preview

Use the drop-down menu to select the trajectory of a specific platform. Selecting "All" displays the trajectories of all platforms.

"Show live position"
Activates the dialog "2D Live Preview Stats" on page 299.

"Start/Stop, Pause, Move to Start/End"
Standard play functions.

"Time Line"
Indicates the current time of the simulation.

"Play slower/faster"
Switches to the next/previous value of the time scale coefficient.

"Time Scale Coefficient"
Applies a coefficient to speed up or slow down the display.

Position Step

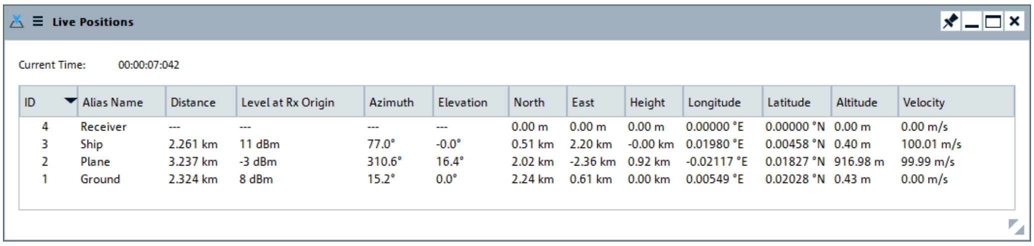
Available if an emitter with "Position > Mode = Steps" is present. You can select a specific step for the emitter on the 2D map.

Simulation Duration

Available if a fixed simulation time is defined for the selected scenario, see "Duration" on page 349.

2D Live Preview Stats

If "Show live position" is activated and a preview is started ("Play"), displays a live track of the positions of the TX items and RX items.



Closing the dialog does not interrupt the movement preview.

Remote command:
`PREView:POSition?` on page 577

Undo/Redo

Reverses the drag&drop operation on the 2D map.

Pan, Zoom

- Pan: Use the mouse to move the displayed map area.

- Zoom in/out: Turn the mouse wheel to zoom in or out.
- Zoom center: Press the zoom icon.

Large Items
Activates the larger size for items displayed in the visualization area.

Visualization settings > Distance, Level, Attitude, EIRP, Frequency, Trajectory, Range, HPBW, Angles, Position, Arrange Labels
To change the displayed information, enable or disable the corresponding parameter.

"Range" If enabled, indicates the dynamic range for the simulation, numerically and as a color-coded indication.

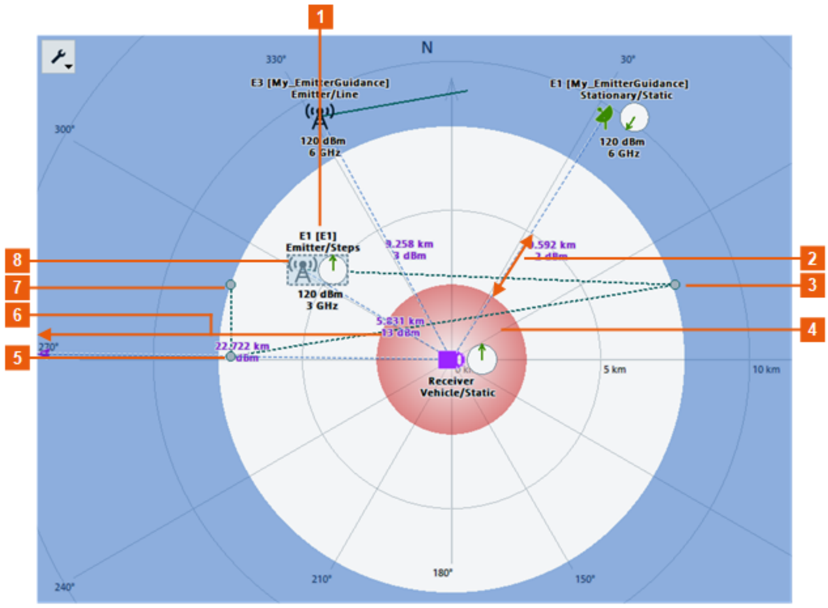


Figure 16-6: 2D map > Visualization > Range: Understanding the displayed information

1	= Selected emitter (Emitter 1, movement = position steps)
2	= Unused area, where the radius (outer border) is the minimum distance between the receiver and the selected emitter; in this example, the position of step 1
3, 5, 7, 8	= Four position steps, emitter currently displayed at position step 3
4	= Minimum distance from the receiver that can be simulated for the selected emitter
6	= Dynamic range of the signal generator (outer border outside of the displayed area)

- Red background
  Range where the maximum simulated output power is exceeded.
- Blue background
  Range where you can place an emitter or emitters and the simulation generates the signal in its dynamic range.
  – Outer border = lowest output power that can be generated for the selected emitter and position mode.
  – Inner border = minimum distance between the receiver and the selected emitter.
    "Position > Mode = Static", the position of the emitter
    "Position > Mode = Steps/Moving", among all steps/trajectory points, the position of the closest to receiver one

  If an emitter moves beyond the outer-border of the blue area, the signal becomes too weak to be generated.
  If emitters are too close to the receiver and enters the red area, the simulated output power can be insufficient to generate such high power. In such cases, the power levels clipped to the value

specified in the signal calculation settings (see "Clipping Level" on page 349).
- White background
  Unused area on the map, showing the cross section between the red and blue area.

"Angels" If enabled, displays the azimuth and elevation angles in a label near the emitter.

"Position" If enabled, displays the positions in terms of north/east coordinates and height.

"Arrange Labels"
If enabled, moves the labels to avoid any overlapping.

Platform, Emitter, Properties
Double-click the "Platform/Emitter" name to access the "Emitter Properties" dialog, see Chapter 16.4.1, "Available TX items", on page 308.

Background Signals
Displays the sequences or the background emitters that compose the background signal.

To add a background emitter, drag and drop a sequence containing the emitter in the block.

Remote command:
See `SCENario:LOCalized:SEQuence` on page 554.
See `SCENario:DF:SEQuence` on page 554.

Platform/Emitter position on the 2D map and distance to the receiver
The 2D map indicates the current distance, calculated from the current platform/emitter position on the 2D map.

To change the current position of the emitter, use one of the following possibilities:
- Drag the platform/emitter to the new position on the 2D map.
- Open its context menu, select "Properties > Position" and change the coordinates, the "Attitude" and the "Distance".
  If several position steps are defined, the changes apply to the current step ("Current Step"), see Discrete position steps.

16.3 3D scan pair view settings

Access:

1. Perform one of the following:
   - In a scenario type "Localized Emitters", select "Map" > " 3D"
   - In a scenario type "DF" (Direction Finding), select "Map" > " 3D"
2. In the 2D map, perform one of the following:
   - In one of the lists, select an emitter or platform > "Properties" > "3D"

- In the map, right-click an emitter or platform > context menu > "3D View with Receiver".

The "3D Scan Pair View" appears.
If you loaded a georeferenced map in the 2D view, it also appears in the 3D view.

The available settings depend on the scenario:
- A simple scenario containing an emitter and a receiver (each with only one antenna) offers fewer settings.
- A complex scenario containing moving platforms (each with multiple emitters) and a scanning DF receiver (with multiple antennas) offers more settings.

However, if you understand how your emitters are configured, the differences are self-explanatory.

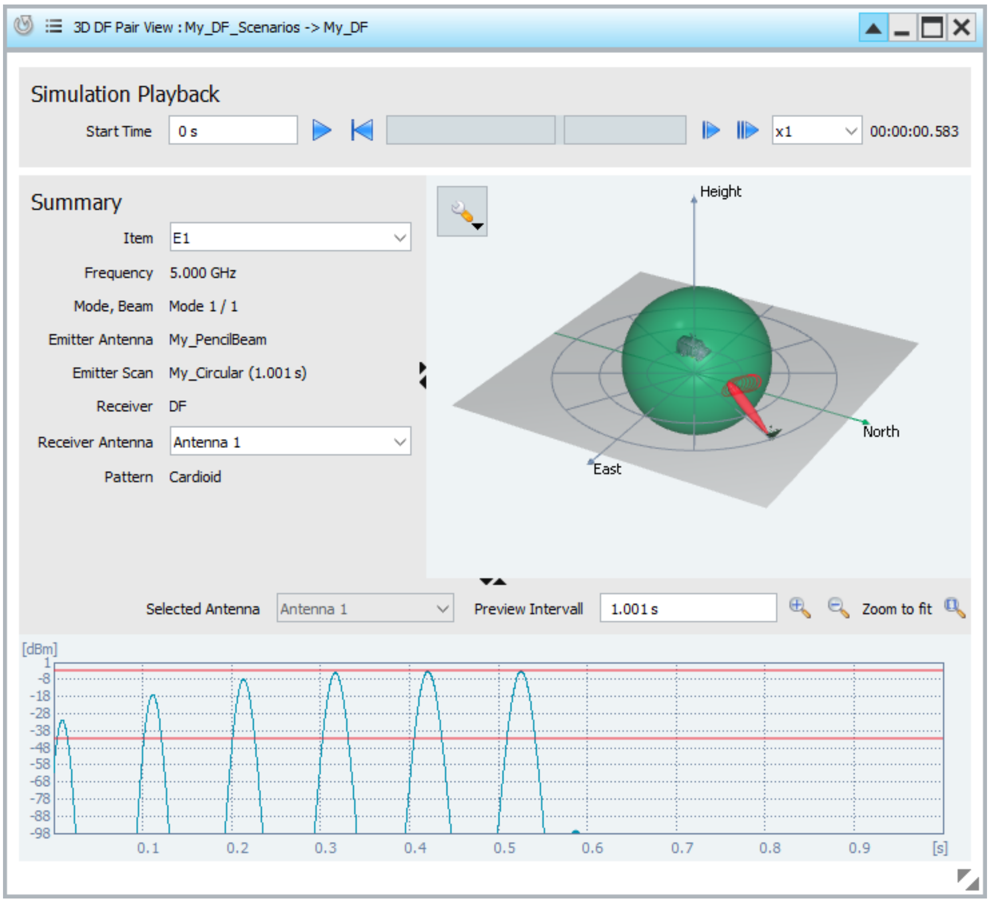


Figure 16-7: Example: 3D Scan Pair View display, Scenario Type = DF

This dialog displays:
- A live plot of the normalized signal power level at the receiver.
- A 3D view of the receiver and emitter antennas with their patterns and scans.
- In "Direction Finding" scenario, the plot and the 3D view correspond to the selected receiver antenna element.

Settings:

Simulation Playback

Controls the timing of the simulation.

For example, with these settings you can analyze when/where/how often the beams overlap while simulating:

- A static scanning RX and a static scanning TX with different scan cycles.
- A moving RX and static TX
- A moving TX with different modes and a static RX

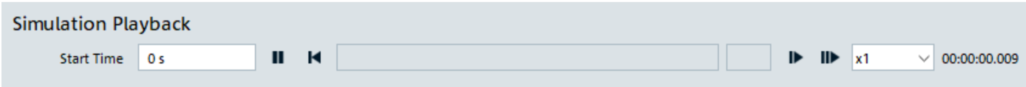


Figure 16-8: Simulation Playback

In addition to setting the overall duration, you can also observe a particular time-segment of the simulation by starting and stopping at particular times.

Note: The duration of a simulation depends on the scenario. An emitter with a long trajectory and low speed causes a long duration. A short trajectory and/or high speed cause a short duration.
The duration provides the maximum value for the "Stop Time" and "Preview Interval" settings.

"Start Time" Set to *0 s* or *minimum*, to play the simulation from the beginning. Typically, set this value to begin the simulation at a particular point along the trajectory.

Buttons Control the simulation using the following buttons:

- Pause/Play - use this toggle to pause the simulation and resume playing.
- Restart - use this button to restart the simulation at the configured "Start Time".
- Play slower/Play faster - decrease/increase the "Speed-up factor"

Summary (live plot and 3D view configuration)

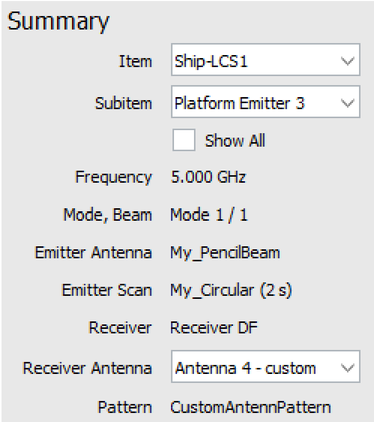


Figure 16-9: Summary

The "Summary" provides the following information fields and drop-down lists:

"Item" Lists all emitters and platforms of the selected scenario.
Select the item that you want to use for the simulation.
If you select a platform, see "Subitem" for a list of the emitters on the platform.

"Subitem" Lists all emitters on the selected platform.
Select the emitter that you want to use for the simulation.

"Frequency" Shows the frequency used by the selected emitter.

"Mode, Beam" Shows the active mode and beam combination (continuously updated) during the simulation.

This information is useful when:

- The selected emitter has more than 1 mode.
- The active mode has more than one beam.

"Emitter Antenna"
Shows the "Antenna Pattern" used by the selected emitter.

"Emitter Scan" Shows the name of the "Antenna Scan" used by the selected emitter.

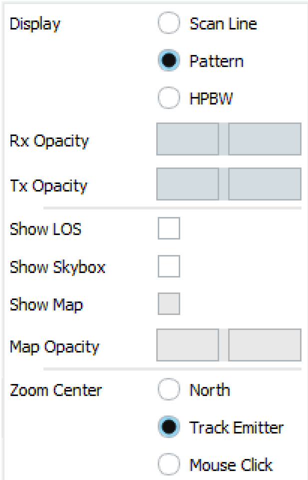
"Receiver" Shows the name of the selected emitter.

"Receiver Antenna"
Lists all antennas configured for the selected receiver.
Select the antenna that you want to use for the simulation.

"Pattern" Shows the "Antenna Pattern" used by the selected "Receiver Antenna".

Settings

Access: click the visualization settings icon to open the settings dialog.



Defines the way that the scan is represented:

"Scan Line" The RX and TX beams are represented by lines.

"Pattern" The RX and TX beams are represented by antenna patterns.

"HPBW" The RX and TX beams are represented by colored eclipses. The size of an eclipse indicates the half power beam width HPBW.

"Show Line of Sight"
Displays the line of sight (LOS) between the emitter and the receiver.

"North"
When zoomed/zooming, the view is focused on the center of the "North" line.
Typically used with maps. Helps you maintain an awareness of the map's orientation.

"Track Emitter"
When zoomed/zooming, the view is focused on the emitter.
Typically used with moving emitters. The view stays focused on the emitter as it moves.

"Mouse Click"
When zoomed/zooming, the view is focused on a user-defined marker. Use the left mouse button to put a marker on the map. Enables you to decide the focus of the view when zoomed/zooming.

Preview graphic settings

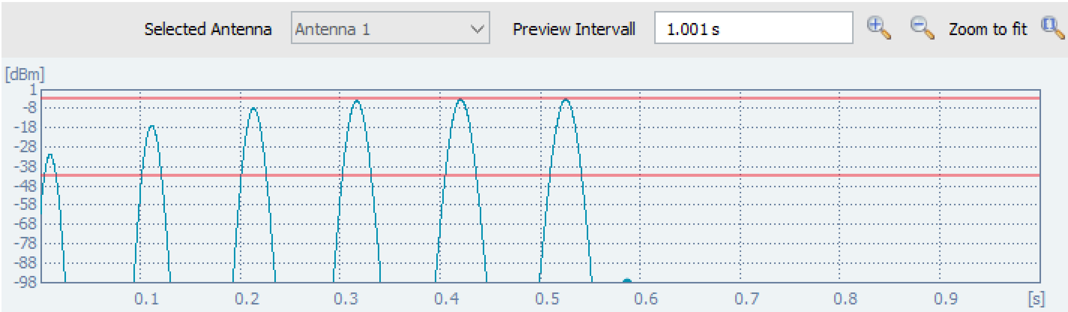


Figure 16-10: Settings for the preview graphic

Define which signal is visualized in the graphic.

See also "How to create the scenario and configure the receiver" on page 292.

"Selected Antenna"

In "Direction Finding" scenario and if "Receiver Antenna > Show that All Antennas" is selected, switches between the receiver antenna elements.

"Preview Interval"

Sets the time interval shown on the preview at the bottom of the view. The horizontal axis of the preview corresponds to the configured "Preview Interval".
Typically, set to a multiple of the TX scan cycle time, to show several sweeps in one preview. For example, for an emitter with a two-second cycle, an interval of 10 seconds shows 5 sweeps.

Note: "Preview Interval" is the time represented by the horizontal axis of the preview. It is not the time taken to fill the preview.
"Preview Duration" is the time taken to fill the preview, which is calculated from your settings, automatically.

"Zoom in", "Zoom out" and "Zoom to fit"

These icons refer to the preview graphic settings time axis:

16.4 Properties of TX items (emitters and platforms)

The "TX Items" dialog provides access to the properties of all the emitters and platforms in the current scenario.

Access:

1. Open a suitable scenario.

   Use any scenario that offers "Map" in the "Scenario Creation" block.
   - "Localized Emitters"
   - "Direction Finding"

2. In the block diagram, select "Map".

The "2D" dialog displays a 2D view of the receiver and currently configured emitters, platforms, together with their main characteristics.

3. In the "2D" dialog, use one of the following to access the "TX Items Properties" dialog:
   - "Properties" list > "TX Items" link
   - Map
     – Double-click a platform/emitter icon
     – Right-click a platform/emitter icon > context menu > "Properties"
   - "TX Items" list
     – Double-click an entry
     – Right-click an entry > context menu > "Properties"

The "TX Items Properties" dialog opens.

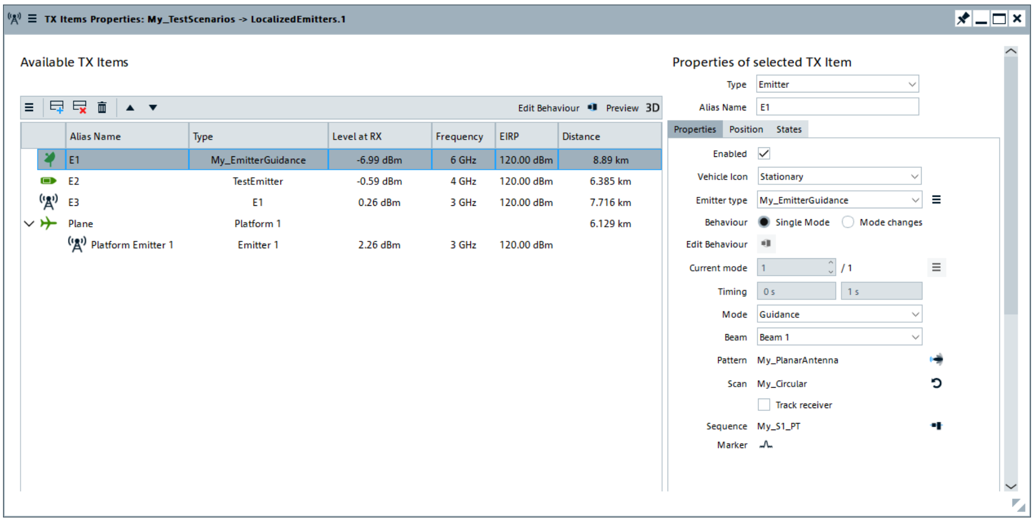


Figure 16-11: TX Items Properties dialog example

As Figure 16-11 shows, this dialog contains 2 parts:
- "Available TX Items" (left)
  Select the item that you want to configure.
- "Properties of selected TX Item" (right)
  This part contains 3 tabs. Use the appropriate tab to configure the "Properties", "Position" or "States" of the selected item.

16.4.1 Available TX items

Access:

► In the "2D" dialog, use one of the following to access the "TX Items Properties" dialog:
   - "Properties" list > "TX Items"

- Map
  – Double-click a platform/emitter icon
  – Right-click a platform/emitter icon > context menu > "Properties"
- "TX Items" list
  – Double-click an entry
  – Right-click an entry > context menu > "Properties"

Append, Delete, Clear, Move Up/Down
Use the standard functions of these buttons to add, reorder, or remove items.

Preview 3D
Opens the "3D Preview" for the selected "Emitter".

See Chapter 16.3, "3D scan pair view settings", on page 302.

Edit Behaviour
Accesses the "Mode Editor" dialog.

The "Mode Editor" is a graphical editor for jointly configuring and defining the duration emitters are working in the different modes.

It works like the Mode Changes settings for each of the emitters but rather than defining the start and end time of a mode, you drag the start and end time on the time scale.

In the dialog, you can configure the mode changes of all emitters available in the scenario.

For a description, see Chapter 10.3.3, "Mode editor settings", on page 230.

List of available TX items
Overview information of the available TX items in table form.

"Alias Name" Displays an alias name.

"Type" Displays the type of TX item.

"Level at RX"

Indicates the maximum level at the receiver origin.
The calculation considers the receiver EIRP and the free-space attenuation. Individual antenna attenuation due to positioning and polarization is omitted.

Remote command:
`SCENario:LOCalized:LEVel` on page 560
`SCENario:DF:LEVel` on page 560

"Frequency" As set with the parameter Frequency.

Remote command:
`SCENario:LOCalized:FREQuency?` on page 555
`SCENario:DF:FREQuency?` on page 555

"EIRP"	As set with the parameter EIRP or with the parameter "EIRP".
"Distance"	The distance from the TX item to the receiver. As set with the East, North, Height/Distance, Azimuth, Elevation parameters.

Remote command:
`SCENario:LOCalized:ADD` on page 429
`SCENario:LOCalized:SELect` on page 430
`SCENario:LOCalized:CURRent` on page 548
`SCENario:LOCalized:DELete` on page 433
`SCENario:LOCalized:CLEar` on page 434
`SCENario:DF:ADD` on page 429
`SCENario:DF:SELect` on page 431
`SCENario:DF:CURRent` on page 548
`SCENario:DF:DELete` on page 433
`SCENario:DF:CLEar` on page 434

16.4.2 Properties of the selected TX item

Access:

1. In the "2D" dialog, use one of the following to access the "TX Items Properties" dialog:
   - "Properties" list > "TX Items"
   - Map
     – Double-click a platform/emitter icon
     – Right-click a platform/emitter icon > context menu > "Properties"
   - "TX Items" list
     – Double-click an entry
     – Right-click an entry > context menu > "Properties"
2. In the list of "Available TX Items", select a platform/emitter.
   For details on the "Properties of Selected TX Item" area, see the following sub-chapters.

In "Properties of selected TX Item", not all TX items support all tabs.

See Table 16-1.

Table 16-1: TX items and tabs

"TX Item"	"Properties" tab	"Position" tab	"States" tab	Remarks
Platform	●	●	○	States depend on emitters. See Chapter 16.4.2.3, "States", on page 321.
Platform emitter	●	○	●	Position depends on the platform. See Chapter 16.4, "Properties of TX items (emitters and platforms)", on page 307.
Emitter	●	●	●	Independent emitter. See Chapter 16.4, "Properties of TX items (emitters and platforms)", on page 307.

The selected "TX Item" is described with the following parameters:

16.4.2.1 Properties (type, alias name and configuration)

The content of the "Properties" tab depends on which TX item you select. Not all the configuration options display for all TX items.

Access:

1. In the list of "Available TX Items", select a platform/emitter.
2. Select "Properties of selected TX Item" > "Properties" tab.

Type

Defines whether an emitter is configured.

Remote command:
`SCENario:LOCalized:TYPE` on page 553
`SCENario:DF:TYPE` on page 553

Alias Name

Enters an alias name.

Remote command:
`SCENario:LOCalized:ALIas` on page 548
`SCENario:DF:ALIas` on page 548

Emitter Name

Selects a platform/emitter.

Remote command:
`SCENario:LOCalized:EMITter` on page 549
`SCENario:DF:EMITter` on page 549

Enabled

Enables selected item for calculation.

Remote command:
`SCENario:DF:EMITter:ENABle` on page 550
`SCENario:LOCalized:EMITter:ENABle` on page 550

Platform type

Defines or edits the platform assigned to the selected "TX Item".

Remote command:
`SCENario:LOCalized:EMITter` on page 549
`SCENario:DF:EMITter` on page 549

Vehicle Icon

Assign different icons to the emitters on the 2D map.

The icons are merely for display; the function and the properties of the emitters do not depend on the icon.

Remote command:
`SCENario:LOCalized:MOVement:VEHicle` on page 574
`SCENario:DF:MOVement:VEHicle` on page 574

Behaviour

Enables emulation of emitters that change their mode over time.

For more information on emitter modes, see "Operational mode" on page 218.

To visualize the effect of the mode switching, select "2D map > Emitter (with several modes) > context menu > 3D View with Receiver".

You can define and configure the duration of the modes in one of the following ways:

- With the settings:
  - "Append, Remove Current, Remove All" on page 314
    "Current Mode" on page 313
    "Timing" on page 314
- With the graphical editor "Edit Modes", see Chapter 10.3.3, "Mode editor settings", on page 230.

Single Mode ← Behaviour
The emitter works in one constant mode, as selected with the "Mode" and "Beam" parameters during the whole scenario.

This setting applies even for emitters for which you defined different modes, possibly each with several beams, in the Emitter dialog.

Remote command:
`SCENario:CEMit:MCHG:STATe` 0
`SCENario:LOCalized:MCHG:STATe` 0
`SCENario:DF:MCHG:STATe` 0

Mode Changes ← Behaviour
If selected, you can define different emitter modes and their timing.

Each mode lasts a defined duration. Once it elapses, the subsequent mode is used.

Use the "Mode changes" function to simulate more realistic scenarios where emitters do not use one fixed mode but change their beam and antenna settings over time.

Remote command:
`SCENario:CEMit:MCHG:STATe` on page 557
`SCENario:LOCalized:MCHG:STATe` on page 557
`SCENario:DF:MCHG:STATe` on page 557

Edit Behaviour ← Behaviour
Accesses the "Mode Editor" dialog.

The "Mode Editor" is a graphical configuration for defining the duration the *selected emitter* is working in the different modes.

Works like the Mode Changes settings but rather than defining the start and end time of a mode, you drag the start and end time on the time scale.

Current Mode ← Behaviour
Selects the entry (mode) for which the displayed and configured settings apply.

The second number indicates the number of enabled modes.

Remote command:
`SCENario:CEMit:MCHG:SELect` on page 431
`SCENario:LOCalized:MCHG:SELect` on page 431
`SCENario:DF:MCHG:SELect` on page 431
`SCENario:CEMit:MCHG:COUNt?` on page 430

SCENario:LOCalized:MCHG:COUNt? on page 430
SCENario:DF:MCHG:COUNt? on page 430

Append, Remove Current, Remove All ← Behaviour
Standard functions for entry handling.

Remote command:
SCENario:CEMit:MCHG:ADD on page 429
SCENario:LOCalized:MCHG:ADD on page 429
SCENario:DF:MCHG:ADD on page 429
SCENario:CEMit:MCHG:DELete on page 433
SCENario:LOCalized:MCHG:DELete on page 433
SCENario:DF:MCHG:DELete on page 433
SCENario:CEMit:MCHG:CLEar on page 434
SCENario:LOCalized:MCHG:CLEar on page 558
SCENario:DF:MCHG:CLEar on page 434

Timing ← Behaviour
Sets the duration of the mode, defined as start and end time.

Where:
- The start value of one entry must be smaller than or equal to the end value.
- The start value of a subsequent entry has to be greater than the end value of the previous one.

Tip: Configure the modes backwards: configure the last mode first, set its end time first.

Remote command:
SCENario:CEMit:MCHG:STARt on page 557
SCENario:CEMit:MCHG:STOP on page 557
SCENario:LOCalized:MCHG:STARt on page 558
SCENario:LOCalized:MCHG:STOP on page 558
SCENario:DF:MCHG:STARt on page 557
SCENario:DF:MCHG:STOP on page 557

EIRP
Sets the EIRP.

Remote command:
EMITter:EIRP on page 469

Frequency
Sets the frequency of the TX item.

Remote command:
EMITter:FREQuency on page 469

Mode
Selects the mode that the emitter is working in.

Remote command:
SCENario:LOCalized:EMITter:MODE on page 550
SCENario:DF:EMITter:MODE on page 550

Beam
Sets the number of currently used beams.

Remote command:
`SCENario:LOCalized:EMITter:MODE:BEAM` on page 550
`SCENario:DF:EMITter:MODE:BEAM` on page 550

Antenna Pattern, Scan, Sequence
Displays the currently selected antenna pattern, antenna scan and sequence.

To change any of them, select the corresponding icon.

Track Receiver
If enabled, the scan follows the receiver automatically.

Remote command:
`SCENario:LOCalized:EMITter:MODE:TRACkrec` on page 551
`SCENario:DF:EMITter:MODE:TRACkrec` on page 551

Emitter Marker
Available in map-based and emitter collection scenarios.

Enables the configuration of up to 4 gate markers in the "Emitter Marker Config" dialog.

The gate marker has the same width as the pulse.

Select the required mode:
- "Unchanged"
  Leaves the marker unchanged, as defined in the pulses and sequences settings of this emitter.
- "Force Marker"
  Forces the selected marker type for every pulse from this emitter.

For details of other types of marker, see Chapter 20, "Defining and enabling marker signals", on page 380.

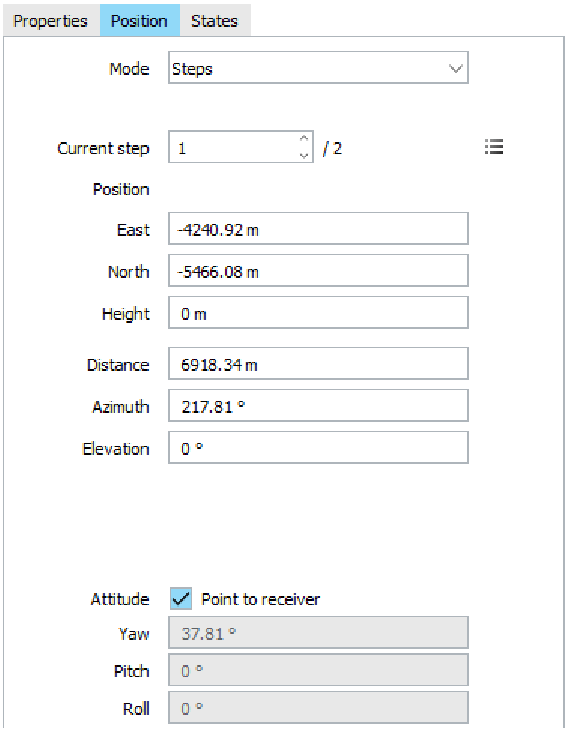
Remote command:
`SCENario:DF:MARKer:FORCe` on page 500
`SCENario:DF:MARKer:GATE` on page 500
`SCENario:LOCalized:MARKer:FORCe` on page 500
`SCENario:LOCalized:MARKer:GATE` on page 500

16.4.2.2 Position settings (static position and discrete position steps)

Access:

1. On the 2D map, select "Properties > TX Items".
2. In the list of "Available TX Items", select an emitter.
3. Select "Propertiers of the Selected Emitters > Position"

   The emitters on the 2D map are static or moving elements with configurable coordinates.



The platform/emitter position can be defined as:

- East/North/Height coordinates relative to the origin
- Azimuth/Distance/Elevation angular positions relative to the origin
- Longitude/Latitude/Altitude absolute coordinates (requires georeferenced map)

The definitions are interdependent.

Mode

Sets if the emitter is static or moving.

Option:R&S PULSE-K39 required for moving emitters.

"Static" The emitter is static; its coordinates are fixed during the whole scenario.
See East, North, Height/Distance, Azimuth, Elevation.

"Steps" You can define several static positions for each emitter.
See "Discrete position steps" on page 319.

"Moving" Emitters are moving objects, defined with their start position and trajectory.
See "Fill Position Steps" on page 320.
See Chapter 16.9, "Creating trajectories on a 2D map", on page 341.

Remote command:
`SCENario:DF:LOCation:PMODe` on page 569
`SCENario:LOCalized:LOCation:PMODe` on page 569

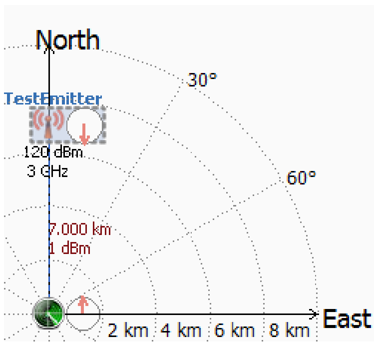
Longitude, Latitude, Altitude
Sets the position of the selected emitter using absolute coordinates (requires georeferenced map).

Remote command:
`SCENario:DF:LOCation:LONGitude` on page 570
`SCENario:DF:LOCation:LATitude` on page 570
`SCENario:DF:LOCation:ALTitude` on page 569
`SCENario:LOCalized:LOCation:LONGitude` on page 570
`SCENario:LOCalized:LOCation:LATitude` on page 570
`SCENario:LOCalized:LOCation:ALTitude` on page 570

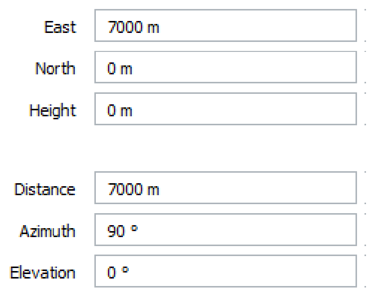
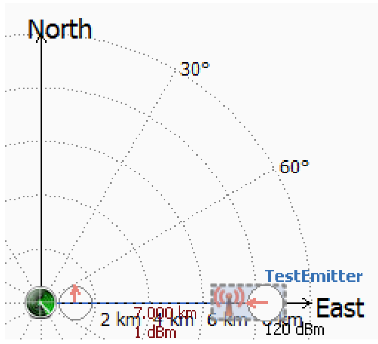
East, North, Height/Distance, Azimuth, Elevation
To change the location of an element:
- Drag this element on the 2D map.
- Use the "Position" parameters to define the location more precisely.

Example:
- An emitter with the East/North coordinates "East = 0 m", "North = 7000 m" is placed on the North, i.e. the "Azimuth Angle = 0°".

<table>
<tr><th>Location parameters</th><th>Map view</th></tr>
<tr><td>East 0 m
North 7000 m
Height 0 m

Distance 7000 m
Azimuth 0 °
Elevation 0 °</td><td>
</td></tr>
</table>

- If the "Azimuth Angle = 90°", the emitter is placed on the east axis, i.e. "North = 0 m" and "East = 7000 m".

Location parameters	Map view
	

Remote command:
SCENario:LOCalized:LOCation:EAST on page 551
SCENario:LOCalized:LOCation:NORTh on page 551
SCENario:LOCalized:LOCation:HEIGht on page 553
SCENario:LOCalized:DISTance on page 551
SCENario:LOCalized:LOCation:AZIMuth on page 551
SCENario:LOCalized:LOCation:ELEVation on page 552
SCENario:DF:LOCation:EAST on page 551
SCENario:DF:LOCation:NORTh on page 551
SCENario:DF:LOCation:HEIGht on page 553
SCENario:DF:DISTance on page 551
SCENario:DF:LOCation:AZIMuth on page 551
SCENario:DF:LOCation:ELEVation on page 552

Attitude
Defines the orientation of the antennas as "Pitch", "Yaw", and "Roll".

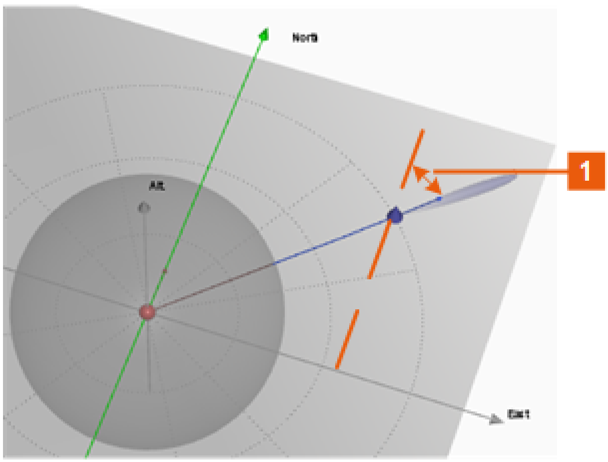


Figure 16-12: Yaw: understanding the displayed information

1 = Yaw is the angle between the antenna's boresight and north

If "Point to Receiver" is enabled, the antenna is automatically turned in the direction of the receiver.

See also "Emitter attitude" on page 218.

Remote command:
SCENario:LOCalized:DIRection:TRACk on page 550
SCENario:LOCalized:DIRection:PITCh on page 549
SCENario:LOCalized:DIRection:YAW on page 548
SCENario:LOCalized:DIRection:ROLL on page 549
SCENario:DF:DIRection:TRACk on page 550
SCENario:DF:DIRection:PITCh on page 549
SCENario:DF:DIRection:YAW on page 548
SCENario:DF:DIRection:ROLL on page 549

Discrete position steps
In "Scenario Type > Localization/Direction Finding", you can define a sequence of **discrete static positions** (steps) to simulate the position change of the emitter. With **"Mode > Steps"**, the emitter is not moving but it changes its position in terms of **coordinates, height and attitude**.

The time duration that the emitter remains at a particular position is defined depending on the selected "Scenario > Signal Calculation > Config > Config > Signal Calculation Settings" > "Duration", see Table 16-2.

Table 16-2: Time duration per position

"Duration"	"Antenna Scan"	"Mode Change"	Time duration (or waveform content) *per position*
"Auto"	Steady (no scan)	Static	One sequence
	Any scan type	Static	One antenna scan
	Any, incl. no scan	Static/Mode Change	All defined modes
"Fixed"	Any, incl. no scan	Static/Mode Change	Time specified with the parameter "Duration > Fixed" divided by the number of steps Longer sequences or scans are truncated; shorter repeated Longer modes are truncated; shorter are not repeated

Note: To prevent that transitions (or jumps) over large distances lead to pulses overtaking one another, distance-related signal propagation delays are not considered.

Current Entry/Number of Entry ← Discrete position steps
Selects the position step for that the displayed position coordinates apply. This is also the position indicated on the 2D map.

The second number indicates the number of available static positions.

Remote command:
SCENario:DF:LOCation:PSTep:SELect on page 431
SCENario:DF:LOCation:PSTep:COUNt? on page 430
SCENario:LOCalized:LOCation:PSTep:SELect on page 431
SCENario:LOCalized:LOCation:PSTep:COUNt? on page 430

Append, Remove Current, Remove All ← Discrete position steps
Standard functions for entry handling.

Remote command:
SCENario:DF:LOCation:PSTep:ADD on page 429
SCENario:DF:LOCation:PSTep:SELect on page 431
SCENario:DF:LOCation:PSTep:DELete on page 433
SCENario:LOCalized:LOCation:PSTep:ADD on page 429
SCENario:LOCalized:LOCation:PSTep:SELect on page 431
SCENario:LOCalized:LOCation:PSTep:DELete on page 433

Auto fill ← Discrete position steps
Opens the "Fill Position Steps" dialog.

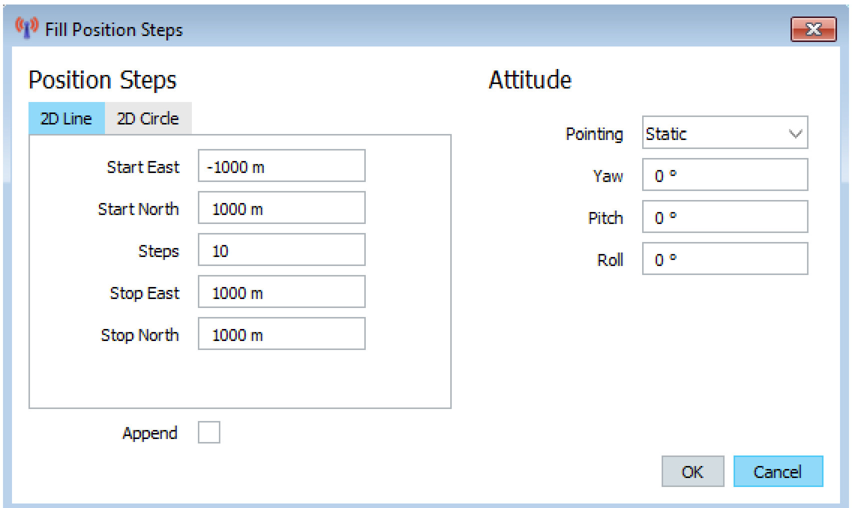
See "Fill Position Steps" on page 320.

Import list X,Y,Z / Dist., Azim., Elev. [Y, P, R] ← Discrete position steps
To define the static positions one by one, you can also import the positions from list files. Lists are simple text files with extension `*.txt`. You can create the lists with any text editor, for example, Notepad.

The list contains the coordinates of each position and (optionally) the yaw, pitch, roll [Y, P, R] values. Position coordinates can be specified in one of the following formats:
- X, Y, Z: East, North, Height
- Dist., Azim., Elev.: Distance, Pitch, Roll

The coordinates of two subsequent positions are separated by line break.

Fill Position Steps
"Fill Position Steps" is a wizard, where you can define steps automatically, based on start and end position and the number of static positions. Static positions can be placed on a straight line or on a circle.



Position Steps > 2D Line ← Fill Position Steps
When distributed on a straight line, the static positions are defined with:
- "Start East/North" coordinates
- "Stop East/North" coordinates

- "Steps" defines the number of static positions between the start and the stop position; positions are spaced equidistant.

Position Steps > 2D Circle ← Fill Position Steps
When distributed on a circle, the static positions are defined with:
- "Distance" sets the circle radius, where the circle center is at the receiver.
- "Start Angle" sets the angle of the start position, where "Start Angel = 0°" corresponds to North.
- "Increment" defines the step size with that the angle is incremented.
- "Steps" defines the number of static positions between the start and the end position; positions are spaced equidistant, where the step size is defined with "Increment".

Attitude > Pointing, Yaw, Pitch, Roll ← Fill Position Steps
Defines the orientation of the antennas:
- "Pointing = Static", the orientation is given as "Pitch", "Yaw", and "Roll".
- "Pointing = to Receiver/ to Front": the antenna attitude is adapted automatically at the specified direction and depending on the current static position.
  This setting is useful if the static positions are distributed on a circle.

Append ← Fill Position Steps
Adds the automatically created positions at the end of the current positions' list.

Ok ← Fill Position Steps
Confirms the settings and apples them.

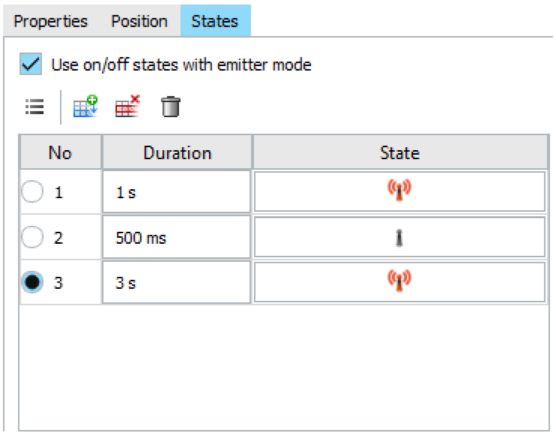
Cancel ← Fill Position Steps
Aborts the changes. Configurations in the "Fill Position Steps" dialog are lost.

16.4.2.3 States

Access:

1. On the 2D map, select "Emitter Properties".
2. In the list of "Available TX Items", select an emitter.
3. Select "Properties of Selected TX Item > Properties > Configuration > Static".

4. Select "States".



Emitters with constant operation mode can be configured to interrupt their signal transmission. You can define the duration of the intervals during that the emitters transmit signal and during which they are muted.

Settings:

Use on/off states with emitter mode

Enables that an emitter can use on and off states.

Remote command:
`SCENario:DF:EMITter:STATe:ENABle` on page 558
`SCENario:LOCalized:EMITter:STATe:ENABle` on page 558

Append, Remove Current, Remove All

Standard functions for entry handling.

Remote command:
`SCENario:DF:EMITter:STATe:ADD` on page 429
`SCENario:DF:EMITter:STATe:INSert` on page 432
`SCENario:DF:EMITter:STATe:COUNt?` on page 430
`SCENario:DF:EMITter:STATe:DELete` on page 433
`SCENario:DF:EMITter:STATe:CLEar` on page 434
`SCENario:LOCalized:EMITter:STATe:ADD` on page 429
`SCENario:LOCalized:EMITter:STATe:INSert` on page 432
`SCENario:LOCalized:EMITter:STATe:COUNt?` on page 430
`SCENario:LOCalized:EMITter:STATe:DELete` on page 433
`SCENario:LOCalized:EMITter:STATe:CLEar` on page 434

No
Subsequent number indicating the state.

Remote command:
SCENario:DF:EMITter:STATe:SELect on page 431
SCENario:LOCalized:EMITter:STATe:SELect on page 431

Duration
Sets the duration during that the emitter remains in the current state.

Remote command:
SCENario:DF:EMITter:STATe:DURation on page 558
SCENario:LOCalized:EMITter:STATe:DURation on page 558

State
Select the icon to toggle the state.

Remote command:
SCENario:DF:EMITter:STATe:VALue on page 558
SCENario:LOCalized:EMITter:STATe:VALue on page 558

Execute States in Loop
Repeats the state definition cyclically.

Remote command:
SCENario:DF:EMITter:STATe:LOOP on page 559
SCENario:LOCalized:EMITter:STATe:LOOP on page 559

16.5 Moving emitters and platforms

Option:R&S PULSE-K39

In addition to the static emitters, and the possibility to define emitters that change their position at discrete steps, you can also define emitters that move along a trajectory with a defined shape. Emitters can move along simple trajectories, such as a straight line, a circle or a part of a circle (arc). Emitters can follow complex trajectories defined by a sequence of map coordinates.

A sequence of map coordinates can be defined in two ways.

- **"Trace points"**

  You can use the "Trace" function to set a sequence of coordinates (known as "Trace points") manually:

  – **In the map**
    Select an emitter > right-click > context menu > "Trajectory" > "Trace"
    Left-click to set trace points. Right-click to apply the trace.

  – **In the position settings of an emitter**
    Select an emitter > right-click > "Properties" > "Position" tab
    Select and set/edit the position of each "Trace point" in turn.

- **"Waypoints"**

You can load a waypoint file which configures a sequence of coordinates automatically. Waypoint files are simple text files with predefined file format and extension. For a description, see Chapter A.3, "Movement files", on page 638.

To improve the visualization, you can assign different icons to the emitters on the 2D map. The icons are merely used for visualization purposes; the function and the properties of the emitters do not depend on the icon.

If at least one moving emitter is configured and its trajectory is defined, you can visualize it on the 2D map. The 2D map also displays the movement of the emitter along the trajectory. In the 3D combined views of the receiver and emitter, you observe the effect of all events influencing the received signal strength. These are: antenna pattern and scan at the emitter and receiver, distance between the emitter and receiver, emitter movement and antenna attitude.

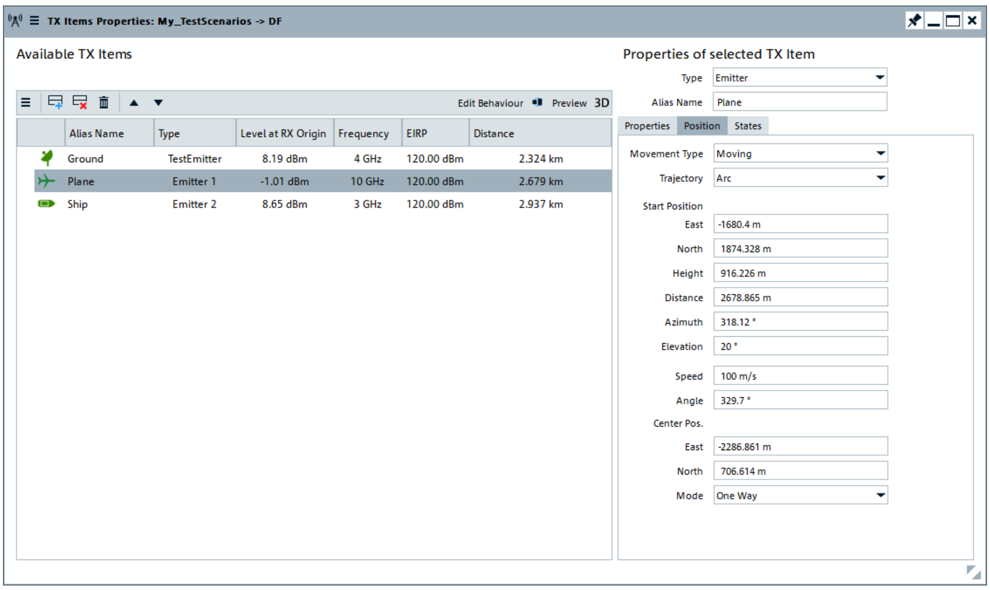
Positions are configured relative to the receiver's origin using the "East", "North" and "Height" settings.

If a georeferenced map is loaded (see Chapter 16.10, "Using georeferenced maps", on page 342), absolute positions can also be defined using the "Longitude", "Latitude" and "Altitude" settings.

Similar to the emitters, the receiver can also be a moving object. The receiver configuration is similar to the configuration of the moving emitters. For settings description, see Chapter 13, "Emulating receivers", on page 259.

Access:

1. On the 2D map, select "Emitter Properties".
2. In the list of "Available TX Items", select an emitter.
3. Select "Properties of the Selected TX Item > Position".
4. Select "Mode > Moving".



5. Configure the "Trajectory" and further settings.

   For step-by-step description, see "How to configure moving emitter and receiver" on page 294.
   For information trajectories, see Chapter 16.9, "Creating trajectories on a 2D map", on page 341.

Settings:

Trajectory
Defines the trajectory shape.

"Line"	The emitter is moving on a straight line, starting from the current position of the emitter (East, North, Height/Distance, Azimuth, Elevation) and ending at the selected "End Position".
"Arc"	The emitter moves on a part of a circle, where the center of the circle is defined with the "Center Position".
"Waypoint"	Emitter with varying coordinates, defined in waypoints files. You find a subset of example files in the installation package but you can also create and load your own files.
"Trace"	Emitter with varying coordinates, defined as a set of "Trace points".

Remote command:
`SCENario:LOCalized:MOVement:TYPE` on page 569
`SCENario:DF:MOVement:TYPE` on page 569

Line trajectory definition
A line is defined with:

Speed ← Line trajectory definition
Sets the speed of the moving emitter.

Remote command:
`SCENario:LOCalized:MOVement:SPEed` on page 572
`SCENario:DF:MOVement:SPEed` on page 572

Acceleration ← Line trajectory definition
Sets the acceleration of the moving emitter.

Remote command:
`SCENario:LOCalized:MOVement:ACCeleration` on page 572
`SCENario:DF:MOVement:ACCeleration` on page 572

End Position ← Line trajectory definition
Sets the "East/North" coordinates and the "Height" of the emitter at the end of the movement.

If "Maps" are enabled, the endpoint can also be defined using "Latitude", "Longitude" and "Altitude".

Remote command:
`SCENario:LOCalized:MOVement:EAST` on page 572
`SCENario:LOCalized:MOVement:NORTh` on page 573
`SCENario:LOCalized:MOVement:HEIGht` on page 573
`SCENario:DF:MOVement:EAST` on page 572
`SCENario:DF:MOVement:NORTh` on page 572
`SCENario:DF:MOVement:HEIGht` on page 573
`SCENario:LOCalized:MOVement:ALTitude` on page 570
`SCENario:LOCalized:MOVement:LATitude` on page 571
`SCENario:LOCalized:MOVement:LONGitude` on page 571
`SCENario:DF:MOVement:ALTitude` on page 570
`SCENario:DF:MOVement:LATitude` on page 570
`SCENario:DF:MOVement:LONGitude` on page 570

Mode ← Line trajectory definition
Defines the behavior of the moving object when the end of the trajectory is reached.

"Cyclic" The trajectory is repeated cyclically. Once the "End Position" is reached, file movement starts again from the beginning.

Using this mode is recommended in the following cases:
- A circle trajectory
- A trajectory in which the start and the end positions are close to each other.

"Round Trip" The emitter moves back and forth on the defined trajectory.

"One Way" The movement is executed once.
When the "End Position" is reached, this position is assumed to be a static one.

Remote command:
`SCENario:DF:MOVement:RMODe` on page 576
`SCENario:LOCalized:MOVement:RMODe` on page 576

Arc trajectory definition
An arc is defined with:

Speed ← Arc trajectory definition
Sets the speed of the moving emitter.

Remote command:
`SCENario:LOCalized:MOVement:SPEed` on page 572
`SCENario:DF:MOVement:SPEed` on page 572

Angle ← Arc trajectory definition
Sets the arc angle and thus defines the arc length.

Remote command:
`SCENario:LOCalized:MOVement:ANGLe` on page 572
`SCENario:DF:MOVement:ANGLe` on page 572

Center Position ← Arc trajectory definition
Sets the "East/North" coordinates of the center of the circle on that the emitter is moving.

If "Maps" are enabled, the center point can also be defined using "Latitude" and "Longitude".

Remote command:
`SCENario:LOCalized:MOVement:EAST` on page 572
`SCENario:LOCalized:MOVement:NORTh` on page 573
`SCENario:LOCalized:MOVement:CLATitude` on page 571
`SCENario:LOCalized:MOVement:CLONgitude` on page 571
`SCENario:DF:MOVement:EAST` on page 572
`SCENario:DF:MOVement:NORTh` on page 572
`SCENario:DF:MOVement:CLATitude` on page 571
`SCENario:DF:MOVement:CLONgitude` on page 571

Mode ← Arc trajectory definition

Defines the behavior of the moving object when the end of the movement is reached.

"Cyclic" The trajectory is repeated cyclically. Once the "End Position" is reached, file movement starts again from the beginning.

Using this mode is recommended in the following cases:

- A circle trajectory
- A trajectory in which the start and the end positions are close to each other.

"Round Trip" The emitter moves back and forth on the defined trajectory.

"One Way" The movement is executed once.
When the "End Position" is reached, this position is assumed to be a static one.

Remote command:
`SCENario:DF:MOVement:RMODe` on page 576
`SCENario:LOCalized:MOVement:RMODe` on page 576

Trace trajectory definition

You can use the "Trace" function to set a sequence of coordinates (known as "Trace points"), manually.

A trace is defined with:

Trace points ← Trace trajectory definition

Selects the trace point that is configured by the following fields.

Remote command:
`SCENario:LOCalized:LOCation:PSTep:SELect` on page 431
`SCENario:DF:LOCation:PSTep:SELect` on page 431

Longitude ← Trace trajectory definition

Sets the longitude of the selected trace point.

Remote command:
`SCENario:LOCalized:LOCation:LONGitude` on page 570
`SCENario:DF:LOCation:LONGitude` on page 570

Latitude ← Trace trajectory definition

Sets the latitude of the selected trace point.

Remote command:
`SCENario:LOCalized:LOCation:LATitude` on page 570
`SCENario:DF:LOCation:LATitude` on page 570

Altitude ← Trace trajectory definition

Sets the altitude above sea-level of the selected trace point.

Remote command:
`SCENario:LOCalized:LOCation:ALTitude` on page 570
`SCENario:DF:LOCation:ALTitude` on page 569

East from RX ← Trace trajectory definition
Sets the number of meters East, from the receiver to the selected trace point.

Remote command:
`SCENario:LOCalized:LOCation:EAST` on page 551
`SCENario:DF:LOCation:EAST` on page 551

North from RX ← Trace trajectory definition
Sets the number of meters North from the receiver to the selected trace point.

Remote command:
`SCENario:LOCalized:LOCation:NORTh` on page 551
`SCENario:DF:LOCation:NORTh` on page 551

Height ← Trace trajectory definition
Sets the number of meters vertically, from the plane of the receiver to the selected trace point.

Remote command:
`SCENario:LOCalized:LOCation:HEIGht` on page 553
`SCENario:DF:LOCation:HEIGht` on page 553

Distance to RX ← Trace trajectory definition
Number of meters along the line of sight from the receiver to the selected trace point.

Remote command:
`SCENario:LOCalized:DISTance` on page 551
`SCENario:DF:DISTance` on page 551

Azimuth from RX ← Trace trajectory definition
Horizontal angle from the receiver to the selected trace point.

Remote command:
`SCENario:LOCalized:LOCation:AZIMuth` on page 551
`SCENario:DF:LOCation:AZIMuth` on page 551

Elevation from RX ← Trace trajectory definition
Vertical angle from the receiver to the selected trace point.

Remote command:
`SCENario:LOCalized:LOCation:ELEVation` on page 552
`SCENario:DF:LOCation:ELEVation` on page 552

Speed ← Trace trajectory definition
Sets the speed of the moving emitter.

Remote command:
`SCENario:LOCalized:MOVement:SPEed` on page 572SCENario:LOCalized:MOVement:SPEed 100
`SCENario:DF:MOVement:SPEed` on page 572

Mode ← Trace trajectory definition
Defines the behavior of the moving object when the end of the movement is reached.

"Cyclic" The trajectory is repeated cyclically. Once the "End Position" is reached, file movement starts again from the beginning.

Using this mode is recommended in the following cases:
- A circle trajectory
- A trajectory in which the start and the end positions are close to each other.

"Round Trip" The emitter moves back and forth on the defined trajectory.

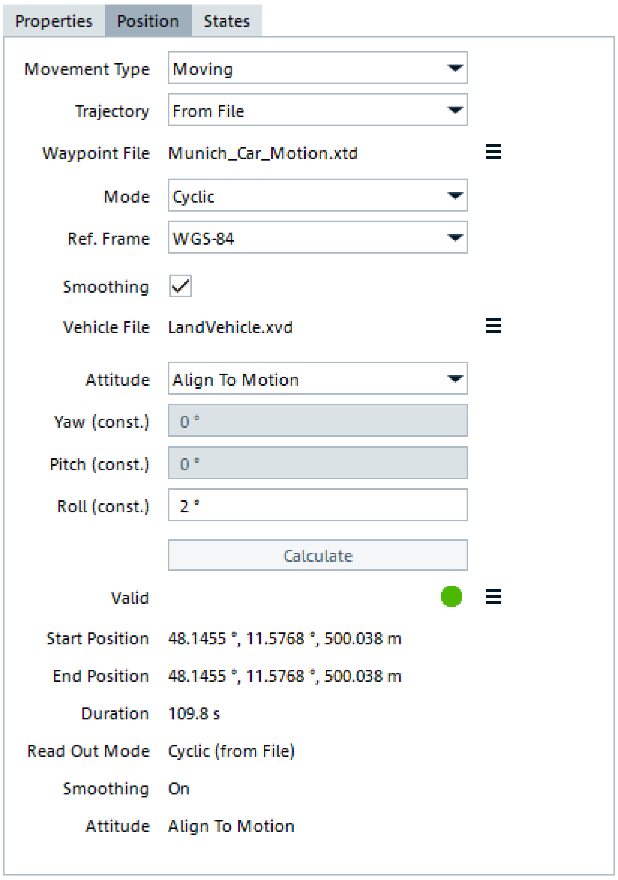
"One Way" The movement is executed once.
When the "End Position" is reached, this position is assumed to be a static one.

Remote command:
`SCENario:LOCalized:MOVement:RMODe` on page 576
`SCENario:DF:MOVement:RMODe` on page 576

Smoothing for traces
The discrete positions (trace points) defined by a "Trace" can cause abrupt changes in the movement direction. This function is identical to the smoothing function used with a waypoints file except that the waypoints are derived from the trace rather than a file (see "Smoothing for waypoints > State" on page 332).

Waypoints trajectory definition
A waypoint file is description of a movement with possibly attitude coordinates that can have different forms, like, for example, a sequence of positions or vector arc movement. A waypoint file must have the extension `*.txt`, `*.kml` or `*.xtd`. For description, see Chapter A.3, "Movement files", on page 638.

You find a subset of example files in the installation package but you can also create and load custom-specific files. Moreover, waypoint smoothing can be activated if vehicle description files are used.



If waypoint files are used, the emitter position and trajectory are retrieved from the file content; the traditional position coordinates parameters are disabled.

Waypoint File ← Waypoints trajectory definition
Indicates the name of the currently used waypoint file.

Load, Delete ← Waypoints trajectory definition
Opens the Windows explorer so that you can navigate to and load waypoint file.

Discards the selected waypoint file.

Remote command:
`SCENario:DF:MOVement:WAYPoint` on page 573
`SCENario:LOCalized:MOVement:WAYPoint` on page 573

Read Out Mode ← Waypoints trajectory definition
Indicates the way the waypoint file is processed.

"Cyclic" The waypoint file is processed cyclically. Once the last waypoint is reached, file processing starts again from the beginning.

Using this mode is recommended if the waypoint file describes one of the following:

- A circle trajectory
- A trajectory in which the start and the end positions are close to each other.

"Round Trip" When the end of the file is reached, the file is processed backwards.

"One Way" The file is processed once.
When the end of the file is reached, the last described position is assumed to be a static one.

Remote command:
`SCENario:DF:MOVement:RMODe` on page 576
`SCENario:LOCalized:MOVement:RMODe` on page 576

Reference Frame ← Waypoints trajectory definition
Select the reference frame used to define the emitters coordinates.

The transformation between the reference frames is performed automatically.

The following applies:
- $X_{WGS84} = (1 - 0.008*10^{-6})*X_{PZ\ 90} - 0.2041*10^{-7}*Y_{PZ\ 90} + 0.1716*10^{-7}*Z_{PZ\ 90} - 0.013$
- $Y_{WGS84} = (1 - 0.008*10^{-6})*Y_{PZ\ 90} - 0.2041*10^{-7}*X_{PZ\ 90} + 0.1115*10^{-7}*Z_{PZ\ 90} + 0.106$
- $Z_{WGS84} = (1 - 0.008*10^{-6})*Z_{PZ\ 90} - 0.1716*10^{-7}*X_{PZ\ 90} - 0.1115*10^{-7}*Y_{PZ\ 90} + 0.022$

Both reference frames are ECEF frames with a set of associated parameters.

"WGS-84" The World Geodetic System WGS-84 is the reference frame used by GPS.

"PZ 90.11 (GLONASS)"
Parametry Zemli PZ (parameters of the Earth) is a reference frame, used, for example, by GLONASS.

Remote command:
`SCENario:DF:MOVement:RFRame` on page 576
`SCENario:LOCalized:MOVement:RFRame` on page 576

Smoothing for waypoints > State ← Waypoints trajectory definition
The discrete positions (waypoints) defined in the waypoints file can cause abrupt changes in the movement direction.

The R&S Pulse Sequencer Digital provides an internal interpolating algorithm that smooths the movement or the trajectory. This algorithm evaluates the dedicated vehicle description (`*.xvd`) file, retrieves the velocity vector and the `<proximity>` value, and inserts waypoints to smooth the trajectory. The resulting movement is more realistic.

To use the algorithm, enable "Smoothing", select "Vehicle File > Load" to load a `*.xvd` file and select "Calculate". See Chapter A.4, "Vehicle description files (Used for smoothening)", on page 645.

Remote command:
`SCENario:DF:MOVement:SMOothening` on page 576
`SCENario:LOCalized:MOVement:SMOothening` on page 576

Vehicle File ← Waypoints trajectory definition
Indicates the name of the currently used vehicle description (`*.xvd`) file.

Load, Delete ← Waypoints trajectory definition

Opens the Windows explorer so that you can navigate to and load vehicle description file.

Discards the selected vehicle description file.

Remote command:
`SCENario:DF:MOVement:VFILe` on page 575
`SCENario:DF:MOVement:VFILe:CLEar` on page 575
`SCENario:LOCalized:MOVement:VFILe` on page 575
`SCENario:LOCalized:MOVement:VFILe:CLEar` on page 575

Attitude ← Waypoints trajectory definition

Defines how the attitude information is defined.

"From Waypoint File"
The attitude parameters are extracted from the selected waypoint file.
Further settings are not required.
This forces the attitude parameters towards motion direction even if the waypoint has attitude information, like, for example, in a `*.xtd` file with `<property waypointformat="position_attitude">`.
For specific emitters like ships or land vehicles, it is realistic to set the yaw and pitch to the vehicle’s motion direction. This is because the usual body axes angles of a land vehicle are in the direction of the velocity vector.
For other emitters, however, like landing planes, this parameter is not useful. As an example, the nose of the plane is in an upward direction at the time when the plane is moving downwards.

"Align to Motion"
Enables a constant rate of change of the roll.

"Constant"
Emitter's attitude is set as the combination of the "Yaw/Heading", "Pitch/Elevation", "Roll/Bank" values.
The resulting attitude is a constant value.

Remote command:
`SCENario:DF:MOVement:ATTitude` on page 574
`SCENario:LOCalized:MOVement:ATTitude` on page 574

Yaw, Pitch, Roll ← Waypoints trajectory definition

Sets the angles of rotation in the corresponding direction, i.e. the rotation around the respective yaw, pitch and roll axes. "Yaw, Pitch, Roll" are defined relative to the local horizon.

Remote command:
`SCENario:DF:MOVement:YAW` on page 575
`SCENario:DF:MOVement:PITCh` on page 575
`SCENario:DF:MOVement:ROLL` on page 575
`SCENario:LOCalized:MOVement:YAW` on page 575
`SCENario:LOCalized:MOVement:PITCh` on page 575
`SCENario:LOCalized:MOVement:ROLL` on page 575

Calculate ← Waypoints trajectory definition
Loads the selected waypoint and vehicle description file.

The "Movement Data" display indicates a summary of the major parameters:
- Start/End Position
- Duration of the movement
- Read Out Mode as defined in the file
- Smoothing state
- Attitude behavior

The status LED indicates the following states:
- **Red**: the waypoint file is not selected, the file is not imported and the movement is not calculated or the file is erroneous.
- **Green**: Movement is calculated

Remote command:
`SCENario:DF:MOVement:IMPort` on page 576
`SCENario:LOCalized:MOVement:IMPort` on page 577

Clear ← Waypoints trajectory definition
Discards the waypoint and vehicle description file.

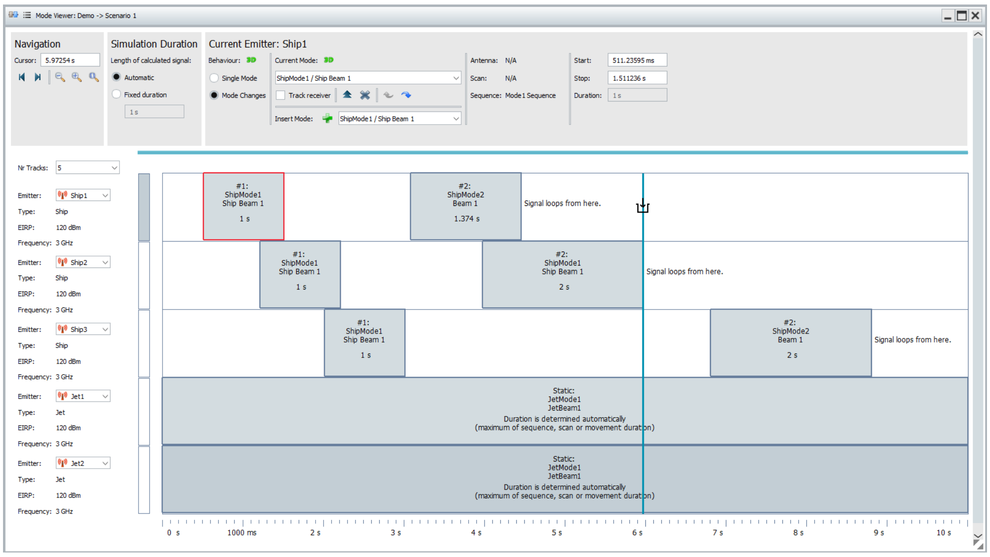
Remote command:
`SCENario:DF:MOVement:CLEar` on page 577
`SCENario:LOCalized:MOVement:CLEar` on page 577

16.6 Mode editor

Access:

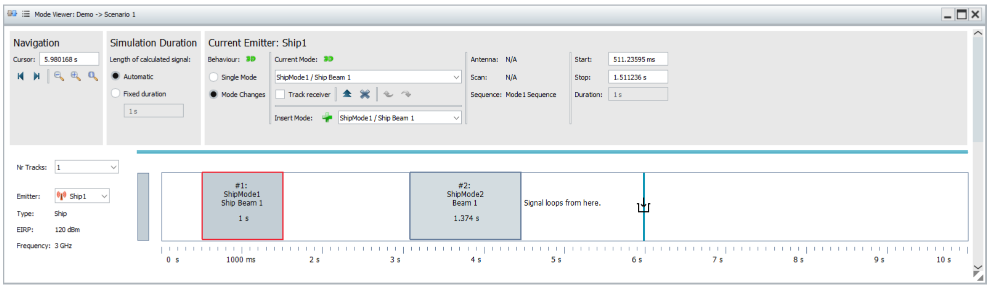
1. Perform one of the following:
   - In a "Scenario Type = Localized Emitters", select "Emitters > 2D".
   - In a "Scenario Type = Direction Finding", select "Map > 2D".
2. In the 2D map, select "Properties > Available TX Items".
3. Select "Edit Modes".

   You access the dialog for configuration of the mode changes of *all emitters* available in the scenario.



4. In the "Properties" of the selected emitter, select "Edit Modes".

   You access the "Mode Editor" dialog for the *current emitter*.



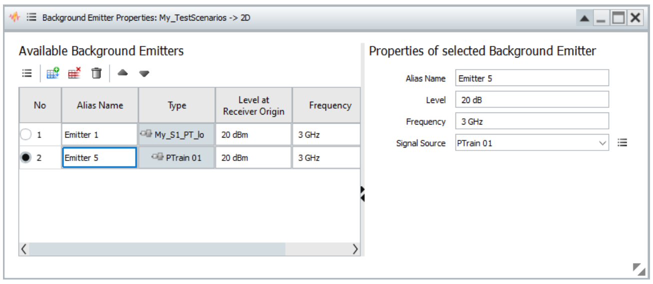
The "Mode Editor" is a graphical editor for joint configuration and defining the duration emitters are working in the different modes.
Operation is similar to Mode Changes settings for each of the emitters. However, rather than defining the start and end time of a mode, you drag the start and end time on the time scale.
The provided settings and configuration possibilities are identical with the configuration of an individual emitter. For description, see Chapter 10.3.3, "Mode editor settings", on page 230.

16.7 Background emitters properties

Access:

1. Perform one of the following:
   - In a "Scenario Type = Localized Emitters", select "Emitters > 2D".
   - In a "Scenario Type = Direction Finding", select "Map > 2D".

2. In the 2D map, double-click the signal in the "Background Signals" block.



Settings:

Available Background Emitters

Displays a list of the sequences used to create the individual background signals.

Use the standard functions in the context menu to add, reorder, or remove items.

Background emitters are identified by their "Alias Name", "Type", "Level" and used "Frequency".

Background emitters do not have antennas, specific positions on the 2D map or direction.

See also "General workflow for creating complex 2D scenarios" on page 290.

Remote command:
`SCENario:LOCalized:ALIas` on page 548
`SCENario:DF:ALIas` on page 548
`SCENario:LOCalized:WAVeform:LEVel` on page 554
`SCENario:DF:WAVeform:LEVel` on page 554
`SCENario:LOCalized:WAVeform:FREQuency` on page 555
`SCENario:DF:WAVeform:FREQuency` on page 555

Properties of selected Background Emitter

Displays and sets the properties for the selected background emitter.

Changing the properties here updates the values for the corresponding emitter in the "Available Background Emitters" list.

Alias Name ← Properties of selected Background Emitter

Sets the name of the selected emitter.

Remote command:
`SCENario:LOCalized:ALIas` on page 548
`SCENario:DF:ALIas` on page 548

Level ← Properties of selected Background Emitter
Sets the level at the receiver for the selected emitter.

The value is the maximum level at the receiver origin. The calculation considers only emitter EIRP and free-space attenuation and omits individual antenna attenuation due to position and polarization.

Remote command:
`SCENario:LOCalized:WAVeform:LEVel` on page 554
`SCENario:DF:WAVeform:LEVel` on page 554

Frequency ← Properties of selected Background Emitter
Sets the frequency of the selected emitter.

Remote command:
`SCENario:LOCalized:WAVeform:FREQuency` on page 555
`SCENario:DF:WAVeform:FREQuency` on page 555

Signal Source ← Properties of selected Background Emitter
Sets the signal source of the selected emitter.

To change the source, select one from the drop-down list.

To edit the current source or add a new one, select the icon and then select the required item from the menu.

Remote command:
`SCENario:LOCalized:WAVeform` on page 554
`SCENario:DF:WAVeform` on page 554

16.8 Platforms with multiple emitters

Access:

1. Open a suitable scenario.

   Use any scenario that offers "Map" in the "Scenario Creation" block.
   - "Localized Emitters"
   - "Direction Finding"

2. In the block diagram, select "Map".

   The "2D" dialog displays a two-dimensional view of the receiver and currently configured emitters and platforms.

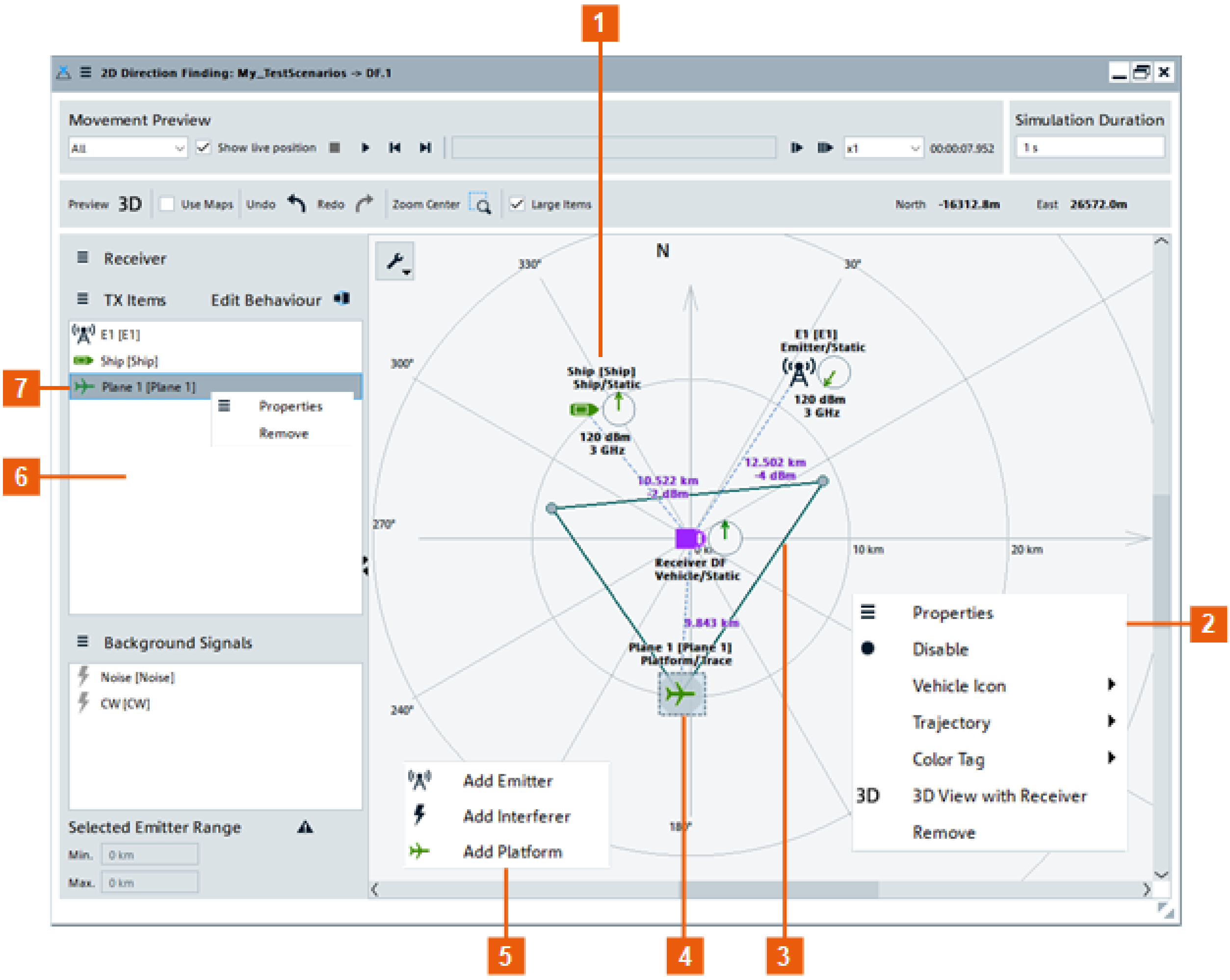


Figure 16-13: Platform-related functions of the 2D dialog

1 = Static platform
2 = Platform context menu
3 = Trajectory of moving platform
4 = Moving platform
5 = Map area context menu (right-click)
6 = "TX Items" list
7 = TX Items context Menu (right-click)

Figure 16-13 shows the platform-related elements of the "2D" dialog.

Adding and configuring platforms

Use the following procedure to add and configure platforms in a scenario:

1. Access the "2D" dialog (as described previously).
2. Add platforms to the list.

   Use one of the following methods:
   - Drag&drop a platform from the repository.

- Right-click in the map area and select "New Platform" in the context menu (see Chapter 16.8.2, "Map area context menu", on page 340).

A platform icon appears in the "Platforms" list and on the map.

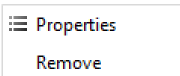
3. Configure the platform.

Use one of the following methods:
- Right-click a platform icon on the map.
A context menu appears. Select "Properties".
See Chapter 16.8.3, "Platform context menu", on page 340.
- Right-click an entry in the "Platforms" list.
A context menu appears. Select "Properties".
See Chapter 16.8.1, "TX items context menu", on page 339.
- Double-click a platform icon or "Platforms" list entry.
Opens the "TX Items Properties" dialog.
See Figure 16-11.
- Click the "TX Items" icon in the list of "Properties" links.
Opens the "TX Items Properties" dialog.
SeeFigure 16-11.

Settings:

16.8.1 TX items context menu

To access this context menu, right-click an entry in the list.



The following functions are available on the context menu:

Properties
Opens the "TX Items Properties" dialog.

See Figure 16-11.

Remove
Removes the selected platform from the list.

16.8.2 Map area context menu

To access this context menu, right-click anywhere in the map area that is not occupied by an object (e.g. an emitter).

The following functions are available on the context menu:

Add Emitter

Adds an emitter icon to the map at the cursor position.

Adds a corresponding entry to the "Emitters" list.

Note: Double-click the entry or the icon to configure it in the TX Items Properties dialog.

Add Platform

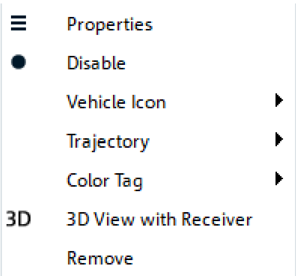
Adds a platform icon to the map at the cursor position.

Adds a corresponding entry to the "Platforms" list.

Note: Double-click the entry or the icon to configure it in the TX Items Properties dialog.

16.8.3 Platform context menu

To access this context menu, right-click a platform icon in the map area.



The following functions are available on the context menu:

Properties
Opens the "TX Items Properties" dialog.

See Chapter 16.4, "Properties of TX items (emitters and platforms)", on page 307.

Enable/Disable
Toggles the platform's emitters on/off within the scenario.

Determines whether the platform is considered in a calculation or not.

Vehicle Icon
Selects the icon for the platform on the 2D map.

The icons are merely indication; the function and the properties of the platform do not depend on the icon.

You can import and use custom icons, see Chapter 16.11, "Importing user icons", on page 345.

Trajectory
Defines the path followed by a moving platform.

R&S Pulse Sequencer Digital supports several trajectory types, including: line, arc, trace and waypoint.

See Chapter 16.5, "Moving emitters and platforms", on page 323.

See Chapter 16.9, "Creating trajectories on a 2D map", on page 341.

Color Tag
A platform icon can have a colored background.

Set the color to: "None", "Green", "Blue", or "Red".

3D View with Receiver
Opens a "3D" view containing the selected platform and the receiver/DF.

See Chapter 16.3, "3D scan pair view settings", on page 302.

Remove
Removes the selected platform from the list.

16.9 Creating trajectories on a 2D map

Option:R&S PULSE-K39

Access:

1. In a "2D Map", right-click to open the context menu.
2. In the context menu, select "Trajectory" to open a list of trajectory types.
3. Select a trajectory type from the list.
4. Use the mouse-keys to create the trajectory on the map.

Settings

Line

A line appears between the current position of the emitter and the mouse pointer.

Move the mouse pointer to adjust the position and direction of the line. Left-click to fix the endpoint of the line trajectory.

Arc

A circle is drawn that centers on the mouse pointer and passes through the emitter position. Left-click to fix the center of the circle. Move the mouse pointer to define the length of the arc. Left-click to fix the endpoint of the arc trajectory.

Trace

A line appears between the current position of the emitter and the mouse pointer.

Move the mouse pointer to adjust the position and direction of the line. Left-click to set a trace point. Set the required number of trace points. Right-click to apply the trace (this action does not set an additional trace point).

Waypoints

This option only defines the type of trajectory. You must also define the waypoints file.

Select the emitter > right-click > "Properties" > "Position" tab > "Waypoint file"

16.10 Using georeferenced maps

The expert option R&S PULSE-K39 enables you to use georeferenced maps with R&S Pulse Sequencer Digital.

The key benefits of this option are that you can:

- Visualize a real-world geographical scenario.
- Accurately position emitters to emulate real-world radar installations:
  – Drag and drop the icons on the map.
  – Use the actual coordinates (lat/lon) of the emitters.
- Create realistic movement paths:
  – Ships moving in shipping lanes.
  – Planes using air corridors and flight levels.
  – Vehicles driving on roads.
- Easy handling:
  – Load and unload maps via toolbar.
  – Reposition map using left mouse-key.
  – Zoom in/out using mouse scroll-button.

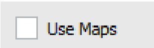
Access:

1. Open a suitable scenario.

   Use any scenario that offers "Map" in the "Scenario Creation" block.
   - "Localized Emitters"
   - "Direction Finding"

2. In the block diagram, select "Map".

   The "2D" dialog displays a 2D view of the receiver and currently configured emitters, together with their main characteristics.



3. In the "2D" dialog, select the "Use Maps" checkbox.

   Note: If you enable this function, the scenario uses the R&S PULSE-K39 option and it **cannot be removed** from the signal generation of the scenario.
   For this reason, a popup message appears asking you to confirm.

   Select "No" if you do not want the scenario to depend on the extended option.

   Select "Yes" to proceed.

The "Map Options" toolbar appears.

Figure 16-14: Map Options toolbar

1 = "Open Map" file browser
2 = Unload map
3 = "Application Colors" dialog
4 = Set opacity of loaded map

Settings:

Use Maps
Option: R&S PULSE-K39

Enables the scenario to use georeferenced maps.

Once enabled, it cannot be disabled (see the note in "Access:" on page 343).

Application Colors
Customizes the colors used in R&S Pulse Sequencer Digital.

Alternative access: select "Configure" > "Colors". See also Changing Colors).

Use the "Map" tab to customize the map-specific colors.

Open Map

Use this dialog to select a georeferenced map file.

The file browser only displays acceptable file formats.

Supported formats:
- `.tif`
- `.tiff`

If the format of the selected file is supported, the map appears in the "2D" diagram and "3D Live Vsiaulization".

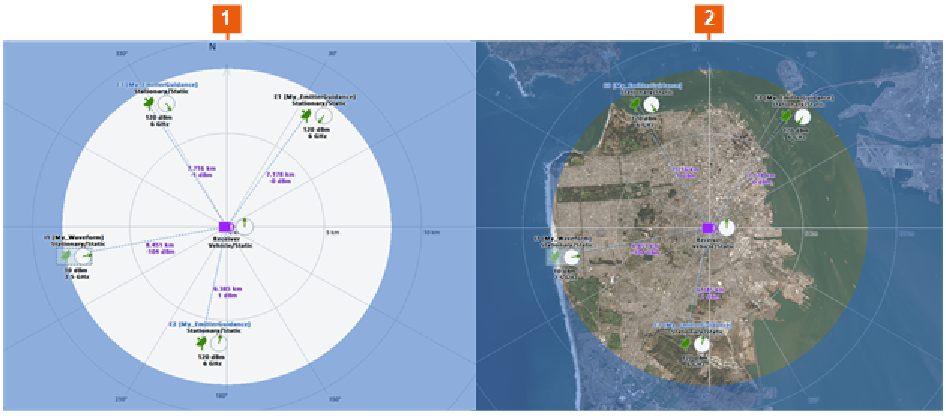


Figure 16-15: 2D diagram before and after loading a map

1 = Scenario with a plain background
2 = Scenario with a georeferenced map

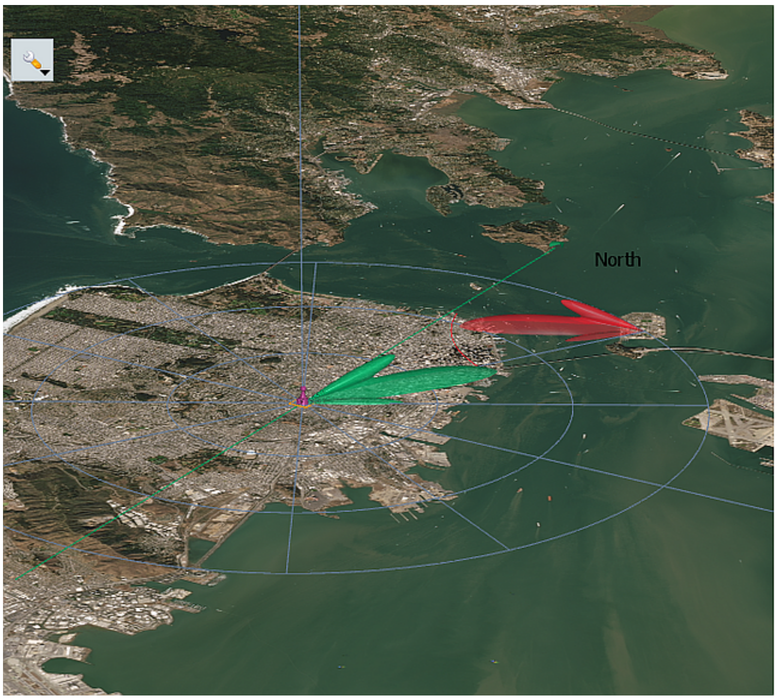


Figure 16-16: 3D Live Visualization with map background

If the selected file does not contain the required georeferencing metadata, R&S Pulse Sequencer Digital outputs error messages in the "Message Log".

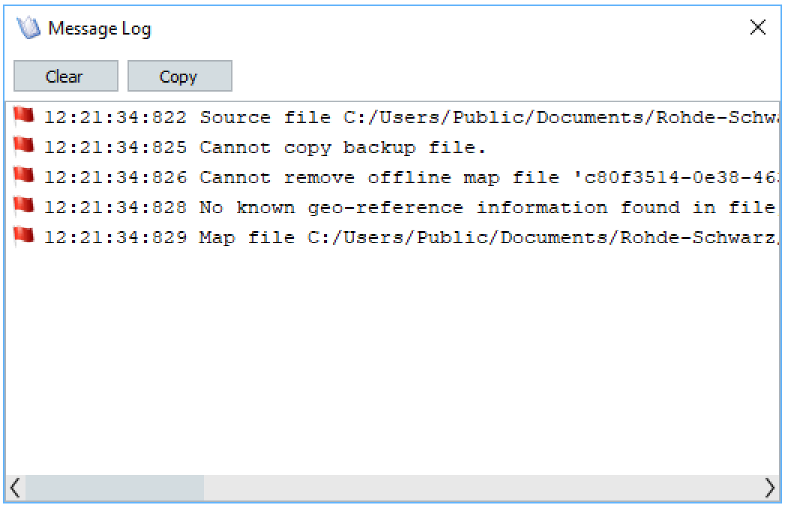


Figure 16-17: Typical errors caused by missing metadata

Unload map
Use this function to unload the current map.

The "2D" diagram is not affected by this action - e.g. the current map centering and zoom level stay the same.

Note: The map is unloaded immediately.
The application does not ask you to confirm the action.

There are no negative consequences if you accidentally unload a map. Use the "Open Map" dialog to load the map again.

Opacity
Use this function to make the map or the scenario more or less prominent.

Move the slider left/right to make the background color more/less visible through the map.

For example:
- To focus on the map, set the slider fully to the right.
- To focus on the scenario, set the slider fully to the left.
- To show the scenario in relation to the environment, set the slider in the middle.

16.11 Importing user icons

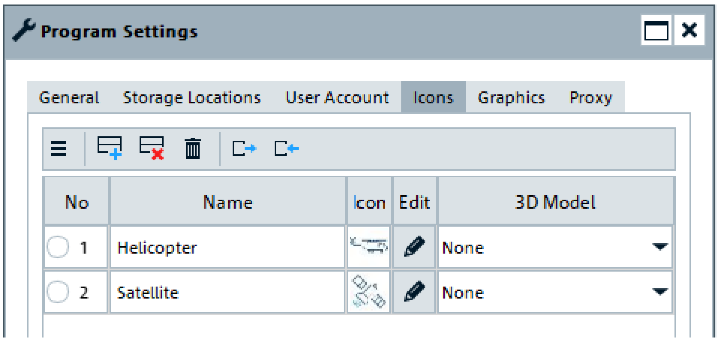
You can import custom icons to expand the list of available icons for displaying RX and TX items in a scenario.

1. In the menu bar, select "Configure > Settings > Icons".

   An empty list is shown.

2. Add icons to your list.

   Use one of the following methods:
   - Add single icons to the list by using the standard functions, see Table 2-4.
   - Import an icon archive that was created by the pulse sequencer software.



3. With the icons in the list you can do the following:
   - Rename the icon.
   - Edit the icon.
   - Select a 3D model for it.
   - Export the icons to create an archive.
4. Finish with "Apply".

   You can now select the user icons from the list of "Vehicle Icons" in a scenario, see Chapter 16.8.3, "Platform context menu", on page 340.

Settings:

Export/import icons as archive
Export the icons to create an archive or import an icon archive that was created by the pulse sequencer software.

Name
Enter the name of the icon.

Icon
Displays the imported user icon.

Edit
Change the displayed user icon.

3D Model
Select a 3D model for the icon from a list of predefined models. The selected 3D model is used to display the TX/RX item in the interactive 3D display.

"None": a cone is shown for the TX/RX item in the interactive 3D display.

17 Configuring the simulation

In order to generate and export the simulated signals from R&S Pulse Sequencer Digital, you need to configure the settings in the "Scenario" dialog.

The following chapters explain the necessary steps.

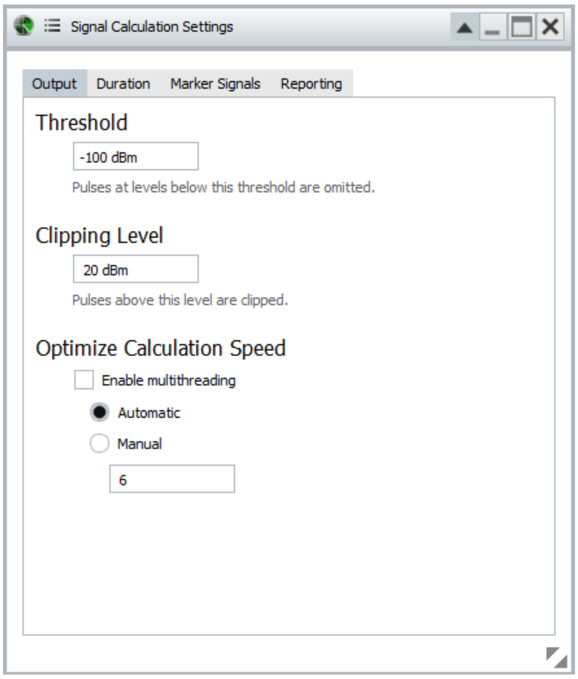
17.1 Signal calculation settings

Access:

► In the "Scenario" dialog, select "Signal Calculation > Config".

Settings:

Output



Threshold ← Output
Enters a level threshold to limit the dynamic range of the signal. Pulses at levels below this threshold are omitted.

The maximum "Threshold" is 0 dBm.

Remote command:
`SCENario:OUTPut:THReshold` on page 564

Clipping Level ← Output
Enters a maximum level to limit the dynamic range of the signal. Pulses at levels above this threshold are reduced (*clipped*) to the configured level.

Remote command:
`SCENario:OUTPut:CLIPping` on page 562

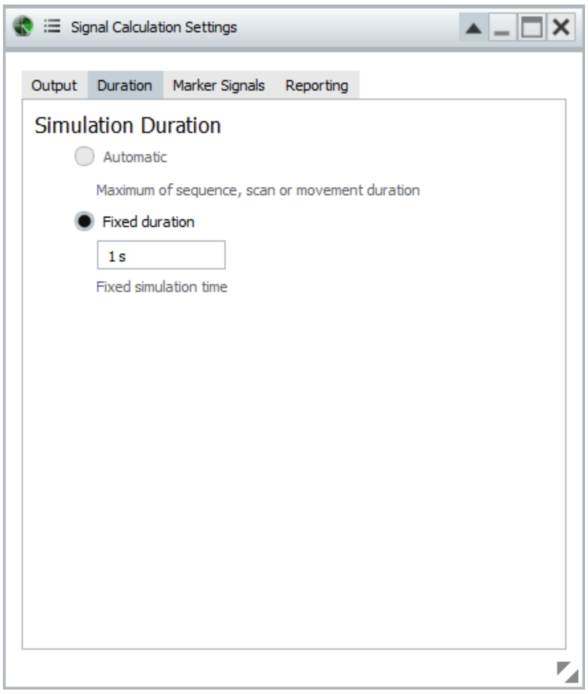
Optimize calculation speed ← Output
Enable this parameter to apply multithreading and decrease the calculation time.

Select one of the options:
- "Automatic"
  The system automatically applies the optimum number of threads for the scenario.
- "Manual"
  Define the number of threads to be used.

See also Chapter 22.4, "Speeding up calculation", on page 404.

Remote command:
`SCENario:OUTPut:MULTithread` on page 564
`SCENario:OUTPut:MTMode` on page 564
`SCENario:OUTPut:MTTHreads` on page 564

Duration



Provides duration-related settings:

"Duration" Defines the simulation content.

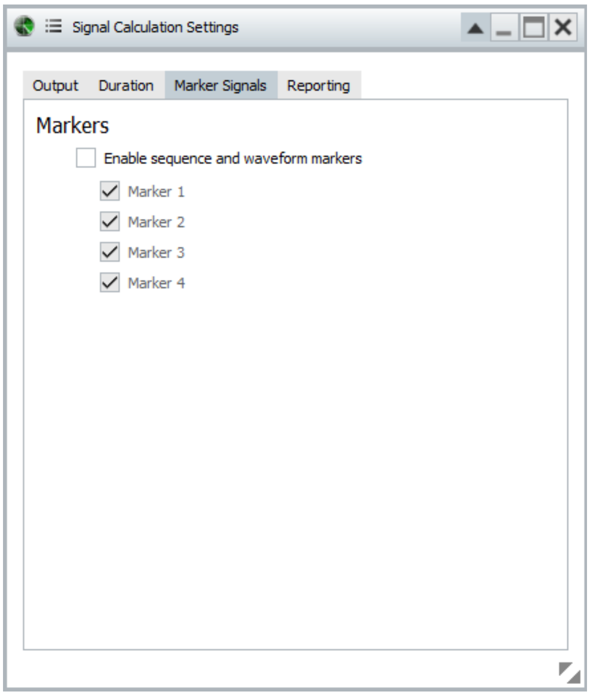
The generated signal can:

- Last a specified duration.
  Short sequences are repeated, longer ones are truncated.
- Have an automatically determined maximum content and duration that best fits the current scenario and configuration.
  Short sequences are repeated, longer ones are truncated.

See also Table 16-2.

Remote command:
`SCENario:OUTPut:DURation:MODE` on page 562
`SCENario:OUTPut:DURation:TIME` on page 562

Marker Signals

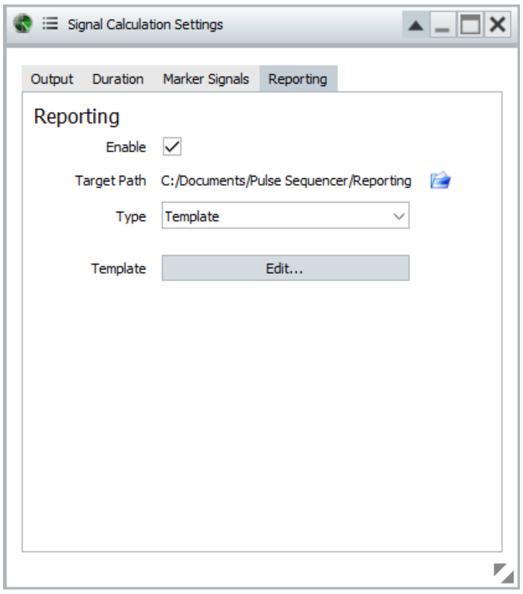


Enables configured markers to be considered in the generated file.

See "Global Marker Signals" on page 381.

Remote command:
`SCENario:OUTPut:MARKer:ENABle` on page 499
`SCENario:OUTPut:MARKer:FLAGs` on page 499

Reporting



Enables and configures report file generation.

Note: Settings change depending on which "Type" option is selected.

See Chapter 21, "Creating reports and documenting measurement results", on page 386.

17.2 Assign signals to destination

In "Localized Emitters" and "Direction Finding" scenarios, you have to select the signal of an emitter or receiver that you want to export to the connected plugin.

In the "Scenario" dialog, perform the following:

1. For scenario type "Localized Emitters", select "Signal Generation Export Plugin > Assign" to open the "Assign Signals to Destinations" dialog.
   For scenario type "Direction Finding", select "Signal Generation Export Plugin > Assign" to open the "Assign Receiver Signals to Destinations" dialog.
2. For all scenario types, select "Signal Generation Export Plugin > Config" to open the "Destinations" dialog.

   The name and the contents of the dialog change depending on the scenario but the settings and the workflow are similar.

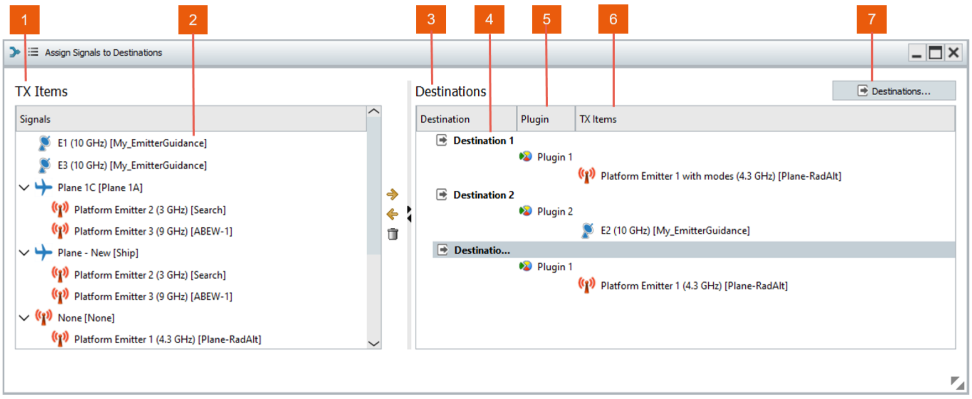


Figure 17-1: Assign Signals to Destinations: understanding the displayed information (Scenario > Signal Generation Export Plugin > Assign)

1 = Available "TX Items" or "Interleaving Groups" if enabled (Option:R&S PULSE-K39 required for interleaving)
2 = Unassigned emitters
3 = List of destinations (entries depend on "Scenario > Signal Generation > Config", see "Available for Assignment" on page 354)
4 = Destination name
5 = Available plugins assigned to destination
6 = Signal of emitter assigned to plugin/details of interleaving groups if enabled (see "Interleaving Groups" on page 353 and "Groups Contents" on page 354)
7 = Opens "Destinations" dialog (see Chapter 3.3, "Destinations settings", on page 56)

For step-by-step instructions, see:

- Chapter 16.1, "How to create scenarios with receiver and TX items", on page 290

Settings:

TX Items

Common representation of the available signals.

Tip: You can assign several signals to the same plugin.

Emitters ← TX Items

Lists the alias names of all configured emitters that are still not assigned to a destination.

Select an emitter and drag and drop it on one of the plugins.

Remote command:
ASSignment:EMITters:LIST? on page 465
ASSignment:EMITters:SELect on page 431

Receiver Signals ← TX Items
Lists all receiver signals that are still not assigned to a destination.

There is one receiver signal per receiver antenna element and emitter combination. The name of the receiver signal follows the syntax:

"<Antenna Element Alias> - <Emitter Name Alias>", as selected with the parameters:
- "Antenna Configuration" on page 266
- "Alias Name" on page 312

Remote command:
ASSignment:ANTennas:LIST? on page 466
ASSignment:ANTennas:SELect on page 431

Interleaving Groups ← TX Items
Option:R&S PULSE-K39

In scenarios that have interleaving enabled, the "Interleaving Groups" list shows all interleaving groups defined for the selected scenario.

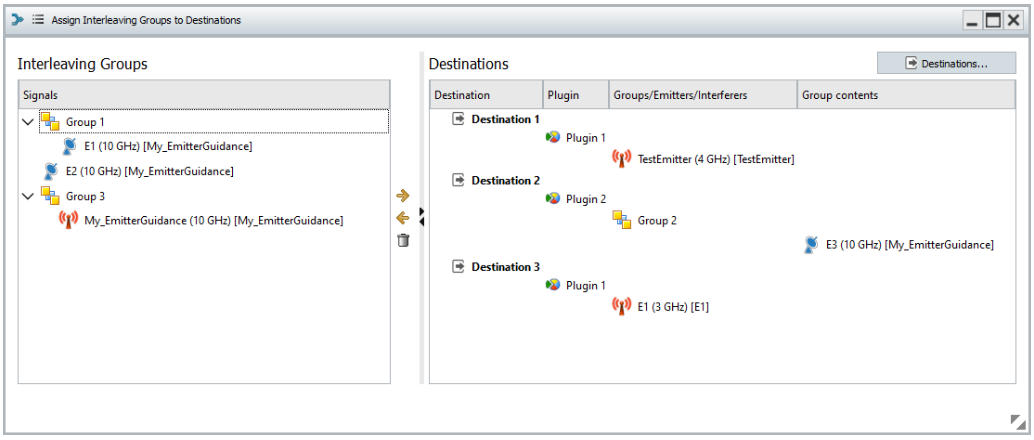


Figure 17-2: Assign Interleaving Groups to Destinations

Click the group name to expand the group content. All emitters assigned to the particular group are shown.

Interleaving groups are assigned jointly. Select the group name and drag and drop it on one of the plugins.

See also "Groups/Emitters" on page 354.

Remote command:
ASSignment:GROup:LIST? on page 467
ASSignment:GROup:SELect on page 467

Add, Remove, Clear
Use these buttons to add a selected emitter to or remove it from the selected plugin.

Alternatively, use the drag and drop method.

Remote command:
ASSignment:DESTination:PATH:EMITter:ADD on page 429
ASSignment:DESTination:PATH:EMITter:SELect on page 431
ASSignment:DESTination:PATH:EMITter:DELete on page 432
ASSignment:DESTination:PATH:EMITter:CLEar on page 433
ASSignment:DESTination:PATH:ANTenna:ADD on page 429
ASSignment:DESTination:PATH:ANTenna:SELect on page 431
ASSignment:DESTination:PATH:ANTenna:DELete on page 432
ASSignment:DESTination:PATH:ANTenna:CLEar on page 434

Destinations
Lists the destinations with their names, available plugins and assigned emitters or interleaving groups with their group content.

Available for Assignment ← Destinations
The number of listed destinations depends on "Scenario > Signal Generation Export Plugin > Config":

- "Scenario > Signal Generation Export Plugin" > **"Assign"**
  Only the plugin destinations that are available in the current setup in the "Destinations" dialog are listed here.
  See Chapter 3.3, "Destinations settings", on page 56.
- "Scenario > Signal Generation Export Plugin" > **"Config"**

Lists all destinations that are available in the repository.

Remote command:
ASSignment:DESTination:LIST? on page 466
ASSignment:DESTination:SELect on page 431
ASSignment:DESTination:PATH:LIST? on page 466
ASSignment:DESTination:PATH:SELect on page 431
ASSignment:DESTination:PATH:EMITter:LIST? on page 467
ASSignment:DESTination:PATH:ANTenna:LIST? on page 467

Plugin ← Destinations
Indicates the available plugins for each destination.

Groups/Emitters ← Destinations
Indicates the alias names of the assigned signal sources. If interleaving is enabled, displays the group names of the assigned signals (see Figure 17-2).

Groups Contents ← Destinations
In scenarios with enabled interleaving, indicates the emitters or PDW lists that are assigned to the particular plugin (see Figure 17-2).

18 Working with PDWs

The pulse description word (PDW) is a commonly used file format that describes radar signals. The PDWs contain the radar signal parameters for each pulse together with a timestamp that defines the pulse start time. Sometimes the PDWs can originate from earlier simulations or live recordings. In other cases, you would like to generate the radar test signal directly in PDW format.

In R&S Pulse Sequencer Digital, you have the following different possibilities to use PDWs:

- **Import custom PDW lists ("Imported Signals")**
  Instead of configuring pulses, MOPs, sequences and IPM effects, you can import your custom PDW lists.
  Based on files with mapping rules, the R&S Pulse Sequencer Digital PDW import interface transforms the custom-specific PDW list into the Rohde & Schwarz proprietary PDW format.
  You can use imported signals in PDW lists in scenario sequences.

  Custom PDWs can be imported in any scenario type, but the created output file differs depending on the scenario type:
  – The "PDW List (Collection)" scenario is dedicated for importing custom PDWs.
  – In scenario other than the "PDW List (Collection)", the created output file is a waveform.
  Waveforms describing long simulation time are usually large.
- **Merging the signals of several emitters** (several PDW lists)
  Per default, you use the "PDW List (Collection)" scenario to configure several PDW lists and switch between them sequentially.
  However, you can also combine them into a single output file. During the interleaving process, overlapping PDWs with lower priority are dropped, where you define the priority of the individual PDW lists (see Figure 18-2).

The solution above introduce the possibilities to generate or regenerate test signals with the R&S Pulse Sequencer Digital.

For step-by-step instructions, see:

- Chapter 18.1, "How to import PDW lists and create output files", on page 356.

For description of the related settings, see:

- Chapter 18.5, "PDW list (Collection) settings", on page 368.
- Chapter 18.7, "PDW interleaving progress statistics", on page 373.
- Chapter 18.8, "Signal preview with interleaved and dropping statistics", on page 374.

18.1 How to import PDW lists and create output files

To import PDW list data

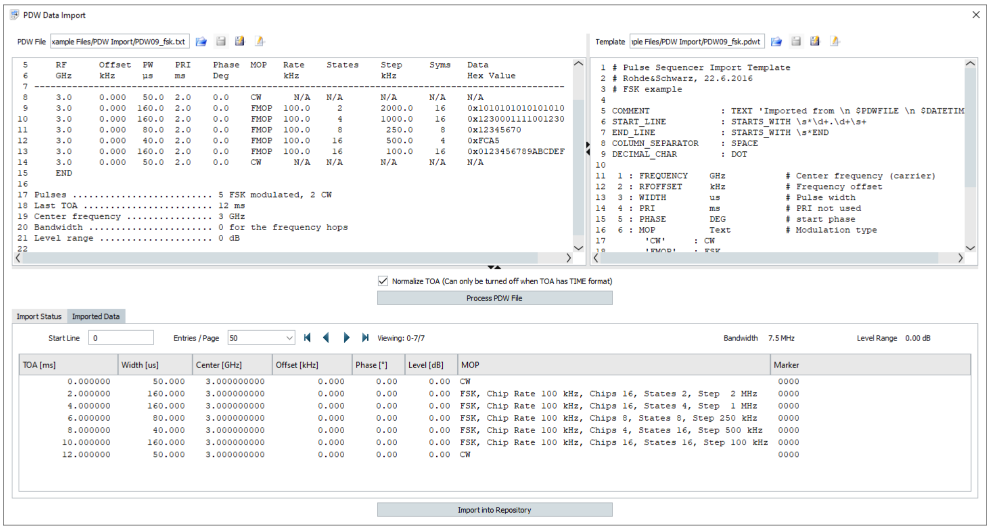
The import of custom PDW list files requires a template that describes how to extract the information form the PDW list file.

The instruction shows you how to operate the software and import the PDW list files. We assume that the template file is created, as described in the document "R&S Pulse Sequencer Digital PDW Import Interface Specification".

1. Select "Repository Tree > Imported Signals > New".
2. Select "Type > PDW Data".
3. Select "Import".
4. In the "PDW Data Import" dialog, load the PDW list file and the PDW template.
   a) Select the "Select PDW File" icon.
      PDW list files are custom text-based PDW lists in ASCII or coma-separated file format (`*.txt` or `*.csv`).
   b) Select the "Import template file" icon.
      Template files are ASCII files with predefined file format and file extension `*.pdwt`.
5. Select "Process PDW File".

   The PDW list file is parsed. The "Import Status" dialog displays parsing process information.
6. Select "Imported Data" to observe detailed information on the imported file.

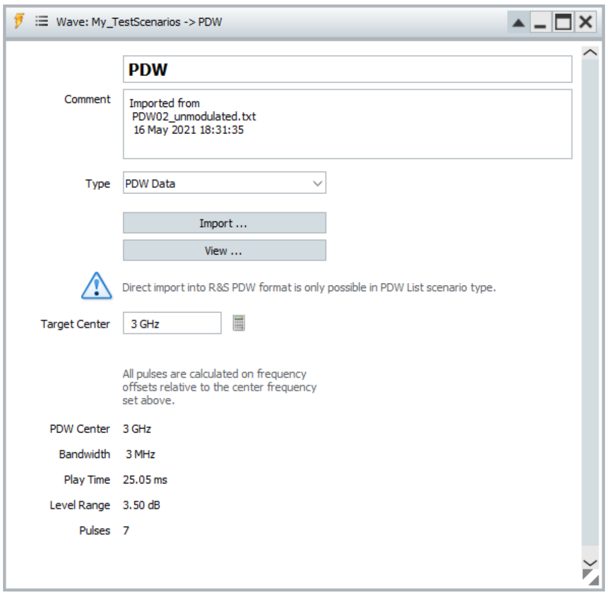
   The imported data is temporarily stored in memory.

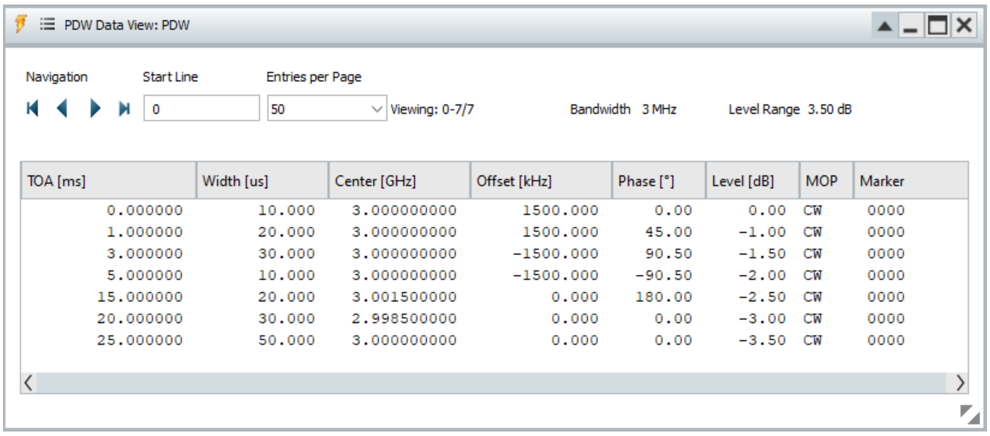
7. Select "Import into Repository" to store the data permanently in the repository.

   The imported PDW information can be used as part of a sequence.

8. Observe the overview information in the "Wave" dialog.



9. Select "View" to display more details on the waveform.



The dialog shows the waveform bandwidth, the level range and the waveform content as a sequence of pulses with their main characteristics.
The displayed information resembles the information in the "PDW Data Import > Imported Data" dialog.

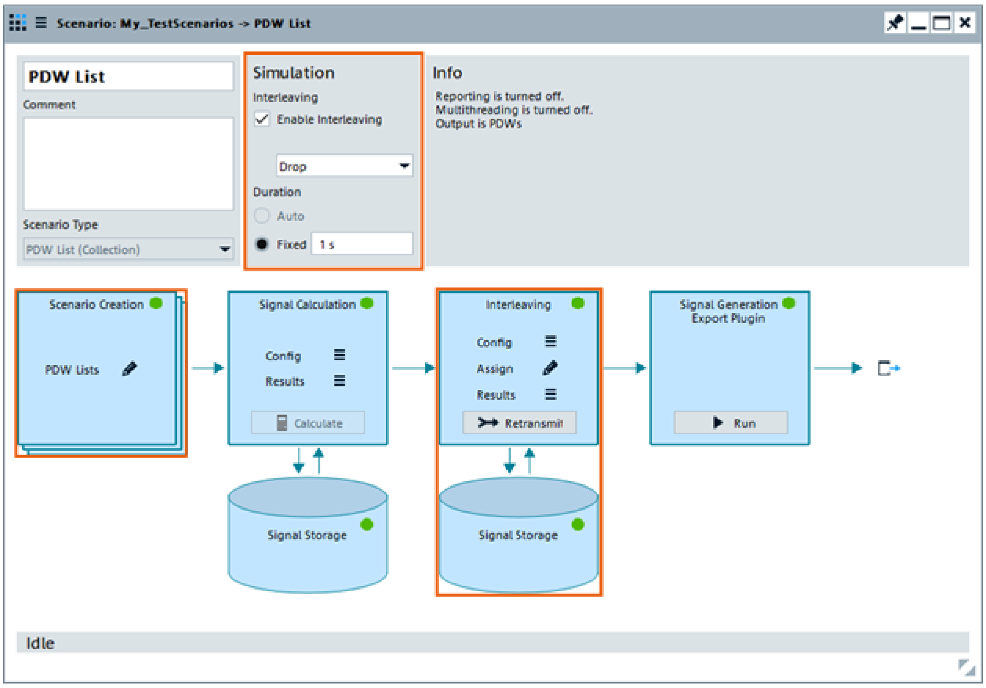
10. Assign the waveform to a sequence.

11. Assign the sequence to a scenario.
    a) Calculate the signal.
    b) In the "Scenario" dialog, select "Signal Calculation > Results > View".

    The "Signal Preview" dialog displays the calculated unmodulated pulses, each with the specified timing settings.
    For more information, see Chapter 15, "Visualizing and analyzing signals", on page 284.

Create PDW list scenario

1. Select "Scenaros > New" > **"Scenario Type = PDW List (Collection)"**.



2. Select "Scenario Creation > PWD Lists" to open the "PDW List (Collection)" dialog.
3. Select "Append" to add PDWs.
4. Enter an alias name.
   Select "PDW List" and select an existing waveform element containing an imported PDW.
5. Option: R&S PULSE-K39.
   To merge the PDW lists into a single output file, in the block diagram select "Simulation > Interleaving > On".

   The "Interleaving" block appears on the block diagram.
6. Select "Interleaving > Config".

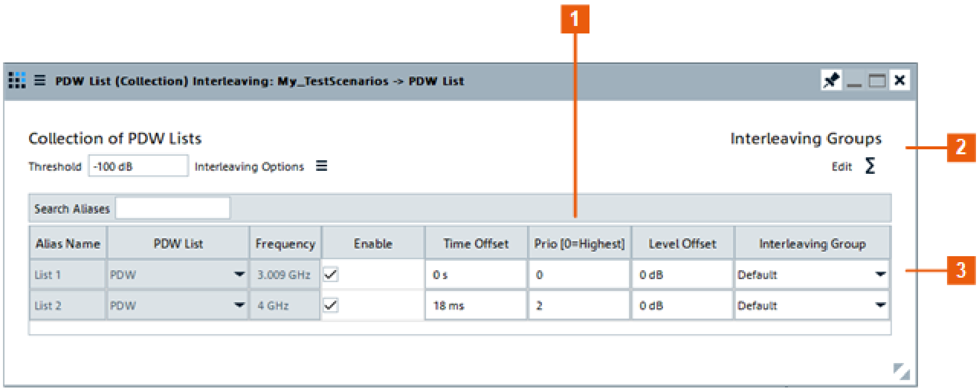


Figure 18-1: PDW List (Collection) Interleaving: understanding the displayed information

1 = Priority of the selected PDW lists
2 = Enables you to change settings for interleaving groups
3 = Enables you to select an interleaving group

7. For each PDW, define a "Priority".

   Multiple lists can be interleaved into a single output file using a priority-based dropping algorithm.

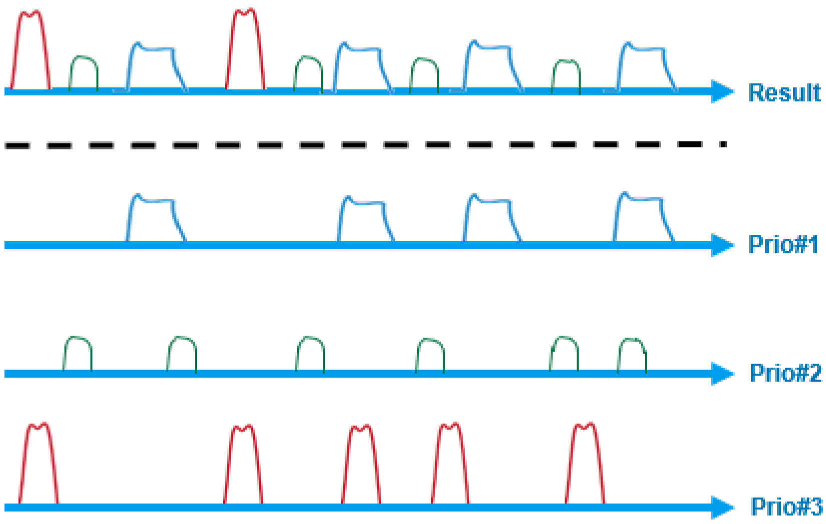


Figure 18-2: Principle of the priority-based interleaving algorithm

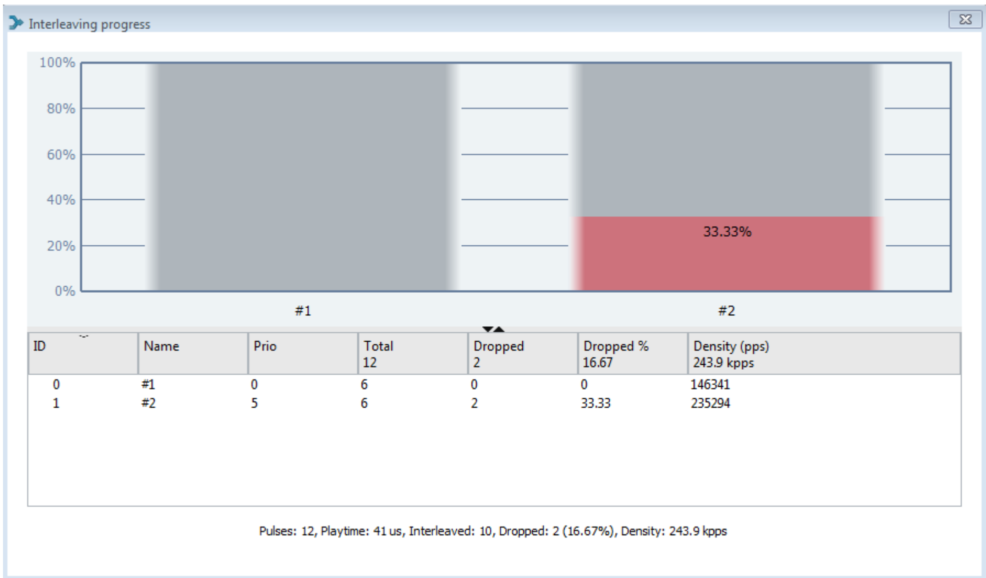
Prio#1 = PDW list with the highest priority (i.e. the priority value is 0 or smaller than the value for the other PDW lists)
Prio#3 = PDW list with the lowest priority (i.e. the highest priority value within the PDW lists)

8. Select a destination that supports the following **mandatory** options:
   - Select a destination with a valid export plugin
   - Multiple emitters
   - Extended sequencing R&S PULSE-K39.
9. Select "Signal Calculation > Calculate".

The signal is calculated and the "Signal Calculation" status indication is green.
The signal calculation and interleaving steps are separated from each other. The interleaving can be applied optionally, as post-processing on the previously calculated data. As long as the "Signal Calculation" settings are not changed, signal recalculation is not required.

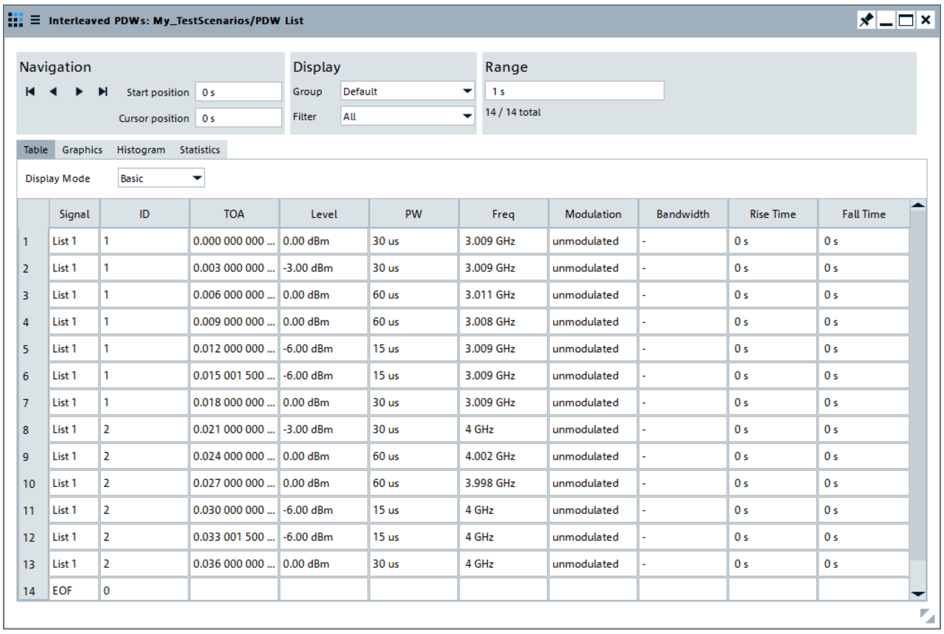
10. Option:R&S PULSE-K39.
Select "Interleaving > Interleave"

During the interleaving, the "Interleaving progress" display indicates the percentage of pulses being processed and dropped in each of the PDW lists.



11. To observe more information on the created output files, in the "Scenario" dialog:

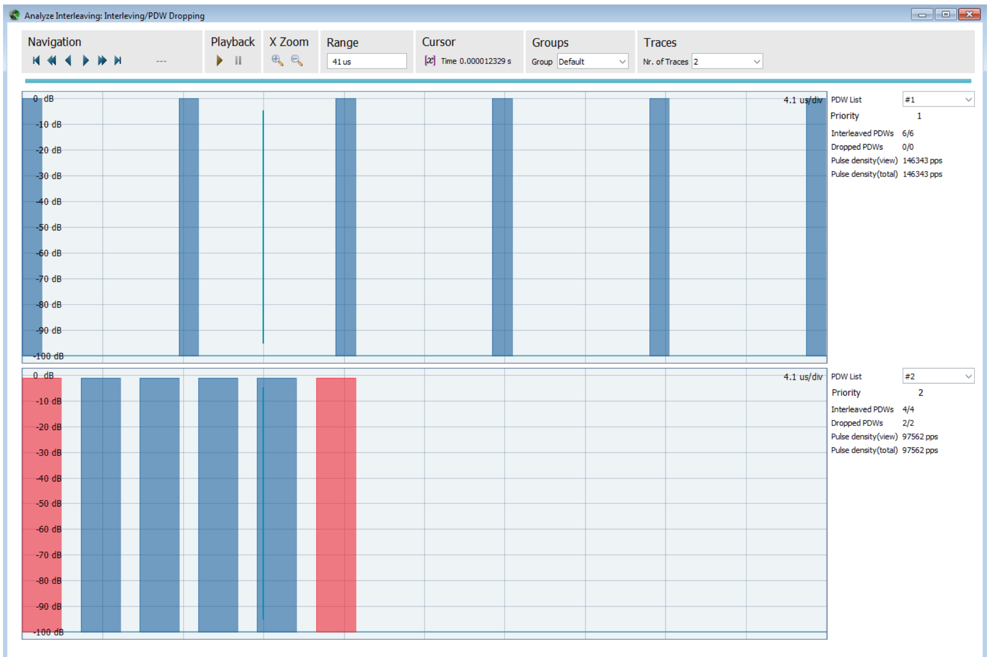
a) Select "Interleaving > Results > View > Interleaved PDW/Dropped PDWs".
Both views display summary information on the interleaved or dropped PDWs and a drop out statistic. The information is represented in a table and in a graphical form ("Table"/"Statistic" and "Graphics"/"Histogram").



	Signal	ID	TOA	Level	PW	Freq	Modulation	Bandwidth	Rise Time	Fall Time
1	List 1	1	0.000 000 000 ...	0.00 dBm	30 us	3.009 GHz	unmodulated	-	0 s	0 s
2	List 1	1	0.003 000 000 ...	-3.00 dBm	30 us	3.009 GHz	unmodulated	-	0 s	0 s
3	List 1	1	0.006 000 000 ...	0.00 dBm	60 us	3.011 GHz	unmodulated	-	0 s	0 s
4	List 1	1	0.009 000 000 ...	0.00 dBm	60 us	3.008 GHz	unmodulated	-	0 s	0 s
5	List 1	1	0.012 000 000 ...	-6.00 dBm	15 us	3.009 GHz	unmodulated	-	0 s	0 s
6	List 1	1	0.015 001 500 ...	-6.00 dBm	15 us	3.009 GHz	unmodulated	-	0 s	0 s
7	List 1	1	0.018 000 000 ...	0.00 dBm	30 us	3.009 GHz	unmodulated	-	0 s	0 s
8	List 1	2	0.021 000 000 ...	-3.00 dBm	30 us	4 GHz	unmodulated	-	0 s	0 s
9	List 1	2	0.024 000 000 ...	0.00 dBm	60 us	4.002 GHz	unmodulated	-	0 s	0 s
10	List 1	2	0.027 000 000 ...	0.00 dBm	60 us	3.998 GHz	unmodulated	-	0 s	0 s
11	List 1	2	0.030 000 000 ...	-6.00 dBm	15 us	4 GHz	unmodulated	-	0 s	0 s
12	List 1	2	0.033 001 500 ...	-6.00 dBm	15 us	4 GHz	unmodulated	-	0 s	0 s
13	List 1	2	0.036 000 000 ...	0.00 dBm	30 us	4 GHz	unmodulated	-	0 s	0 s
14	EOF	0								

b) Select "Interleaving > Results > View > Analyze".

The "Analyze Dropped PDWs" dialog indicates the PDWs that are and are not included in the resulting output file. PDWs displayed in **red are dropped out**.



12. To increase the number of interleaved PDWs, add a delay to one of the PDW lists. Set, for example, "Interleaving > Config > PDW_1 > Time Offset = 11 us".

Create the output signal in the same manner. Observe the resulting output signal.

Depending on the PDW list, multiple or all PDWs from all PDW lists are included in the output file.

18.2 PDW import mechanism

R&S Pulse Sequencer Digital uses a template-based import mechanism for the PDW import. Import templates are human readable text files that describe how information is extracted from the PDW list file. The PDW list file is also a human readable text file using one single row per PDW. The columns contain the various parameters related to the PDW.

The diagram on Figure 18-3 shows the principal concept.

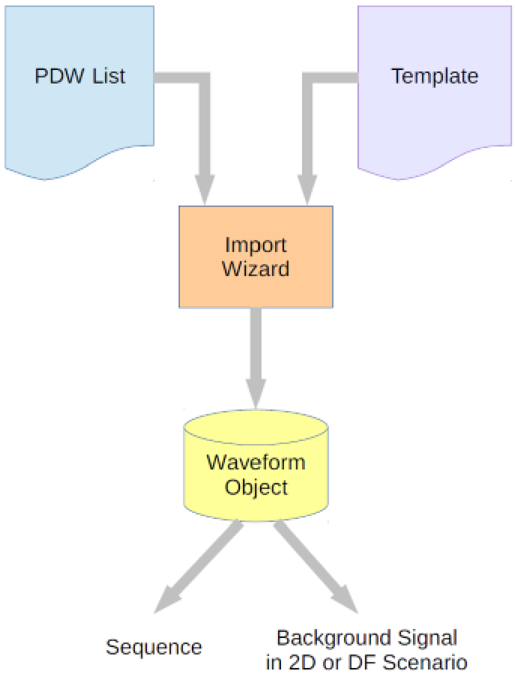


Figure 18-3: Import mechanism principle

The "PDW Data Import" wizard is a dialog that loads both files and shows their content. This dialog is also used to control the import process and accept the imported data.

The import process starts by parsing the text-based template file. Once parsing the template is complete, R&S Pulse Sequencer Digital loads and analyzes the PDW list file. All imported data is temporarily stored in memory until you choose to store the data permanently in the repository.

Storing the data lets R&S Pulse Sequencer Digital copy the extracted PDW data to a waveform object within the R&S Pulse Sequencer Digital repository. The internal storage format is a proprietary binary stream. If future extensions to the data stream are required, R&S Pulse Sequencer Digital automatically converts existing streams to the newer format.

After the import process has completed the original template and PDW files are no longer required.

For description of the related settings, see:

- Chapter 18.4, "PDW data import wizard settings", on page 365.

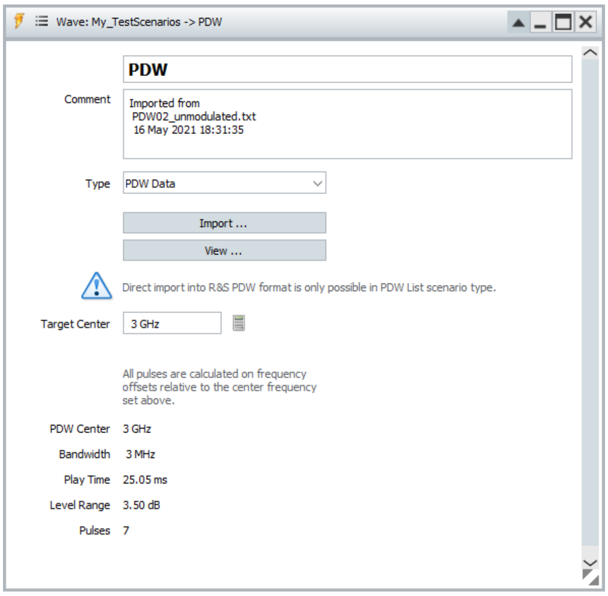
For step-by-step instruction, see:

- "To import PDW list data" on page 356.

18.3 PDW data settings

Access:

1. Select "Repository Tree > Imported Signals > New".
2. Select "Type = PDW Data".



This section describes the available settings. For step-by-step instructions, see "To import PDW list data" on page 356.

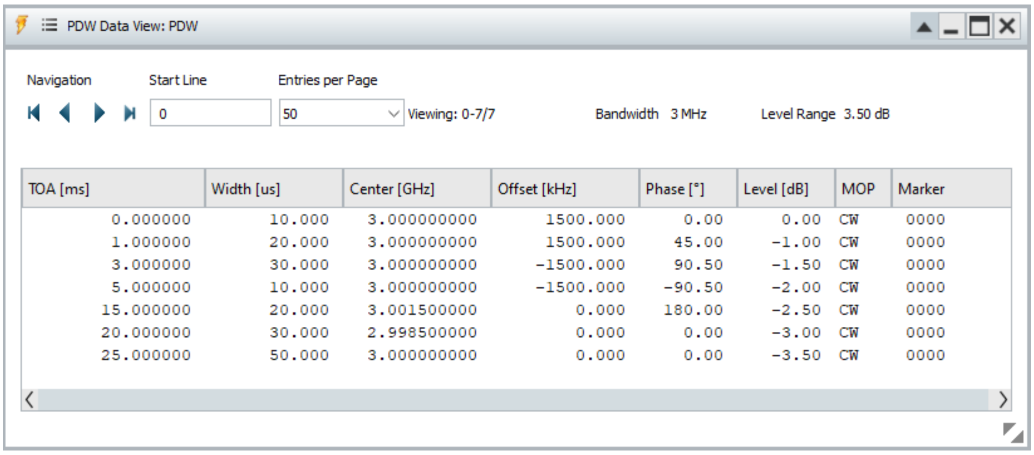
Settings:

Import

Accesses a file import wizard, see Chapter 18.4, "PDW data import wizard settings", on page 365.

View > PDW Data View

If the PDW list file is imported in the repository, it opens the "PDW Data View".



The dialog shows the waveform bandwidth, the level range and the waveform content as a sequence of pulses with their main characteristics.

The displayed information resembles the information in the "PDW Data Import > Imported Data" dialog.

"Start Line" Sets the first line to be displayed.

Remote command:
`IMPort:VIEW:TIME:STARt` on page 495

"Entries per Page"

Sets the number of lines displayed per page.

Remote command:
`IMPort:VIEW:COUNt` on page 495

"Start, End, Forwads, Backwords"

Scrolls through the entries in the selected way.

Remote command:
`IMPort:VIEW:MOVE:STARt` on page 495
`IMPort:VIEW:MOVE:FORWard` on page 495
`IMPort:VIEW:MOVE:BACKwards` on page 495
`IMPort:VIEW:MOVE:END` on page 495

Target Center

Sets the center frequency for the output waveform.

When a PDW list file is imported, the value is set to the automatically calculated center frequency of the PDW file. The value resembles the value indicated as "PDW Center".

The frequency offsets of the pulses are recalculated. The bandwidth increases.

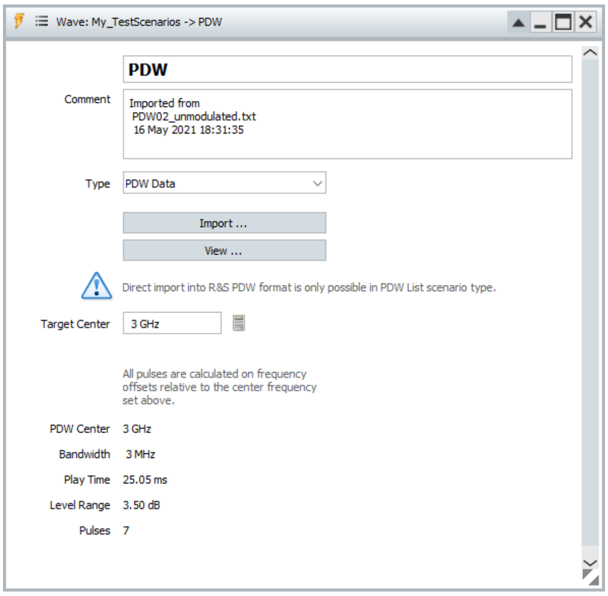
Calculate
The "Calculate" icon resets the "Target Center" field to the value displayed as "PDW Center".

PDW Center, Bandwidth, Play Time, Level Range, Pulses
If the PDW list file is imported in the repository, the dialog displays information on the waveform:
- "PDW Center": *Calculated center frequency*
  The indicated value is the center frequency the software uses during the import process. The pulses are calculated relatively to this value.
  The "PDW Center" is calculated as the middle frequency between the min and the max frequency values included in the PDW file.
  Chirp frequency deviations are considered.
- "Bandwidth": Calculated bandwidth
- "Play Time": Duration
- "Level Range": Calculated level range
- "Pulses": Number of pulses in the waveform.

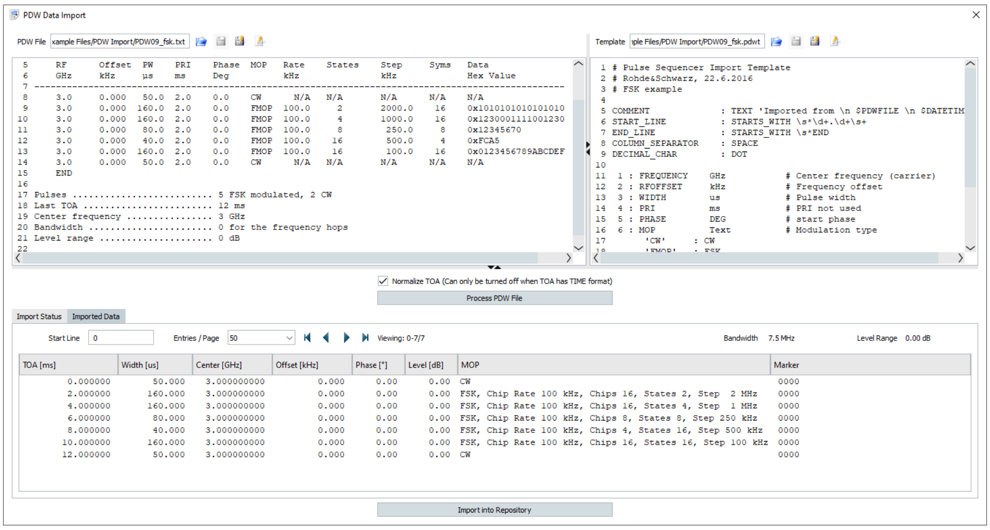
18.4 PDW data import wizard settings

Access:

1. Select "Repository Tree > Waveform > New".
2. Select "Type = PDW Data".



3. Select "Import".



If the required template file is provided, the "PDW Data Import" wizard converts custom PDW list files into a Rohde & Schwarz waveform file.
This section describes the available settings. For step-by-step instructions, see "To import PDW list data" on page 356.

Settings:

PDW File

File path and filename of the custom text-based PDW list file.

PDW list files are files in ASCII or coma-separated file format (`*.txt` or `*.csv`).

Remote command:
`IMPort:PDW:FILE:PDW` on page 491

Select PDW data file, Create new PDW data file, Save/Save as

Icons with standard file handling functions.

Note: The "Save/Save as" function stores changes in the PDW file or template itself. To parse the data, import it and store it into the repository, select "Process PDW File" and "Import into repository".

Remote command:
IMPort:PDW:FILE:PDW:LOAD on page 492
IMPort:PDW:FILE:PDW:SAVE on page 492

Template
File path and filename of the import template file.

Template files are ASCII files with predefined file format and file extension `*.pdwt`.

For more information, see document "R&S Pulse Sequencer Digital PDW Import Interface Specification".

Remote command:
IMPort:PDW:FILE:TEMPlate on page 491

Select import template file, Create new import template, Save/Save as
Icons with standard file handling functions.

Remote command:
IMPort:PDW:FILE:TEMPlate:LOAD on page 492
IMPort:PDW:FILE:TEMPlate:SAVE on page 492

PDW and template files content editor
Displays the content of the loaded file in the editor.

If necessary, edit the files and store the changes.

Normalize TOA
If selected, it normalizes the time of arrival (TOA) of the first entry to zero. The following timestamps are considered relative to the first timestamp.

Remote command:
IMPort:PDW:NORM on page 492

Process PDW File
Starts parsing the PDW file.

Observe the parsing process information in the "Import Status" dialog.

The imported data is temporarily stored in memory.

Remote command:
IMPort:PDW:EXECute on page 492

Import Status
Indicates the parsing and import status.

Remote command:
IMPort:PDW:STATus? on page 492

Imported Data
Shows summary information retrieved from the imported waveform, such as the waveform bandwidth, the level range and the waveform content as a sequence of pulses with their main characteristics.

Remote command:
`IMPort:PDW:DATA:SEL` on page 493
`IMPort:PDW:DATA:MOP?` on page 493
`IMPort:PDW:DATA:WIDTh?` on page 494
etc.

Import into Repository
Stores the data permanently in the repository.

As any other waveform element, the imported PDW information can be used as part of a sequence.

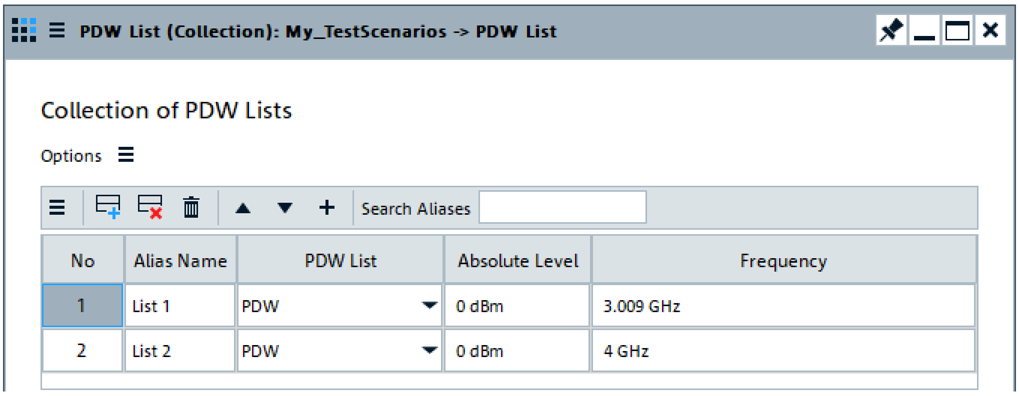
Remote command:
`IMPort:PDW:STORe` on page 494

18.5 PDW list (Collection) settings

Access:

► In a "Scenario Type = PDW List", select "PWD Lists > Edit".

Dialog displays overview information of the available PDW lists in table form.



Settings:

Options
Joint configuration of all PDW lists at once.

- "Reset Frequency"

- "Reset Abs. Level"

Append, Delete, Clear, Move Up/Down, Copy and append
Use the standard functions in the context menu to add, reorder, or remove items.

Remote command:
SCENario:CPDW:ADD on page 429
SCENario:CPDW:DELete on page 433
SCENario:CPDW:CLEar on page 434

Search Aliases
Allows fast search of the PDW list; useful in scenarios with many PDW lists.

Filters and displays PDW lists in table rows according to the entered search string.

No
Subsequent number.

Remote command:
SCENario:CPDW:SELect on page 431

Alias Name
Enters an alias name.

Remote command:
SCENario:CPDW:ALIas on page 559

PDW List
Selects the waveform element, used to import the PDW list.

Remote command:
SCENario:CPDW:NAME on page 559

Absolute Level
Sets the absolute level for the selected PDW list.

Thus, you can combine PDW lists with different levels into one output signal.

Remote command:
SCENario:CPDW:LVABs on page 556

Frequency
Sets the frequency for the selected PDW list.

Thus, you can combine PDW lists with different frequencies into one output signal.

Remote command:
SCENario:CPDW:FREQ on page 555

18.6 PDW list (Collection) interleaving settings

Access:

1. In a "Scenario Type = PDW List", select "PWD Lists > Interleaving > On".
2. Select "Interleaving > Config".

   Dialog displays the available PDW lists as configured in the PDW List (Collection) dialog, together with interleaving related settings, like "Time Offset" and "Priority".

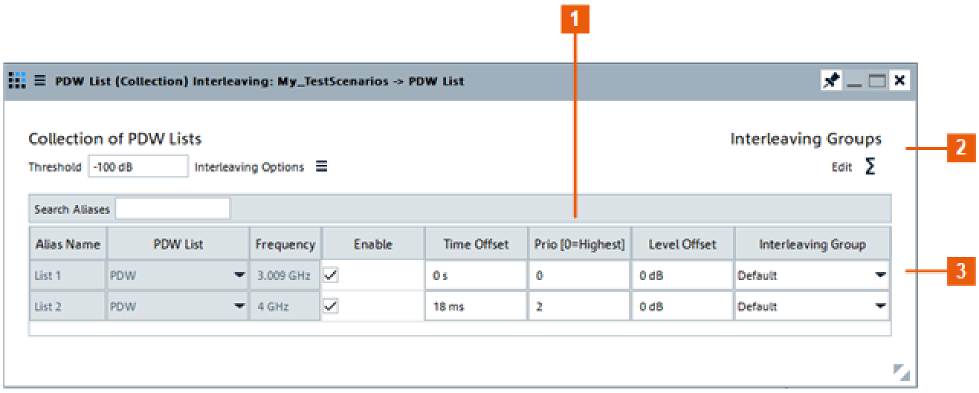


Figure 18-4: PDW List (Collection) Interleaving: understanding the displayed information

1 = Priority of the selected PDW lists; applies only if PDW lists are merged, see Figure 18-2.
2 = Opens the "Interleaving Groups" dialog, where you can define groups within that the interleaving is applied
3 = Each PDW list is assigned to an interleaving group

Settings:

Threshold

Enters a level threshold to limit the dynamic range of the signal. Pulses at levels below this threshold are omitted.

Resembles the value set with the parameter "Scenario > Signal Calculation > Config > Config > Signal Calculation Settings > Output > Threshold".

Remote command:
`SCENario:CPDW:THReshold` on page 563

Search Aliases
Allows fast search of the PDW list; useful in scenarios with many PDW lists.

Filters and displays PDW lists in table rows according to the entered search string.

Interleaving Options
Joint configuration of all PDW lists at once.

- "Enable All"/"Disable All"
- "Reset Time offset"/"Reset Level offset"
- "Reset Prio"

Alias Name
Displays the alias name, as set with the parameter "Scenario > PDW Lists > Edit" > Alias Name.

Remote command:
`SCENario:CPDW:ALIas` on page 559

PDW List
Displays the waveform element, used to import the PDW list.

Remote command:
`SCENario:CPDW:NAME` on page 559

Frequency
Indicates the frequency of the particular PDW list, as set with the parameter "Scenario > PDW Lists > Edit" > Frequency.

Enable
Includes the PDW list in the output file.

Remote command:
`SCENario:CPDW:ENABle` on page 559

Time Offset
Shifts the processing of the PDW list in time.

Use this function, for example, to decrease the number of dropped PDWs.

Remote command:
`SCENario:CPDW:LDELay` on page 560

Priority [0 = Highest]
Sets the priority of the selected PDW list. Higher value means lower priority. If PDWs are overlapping, the PDWs from the PDW list with lower priority are discarded.

See Figure 18-2.

Remote command:
`SCENario:CPDW:PRIority` on page 560

Level Offset

Adds a level offset.

The value is set relative to the value set with the parameter "Scenario > PDW Lists > Edit" > Absolute Level.

Remote command:
`SCENario:CPDW:LEVel` on page 560

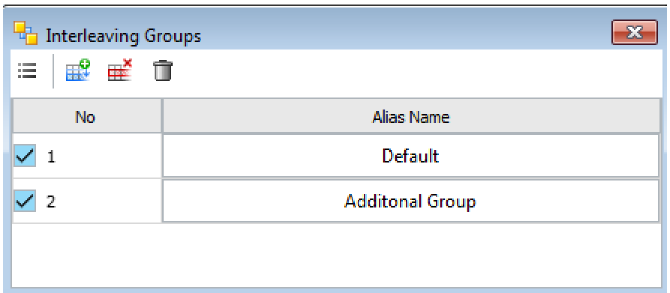
Group

Assigns the emitter to one of the available interleaving groups, defined in the Edit Interleaving Groups dialog.

Remote command:
`SCENario:CPDW:GROup` on page 561

Edit Interleaving Groups

Select the icon to access the "Edit Interleaving Groups" dialog.



Interleaving groups allow you to combine the emitter signals into different output signals.

Use the standard functions (icons) to add a new group, rearrange the groups, delete the existing one or all groups.

Select, Insert, Append/Prepend, Remove, Clear ← Edit Interleaving Groups

Standard functions for items handling, see "Standard function in the context menus" on page 33.

Select a row, for example, to delete it or to insert a row before it.

Remote command:
`SCENario:CPDW:GROup:CATalog?` on page 561
`SCENario:CPDW:GROup:SELect` on page 430
`SCENario:CPDW:GROup:ADD` on page 429
`SCENario:CPDW:GROup:DELete` on page 432
`SCENario:CPDW:GROup:CLEar` on page 433

No. ← Edit Interleaving Groups

Select a row to perform any row-based actions.

Interleaving groups are indicated by consecutive number; also used for indication in remote control.

Remote command:
`SCENario:CPDW:GROup:COUNt?` on page 430
`SCENario:CPDW:GROup:SELect` on page 430

Alias Name ← Edit Interleaving Groups
Enter an alias name for the interleaving group.

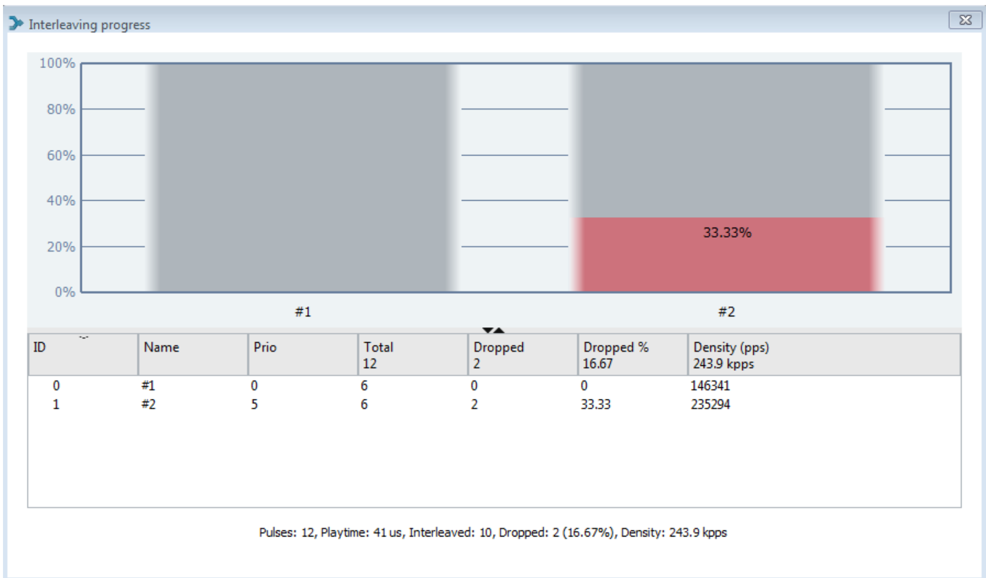
Remote command:
`SCENario:CPDW:GROup:ALIas` on page 561

18.7 PDW interleaving progress statistics

Access:

1. In a "Scenario Type = PDW List", select "Interleaving > On".
2. Select "Signal Control > Prepare".
3. Select "Signal Control > Interleave".

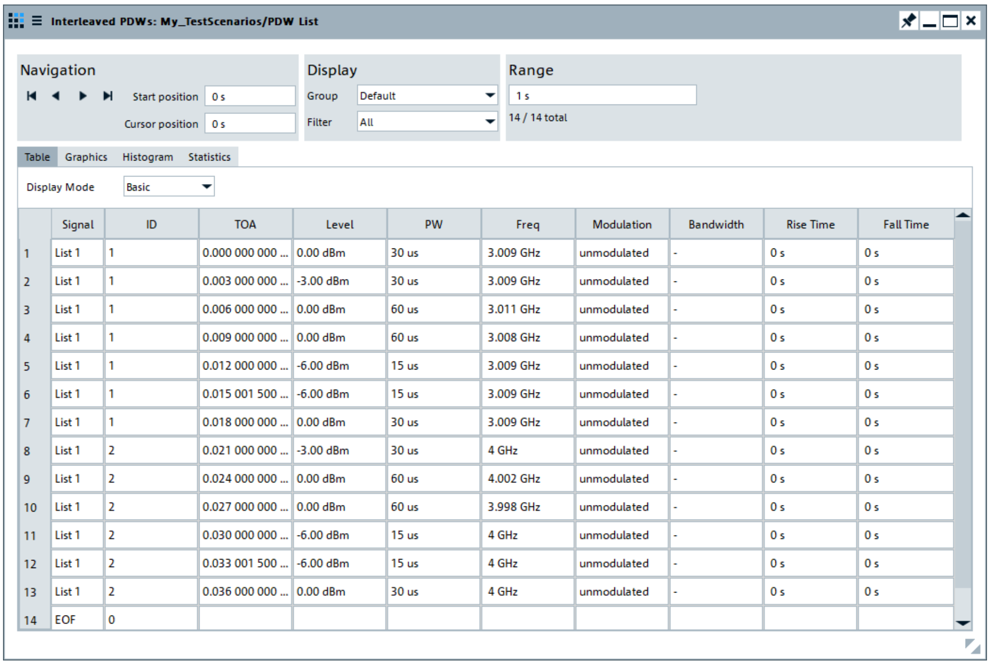
   The "Interleaving Progress" display opens automatically during the signal calculation, if PDW lists are merged. It indicates the percentage of pulses being processed and dropped in each of the PDW lists.



18.8 Signal preview with interleaved and dropping statistics

Access:

1. In a "Scenario Type = PDW List", select "Interleaving > On".
2. Select "Signal Control > Prepare".
3. Select "Signal Control > Interleave".
4. Select "Interleave > Results > Interleaved PDW/Dropped PDWs".



	Signal	ID	TOA	Level	PW	Freq	Modulation	Bandwidth	Rise Time	Fall Time
1	List 1	1	0.000 000 000 ...	0.00 dBm	30 us	3.009 GHz	unmodulated	-	0 s	0 s
2	List 1	1	0.003 000 000 ...	-3.00 dBm	30 us	3.009 GHz	unmodulated	-	0 s	0 s
3	List 1	1	0.006 000 000 ...	0.00 dBm	60 us	3.011 GHz	unmodulated	-	0 s	0 s
4	List 1	1	0.009 000 000 ...	0.00 dBm	60 us	3.008 GHz	unmodulated	-	0 s	0 s
5	List 1	1	0.012 000 000 ...	-6.00 dBm	15 us	3.009 GHz	unmodulated	-	0 s	0 s
6	List 1	1	0.015 001 500 ...	-6.00 dBm	15 us	3.009 GHz	unmodulated	-	0 s	0 s
7	List 1	1	0.018 000 000 ...	0.00 dBm	30 us	3.009 GHz	unmodulated	-	0 s	0 s
8	List 1	2	0.021 000 000 ...	-3.00 dBm	30 us	4 GHz	unmodulated	-	0 s	0 s
9	List 1	2	0.024 000 000 ...	0.00 dBm	60 us	4.002 GHz	unmodulated	-	0 s	0 s
10	List 1	2	0.027 000 000 ...	0.00 dBm	60 us	3.998 GHz	unmodulated	-	0 s	0 s
11	List 1	2	0.030 000 000 ...	-6.00 dBm	15 us	4 GHz	unmodulated	-	0 s	0 s
12	List 1	2	0.033 001 500 ...	-6.00 dBm	15 us	4 GHz	unmodulated	-	0 s	0 s
13	List 1	2	0.036 000 000 ...	0.00 dBm	30 us	4 GHz	unmodulated	-	0 s	0 s
14	EOF	0								

The dialog is similar to the "Signal Preview" dialog.

This section describes the settings dedicated to the "Interleaved/Dropped PDWs" dialogs. For description of all other settings, see Chapter 15.1, "Signal preview settings", on page 284.

Settings:

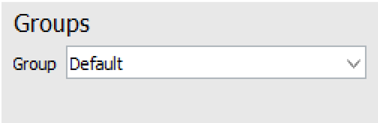
Display > Filter

If interleaving is used, the calculated output signal contains pulses originating for multiple PDW lists or emitters. By default, all PDWs are displayed. Use the "Filter" parameter to narrow down the displayed information to the PDWs of one of the PDW lists.

The filter does not change the zoom level or the scaling on both axis.

Display > Groups

If interleaving groups are used, selects the group for which pulses are displayed.



Table

Provides summary information on the number of processes PDWs and the content of the output file.

Graphics

Visualize the variation of the parameters over time.

For details, see Chapter 15.1, "Signal preview settings", on page 284.

Graphics > Highlight

Highlights pulses belonging to the selected PDW list.

Histogram

Resembles the information displayed during the signal processing, see Chapter 18.7, "PDW interleaving progress statistics", on page 373.

Use the filtering option to narrow down to the displayed information according to one of the predefined criteria.

Statistics

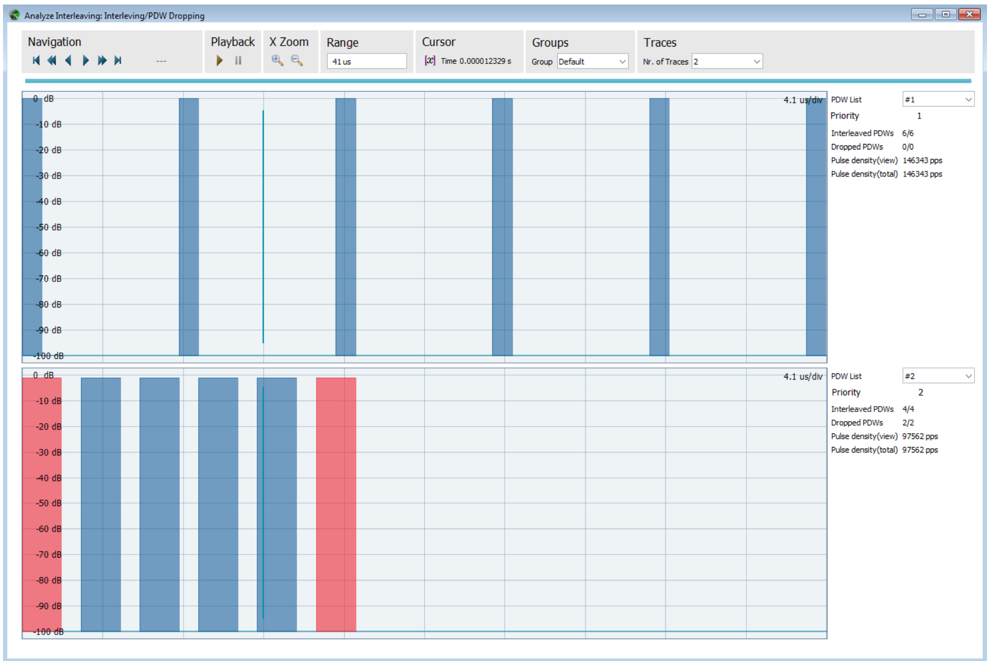
The drop out statistic is the table form representation of the same information as the Histogram.

18.9 Analyze interleaving

Access:

1. In a "Scenario Type = PDW List", select "Interleaving > On".
2. Select "Signal Control > Prepare".
3. Select "Signal Control > Interleave".
4. Select "Interleaving > Results > View > Analyze".

   The "Analyze Interleaving" dialog indicates the PDWs that are and are not included in the resulting output file. PDWs displayed in **red color are dropped out**.



The provided settings are a subset of the settings available in the "Waveform View/ Signal Preview" displays and they have the same effect.
For description, see Chapter 15.1, "Signal preview settings", on page 284.
The following settings are dedicated to this dialog.

Display > Groups
If interleaving groups are used, selects the group whose pulses are displayed.

Traces
Sets the number of traces to be displayed.

If the selected interleaving group comprises many PDW lists, this function can be used to narrow down the number of simultaneously displayed traces and hence increase the zoom level on the y-axis.

Information per trace
The following information is displayed:
- Trace name
- Priority
- Interleaved PDWs
- Dropped PDWs
- Pulse density (view)
- Pulse density (total)

19 Defining complex modulation schemes and IPM profiles

R&S Pulse Sequencer Digital enables you to define custom antenna patterns and IPM profiles using an external plugin.

Plugins are Microsoft Windows DDL modules that contain the maths that is required for the envelope shaping and the modulation on pulse.

Some example plugins are provided with the software as binary and source code. These examples can serve as a starting point for your own applications. Once imported, the software handles the plugins automatically.

Each plugin:

- Has to provide a range of functions to identify itself
- Can register a set of configuration parameters that can be used as variables inside the plugin.

You can load plugins to:

- Define a custom inter-pulse modulation profile (IPM)
- Create a report file during the signal calculation process
- Define the file format of the custom antenna pattern files.

How to import and assign user defined plugins

See:

- Chapter C, "Plug-in programming API", on page 650 for a description of the plugin programming API and instructions on how to load the plugin examples.
- "To import a plugin" on page 377
- "To assign the user defined IPM profile" on page 378

See also:

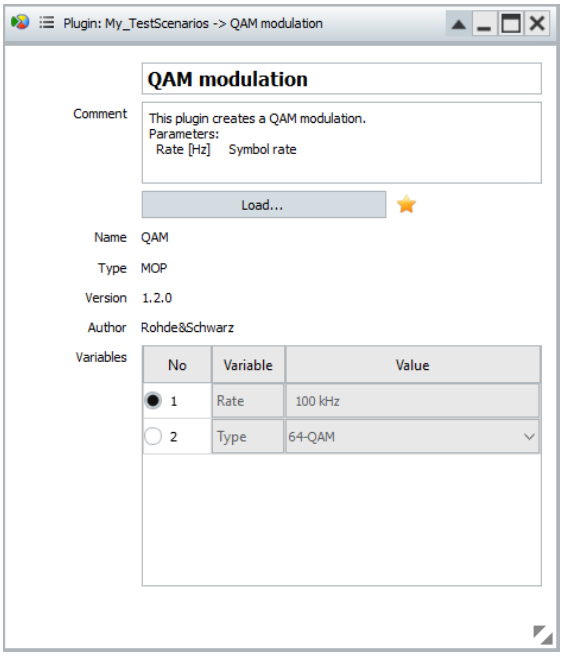
- "To import an antenna pattern file in custom file format" on page 215
- Chapter 21, "Creating reports and documenting measurement results", on page 386

To import a plugin

1. In the repository tree, select "Plugins > New".
2. In the "Plugin" dialog, enter a name and description.
3. Select "Load", navigate to the `*.dll` file, and select it.

   Only files containing modulation data are listed and can be loaded.

   The software loads the `*.dll` file, retrieves information from it and displays it in the "Plugin" dialog.
   The "Variables" table lists the plugin variables. You can edit them when you load the plugin into a repository element.



You can use the plugins to create:

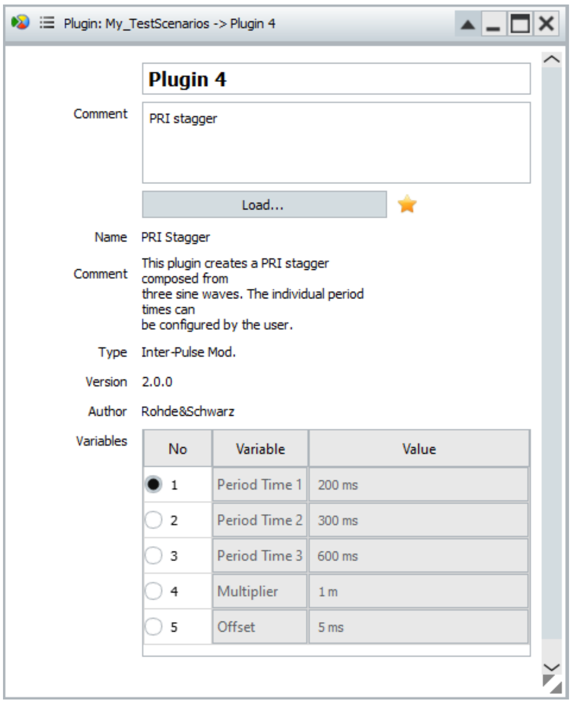
- Create custom IPM profile, see "To assign the user defined IPM profile" on page 378
- Import custom antenna pattern file formats, see "To import an antenna pattern file in custom file format" on page 215.

Remote Commands:

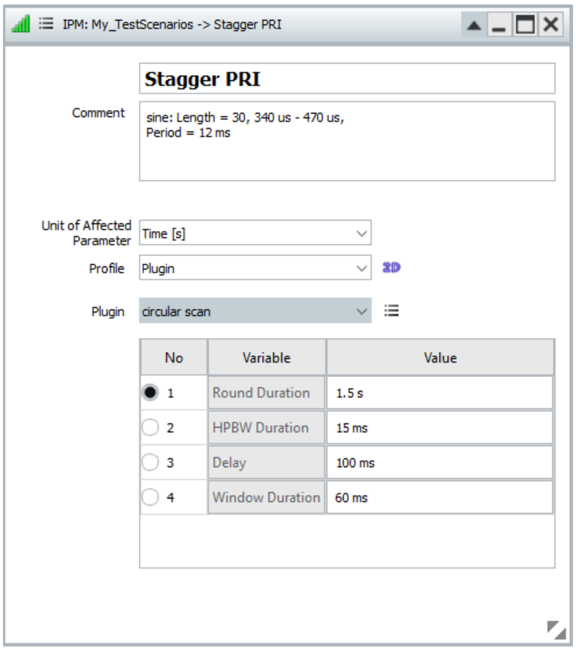
- PLUGin:CREate on page 426
- PLUGin:CATalog? on page 426
- PLUGin:NAME on page 427
- PLUGin:COMMent on page 428
- PLUGin:LOAD on page 502
- PLUGin:SELect on page 427
- PLUGin:REMove on page 428
- PLUGin:MODule:NAME? on page 427
- PLUGin:MODule:TYPE? on page 502
- PLUGin:MODule:VERSion? on page 502
- PLUGin:MODule:AUTHor? on page 502
- PLUGin:MODule:COMMent? on page 502
- PLUGin:MODule:DATA? on page 503

To assign the user defined IPM profile

1. Import a suitable plugin.



2. In the repository tree, select "Inter-Pulse Mods > IPM-Name".
3. Select "Profile > Plugin".
4. Select "Plugin" and select a loaded user-defined IPM profile, e.g. `Stagger PRI`.



5. Select "2D" to visualize the IPM profile.
6. If required, change the values of the used variables and observe the effect on the "2D" diagram.

20 Defining and enabling marker signals

Markers signals or markers are binary signals that are additionally generated and added to the generated signal.

For details, see the user manual of the corresponding base unit.

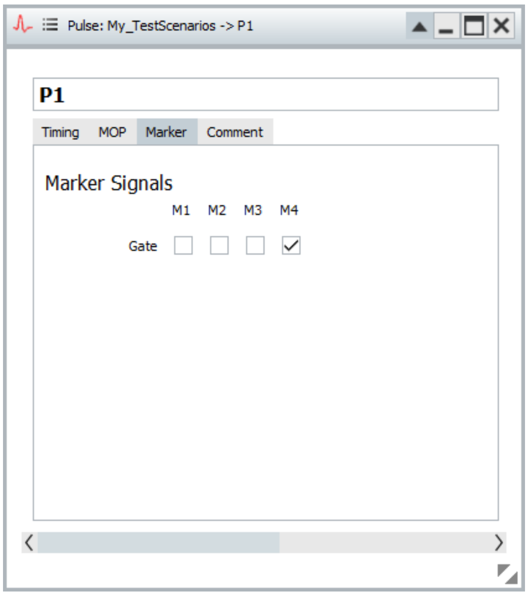
20.1 Marker settings

In the R&S Pulse Sequencer Digital, you define markers on four levels: on a pulse basis, for emitters, in the sequence or define global conditions like the start of a scenario.

Access:

1. Select "Repository Tree > Pulse > Marker".
   See "Pulse Markers" on page 380.
2. Select "Repository Tree > Sequence > Sequence Description > Marker".
   See "Sequence Markers" on page 381.
3. Select "Map > TX Items > Properties > TX Items Properties".
   See "Emitter Marker" on page 315
4. Select "Repository Tree > Scenario > Signal Calculation > Config > Marker Signals > Enable sequence and waveform markers".
   See "Global Marker Signals" on page 381.

Pulse Markers
You can assign up to 4 gate markers to the pulse.

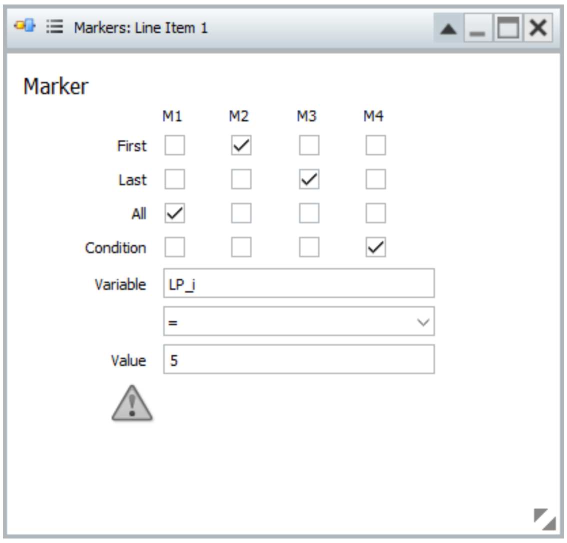


Marker information is directly added to the resulting waveform and the marker signal output is therefore synchronous with the waveform playback.

See also "Emitter Marker" on page 315

Remote command:
`PULSe:MARKer:GATE` on page 498

Sequence Markers



"First, Last, All" If pulse markers are defined, you can also define markers on multiple repeating pulses and mark the first, last or all pulses.

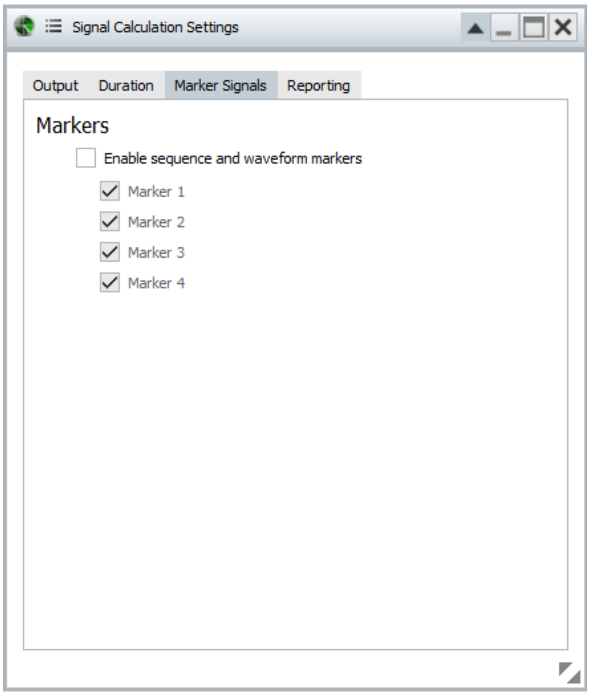
"Condition, Variable, Value"
A marker signal is generated if a defined condition is fulfilled. Conditions are defined as logical expressions, where the variable value ("Variable") is compared to a fixed value ("Value"). The variable must exist.

Example:

If a sequence item is included into a loop, you can create marker signals that mark for example one specific loop run number.

See:
- Chapter 7.2.3, "Loop settings", on page 127
- "To enable sequence markers" on page 383

Global Marker Signals
If the selected sequence in a scenario contains markers, you can define that these markers be considered in the output from the signal calculation.



Remote command:
SCENario:OUTPut:MARKer:ENABle on page 499
SCENario:OUTPut:MARKer:FLAGs on page 499

20.2 How to configure and visualize markers

To enable pulse markers

1. Select "Repository Tree > Pulse > Marker".
2. Enable up to four "Gate" markers for the pulse.

To enable sequence markers

1. Select "Repository Tree > Sequence".

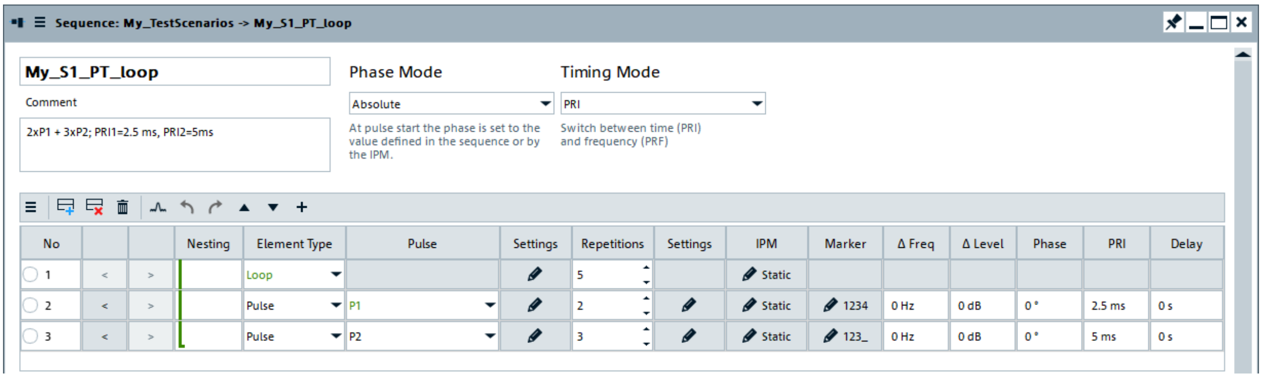
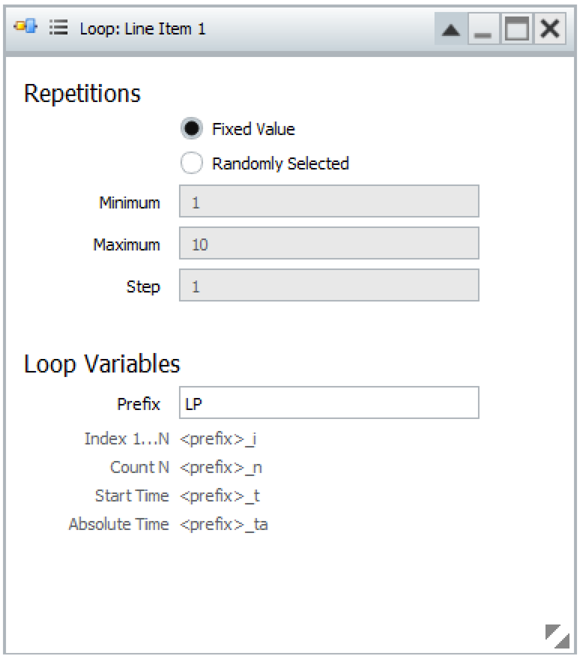
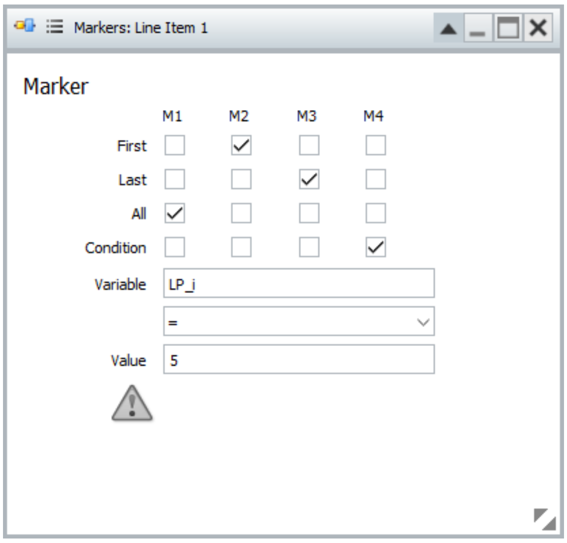


Figure 20-1: Example of a sequence composed of two pulses included into a loop

2. Select "Loop > ..." and enter "Loop Variables > Prefix = LP".



3. Select "Sequence Description table > Item#2 > Marker".
4. Select "M1 > All", "M2 > First", "M3 > Last".



The Marker 2 is generated only for the first pulse out of all repetitions.

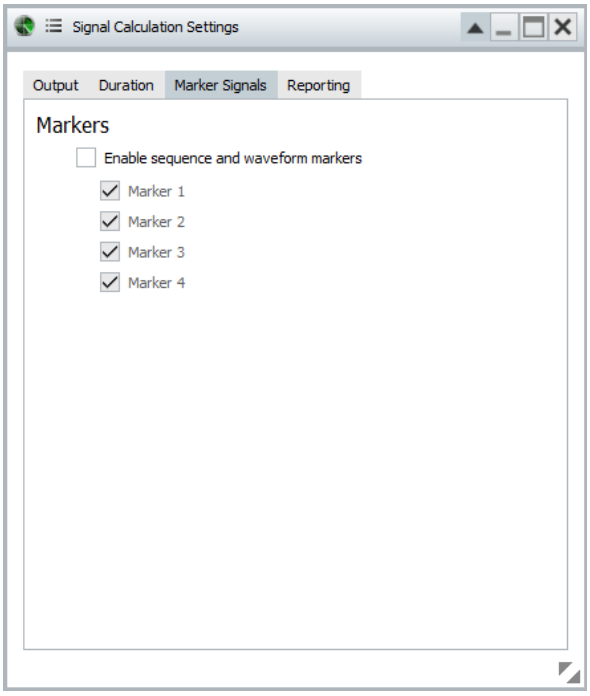
5. Select "M4 > Condition", "Variable = LP_i" and "Value = 5".

   If the condition LP_i = 5 is fulfilled, a marker is generated. That is, marker 4 is generated during the fifth loop run.

To enable marker signals in the output signal

1. Select "Repository Tree > Scenario > Signal Calculation > Config > Config > Signal Calculation Settings".

2. Select "Marker Signals > Enable sequence and waveform markers".



21 Creating reports and documenting measurement results

In R&S Pulse Sequencer Digital, you can document your measurement results and software configuration in the following ways:

- Create and export a repository archive, so that you can repeat measurements under the same conditions
- Create automatically scaled hardcopy of current screens or dialogs
- Create and store reports in text files, in `HTML` format or in Excel files with several spreadsheets.

This section focuses on the built-in report generation function. For information on the other functions, see:

- "To create and export a repository archive" on page 71
- "To set the size and color scheme of the hardcopies" on page 47

Report generation

The R&S Pulse Sequencer Digital provides a built-in report generation function so that all parameters used during the signal generation are automatically saved to a printable file. You can configure the target directory in which the generated reports are stored.

Report files can be formatted based on:

- **Template**
  The reporting uses a predefined but editable template. Generated is an ASCII text file or an `HTML` page, where data is formatted in columns, including header description. The reporting files in ASCII format use the predefined file extension `*.pwd`
  – ASCII text file

```
This is a template-based report.

User:        R&S
Rep Path:    C:\_PS_files\reports

Date:        28.10.2015 10:07:25
ISO Date:    2015-10-28T10:07:25

Repository: My_TestScenarios
 Comment:
 Version:    2.3
 Author:     Rohde&Schwarz
 Created:    25 Jun 2014 15:27:04
 Path:       C:\Users\Public\Documents\Rohde-Schwarz\Pulse Sequencer\Repositories\20141124_100351

Scenario:    SimplePulseTrain
Comment:

         TOA |          RF |           PW |       PA | MF |  MOP |       BW | Rep |
          ns |         GHz |           us |      dBm |    |      |      kHz |     |
---------------------------------------------------------------------------------
           0 |         3.0 |      120.000 |    -30.0 |  0 | NONE |        0 |   1 |
     2500000 |         3.0 |      120.000 |    -30.0 |  0 | NONE |        0 |   2 |
     5000000 |         3.0 |      220.000 |    -30.0 |  0 | NONE |        0 |   1 |
    10000000 |         3.0 |      220.000 |    -30.0 |  0 | NONE |        0 |   2 |
    15000000 |         3.0 |      220.000 |    -30.0 |  0 | NONE |        0 |   3 |
    22000000 |         3.0 |      120.000 |    -40.0 |  0 | NONE |        0 |   1 |
    24500000 |         3.0 |      120.000 |    -40.0 |  0 | NONE |        0 |   2 |
```

Figure 21-1: Example of an ASCII report file ("Type > Template")

  – HTML page

PDW Report Generated with PS Example

User	R&S
Repository Path	C:_PS_files\reports
Date	28.10.2015 08:15:18
ISO Date	2015-10-28T08:15:18

Repository	My_TestScenarios
Comment	
Version	2.3
Author	Rohde&Schwarz
Created	25 Jun 2014 15:27:04
Path	C:\Users\Public\Documents\Rohde-Schwarz\Pulse Sequencer\Repositories\20141124_100351

SimplePulseTrain

TOA	RF	PW	PA	MF	MOP	BW	Rep
ns	GHz	us	dBm			kHz	
0	3.0	120.000	-30.0	0	NONE	0	1
2500000	3.0	120.000	-30.0	0	NONE	0	2
5000000	3.0	220.000	-30.0	0	NONE	0	1
10000000	3.0	220.000	-30.0	0	NONE	0	2
15000000	3.0	220.000	-30.0	0	NONE	0	3
22000000	3.0	120.000	-40.0	0	NONE	0	1
24500000	3.0	120.000	-40.0	0	NONE	0	2

Figure 21-2: Example of an HTML report file ("Type > Template")

- **User-defined plugin**
  This reporting creates reports according to custom templates, e.g. DFS.
  You can, for example, create a Microsoft Excel plugin that retrieves the pulse parameters and enters them in a spreadsheet.

For a detailed description of the file formats, the template and the plugin, see:

- Chapter C, "Plug-in programming API", on page 650
- Chapter 19, "Defining complex modulation schemes and IPM profiles", on page 377.

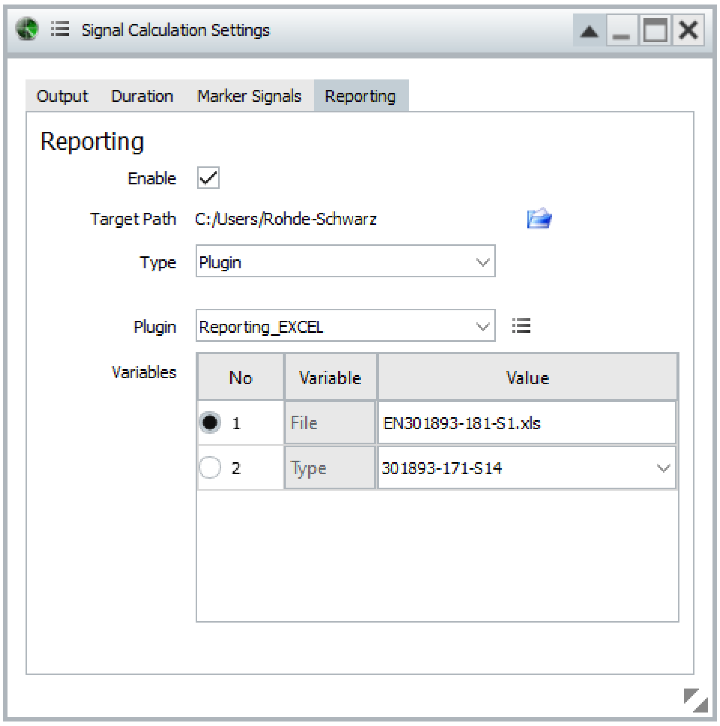
See also Chapter 21.2, "How to create test reports", on page 392.

21.1 Reporting settings

Access:

1. In the "Scenario" dialog, select "Signal Calculation > Config > Config".

2. Select "Reporting".



Available settings:

Reporting Enable

Enables generation of reports.

Remote command:
SCENario:PDW:ENABle on page 504

Target Path, Set Path

Indicates the current storage location.

Per default, report logs are stored in the user directory of the current user, that is the %HOMEPATH% directory.

To change the storage location, select the "Set the target path" icon.

See also "To set the storage location for the generated report files" on page 392.

Remote command:
`SCENario:PDW:PATH` on page 504

Type

Sets the template used be the reporting function.

"Default" Loads the predefined template.
The report is a plain text file. The format is simple, column-based and cannot be modified. Further settings are not required.

"Template" Enables the predefined template for editing (see "Edit Template" on page 389).

"Plugin" Generates report in custom format by using a plugin.
Custom templates are used in the same way as the predefined template (see "Plugin and plugin variables" on page 391).
For details on the plugin API, see Chapter C, "Plug-in programming API", on page 650.

"AMMOS PDW"
Writes PDW reports in the Rohde & Schwarz proprietary PDW format. This format is used by Rohde & Schwarz monitoring products.

Remote command:
`SCENario:PDW:TYPE` on page 505

Edit Template

Opens the predefined template in edit mode.

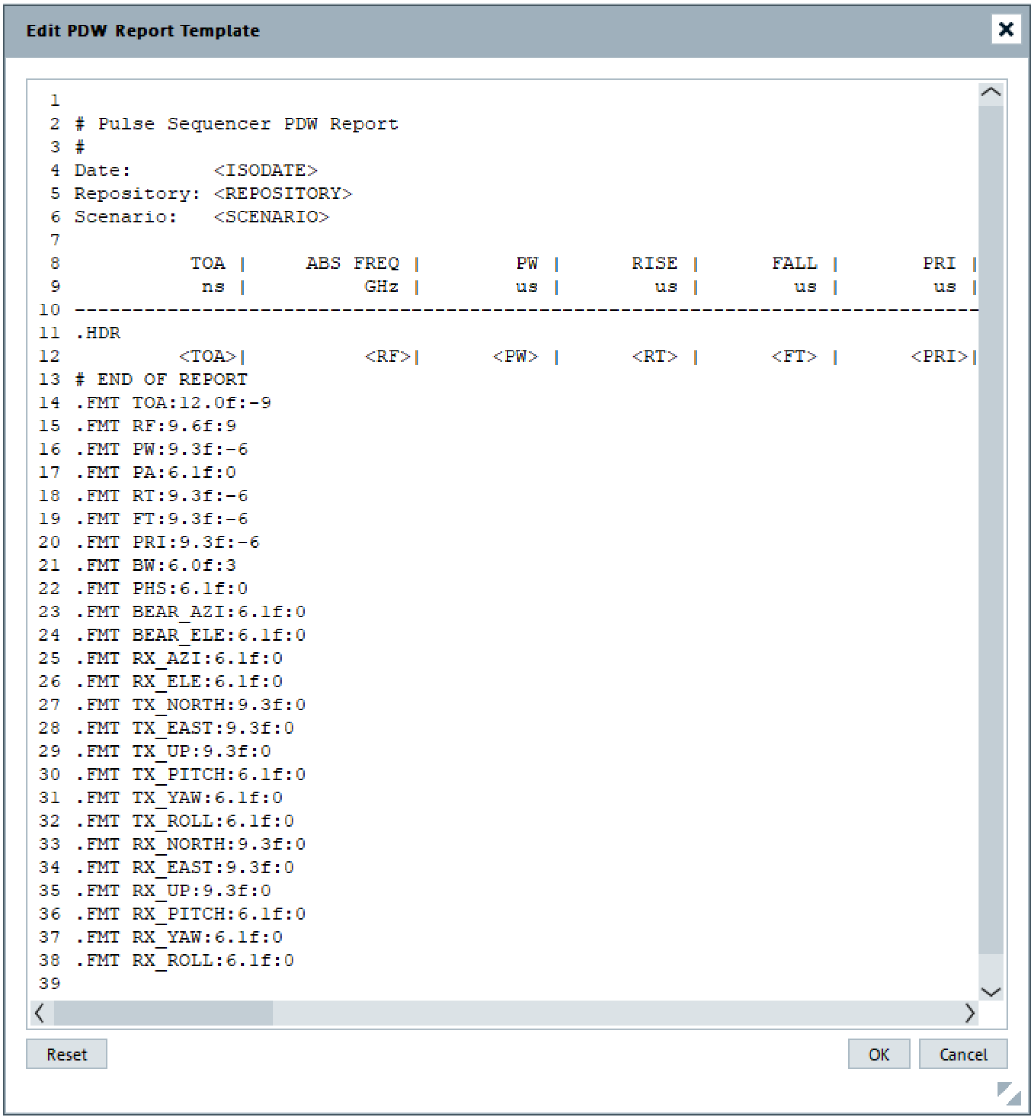


Figure 21-3: Example: Template that creates text reports in ASCII format

Note: The software uses the last modified template version. Previous versions cannot be recalled.
Before you modify the template, store the template content as a text file.

Remote command:
`SCENario:PDW:TEMPlate` on page 506

Specific settings for Type > AMMOS PDW

The following settings are dedicated to "Type > AMMOS PDW":

Format ← Specific settings for Type > AMMOS PDW

Selects format of the AMMOS file.

"PDW"	Standard AMMOS PDWs, including header information.
"PPDW"	Shortened AMMOS PDWs, without header information.

Remote command:
SCENario:PDW:AMMos:PPDW on page 505

Frame Length ← Specific settings for Type > AMMOS PDW
Sets the frame length.

A frame can contain several PDWs but it can also be empty.

Remote command:
SCENario:PDW:AMMos:FRAMe on page 505

Start Time, User Set Value ← Specific settings for Type > AMMOS PDW
Defines how the report start time is set.

- "User Defined Value > Off"
  The reporting start time is time at that the scenario calculation starts.
- "User Defined Value > On"
  The reporting starts at user-defined moment. This setting is useful, for example, if particular event or moment of time is to be captured.

Remote command:
SCENario:PDW:AMMos:UTIMe:ENABle on page 506
SCENario:PDW:AMMos:UTIMe:ISO on page 506

Azimuth ← Specific settings for Type > AMMOS PDW
Defines whether the angle of the RX antenna or the bearing (i.e. the azimuth direction of the emitter) is reported.

The former is suitable if spinning dish antennas are used. Use the bearing for direction finding.

Remote command:
SCENario:PDW:AMMos:AZIMuth on page 506

Plugin and plugin variables
Selects and loads a reporting template form a `*.dll` file. This template must exist in the "plugin" library.

Report plugins can register a set of variables that can be used to control the plugin's algorithms.

See:
- Chapter C, "Plug-in programming API", on page 650
- "To import a plugin" on page 377.

Remote command:
SCENario:PDW:PLUGin:NAME on page 505
SCENario:PDW:PLUGin:VARiable:CATalog? on page 503
SCENario:PDW:PLUGin:VARiable:SELect on page 503
SCENario:PDW:PLUGin:VARiable:VALue on page 504

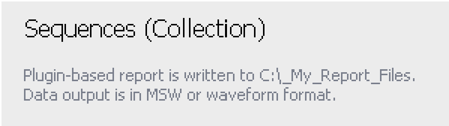
21.2 How to create test reports

This section shows how to:

- "To enable report file generation" on page 392
- "To set the storage location for the generated report files" on page 392
- "To change the report type" on page 393

To enable report file generation

The "Scenario" dialog provides information on the report logging status. The related information is displayed to the right of the "Comment" field.



To enable or disable the report file generation, proceed as following:

1. In the "Scenario" dialog, select "Waveform Generation > Config".
2. In the "Signal Calculation Settings" dialog, select "Reporting > Reporting Enable > On".

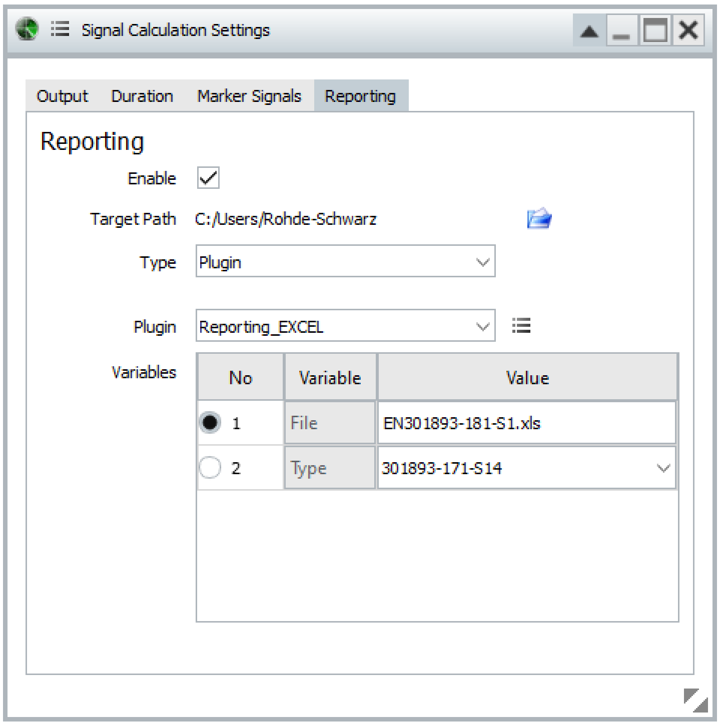
To set the storage location for the generated report files

Per default, report files are stored in the user home directory of the current user, that is the `%HOMEPATH%` directory.

1. Open the "Scenario" dialog to retrieve information on the current storage location.

   Information is displayed to the right of the "Comment" field.
2. Select "Signal Calculation > Config > Config".
3. In the "Signal Calculation Settings" dialog, select "Reporting".

   The parameter "Target Path" indicates the current storage location.



4. To change this location, select the "Set target path" icon, navigate to the directory, and confirm with "Select Folder".

To change the report type

You can also store the test reports in text form or as HTML pages (e.g. if Microsoft Excel is not installed on your PC).

To change the type of the generated reports, proceed as following:

1. In the "Scenario" dialog, select "Signal Calculation > Config > Config > Report".
2. Select "Type > Template".
3. Select "Edit" to display the current template.

Edit PDW Report Template

```
# Pulse Sequencer PDW Report
#
Date:        <ISODATE>
Repository:  <REPOSITORY>
Scenario:    <SCENARIO>

          TOA |          RF |          PW |       PA | MF |    MOP |      BW |
           ns |         GHz |          us |      dBm |    |        |     kHz |
---------------------------------------------------------------------
.HDR
 <TOA> | <RF> | <PW> | <PA> |  <MF> | <MOP> | <BW> |
# END OF REPORT
.OPT <TOA:12.0f:-9>
.OPT <RF:9.6f:9>
.OPT <PW:9.3f:-6>
.OPT <PA:6.1f:0>
.OPT <BW:6.0f:3>

```

Reset OK Cancel

You can also change the used templates or create and load your plugin.

See:
- Chapter C, "Plug-in programming API", on page 650
- Chapter 19, "Defining complex modulation schemes and IPM profiles", on page 377.

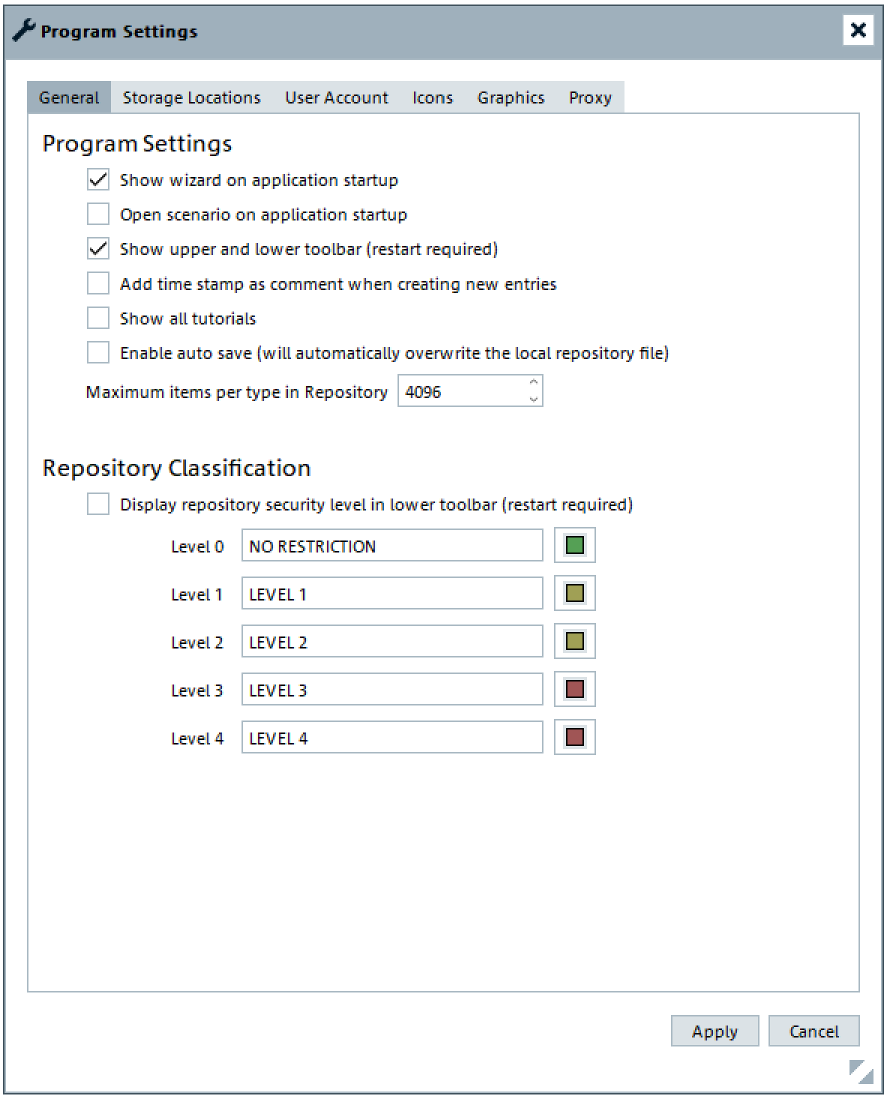
22 Performing administration tasks

This section informs you about file handling, user management and settings you can use to speed up calculation.

22.1 Setting general program settings

Access:

► In the menu bar, select "Configure > Settings".



Program Settings
Enable each of the optional configurations according to your needs.

Set the value for the maximum items per type in the repository. The limit for items per type is 4096.

Remote command:
`PROGram:STARtup:WIZard:ENABle` on page 436
`PROGram:STARtup:LOAD:ENABle` on page 436
`PROGram:TOOLbar:ENABle` on page 436
`PROGram:COMMent:ENABle` on page 434
`PROGram:TUTorials:SHOW:ENABle` on page 436

Repository Classification
Define whether the classification level is displayed in the repository and sets the used colors, see "To change the general program settings" on page 46.

See also Chapter 2.9, "Customizing the software", on page 45.

Remote command:
`PROGram:CLASs:ENABle` on page 434

Apply, Cancel
Accept or reject the changed settings.

Remote command:
`PROGram:SETTings:ACCept` on page 435
`PROGram:SETTings:REJect` on page 435

22.2 User administration

The R&S Pulse Sequencer Digital uses its own **user rights management** that is independent from the user administration of the operating system but follows the same principles: the current user cannot delete its own account; users with lower access rights cannot delete accounts with higher rights.

User roles

Repositories support simultaneous access from several users. Per default, all users are granted with the same (administrator) rights. The user rights can be restricted by assigning of a different user role, other than the creator role. At least one user with administrator rights must always exist.

To protect a repository from accidental changes, the repository data can be password protected. Passwords are then required to access a repository. If the password is lost, access to the repository is no longer possible.

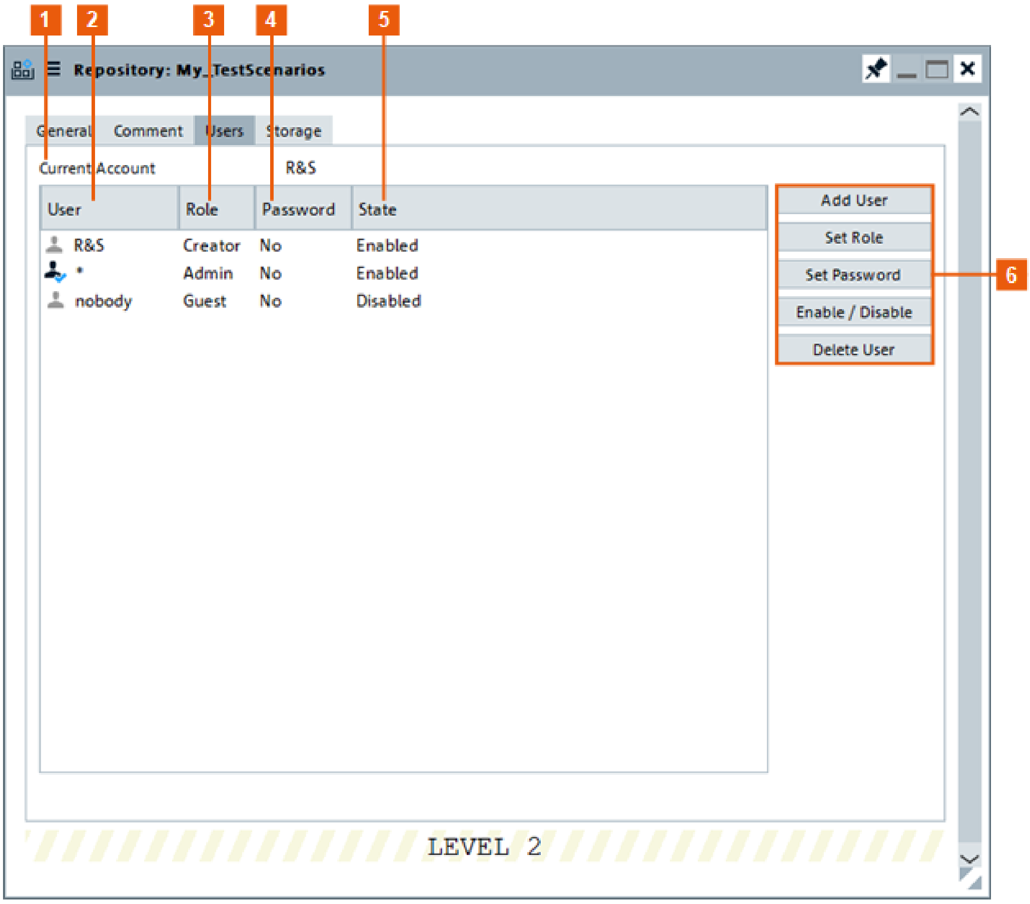


Figure 22-1: Repository > Users: understanding the displayed information

1 = Indicates the user name of the logged in user
2 = Login name or custom user name
3 = Assigned user role; the creator role is assigned automatically and cannot be changed (see also Table 22-1)
4 = Identifies if the repository is password protected or not
5 = Current user state
6 = Standard functions for user management

Table 22-1: Overview of the available user roles and their access rights

User role	Read access	Write access	Create a scenario from the existing elements	Delete elements	Copy repository elements
Creator	x	x	x	x	x
Admin	x	x	x	x	x
User	x	-	x	-	x
Guest	x	-	-	-	x

Per default, the R&S Pulse Sequencer Digital uses your Windows login name to access the repositories but you can also use a custom name, see "User Account" on page 401.

See also Chapter 22.2.1, "How to set and change user passwords", on page 398.

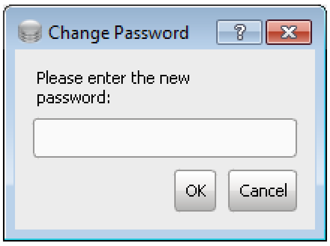
22.2.1 How to set and change user passwords

See:

- "To protect a repository from accidental changes" on page 398
- "To remove the password protection on a repository" on page 399
- "To change the default user used to access repositories" on page 399

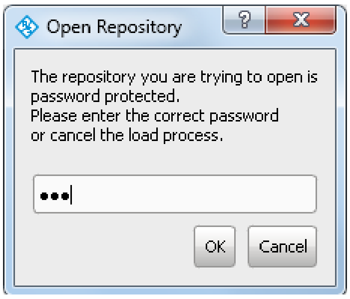
To protect a repository from accidental changes

1. In the project tree, double-click the repository name.
2. In the "Repository" dialog, select "Users".
3. To protect the repository with a password:
   a) Select "Users > <current Account>" and select "Set Password".
   The "Change Password" dialog opens.



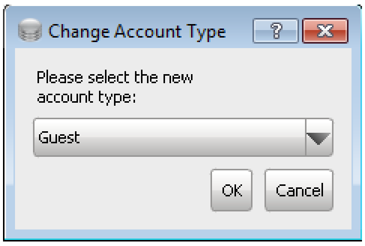
   b) Enter a password and click "Ok".

   On a new startup or when this repository is loaded, the R&S Pulse Sequencer Digital requires the correct password to load the repository.



4. To allow read-only access:

a) Select "Users > *" and select "Set Role".
The "Change Account Type" dialog opens.



b) Select a new role with read-only access, e.g. "Guest" and click "Ok".

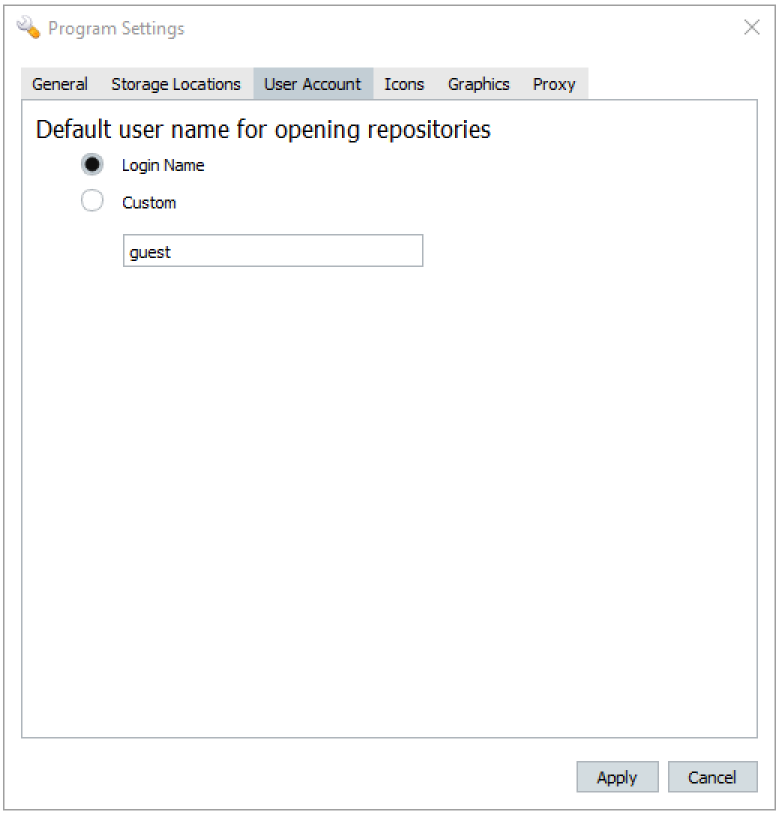
To remove the password protection on a repository

1. In the project tree, double-click the repository name.
2. In the "Repository" dialog, select "Users".
3. Select "Users > <Current Account>" and select "Set Password".
4. In the "Change Password" dialog, leave the password field blank.
5. Click"Ok".

   The "Users" dialog confirms that the repository is not protected with a password.

To change the default user used to access repositories

1. In the toolbar, select "Configure > Settings > User Account".



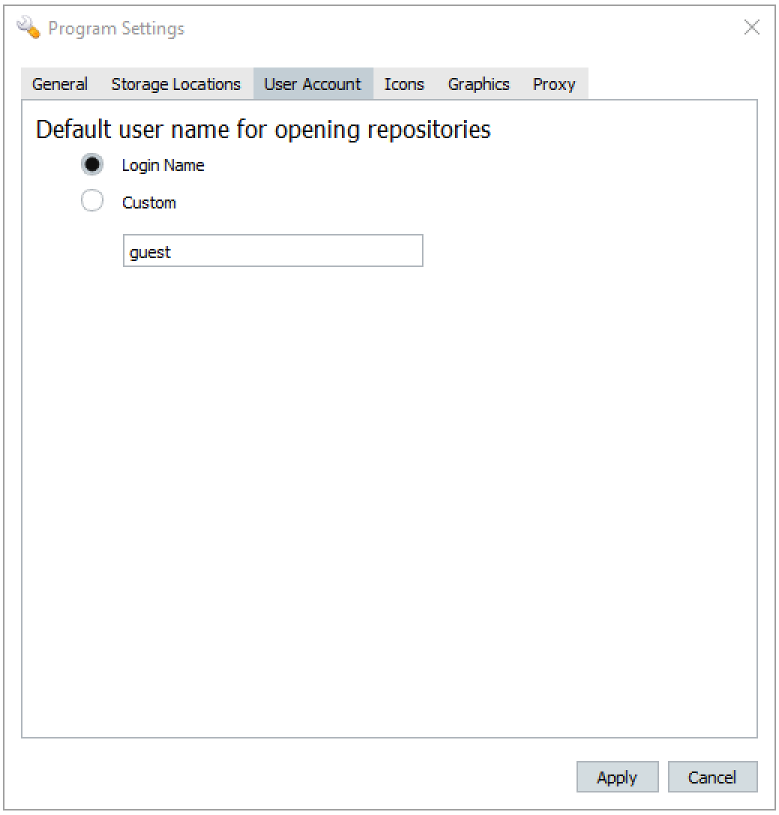
2. Select "Custom > On" and enter an existing user name.

22.2.2 Related settings

Access:

1. To open the "Repository" user management dialog:
   a) In the project tree, double-click a repository name.
      In the "Repository" dialog, select "Users".

   For example, see Figure 22-1.
2. To change the default account used to open repositories, perform the following:
   a) In the menu bar, select "Configure > Settings".
   b) Select "User Account".



Users
Indicates the user name of the currently used account and an overview of all defined users together with their roles.

For more information, see Chapter 22.2, "User administration", on page 396.

Remote command:
`REPository:ACCess?` on page 527

User Account
Set the user name for opening the repositories. Per default, your Windows login name is used, but you can also define and use a custom user name.

For more information, see Chapter 22.2, "User administration", on page 396.

22.3 Defining storage locations

During software installation, the R&S Pulse Sequencer Digital creates a predefined folder structure, concerning program and project data, report files, startup log file etc. The default folders depend on how the software is installed, for all users or for a particular one, but you can change some of them, if necessary.

Table 22-2 resumes the information provided in Installing the software and extends it with information on whether and how file paths can be modified.

Table 22-2: Overview of file types, default storage locations and where to change the file paths

File type	Default location*	Modification	Information indicated in
Program data	`%PROGRAMFILES(X86)%\Rohde-Schwarz\Pulse Sequencer Digital` (`Path`)	not possible	
Project data (repository database) SDK files	`%PUBLIC%\Public\Documents\Rohde-Schwarz\Pulse Sequencer Digital` (`DataPath`)	"Repository Manager > Add Path/Add Home Path" See "To set the storage location for repositories" on page 402.	• "Program Settings > Storage Location > Repository Databases" • For a particular repository: "Repository > Storage" See "To find out the current storage location for reports, program and project data" on page 402.
Report files	`%HOMEPATH%` (`HomePath` or `ReportPath`)	• "Program Settings > Storage Location > Reports Path" • "Scenario > Signal Calculation > Config > Reports > Target Path" See "To set the storage location for the generated report files" on page 392.	"Scenario > Status Information"
Startup log file	`%HOMEPATH%`	not possible	-
User settings (workspace, etc.)	`%HOMEPATH%\AppData\Roaming\Rohde-Schwarz\Pulse Sequencer Digital`	not possible	-

*) Software installation for all users.

22.3.1 How to set and change storage locations

See:

- "To find out the current storage location for reports, program and project data" on page 402
- "To set the storage location for repositories" on page 402
- See also Chapter A, "Supported file types and file formats", on page 625.

To find out the current storage location for reports, program and project data

Use one of the following:

1. Select "Menu bar > Help > Debug Information".
   Observe the information in the R&S Pulse Sequencer Digital section.
   Storage locations are indicated for *reports, program and project data*.
2. Select "Menu bar > Configure > Settings > Storage Locations".
   Observe the indication for "Repository Databases > Install Path" and "Reports".
   Storage locations are indicated for *reports and project data*.
3. Double-click a repository name and select "Repository > Storage".
   The storage location of the *particular repository* is indicated.
4. Open a "Scenario" and observe the displayed "Info".
   The *reports* storage location is indicated.

To set the storage location for repositories

Per default, repositories are saved in the `DataPath` directory (see Table 22-2). Alternatively, you can define a specific path and, for example, save repositories in a network folder. The latter allows for repository sharing among different users or access from different computers.

1. Select "Menu bar > Configure > Settings > Storage Locations".
   Select the respective storage locations, e.g. for reports.
2. Select "Menu bar > Repository > Load + Manage".
   The "Repository Manager" opens.
3. In the "Repository Manager", select "Add Home Path".
4. In the "Repository Manager", select "Add Path".
5. Navigate to the required directory. Select it.
   Found repository files are displayed in the list.

22.3.2 Storage locations settings

Access:

1. In the menu bar, select "Configure > Settings".
2. In the "Program Settings" dialog, select "Storage Location".

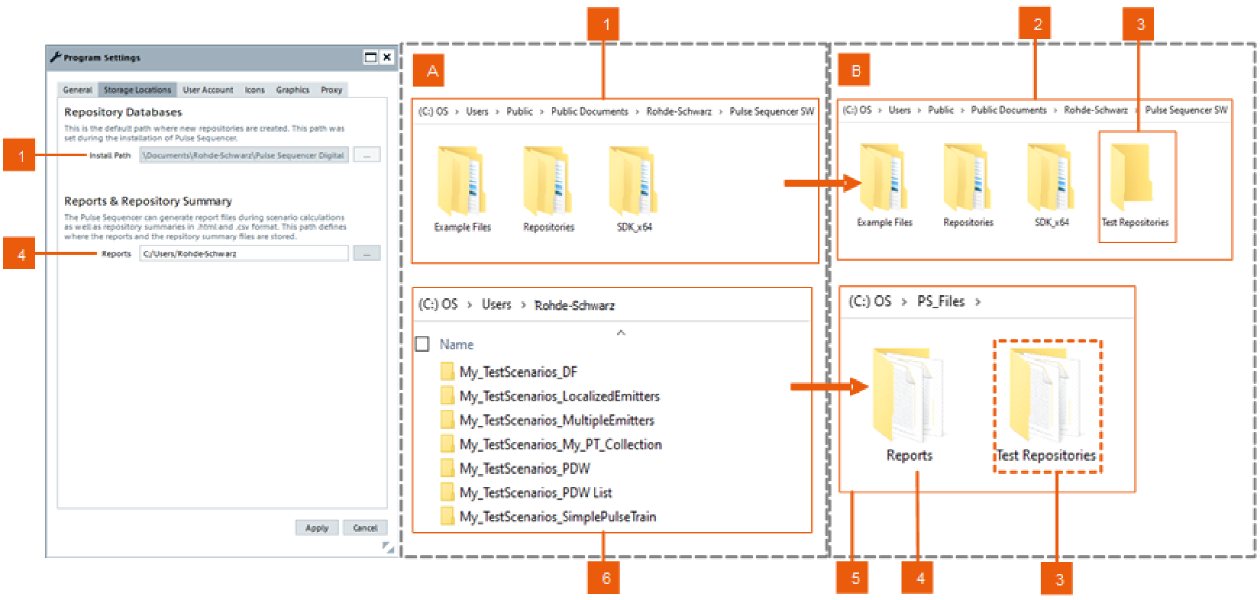


Figure 22-2: Storage Location: Understanding the displayed information

A = Default storage location
B = User-defined storage location or subdirectories
1 = Default "Install Path" / "DataPath" - cannot be changed; subdirectories are permitted (see "To set the storage location for repositories" on page 402)
2 = "DataPath"
3 = User-defined repository storage location, as a subdirectory of the default "DataPath" or "HomePath" directory, or in a freely defined location (e.g. in the UserDirectory)
4 = Reports, default and user-defined location
5 = UserDirectory, user-defined storage location
6 = "%HomePath%"

For step-by-step descriptions, see Chapter 22.3.1, "How to set and change storage locations", on page 402.

Settings:

Repository Databases
Indicates information concerning the repository storage place.

"Install Path" Storage location for repository files.
Resembles the information displayed in the "Repository Manager" (see Figure 4-1).

Remote command:
`PROGram:PATH:INSTall?` on page 435

Reports

Defines the directory that holds generated reports.

Works like "Signal Calculation > Config > Config > Reporting > Target Path".

Remote command:
`PROGram:PATH:REPort` on page 435

22.4 Speeding up calculation

In R&S Pulse Sequencer Digital, you have the different possibilities to speed up the calculation process. Use them exclusively or simultaneously.

Overview

To improve performance, especially if movements are simulated and large waveforms are calculated, we recommend that you:

- Optimize calculation speed by enabling multiple threads.
- Enable GPU-based antenna pattern calculation

Step-by-step instructions

See Chapter 22.4.1, "How to optimize performance", on page 404.

Settings

See Chapter 22.4.2, "Related settings", on page 405.

22.4.1 How to optimize performance

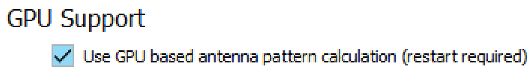
To speed up calculation by enabling multiple threads

1. In the "Scenario" dialog, select "Signal Calculation > Config > Config > Output".
2. Select "Optimize calculation speed > Enable multithreading > On".

To accelerate the antenna pattern calculation and display

The calculation of antenna patterns can be a time-consuming operation. To accelerate this operation and the 2D and 3D display of the antenna patterns, proceed as follows:

1. In the menu bar, select "Configure > Settings > Graphics".
2. Select "GPU Support > Use GPU based antenna pattern calculation > On".



3. Click "Apply".
4. Restart the software.

   The GPU (graphics processing unit) takes over the calculation. The calculation is accelerated around 800 times.

22.4.2 Related settings

Optimize calculation speed

Enable this parameter to apply multithreading and decrease the calculation time.

Select one of the options:

- "Automatic"
  The system automatically applies the optimum number of threads for the scenario.
- "Manual"
  Define the number of threads to be used.

See also Chapter 22.4, "Speeding up calculation", on page 404.

Remote command:
`SCENario:OUTPut:MULTithread` on page 564
`SCENario:OUTPut:MTMode` on page 564
`SCENario:OUTPut:MTTHreads` on page 564

GPU Support

Access: "Configure > Settings > Graphics".

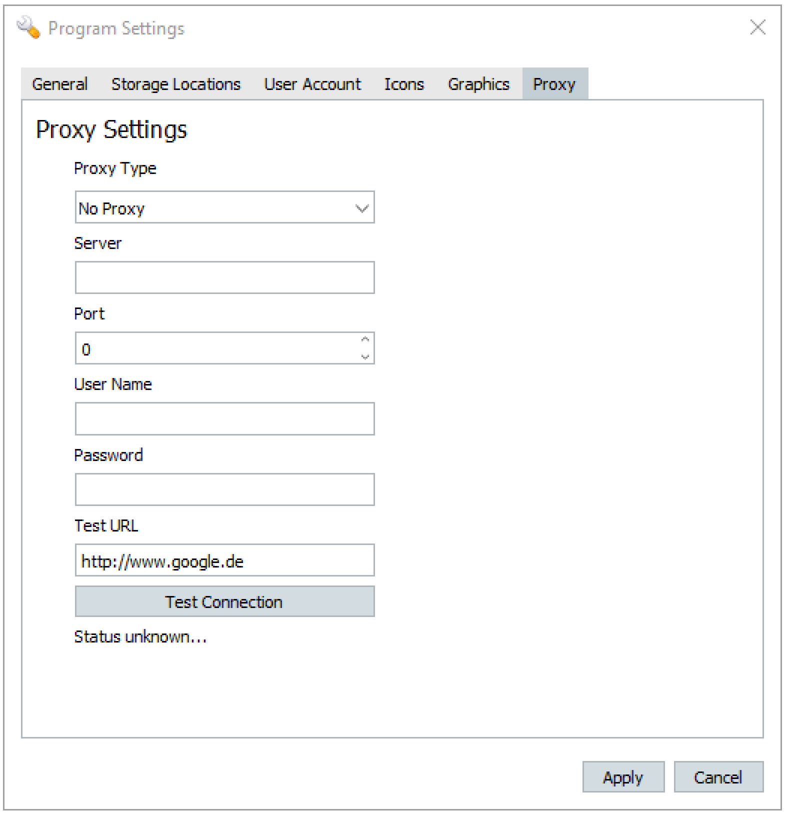
If enabled, this accelerates the antenna pattern calculation and display.

See also Chapter 22.4, "Speeding up calculation", on page 404.

Remote command:
`PROGram:GPU:ENABle` on page 435

22.5 Configuring a proxy server

Access: in the menu bar, select "Configure > Settings > Proxy".



A proxy server is useful when a PC running the R&S Pulse Sequencer Digital in a local network needs secure access to the internet, for example, to download georeferenced maps.

To configure a proxy server

1. Set "Proxy Type" = "HTTP Proxy".
2. Enter the IP and port of the proxy server.
3. Enter the user name and password to access the proxy server.
4. Enter a test URL of your choice.
5. Select "Test Connection" to attempt connection to the specified test URL.

   The dialog displays the status of the connection test.
6. Select "Apply".
7. To reject the changes and close the dialog, select "Cancel".
   The settings remain saved in the dialog fields if you do not clear them.

Select "Proxy Type = No Proxy" to deactivate the proxy settings.

22.6 Updating the software

You find the last released software version and the release notes on the R&S Pulse Sequencer Digital product page at:

https://www.rohde-schwarz.com/software/pulse-sequencer/.

To update the software version

1. Download the software and the release notes.
2. Follow the installation instructions provided in the release notes.

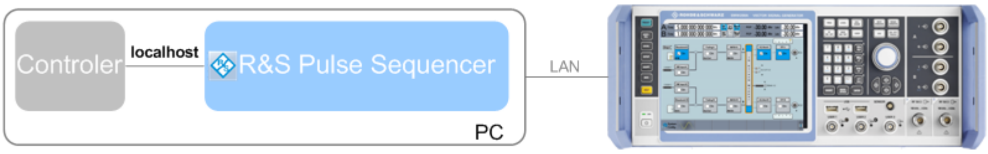
23 Automation of R&S Pulse Sequencer Digital

In addition to working with the R&S Pulse Sequencer Digital software interactively, it is also possible to operate and control it from a remote PC. Remote control operation allows you to automate the configuration process and is useful when a higher configuration speed is required.

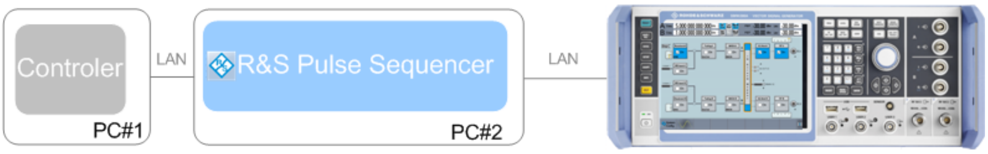
Remote control is an operation by which the software is operated remotely from a so called controller. A controller can be any PC on which some required software is installed and configured. R&S Pulse Sequencer Digital can be but not necessarily is installed on the same PC. In case R&S Pulse Sequencer Digital is not physically located on the controller's PC, a LAN connection between both PCs is required.

The following figures depict the following three possible cases:

- The controller is a script in the R&S Pulse Sequencer Digital itself and controls the software and possibly also other instruments.
- The controller and R&S Pulse Sequencer Digital are on the same PC.



- The controller and the software are installed on two separate PCs.



In remote control operation, the software itself is operated by remote control commands. For description of specific SCPI commands, refer to Chapter 24, "Remote control commands", on page 416.

A knowledge about the remote control operation and the SCPI command syntax is assumed.

Supported interfaces and protocols

R&S Pulse Sequencer Digital support remote control over the LAN interface and with the socket protocol. The socket controller is sufficient; a VISA (Virtual Instrument Software Architecture) library is not required.

Socket communication requires the specification of the port (commonly referred to as port number); the registered default port is 5025 but it can be changed if necessary.

Socket communication

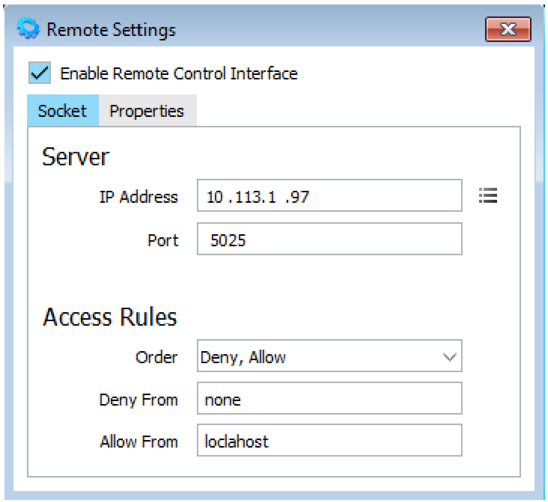
Socket communication is a simple network communication and is also referred to as "Raw Ethernet communication". It is available by default on all operating systems.

The simplest way to establish socket communication is to use the built-in telnet program. The telnet program is part of every operating system and supports a communication with the software on a command-by-command basis. For more convenience and to enable automation by programs, user-defined sockets can be programmed.

23.1 Remote settings

Access:

► In the menu bar, select "Configure > Remote Control".



Settings:

Enable Remote Control Interface
Activates the remote control interface.

If enabled, the "IP Address" of the server and the used port are indicated in the taskbar.

Socket
Configures the remote control interface.

Server ← Socket
Configures the TCP/IP server to be used for remote control:

"IP Address" Enter the IP address of the TCP/IP server or select it from the list of automatically retrieved values.

"Port" Sets the port number for the remote control interface.

Access Rules ← Socket
Defines the access rules:

"Order" Sets the order of the deny/allow rules.

"Deny from/Allow from"
Deny or allow remote access.

Allowed values:
- none
- all
- localhost i.e. 127.0.0.1
- If DNS supported, the hostname of a particular computer
- IPv4 address
- Address area, e.g. 192.168.10

Properties
Set the serial number of the software. The serial number is a unique 6-digit number, returned as a result of the `*IDN?` query.

See "To identify the R&S Pulse Sequencer Digital by its serial number" on page 413.

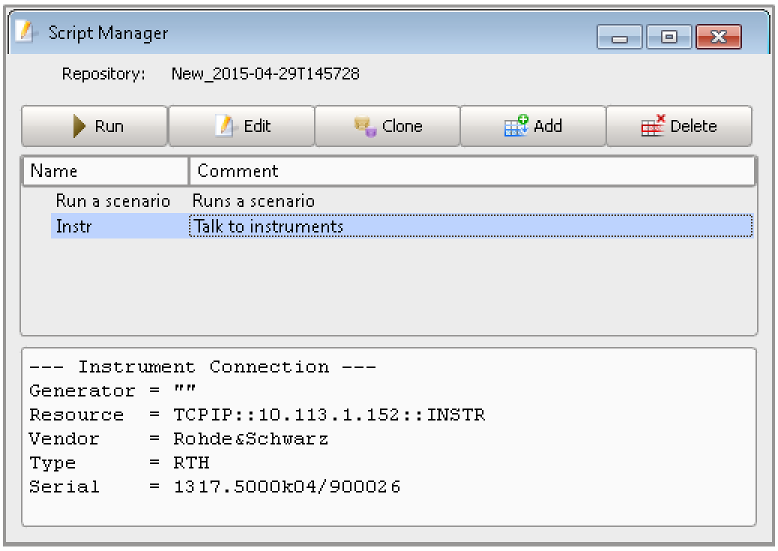
23.2 Script manager settings

The R&S Pulse Sequencer Digital provides also a built-in java script editor. With this function, you can write scripts that control the software.

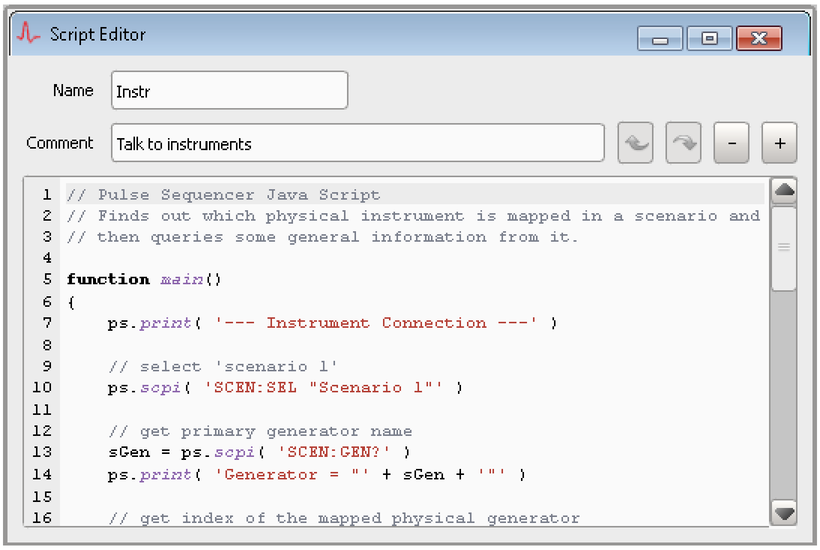
To access the script manager

1. In the menu bar, select "Tools > Script".

   The "Script Manager" lists the available scripts with their name and comment.



2. To execute a script, select "Run".
   The script output from the `ps.print` function is displayed in the "Script Manager".
3. To create a script, select "Add" or use the "Clone" function.
4. To edit a script, double-click a script in the script list or select a script and select "Edit".



   You can write or copy your java script file here.

For description of the script syntax and script examples, see:

- Chapter D, "Scripting API", on page 666
- Chapter D.3, "Script examples", on page 671

Script Manager
The dialog indicates the current repository and lists the available scripts with their name and comment.

If a script is executed, the script output, i.e. the output from the `ps.print` function, is displayed in the lower part of the dialog.

"Run" Executes the selected script.

"Edit" Opens the selected script for editing in the "Script Editor".

"Clone, Add, Delete"
Performs the corresponding action.

Script Editor
The script editor is a dialog, where you can change the name of a script, add a comment and create or edit the script code.

The icons provide common "Undo/Redo" and "Zoom In/Out" functions.

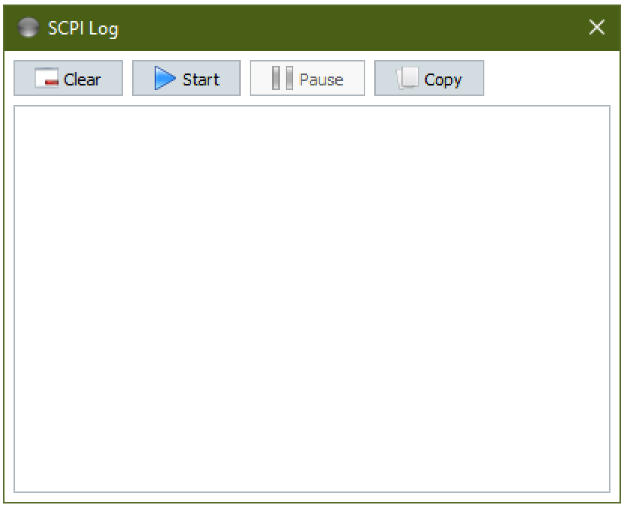
23.3 SCPI log settings

The "SCPI Log" dialog provides a sequential register of the SCPI commands associated with actions performed on the user interface. Copy the list of recorded SCPI commands to the clipboard and create scripts to automate the usage of the R&S Pulse Sequencer Digital.

To start logging SCPI commands

1. Select the "SCPI Log" dialog icon on the taskbar.

   The "SCPI Log" dialog opens.

2. Select the "Start" button.

   The "SCPI Log" dialog button on the taskbar changes to .
   The "Start" button is grayed out and the "Pause" button becomes available.

3. To skip logging, select "Pause".

4. To resume logging, select "Start".

See also Chapter 23.5, "How to log SCPI commands", on page 414.

Copy

Copies the copy the content of the "SCPI Log" dialog to the clipboard.

Clear

Deletes the content of the "SCPI Log" dialog.

Context menu

In the "SCPI Log" dialog, open the context menu (right click) to access a context menu with basic text editing capabilities.

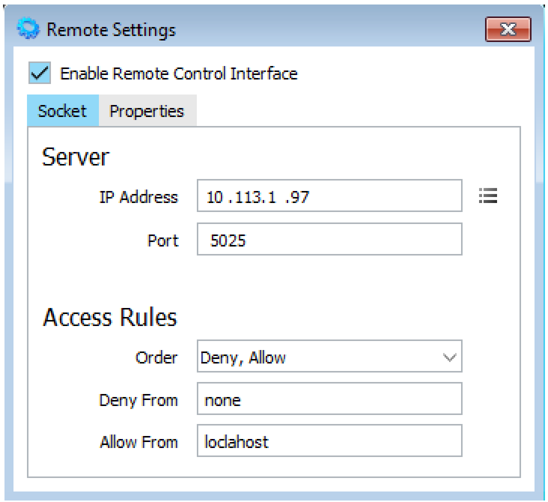
23.4 How to configure and enable remote control of R&S Pulse Sequencer Digital

See:

- "To access the required settings" on page 413
- "To use the console to test SCPI commands" on page 414
- "To find the SCPI command corresponding to a parameter on the user interface" on page 414

To access the required settings

1. In the menu bar, select "Configure > Remote Control".
2. Select "Enable Remote Control Interface".
3. Do not change the default socket port.

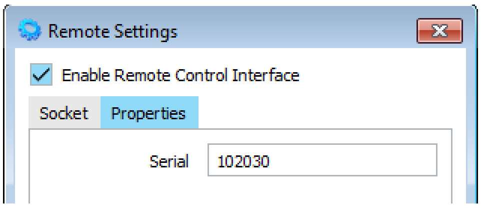


To identify the R&S Pulse Sequencer Digital by its serial number

If you remotely control several R&S Pulse Sequencer Digital from the same controller, it is useful to use serial numbers to distinguish between the different installations.

To set the serial number of the software:

1. In the menu bar, select "Configure > Remote Control".
2. Open the "Properties" tab.
3. In the "Serial" field, enter a unique 6-digit number.

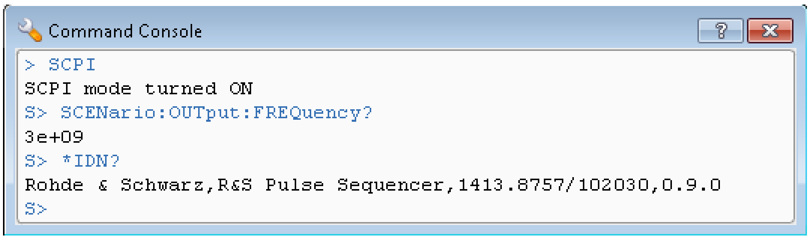


To query the serial number, use the command `*IDN?`.
See also "To use the console to test SCPI commands" on page 414.

To use the console to test SCPI commands

R&S Pulse Sequencer Digital provides a built-in console window that allows you to test SCPI commands or commands sequences.

1. To access this window, in the menu bar, select "Tools > Console"
2. In the "Command Console" window, type `SCPI` to start the remote control mode.
3. Enter any SCPI command.
   See "To use the console to test SCPI commands" on page 414.



To find the SCPI command corresponding to a parameter on the user interface

Perform one of the following:

1. Right mouse click to open the context menu of the parameter.
2. Press F1 to open the "Help" window. You find a link to the description of the SCPI command right after the parameter description.

   For a concise description of the available SCPI commands, see Chapter 24, "Remote control commands", on page 416.

23.5 How to log SCPI commands

To start logging, see To start logging SCPI commands.

1. Create the emitter.

   For details on creating scenarios and setting emitter properties, see:
   - Chapter 16, "Creating complex 2D scenarios with receiver and TX items", on page 289
   - Chapter 16.7, "Background emitters properties", on page 335

The "SCPI Log" dialog displays:

```
SCENario:CEMit:ADD
SCENario:CEMit:SELect 4
SCENario:LOCalized:TYPE BACKGROUND
```

A new emitter is added.

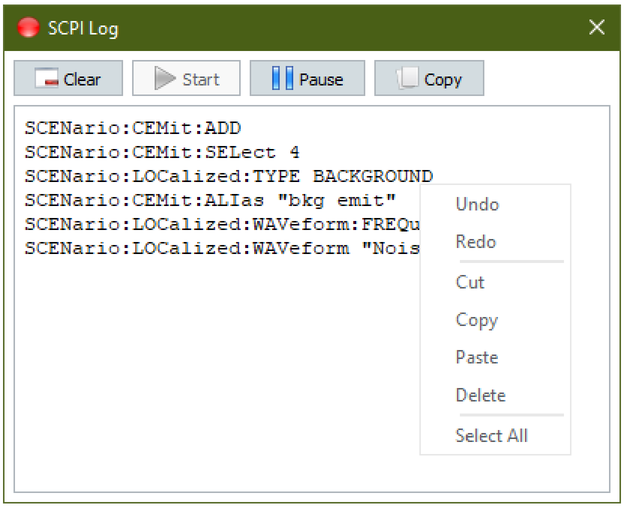
2. Rename the emitter.

   A new line appears in the "SCPI Log" dialog:

   ```
   SCENario:CEMit:ALIas "bkg emit"
   ```

3. To edit the logged SCPI command list, open the context menu (right-click). The provided functions are self-explanatory.

   The figure below shows the final result.



The context menu only edits the item or items selected in the SCPI log list. It does not change the GUI.

24 Remote control commands

The following commands are required to perform signal generation with the R&S Pulse Sequencer Digital option in a remote environment. We assume that the R&S Pulse Sequencer Digital has already been set up for remote operation in a network as described in the R&S Pulse Sequencer Digital documentation. Knowledge about the remote control operation and the SCPI command syntax is assumed.

24.1 Conventions used in SCPI command descriptions

The following conventions are used in the remote command descriptions:

- **Command usage**
  If not specified otherwise, commands can be used both for setting and for querying parameters.
  If a command can be used for setting or querying only, or if it initiates an event, the usage is stated explicitly.
- **Parameter usage**
  If not specified otherwise, a parameter can be used to set a value, and it is the result of a query.
  Parameters required only for setting are indicated as **Setting parameters**.

Parameters required only to refine a query are indicated as **Query parameters**. Parameters that are only returned as the result of a query are indicated as **Return values**.

- **Conformity**
  Commands that are taken from the SCPI standard are indicated as **SCPI confirmed**. All commands used by the R&S Pulse Sequencer Digital follow the SCPI syntax rules.
- **Asynchronous commands**
  A command which does not automatically finish executing before the next command starts executing (overlapping command) is indicated as an **Asynchronous command**.
- **Reset values (*RST)**
  Default parameter values that are used directly after resetting the instrument (`*RST` command) are indicated as ***RST** values, if available.
- **Factory preset values**
  Default parameter values that are reset only by factory preset.
- **Default unit**
  The default unit is used for numeric values if no other unit is provided with the parameter.
- **Manual operation**
  If the result of a remote command can also be achieved in manual operation, a link to the description is inserted.

24.2 Programming examples

The following sections provide simple programming examples for the R&S Pulse Sequencer Digital. The purpose of the examples is to present **all** commands for a given task. In real applications, one would rather reduce the examples to an appropriate subset of commands.

The programming examples have been tested with the built-in console which provides an environment for the development and execution of remote tests. To keep the examples as simple as possible, only the "clean" SCPI-like syntax elements are reported. The // character combination ("double-slash") indicates a non-executable command line (e.g. a comment).

24.3 Common commands

Common commands are described in the IEEE 488.2 (IEC 625-2) standard. These commands have the same effect and are employed in the same way on different devices. The headers of these commands consist of "*" followed by three letters. Many common commands are related to the Status Reporting System.

Available common commands:

***IDN?**

Identification

Returns the instrument identification.

Return values:

<ID> "Rohde&Schwarz,<device type>,<part number>/<serial number>,<firmware version>"

Usage: Query only

***OPT?**

Option identification query

Queries the options included in the instrument. For a list of all available options and their description, refer to the specifications document.

Return values:

<Options> The query returns a list of options. The options are returned at fixed positions in a comma-separated string. A zero is returned for options that are not installed.

Usage: Query only

***RST**

Reset

Sets the software to a defined default status. The default settings are indicated in the description of commands.

Unsaved settings are lost.

Usage: Setting only

24.4 Commands with similar syntax

This section describes commands, that follow similar syntax and are common to the command groups listed in the following sections.

- `<COMMAND>:ADD`
- `<COMMAND>:CATalog`
- `<COMMAND>:CLEar`
- `<COMMAND>:COMMent`
- `<COMMAND>:COUNt`

- `<COMMAND>:CREate`
- `<COMMAND>:DELete`
- `<COMMAND>:INSert`
- `<COMMAND>:NAME`
- `<COMMAND>:REMove`
- `<COMMAND>:SELect`

Commands for handling of repository elements

Repository elements are referenced by their *unique name*. The name can contain empty spaces. In each of the command groups, there is a `...:SELect` command. Use this command to select one repository element to which the subsequent commands apply.

A typical remote control sequence would look like in the Example"Handling repository elements" on page 419.

Example: Handling repository elements

The following is a simple example that explains the principle of repository elements handling by using the common commands.

```
// activate the remote control mode
SCPI
// create new empty repository
REPository:CREate "Repository for tests"
REPository:CATalog?
// "Repository for tests"
REPository:SAVE

// create two new scenarios
SCENario:CREate "New 1"
SCENario:CREate "New 2"
SCENario:SELect "New 1"
SCENario:COMMent "simple test scenario"
// rename the scenario
SCENario:NAME "New"
// remove it from the repository
SCENario:CATalog?
// "New", "New 2"
SCENario:REMove "New 2"
SCENario:CATalog?
// "New"

// create new pulse
// PULSe:CREate "New"
PULSe:CATalog?
// "New"
// although the pulse and the scenario use the same name,
// they are different and unique elements for the repository
```

Commands for handling of list and table items

Several repository elements are described in table form or as lists. For example, the sequence description table, the FM and AM Step modulations, IPM profiles in list form, or lists of multiple emitters.

Lists and tables are composed of items, where items are referenced by their *number*. Use the `...:SELect` command to select one item to which the subsequent commands apply.

A typical remote control sequence would look like in the Example"Handling items" on page 420.

Example: Handling items

The following is a simple example that explains the principle of items handling by using the common commands.

```
// activate the remote control mode
SCPI
// creates a pulse-based sequence with 2 items (i.e. segments)
SEQuence:CREate "Seq1"
SEQuence:SELect "Seq1"
SEQuence:TYPE PULSe
SEQuence:ITEM:ADD
SEQuence:ITEM:COUNt?
// 1
SEQuence:ITEM:SELect 1; SEQuence:ITEM:TYPE PULSe
SEQuence:ITEM:PULSe "P1"
SEQuence:ITEM:ADD
SEQuence:ITEM:COUNt?
// 2
SEQuence:ITEM:DELete 2
SEQuence:ITEM:COUNt?
// 1
SEQuence:ITEM:CLEar
SEQuence:ITEM:COUNt?
// 0
```

ANTenna:CATalog?
EMITter:CATalog?

RECeiver:CATalog?
IPM:CATalog?
PLATform:CATalog?
PLUGin:CATalog?
PULSe:CATalog?
SCAN:CATalog?
SCENario:CATalog?
SEQuence:CATalog?
WAVeform:CATalog?
REPository:CATalog?

Queries the available repository elements in the database.

Return values:

<Catalog>	string
Example:	See Example"Working with repositories" on page 526
Usage:	Query only
Manual operation:	See "Info" on page 61

ANTenna:CREate <Create>
EMITter:CREate <Create>
RECeiver:CREate <Create>
IPM:CREate <Create>
PLATform:CREate <Create>
PLUGin:CREate <Create>
PULSe:CREate <Create>
SCAN:CREate <Create>
SCENario:CREate <Create>
SEQuence:CREate <Create>
WAVeform:CREate <Create>
REPository:CREate <Create>

Creates a repository element with the selected name.

Setting parameters:

<Create>	string Must be unique for the particular type of repository elements. May contain empty spaces.
Example:	See Example"Working with repositories" on page 526
Usage:	Setting only
Manual operation:	See "Info" on page 61

ANTenna:SELect <Select>
EMITter:SELect <Select>
PLATform:EMITter:SELect <Select>
PLATform:SELect <Select>
RECeiver:SELect <Select>

DESTination:SELect <Select>
IPM:SELect <Select>
PLUGin:SELect <Select>
PULSe:SELect <Select>
SCAN:SELect <Select>
SCENario:SELect <Select>
SEQuence:SELect <Select>
WAVeform:SELect <Select>
REPository:SELect <Select>

Selects the repository element to which the subsequent commands apply.

Parameters:

<Select> string

Element name, as defined with the `...:CREate` or `...:NAME` command.
To query the existing elements, use the `...:CATalog?` command.
For example, `REPository:CATalog?`.

Example: See Example"Working with repositories" on page 526.

Manual operation: See "Info" on page 61

ANTenna:NAME <Name>
EMITter:MODE:NAME <Name>
EMITter:NAME <Name>
EMITter:MODE:BEAM:NAME <Name>
PLATform:NAME <Name>
RECeiver:NAME <Name>
DESTination:NAME <Name>
IPM:NAME <Name>
IPM:PLUGin:NAME <Name>
PLUGin:MODule:NAME?
PLUGin:NAME <Name>
PULSe:NAME <Name>
SCAN:NAME <Name>
SCENario:NAME <Name>
SEQuence:NAME <Name>
WAVeform:NAME <Name>

Renames the selected repository element.

Parameters:

<Name> string

Must be unique for the particular type of repository elements.
May contain empty spaces.

Example: See Example"Handling repository elements" on page 419

ANTenna:COMMent <Comment>
EMITter:COMMent <Comment>
INSTrument:COMMent <Comment>
PLATform:COMMent <Comment>
RECeiver:COMMent <Comment>
IPM:COMMent <Comment>
PLATform:COMMent <Comment>
PLUGin:COMMent <Comment>
PULSe:COMMent <Comment>
PULSe:MOP:COMMent <Comment>
SCAN:COMMent <Comment>
SCENario:COMMent <Comment>
SEQuence:COMMent <Comment>
WAVeform:COMMent <Comment>
REPository:COMMent <Comment>

Adds a description to the selected repository element.

Parameters:

<Comment> string

Example: See Example"Working with repositories" on page 526

Manual operation: See "Comment" on page 62

ANTenna:REMove <Remove>
EMITter:REMove <Remove>
PLATform:REMove <Remove>
RECeiver:REMove <Remove>
IPM:REMove <Remove>
PLUGin:REMove <Remove>
PULSe:REMove <Remove>
SCAN:REMove <Remove>
SCENario:REMove <Remove>
SEQuence:REMove <Remove>
WAVeform:REMove <Remove>
REPository:REMove <File>[,<Username>,<Passwd>]

Removes the selected element from the workspace.

The element must not reference any child elements. Remove the referenced elements first.

Setting parameters:

<File> string

Element name, as defined with the `...:CREate` or `...:NAME` command.

<Username> string

Required if the repository is password protected

<Passwd>	string Required if the repository is password protected
Example:	See Example"Working with repositories" on page 526
Usage:	Setting only
Manual operation:	See "Info" on page 61

SCENario:CPDW:GROup:ADD
SCENario:CEMit:GROup:ADD
SCENario:DF:GROup:ADD
SCENario:LOCalized:GROup:ADD
ASSignment:DESTination:PATH:EMITter:ADD
ASSignment:DESTination:PATH:ANTenna:ADD
EMITter:MODE:ADD
EMITter:MODE:BEAM:ADD
PLATform:EMITter:ADD
PLATform:EMITter:BLANkranges:ADD
RECeiver:ANTenna:ADD
PULSe:MOP:AMSTep:ADD
PULSe:MOP:FMSTep:ADD
PULSe:MOP:PCHirp:ADD
PULSe:MOP:PIECewise:ADD
PULSe:MOP:CCHirp:ADD
PULSe:MOP:PLISt:ADD
SCENario:LOCalized:ADD
SCENario:DF:ADD
SCENario:CSEQuence:ADD
SCENario:CEMit:ADD
SCENario:CEMit:MCHG:ADD
SCENario:DF:MCHG:ADD
SCENario:DF:EMITter:STATe:ADD
SCENario:LOCalized:MCHG:ADD
SCENario:LOCalized:EMITter:STATe:ADD
SCENario:DF:LOCation:PSTep:ADD
SCENario:LOCalized:LOCation:PSTep:ADD
SCENario:DF:RECeiver:MOVement:PSTep:ADD
SCENario:LOCalized:RECeiver:MOVement:PSTep:ADD
SCAN:CUSTom:ENTRy:ADD
SCENario:CPDW:ADD
IPM:LIST:ITEM:ADD
SEQuence:ITEM:IPM:ADD
SEQuence:ITEM:ADD

Appends new item.

Example:	See Example"Handling items" on page 420
Usage:	Event

Manual operation: See "New, Insert, Append, Remove, Clear Items, Undo/Redo" on page 123

PLATform:EMITter:BLANkranges:COUNt?
SCENario:CPDW:GROup:COUNt?
SCENario:CEMit:GROup:COUNt?
SCENario:DF:GROup:COUNt?
SCENario:LOCalized:GROup:COUNt?
INSTrument:COUNt?
EMITter:MODE:BEAM:COUNt?
EMITter:MODE:COUNt?
PULSe:MOP:AMSTep:COUNt?
PULSe:MOP:FMSTep:COUNt?
PULSe:MOP:PCHirp:COUNt?
PULSe:MOP:PLISt:COUNt?
PULSe:MOP:CCHirp:COUNt?
PULSe:MOP:PIECewise:COUNt?
IPM:LIST:ITEM:COUNt?
SCENario:CEMit:MCHG:COUNt?
SCENario:DF:MCHG:COUNt?
SCENario:LOCalized:MCHG:COUNt?
SCENario:LOCalized:EMITter:STATe:COUNt?
SCENario:DF:EMITter:STATe:COUNt?
SCENario:DF:LOCation:PSTep:COUNt?
SCENario:LOCalized:LOCation:PSTep:COUNt?
SCAN:CUSTom:ENTRy:COUNt?
SETup:COUNt?
SEQuence:ITEM:IPM:COUNt?
SEQuence:ITEM:COUNt?

Queries the number of existing items.

Return values:

<Count> integer

*RST: 0

Example: See Example"Handling items" on page 420

Usage: Query only

Manual operation: See "Select, No." on page 123

PLATform:EMITter:BLANkranges:SELect <Select>
PLATform:EMITter:SELect <Select>
PLATform:SELect <Select>
SCENario:CPDW:GROup:SELect <Select>
SCENario:CEMit:GROup:SELect <Select>
SCENario:DF:GROup:SELect <Select>
SCENario:LOCalized:GROup:SELect <Select>
SCENario:LOCalized:SELect <Select>

SCENario:DF:SELect <Select>
SCENario:CSEQuence:SELect <Select>
SCENario:CEMit:SELect <Select>
EMITter:MODE:BEAM:SELect <Select>
EMITter:MODE:SELect <Select>
RECeiver:ANTenna:SELect <Select>
PULSe:MOP:AMSTep:SELect <Select>
PULSe:MOP:FMSTep:SELect <Select>
PULSe:MOP:PCHirp:SELect <Select>
PULSe:MOP:PLISt:SELect <Select>
PULSe:MOP:CCHirp:SELect <Select>
PULSe:MOP:PIECewise:SELect <Select>
SCAN:CUSTom:ENTRy:SELect <Select>
IPM:LIST:ITEM:SELect <Select>
SCENario:CEMit:MCHG:SELect <Select>
SCENario:DF:MCHG:SELect <Select>
SCENario:DF:LOCation:PSTep:SELect <Select>
SCENario:DF:EMITter:STATe:SELect <Select>
SCENario:LOCalized:LOCation:PSTep:SELect <Select>
SCENario:LOCalized:EMITter:STATe:SELect <Select>
SCENario:LOCalized:MCHG:SELect <Select>
SCENario:CPDW:SELect <Select>
SETup:SELect <Select>
SEQuence:ITEM:IPM:SELect <Select>
SEQuence:ITEM:SELect <Select>

Selects the item to which the subsequent commands apply.

Parameters:

<Select> float

Item number within the range 1 to `...:COUNt`.
For example, `SEQuence:ITEM:COUNt?`.

Range: 1 to 4096

Example: See Example"Handling items" on page 420

Manual operation: See "Select, No." on page 123

INSTrument:SELect <Select>
ASSignment:ANTennas:SELect <Select>
ASSignment:EMITters:SELect <Select>
ASSignment:DESTination:SELect <Select>
ASSignment:DESTination:PATH:SELect <Select>
ASSignment:DESTination:PATH:EMITter:SELect <Select>
ASSignment:DESTination:PATH:ANTenna:SELect <Select>

Selects the element to which the subsequent commands apply.

Parameters:

<Select> string

Available element as queried with the corresponding `...:LIST` command.
For example, `ASSignment:DESTination:PATH:ANTenna:LIST?` on page 467

Example: See Chapter 17.2, "Assign signals to destination", on page 351.

Manual operation: See "Add, Remove, Clear" on page 353

SCAN:CUSTom:ENTRy:INSert <Insert>
SCENario:DF:EMITter:STATe:INSert <Insert>
SCENario:LOCalized:EMITter:STATe:INSert <Insert>
PULSe:MOP:AMSTep:INSert <Insert>
PULSe:MOP:FMSTep:INSert <Insert>
PULSe:MOP:PCHirp:INSert <Insert>
PULSe:MOP:CCHirp:INSert <Insert>
PULSe:MOP:PIECewise:INSert <Insert>
PULSe:MOP:PLISt:INSert <Insert>

Inserts a new item before the selected one.

Setting parameters:

<Insert> float

*RST: 0

Usage: Setting only

Manual operation: See "Custom Phase" on page 109

PLATform:EMITter:BLANkranges:DELete <Delete>
PLATform:EMITter:DELete <Delete>
SCENario:CPDW:GROup:DELete <Delete>
SCENario:CEMit:GROup:DELete <Delete>
SCENario:DF:GROup:DELete <Delete>
SCENario:LOCalized:GROup:DELete <Delete>
ASSignment:DESTination:PATH:EMITter:DELete
ASSignment:DESTination:PATH:ANTenna:DELete
SCAN:CUSTom:ENTRy:DELete <Delete>
EMITter:MODE:BEAM:DELete <Delete>
EMITter:MODE:DELete <Delete>
RECeiver:ANTenna:DELete
IPM:LIST:ITEM:DELete <Delete>
PULSe:MOP:AMSTep:DELete <Delete>
PULSe:MOP:FMSTep:DELete <Delete>
PULSe:MOP:PCHirp:DELete <Delete>
PULSe:MOP:CCHirp:DELete <Delete>
PULSe:MOP:PLISt:DELete <Delete>
PULSe:MOP:PIECewise:DELete <Delete>

SCENario:LOCalized:DELete <Delete>
SCENario:CEMit:MCHG:DELete <Delete>
SCENario:DF:MCHG:DELete <Delete>
SCENario:LOCalized:MCHG:DELete <Delete>
SCENario:LOCalized:EMITter:STATe:DELete <Delete>
SCENario:DF:EMITter:STATe:DELete <Delete>
SCENario:DF:DELete <Delete>
SCENario:CSEQuence:DELete <Delete>
SCENario:CEMit:DELete <Delete>
SCENario:DF:LOCation:PSTep:DELete <Delete>
SCENario:LOCalized:LOCation:PSTep:DELete <Delete>
SCENario:DF:RECeiver:MOVement:PSTep:DELete <Delete>
SCENario:LOCalized:RECeiver:MOVement:PSTep:DELete <Delete>
SCENario:CPDW:DELete <Delete>
SEQuence:ITEM:IPM:DELete <Delete>
SEQuence:ITEM:DELete <Delete>

Deletes the particular item.

Setting parameters:

<Delete> float
*RST: 0

Example: See Example"Handling items" on page 420

Usage: Setting only

Manual operation: See "New, Insert, Append, Remove, Clear Items, Undo/Redo" on page 123

PLATform:EMITter:BLANkranges:CLEar
PLATform:EMITter:CLEar
SCENario:CPDW:GROup:CLEar
SCENario:CEMit:GROup:CLEar
SCENario:DF:GROup:CLEar
SCENario:LOCalized:GROup:CLEar
ANTenna:MODel:USER:CLEar
SCAN:CUSTom:ENTRy:CLEar
EMITter:MODE:ANTenna:CLEar
EMITter:MODE:BEAM:CLEar
EMITter:MODE:CLEar
EMITter:MODE:SCAN:CLEar
RECeiver:ANTenna:CLEar
IPM:LIST:CLEar
PULSe:MOP:AMSTep:CLEar
PULSe:MOP:FMSTep:CLEar
PULSe:MOP:PCHirp:CLEar
PULSe:MOP:PLISt:CLEar
PULSe:MOP:CCHirp:CLEar
PULSe:MOP:PIECewise:CLEar
ASSignment:DESTination:PATH:EMITter:CLEar

ASSignment:DESTination:PATH:ANTenna:CLEar
SCENario:EMITter:CLEar
SCENario:CSEQuence:CLEar
SCENario:CEMit:CLEar
SCENario:CEMit:MCHG:CLEar
SCENario:LOCalized:CLEar
SCENario:DF:MCHG:CLEar
SCENario:DF:EMITter:STATe:CLEar
SCENario:LOCalized:EMITter:STATe:CLEar
SCENario:CPDW:CLEar
SCENario:DF:CLEar
SEQuence:ITEM:CLEar

Deletes all items from the list or the table.

Example: See Example"Handling items" on page 420

Usage: Event

Manual operation: See "New, Insert, Append, Remove, Clear Items, Undo/Redo" on page 123

24.5 Program settings

PROGram:CLASs:ENABle <Enable>

Enables whether the workspace classification level appears in the lower window (restart required).

Parameters:
<Enable> ON | OFF | 1 | 0

Manual operation: See "Repository Classification" on page 396

PROGram:COMMent:ENABle <Enable>

Add timestamp as comment when creating entries.

Parameters:
<Enable> ON | OFF | 1 | 0

Manual operation: See "Program Settings" on page 396

PROGram:GPU:ENABle <Enable>

Enables the GPU (Graphics Processing Unit) to be used for antenna pattern calculations.

Using the GPU accelerates the calculation.

Requires a restart.

Parameters:
<Enable> ON | OFF | 1 | 0

Manual operation: See "GPU Support" on page 405

PROGram:PATH:INSTall?

Queries the storage location for repository files.

Return values:
<Install> string

Usage: Query only

Manual operation: See "Repository Databases" on page 403

PROGram:MODE <Mode>

Selects the operation mode on start-up.

Parameters:
<Mode> DEMO | STANdard | EXPert

PROGram:PATH:REPort <Report>

Sets the directory that holds generated reports.

Parameters:
<Report> string

Manual operation: See "Reports" on page 404

PROGram:SETTings:ACCept
PROGram:SETTings:REJect

Reject changes to program settings.

Usage: Event

Manual operation: See "Apply, Cancel" on page 396

PROGram:STARtup:LOAD:ENABle <Enable>

Sets if a scenario is opened each time the software is started up.

Parameters:
<Enable> ON | OFF | 1 | 0

Manual operation: See "Program Settings" on page 396

PROGram:STARtup:WIZard:ENABle <Enable>

Enable this command, if you wish the wizard to open when the software starts.

Parameters:
<Enable> ON | OFF | 1 | 0

Manual operation: See "Program Settings" on page 396

PROGram:TOOLbar:ENABle <Enable>

Parameters:
<Enable> ON | OFF | 1 | 0

Manual operation: See "Program Settings" on page 396

PROGram:TUTorials:SHOW:ENABle <Enable>

This setting re-enables all tutorials.

Tutorials are shown upon opening certain dialogs for the first time (e.g. 2D Map). When the tutorial has been viewed, it is then disabled.

Parameters:
<Enable> ON | OFF | 1 | 0

Example: PROG:TUT:SHOW:ENAB ON

Manual operation: See "Program Settings" on page 396

24.6 Antenna pattern commands

Example: Configuring antenna patterns

```
SCPI
ANTenna:CREate "My_PlanarAntenna"
ANTenna:COMMent "planar phased antenna with parabolic aperture distribution"
ANTenna:MODel:TYPE ARRay
ANTenna:MODel:FREQuency 1e+10
ANTenna:MODel:BANDwidth 1e+9
ANTenna:MODel:POLarization HORizontal
ANTenna:MODel:ARRay:DISTribution PARabolic
ANTenna:MODel:ARRay:NX 12
ANTenna:MODel:ARRay:NZ 12
ANTenna:MODel:ARRay:XDIStance 0.6
ANTenna:MODel:ARRay:ZDIStance 0.3
ANTenna:MODel:ARRay:ELEMent:COSine 0
ANTenna:MODel:ARRay:PEDestal 0.1

ANTenna:CREate "Test"
ANTenna:MODel:TYPE HORN
ANTenna:MODel:ROTation:Z 0
ANTenna:MODel:ROTation:X 0
ANTenna:MODel:FREQuency 1e+10
ANTenna:MODel:POLarization HORizontal
ANTenna:MODel:HORN:LX 0.076
ANTenna:MODel:HORN:LZ 0.05
ANTenna:MODel:HORN:RESolution 5.E-01

ANTenna:CREate "My_PencilBeam"
ANTenna:MODel:TYPE SINC
ANTenna:MODel:ROTation:Z 0
ANTenna:MODel:ROTation:X 0
ANTenna:MODel:POLarization HORizontal
ANTenna:MODel:SINC:HPBW 3

ANTenna:CREate "Test CSC"
ANTenna:MODel:TYPE COSecant
ANTenna:MODel:POLarization HORizontal
ANTenna:MODel:COSecant:HPBW 2
ANTenna:MODel:COSecant:T1 5
ANTenna:MODel:COSecant:T2 70

ANTenna:CREate "User"
ANTenna:MODel:TYPE USER
ANTenna:MODel:USER:LOAD "C:/_PS_files/antenna.ant_pat"
ANTenna:MODel:ROTation:Z -90
ANTenna:MODel:ROTation:X 0
ANTenna:MODel:POLarization HORizontal
```

```
ANTenna:MODel:USER:CLEAr

ANTenna:CREate "Custom"
ANTenna:MODel:TYPE CUSTom
ANTenna:MODel:POLarization HORizontal
ANTenna:MODel:CUSTom:HPBW:XY 10
ANTenna:MODel:CUSTom:HPBW:YZ 3
ANTenna:MODel:CUSTom:SLSTart 30
ANTenna:MODel:CUSTom:SLRolloff 10
ANTenna:MODel:CUSTom:SLSCale 0.5
ANTenna:MODel:BACKlobe:ENABle 1
ANTenna:MODel:BACKlobe:TYPE MIRRor
ANTenna:MODel:BACKlobe:ATTenuation 40

ANTenna:NAME "Custom Phased Array"
ANTenna:MODel:TYPE CARRAY
ANTenna:MODel:POLarization VERTical
ANTenna:MODel:CARRay:GEOMetry RECTANGULAR
ANTenna:MODel:CARRay:RECTangular:NX 25
ANTenna:MODel:CARRay:RECTangular:NZ 25
ANTenna:MODel:CARRay:RECTangular:XDIStance 0.015
ANTenna:MODel:CARRay:RECTangular:ZDIStance 0.015
ANTenna:MODel:CARRay:ELEMent:COSine 1
ANTenna:MODel:CARRay:RECTangular:LATTice RECTANGULAR
ANTenna:MODel:CARRay:DIStribution COSN
ANTenna:MODel:CARRay:COSN 4
```

ANTenna:MODel:TYPE <Type>

Sets the antenna pattern.

Parameters:

<Type> DIPole | PARabolic | GAUSsian | SINC | HORN | COSecant | ARRay | USER | CUSTom | CARRay | CARDoid | PLUGin

Example: See Example"Configuring antenna patterns" on page 437.

Manual operation: See "Type" on page 175

ANTenna:MODel:POLarization <Polarization>

Sets the antenna polarization.

Parameters:

<Polarization> VERTical | HORizontal | CRIGht | CLEFt | SRIGht | SLEFt

Example: Example"Configuring antenna patterns" on page 437

Manual operation: See "Polarization" on page 176

ANTenna:MODel:CARRay:COSN <Cosn>
ANTenna:MODel:ARRay:COSN <Cosn>

Sets the value of the coefficient N in the $\cos^N$ distribution.

Parameters:

<Cosn> float

Range: 2 to 10

Example: Example"Configuring antenna patterns" on page 437

Manual operation: See "Planar Phased Array Antenna Settings" on page 182

ANTenna:MODel:ARRay:COSN:Z <Z>
ANTenna:MODel:ARRay:COSN:X <X>

Requires `ANTenna:MODel:ARRay:DISTribution:TYPE 1`.

Sets the individual value of the coefficient N in the $\cos^N$ distribution for X and Z direction.

Parameters:

<X> float

Range: 2 to 10

Manual operation: See "Planar Phased Array Antenna Settings" on page 182

ANTenna:MODel:CARRay:COSN:Z <Z>
ANTenna:MODel:CARRay:COSN:X <X>

Requires `ANTenna:MODel:CARRay:DISTribution:TYPE 1`.

Sets the individual value of the coefficient N in the cos^N distribution for X and Z direction.

Parameters:

<X> float

Range: 2 to 10

Manual operation: See "Custom Phased Array Settings" on page 185

ANTenna:MODel:CARRay:DISTribution:TYPE <Type>
ANTenna:MODel:ARRay:DISTribution:TYPE <Type>

Enables using the individual distribution function for X and Z direction.

Parameters:

<Type> ON | OFF | 1 | 0

Manual operation: See "Planar Phased Array Antenna Settings" on page 182

ANTenna:MODel:ARRay:DISTribution <Distribution>

Sets the aperture distribution of the Planar Phased Array antenna.

Parameters:

<Distribution> UNIForm | PARabolic | COSine | CSQuared | COSN | TRIangular | HAMMing | HANN

Example: Example"Configuring antenna patterns" on page 437

Manual operation: See "Planar Phased Array Antenna Settings" on page 182

ANTenna:MODel:CARRay:DISTribution <Distribution>

Sets the aperture distribution of the Custom Phased Array antenna.

Parameters:

<Distribution> UNIForm | PARabolic | COSine | CSQuared | COSN | TRIangular | HAMMing | HANN

Example: See Example"Configuring antenna patterns" on page 437

Manual operation: See "Custom Phased Array Settings" on page 185

ANTenna:MODel:ARRay:DISTribution:Z <Z>
ANTenna:MODel:ARRay:DISTribution:X <X>

Requires `ANTenna:MODel:ARRay:DISTribution:TYPE 1`.

Sets the individual aperture distribution function for X and Z direction.

Parameters:

<X> UNIForm | PARabolic | COSine | CSQuared | COSN | TRIangular | HAMMing | HANN

Manual operation: See "Planar Phased Array Antenna Settings" on page 182

ANTenna:MODel:CARRay:DISTribution:Z <Z>
ANTenna:MODel:CARRay:DISTribution:X <X>

Requires `ANTenna:MODel:ARRay:DISTribution:TYPE 1`.

Sets the individual aperture distribution function for X and Z direction.

Parameters:

<X> UNIForm | PARabolic | COSine | CSQuared | COSN | TRIangular | HAMMing | HANN

Manual operation: See "Custom Phased Array Settings" on page 185

ANTenna:MODel:ARRay:ELEMent:COSine <Cosine>
ANTenna:MODel:CARRay:ELEMent:COSine <Cosine>

Sets the characteristic of individual antenna elements.

Parameters:

<Cosine> ON | OFF | 1 | 0

0|OFF
Omnidirectional characteristic

1|ON
Cosine characteristic

Example: See Example"Configuring antenna patterns" on page 437

Manual operation: See "Custom Phased Array Settings" on page 185

ANTenna:MODel:CARRay:LINear:N <N>
ANTenna:MODel:CARRay:HEXagonal:N <N>
ANTenna:MODel:CARRay:RECTangular:NX <Nx>
ANTenna:MODel:CARRay:RECTangular:NZ <Nz>
ANTenna:MODel:ARRay:NX <Nx>
ANTenna:MODel:ARRay:NZ <Nz>

Sets the number of elements of the antenna array.

Parameters:

<Nz> float

Range: 2 to 1000 (planar phased array; linear phase array), 100 (rectangular phase array), 50 (hexagonal phase array)

Example: Example"Configuring antenna patterns" on page 437

Manual operation: See "Planar Phased Array Antenna Settings" on page 182

ANTenna:MODel:CARRay:PEDestal <Pedestal>
ANTenna:MODel:ARRay:PEDestal <Pedestal>

Sets the pedestal level of the antenna array.

Parameters:

<Pedestal> float

Range: 0 to 1

Example: Example"Configuring antenna patterns" on page 437

Manual operation: See "Planar Phased Array Antenna Settings" on page 182

ANTenna:MODel:ARRay:PEDestal:Z <Z>
ANTenna:MODel:ARRay:PEDestal:X <X>

Requires `ANTenna:MODel:ARRay:DISTribution:TYPE 1`.

Sets the individual pedestal level of the antenna array in X or Z direction.

Parameters:

<X> float

Range: 0 to 1

Manual operation: See "Planar Phased Array Antenna Settings" on page 182

ANTenna:MODel:CARRay:PEDestal:Z <Z>
ANTenna:MODel:CARRay:PEDestal:X <X>

Requires `ANTenna:MODel:ARRay:DISTribution:TYPE 1`.

Sets the individual pedestal level of the antenna array in X or Z direction.

Parameters:

<X> float

Range: 0 to 1

Manual operation: See "Custom Phased Array Settings" on page 185

ANTenna:MODel:CARRay:CIRCular:DISTance <Distance>
ANTenna:MODel:CARRay:HEXagonal:DISTance <Distance>
ANTenna:MODel:CARRay:LINear:DISTance <Distance>

ANTenna:MODel:CARRay:RECTangular:XDIStance <Xdistance>
ANTenna:MODel:CARRay:RECTangular:ZDIStance <Zdistance>
ANTenna:MODel:ARRay:XDIStance <Xdistance>
ANTenna:MODel:ARRay:ZDIStance <Zdistance>

Sets the spacing between the elements of the array antenna.

Parameters:

<Zdistance> float

Range: 0.0001 to 1

Example: Example"Configuring antenna patterns" on page 437

Manual operation: See "Planar Phased Array Antenna Settings" on page 182

ANTenna:MODel:BACKlobe:ENABle <Enable>

Enables the simulation of a back lobe.

Parameters:

<Enable> ON | OFF | 1 | 0

Example: Example"Configuring antenna patterns" on page 437

Manual operation: See "Simulate Back Lobe, Attenuation, Type" on page 177

ANTenna:MODel:BACKlobe:TYPE <Type>

Sets the shape of the back lobe pattern.

Parameters:

<Type> MIRRor | OMNidirect

Example: Example"Configuring antenna patterns" on page 437

Manual operation: See "Simulate Back Lobe, Attenuation, Type" on page 177

ANTenna:MODel:BACKlobe:ATTenuation <Attenuation>

Sets the attenuation of the back lobe.

Parameters:

<Attenuation> float

Range: 0 to 100

Example: Example"Configuring antenna patterns" on page 437

Manual operation: See "Simulate Back Lobe, Attenuation, Type" on page 177

ANTenna:MODel:COSecant:T1 <T1>
ANTenna:MODel:COSecant:T2 <T2>

Sets the Theta parameters.

Parameters:

<T2> float

Range: 1 to 90

Example: Example"Configuring antenna patterns" on page 437

Manual operation: See "Cosecant Squared Antenna Settings" on page 181

ANTenna:MODel:CARRay:CIRCular:LATTice <Lattice>
ANTenna:MODel:CARRay:RECTangular:LATTice <Lattice>

Sets the lattice.

Parameters:

<Lattice> RECTangular | TRIangular

Example: Example"Configuring antenna patterns" on page 437

Manual operation: See "Custom Phased Array Settings" on page 185

ANTenna:MODel:CARRay:CIRCular:RADius <Radius>

Set the radius of the circular phased array antenna.

Parameters:

<Radius> float

Range: 1 to 50

Manual operation: See "Custom Phased Array Settings" on page 185

ANTenna:MODel:CARRay:GEOMetry <Geometry>

Sets the geometry of the custom phased array antenna.

Parameters:

<Geometry> RECTangular | LINear | HEXagonal | CIRCular

Example: Example"Configuring antenna patterns" on page 437

Manual operation: See "Custom Phased Array Settings" on page 185

ANTenna:MODel:CARRay:ELEMent <Element>

Set the state of one or more antenna elements.

Antenna elements are indicated by their subsequent number, as it is displayed in the "Antenna Geometry" dialog (Figure 9-3).

Parameters:

<Element> numeric list

A comma-separated list of numbers or of range of numbers enclosed in brackets. In the case of a range, the end points are inclusive.
The listed numbers indicate the *active* antenna elements; if a number is omitted, the antenna element is disabled.

Example:

```
ANTenna:MODel:CARRay:GEOMetry LINEAR
ANTenna:MODel:CARRay:LINear:N 7
ANTenna:MODel:CARRay:ELEMent? (1,3:7)
// element 1 and 3 through 7 are active
ANTenna:MODel:CARRay:ELEMent (1,3,5,7)
// antenna elements 1, 3, 5 and 7 are active
// antenna elements 2, 4 and 6 are deactivated
```

Manual operation: See "Custom Phased Array Settings" on page 185

ANTenna:MODel:CUSTom:HPBW:XY <Xy>
ANTenna:MODel:CUSTom:HPBW:YZ <Yz>

Sets the required HPBW of the custom antenna.

Parameters:

<Yz> float

Range: 0.1 to 45

Example: Example"Configuring antenna patterns" on page 437

Manual operation: See "Custom Antenna Settings" on page 184

ANTenna:MODel:CUSTom:SLSTart <Slstart>

Sets the power level of the first pairs of side lobes.

Parameters:

<Slstart> float

Range: 1 to 90

Example: Example"Configuring antenna patterns" on page 437

Manual operation: See "Custom Antenna Settings" on page 184

ANTenna:MODel:CUSTom:SLRolloff <Slrolloff>

Sets the factor used to calculate the HPBW of the side lobes.

Parameters:

<Slrolloff> float

Range: 1 to 45

Example: Example"Configuring antenna patterns" on page 437

Manual operation: See "Custom Antenna Settings" on page 184

ANTenna:MODel:CUSTom:SLSCale <Slscale>

Sets the step size to calculate the power level of the side lobes.

Parameters:

<Slscale> float

Range: 0.01 to 10

Example: Example"Configuring antenna patterns" on page 437

Manual operation: See "Custom Antenna Settings" on page 184

ANTenna:MODel:BANDwidth <Bandwidth>

Sets the antenna bandwidth.

Parameters:

<Bandwidth> float

Range: 1e+06 to 1e+11

Example: Example"Configuring antenna patterns" on page 437

Manual operation: See "Frequency, Bandwidth" on page 176

ANTenna:MODel:FREQuency <Frequency>

Sets the frequency.

Parameters:

<Frequency> float

Range: 1e+06 to 1e+11
Default unit: Hz

Example: See Example"Configuring antenna patterns" on page 437

Manual operation: See "Frequency, Bandwidth" on page 176

ANTenna:MODel:COSecant:HPBW <Hpbw>

Sets the Half-Power Beam Width Cosecant Squared antenna.

Parameters:

<Hpbw> float

Range: 0.01 to 30

Example: See Example"Configuring antenna patterns" on page 437.

Manual operation: See "Cosecant Squared Antenna Settings" on page 181

ANTenna:MODel:GAUSsian:HPBW:AZIMuth <Azimuth>
ANTenna:MODel:GAUSsian:HPBW:ELEVation <Elevation>
ANTenna:MODel:SINC:HPBW:AZIMuth <Azimuth>
ANTenna:MODel:SINC:HPBW:ELEVation <Elevation>

Sets the Half-Power Beam Width in azimuth and elevation direction for the Gaussian and Sin(x)/x antennas.

Parameters:

<Elevation> float

Range: 0.1 to 45
Default unit: degree

Example: See Example"Configuring antenna patterns" on page 437.

Manual operation: See "Sin(x)/x Antenna Settings" on page 180

ANTenna:MODel:ARRay:RESolution <Resolution>
ANTenna:MODel:CARRay:RESolution <Resolution>
ANTenna:MODel:CARDoid:RESolution <Resolution>
ANTenna:MODel:COSecant:RESolution <Resolution>
ANTenna:MODel:CUSTom:RESolution <Resolution>
ANTenna:MODel:DIPole:RESolution <Resolution>
ANTenna:MODel:GAUSsian:RESolution <Resolution>
ANTenna:MODel:HORN:RESolution <Resolution>
ANTenna:MODel:PARabolic:RESolution <Resolution>
ANTenna:MODel:SINC:RESolution <Resolution>

Sets a custom resolution for the antenna pattern simulation.

Parameters:

<Resolution> float

Range: 0.1 to 1

Example: See Example"Configuring antenna patterns" on page 437.

Manual operation: See "Resolution" on page 178

ANTenna:MODel:HORN:LX <Lx>
ANTenna:MODel:HORN:LZ <Lz>

Sets the length of the rectangular sides of the Pyramidal Horn antenna.

Parameters:

<Lz> float

Range: 0.01 to 100
Default unit: m

Example: Example"Configuring antenna patterns" on page 437

Manual operation: See "Pyramidal Horn Antenna Settings" on page 181

ANTenna:MODel:PARabolic:DIAMeter <Diameter>

Sets the diameter of the parabolic dish antenna.

Parameters:

<Diameter> float

Range: 0.05 to 100
Default unit: m

Example: Example"Configuring antenna patterns" on page 437

Manual operation: See "Parabolic Antenna Settings" on page 179

ANTenna:MODel:ROTation:X <X>
ANTenna:MODel:ROTation:Z <Z>

Sets the X and Z antenna rotation.

Parameters:

<Z> float

Range: -180 to 180
Default unit: degree

Example: Example"Configuring antenna patterns" on page 437

Manual operation: See "Z-Rotation, X-Rotation" on page 176

ANTenna:MODel:CARDoid:EXPonent <Exponent>

Use values greater than 1 to narrow the antenna beam.

Parameters:

<Exponent> float

Range: 1 to 20

Manual operation: See "Cardioid Antenna Settings" on page 190

ANTenna:MODel:USER:LOAD <Load>

Loads a custom antenna pattern file.

Setting parameters:

<Load> string

Example: Example"Configuring antenna patterns" on page 437

Usage: Setting only

Manual operation: See "Load/Purge" on page 191

PLOT:POLar:TYPE <Type>

Sets the coordinates of the 2D antenna pattern diagram.

Setting parameters:
<Type> POLar | CARTesian

Example:
```
PLOT:POLar:TYPE CARTesian
PLOT:POLar:CUT XY
PLOT:POLar:LOG:MIN -90
```

Usage: Setting only

Manual operation: See "2D" on page 178

PLOT:POLar:CUT <Cut>

Sets the diagram cut.

Setting parameters:
<Cut> XY | YZ

Example: See PLOT:POLar:TYPE on page 449

Usage: Setting only

Manual operation: See "2D" on page 178

PLOT:POLar:LOG:MIN <Min>

Sets the minimum value displayed on the y axis.

Setting parameters:
<Min> float

Example: See PLOT:POLar:TYPE on page 449

Usage: Setting only

Manual operation: See "2D" on page 178

24.7 Antenna scan commands

Example: Defining antenna scans

```
SCPI
SCAN:CREate "My_RasterScan"
SCAN:TYPE RASTer
SCAN:RASTer:WIDTh 70
SCAN:RASTer:BARWidth 3
SCAN:RASTer:RATE 15
SCAN:RASTer:BARS 10
SCAN:RASTer:DIRection HORizontal
SCAN:RASTer:RETRace 0.001
SCAN:RASTer:UNIDirection 1
SCAN:RASTer:FLYBack 0.001
SCAN:RASTer:PALMer 1
SCAN:RASTer:PRATe 5
SCAN:RASTer:PSQuint 1.5

SCAN:CREate "Test Antenna Scan"
SCAN:TYPE HELical
SCAN:HELical:RPM 45
SCAN:HELical:TURNs 8
SCAN:HELical:ELEVation:STEP 3
SCAN:HELical:RETRace 1
SCAN:HELical:ROTation CW

// circular scan
SCAN:CIRCular:MODE RPM
SCAN:CIRCular:RPM 6.E+01
SCAN:CIRCular:PERiod?
// 1.E+00
```

SCAN:CIRCular:ROTation <Rotation>
SCAN:CONical:ROTation <Rotation>
SCAN:HELical:ROTation <Rotation>
SCAN:LSW:ROTation <Rotation>
SCAN:SIN:ROTation <Rotation>
SCAN:SPIRal:ROTation <Rotation>

Sets the rotation direction of the antenna.

Parameters:

<Rotation> CW | CCW

Example: See Example"Defining antenna scans" on page 451

Manual operation: See "Spiral Scan" on page 200

SCAN:CIRCular:MODE <Mode>

Sets if the scan turning speed is set as a scans rate or as a period.

Parameters:

<Mode> RPM | SEC

RPM
Scan rate, set with the command `SCAN:CIRCular:RPM`.

SEC
Scan period, set with the command `SCAN:CIRCular:PERiod`.

Example: See Chapter 24.7, "Antenna scan commands", on page 451.

Manual operation: See "Circular Scan" on page 195

SCAN:CIRCular:PERiod <Period>

Sets the time it takes for the antenna to turn once.

Parameters:

<Period> float

Range: 0.006 to 6000

Example: See Chapter 24.7, "Antenna scan commands", on page 451.

Manual operation: See "Circular Scan" on page 195

SCAN:CIRCular:RPM <Rpm>
SCAN:HELical:RPM <Rpm>

Sets the rotation speed of the antenna.

Parameters:

<Rpm> float

Range: 0.01 to 1000
Default unit: degree/s

Example: Example"Defining antenna scans" on page 451

Manual operation: See "Helical Scan" on page 199

SCAN:CONical:RATE <Rate>
SCAN:SECTor:RATE <Rate>
SCAN:SIN:RATE <Rate>
SCAN:RASTer:RATE <Rate>

Sets the turning speed.

Parameters:

<Rate> float

Range: 0.01 to 100000
Default unit: degree/s

Example: Example"Defining antenna scans" on page 451

Manual operation: See "Raster Scan" on page 198

SCAN:HELical:ELEVation:STEP <Step>

Sets the step width with that the antenna changes its elevation.

Parameters:

<Step> float

Range: 0.01 to 11.25
Default unit: degree

Example: See Example"Defining antenna scans" on page 451

Manual operation: See "Helical Scan" on page 199

SCAN:HELical:TURNs <Turns>

Sets the number of turns.

Parameters:

<Turns> float

Range: 1 to 30

Example: Example"Defining antenna scans" on page 451

Manual operation: See "Helical Scan" on page 199

SCAN:LSW:DIRection <Direction>

Sets the horizontal or vertical switching direction.

Parameters:
<Direction> H | V

Example: Example"Defining antenna scans" on page 451

Manual operation: See "Lobe Switching Scan" on page 201

SCAN:LSW:DWELl <Dwell>

Sets the speed with that the antenna switches between the lobes.

Parameters:
<Dwell> float

Range: 1e-06 to 1

Example: See Example"Defining antenna scans" on page 451

Manual operation: See "Lobe Switching Scan" on page 201

SCAN:LSW:LOBes <Lobes>

Set the number of lobes.

Parameters:
<Lobes> 2 | 4

Example: Example"Defining antenna scans" on page 451

Manual operation: See "Lobe Switching Scan" on page 201

SCAN:CONical:SQUint <Squint>
SCAN:LSW:SQUint <Squint>

Sets the offset angle of the antenna beam, that means for the conical antenna the parameter sets the radius of the scanned circle.

Parameters:
<Squint> float

Range: 0.05 to 15
Default unit: degree

Example: Example"Defining antenna scans" on page 451

Manual operation: See "Lobe Switching Scan" on page 201

SCAN:RASTer:BARS <Bars>

Sets the number of scanned bars (sectors).

Parameters:

<Bars>	float Range: 1 to 30

Example: Example"Defining antenna scans" on page 451

Manual operation: See "Raster Scan" on page 198

SCAN:RASTer:BARTranstime <Bartranstime>

Transition time between two bars in bidirectional scan mode.

Parameters:

<Bartranstime>	float Range: 0 to 1 Increment: 1 ms Default unit: seconds

---

SCAN:RASTer:BARWidth <Barwidth>

Sets the distance between two consecutive scanned bars (sectors).

Parameters:

<Barwidth>	float Range: 0.1 to 9 Default unit: m

Example: Example"Defining antenna scans" on page 451

Manual operation: See "Raster Scan" on page 198

SCAN:RASTer:REWind <Rewind>

If enabled, the antenna scans forwards and backwards.

Parameters:

<Rewind>	ON | OFF | 1 | 0

Example: See Example"Defining antenna scans" on page 451.

Manual operation: See "Raster Scan" on page 198

SCAN:RASTer:FLYBack <Flyback>
SCAN:SECTor:FLYBack <Flyback>

Sets the Flyback time for the antenna working in unidirectional mode.

Parameters:

<Flyback>	float Range: 0 to 1 Default unit: s

Example: Example"Defining antenna scans" on page 451

Manual operation: See "Sector Scan" on page 197

SCAN:HELical:RETRace <Retrace>
SCAN:RASTer:RETRace <Retrace>
SCAN:SPIRal:RETRace <Retrace>

Sets the speed for the antenna to return to the initial orientation.

Parameters:
<Retrace> float

Range: 0 to 1

Example: Example"Defining antenna scans" on page 451

Manual operation: See "Spiral Scan" on page 200

SCAN:SPIRal:UNIDirection <Unidirection>
SCAN:RASTer:UNIDirection <Unidirection>
SCAN:SIN:UNIDirection <Unidirection>
SCAN:SECTor:UNIDirection <Unidirection>

Enables a unidirectional scan mode.

Parameters:
<Unidirection> ON | OFF | 1 | 0

Example: See Example"Defining antenna scans" on page 451

Manual operation: See "Sector Scan" on page 197

SCAN:RASTer:WIDTh <Width>
SCAN:SECTor:WIDTh <Width>

Sets the width of the sector to be scanned.

Parameters:
<Width> float

Range: 0.1 to 360
Default unit: degree

Example: Example"Defining antenna scans" on page 451

Manual operation: See "Sector Scan" on page 197

SCAN:RASTer:DIRection <Direction>

Sets the scanning direction.

Parameters:
<Direction> HORizontal | VERTical

Manual operation: See "Raster Scan" on page 198

see Example"Defining antenna scans" on page 451

SCAN:SPIRal:ROUNds <Rounds>

Sets the number of rounds the antenna performs.

Parameters:

<Rounds> float

Range: 0.1 to 15

Example: Example"Defining antenna scans" on page 451

Manual operation: See "Spiral Scan" on page 200

SCAN:SPIRal:RTIMe <Rtime>

Sets the turning speed of the antenna.

Parameters:

<Rtime> float

Range: 0.01 to 100
Default unit: degree/s

Example: Example"Defining antenna scans" on page 451

Manual operation: See "Spiral Scan" on page 200

SCAN:SPIRal:STEP <Step>

Determines the step size to increase the scan radius.

Parameters:

<Step> float

Range: 1 to 11.25
Default unit: degree

Example: Example"Defining antenna scans" on page 451

Manual operation: See "Spiral Scan" on page 200

SCAN:TYPE <Type>

Sets the scan type.

Parameters:

<Type> CIRCular | SECTor | RASTer | CONical | HELical | SPIRal | LSW | SIN | CUSTom | LISSajous

Example: Example"Defining antenna scans" on page 451

Manual operation: See "Scan Type" on page 194

SCAN:CIRCular:NODDing <Nodding>
SCAN:SECTor:NODDing <Nodding>

Enables superimposing a horizontal nodding on the scan.

Parameters:

<Nodding> ON | OFF | 1 | 0

Example:
```
SCAN:SECTor:NODDing 1
SCAN:SECTor:NRATe 500
SCAN:SECTor:NELevation 15
```

Manual operation: See "Sector Scan" on page 197

SCAN:CIRCular:NELevation <Nelevation>
SCAN:SECTor:NELevation <Nelevation>

Sets the elevation angle.

Parameters:

<Nelevation> float

Range: 0.01 to 90

Example: See SCAN:SECTor:NODDing on page 459

Manual operation: See "Sector Scan" on page 197

SCAN:CIRCular:NRATe <Nrate>
SCAN:SECTor:NRATe <Nrate>

Sets the elevation rate.

Parameters:

<Nrate> float

Range: 0.01 to 2000

Example: See SCAN:SECTor:NODDing on page 459

Manual operation: See "Sector Scan" on page 197

SCAN:CIRCular:PALMer <Palmer>
SCAN:RASTer:PALMer <Palmer>
SCAN:SPIRal:PALMer <Palmer>
SCAN:SECTor:PALMer <Palmer>

Enables superimposing a conical scan on the current scan.

Parameters:

<Palmer> ON | OFF | 1 | 0

Manual operation: See "Sector Scan" on page 197

see Example"Defining antenna scans" on page 451

SCAN:CIRCular:PRATe <Prate>
SCAN:RASTer:PRATe <Prate>
SCAN:SPIRal:PRATe <Prate>
SCAN:SECTor:PRATe <Prate>

Sets the scan rate.

Parameters:

<Prate> float

Range: 0.1 to 1000

Manual operation: See "Sector Scan" on page 197

see Example"Defining antenna scans" on page 451

SCAN:CIRCular:PSQuint <Psquint>
SCAN:RASTer:PSQuint <Psquint>
SCAN:SPIRal:PSQuint <Psquint>
SCAN:SECTor:PSQuint <Psquint>

Sets the squint angle.

Parameters:

<Psquint> float

Range: 0.05 to 45

Manual operation: See "Sector Scan" on page 197

see Example"Defining antenna scans" on page 451

SCAN:CUSTom:IMPort:FILE <File>

Sets the file to import.

Parameters:

<File> string

Example: `SCAN:CUSTom:IMPort:FILE "D:/PS/scan1.csv"`

Manual operation: See "Custom Scan" on page 203

SCAN:CUSTom:IMPort:EXEC

Starts importings the file.

Usage: Event

Manual operation: See "Custom Scan" on page 203

SCAN:CUSTom:ENTRy:AZIMuth <Azimuth>

Sets the azimuth of the scan position.

Parameters:

<Azimuth> float

Range: -180 to 180

Example:

```
SCAN:TYPE CUSTOM
SCAN:CUSTom:ENTRy:ADD
SCAN:CUSTom:ENTRy:COUNt?
// 4
SCAN:CUSTom:ENTRy:SELect 1
SCAN:CUSTom:ENTRy:AZimuth 90
SCAN:CUSTom:ENTRy:ELevation 25
SCAN:CUSTom:ENTRy:DWELl 10
SCAN:CUSTom:ENTRy:JUMPtype JUMP
SCAN:CUSTom:ENTRy:SELect 2
SCAN:CUSTom:ENTRy:AZimuth -120
SCAN:CUSTom:ENTRy:ELevation 5
SCAN:CUSTom:ENTRy:DWELl 25
SCAN:CUSTom:ENTRy:JUMPtype MOVE
SCAN:CUSTom:ENTRy:TRANstime 40
...
```

Manual operation: See "Custom Scan" on page 203

SCAN:CUSTom:ENTRy:ELEVation <Elevation>

Sets the elevation of the scan position.

Parameters:

<Elevation> float

Range: -90 to 90

Example: See SCAN:CUSTom:ENTRy:AZIMuth on page 460

Manual operation: See "Custom Scan" on page 203

SCAN:CUSTom:ENTRy:DWELl <Dwell>

Sets how long the scan stays in a position.

Parameters:

<Dwell> float

Range: 0 to 3600

Example: See SCAN:CUSTom:ENTRy:AZIMuth on page 460

Manual operation: See "Custom Scan" on page 203

SCAN:CUSTom:ENTRy:JUMPtype <Jumptype>

Defines how to move to the next position, either with a jump or with a transition.

For transitions, you need to define a transition time.

Parameters:
<Jumptype> ON | OFF | 1 | 0

ON | 1
Jump enabled.

OFF | 0
Transition enabled.

Example: See `SCAN:CUSTom:ENTRy:AZIMuth` on page 460

Manual operation: See "Custom Scan" on page 203

SCAN:CUSTom:ENTRy:TRANstime <Transtime>

Sets the time for the transition between two positions.

Parameters:
<Transtime> float

Range: 0 to 3600

Example: See `SCAN:CUSTom:ENTRy:AZIMuth` on page 460

Manual operation: See "Custom Scan" on page 203

SCAN:LISSajous:AMPX <Ampx>
SCAN:LISSajous:AMPZ <Ampz>

Sets the magnitudes of two harmonic vibrations.

Parameters:
<Ampz> float

Range: 0.01 to 45

Example:

```
SCAN:LISSajous:AMPX 45
SCAN:LISSajous:AMPZ 30
SCAN:LISSajous:FREQ 10
SCAN:LISSajous:XFACtor 9
SCAN:LISSajous:ZFACtor 3
SCAN:LISSajous:PHIX 15
SCAN:LISSajous:PHIZ 30
```

Manual operation: See "Lissajous" on page 205

SCAN:LISSajous:FREQ <Freq>

Sets the base frequency.

Parameters:
<Freq> float

Range: 0.01 to 1000

Example: See `SCAN:LISSajous:AMPZ` on page 462

Manual operation: See "Lissajous" on page 205

SCAN:LISSajous:PHIX <Phix>
SCAN:LISSajous:PHIZ <Phiz>

Sets the phases of the two harmonic vibrations.

Parameters:

<Phiz> float

Range: 0 to 360

Example: See `SCAN:LISSajous:AMPZ` on page 462

Manual operation: See "Lissajous" on page 205

SCAN:LISSajous:XFACtor <Xfactor>
SCAN:LISSajous:ZFACtor <Zfactor>

Sets the ratio between the two angular frequencies.

Parameters:

<Zfactor> float

Range: 1 to 10

Example: See `SCAN:LISSajous:AMPZ` on page 462

Manual operation: See "Lissajous" on page 205

SCAN:SIN:WIDTh <Width>

Sets the angle on the XY plane between the origin and the end of the scan.

Parameters:

<Width> float

Range: 1 to 180

Example:

```
SCAN:TYPE SIN
SCAN:SIN:WIDTh 90
SCAN:SIN:HEIGht 30
SCAN:SIN:RATE 100
SCAN:SIN:ROTation CW
SCAN:SIN:UNIDirection 0
SCAN:SIN:INVersion 0
```

Manual operation: See "Sine Scan" on page 202

SCAN:SIN:HEIGht <Height>

Sets the amplitude of the sine wave.

Parameters:

<Height> float

Range: 1 to 90

Example: See SCAN:SIN:WIDTh on page 463

Manual operation: See "Sine Scan" on page 202

SCAN:SIN:INVersion <Inversion>

Sets whether the upper or the down (mirrored) sine wave is used first.

Parameters:

<Inversion> ON | OFF | 1 | 0

OFF
Upper sine first

ON
Down sine first

Example: See SCAN:SIN:WIDTh on page 463

Manual operation: See "Sine Scan" on page 202

SCAN:STEering <Steering>

Defines whether electronic steering is used.

Electronic steering is only available for scan types that use phased array antennas.

Parameters:

<Steering> ON | OFF | 1 | 0

Example: SCAN:STEering ON

Manual operation: See "Electronic Scan" on page 195

24.8 Signal mapping commands

Example: Performing signal to destination mapping in 2D scenario

```
ASSignment:DESTination:LIST?
// "Destination 1"
ASSignment:Destination:SELect "Destination 1"

ASSignment:Destination:PATH:LIST?
// "Plugin 1"
ASSignment:Destination:PATH:SELect "Plugin 1"

ASSignment:EMITters:SELect "Bkg. Emitter 1"
ASSignment:Destination:PATH:EMITter:ADD
ASSignment:Destination:PATH:EMITter:LIST?
// "Bkg. Emitter 1"
ASSignment:EMITters:LIST?
// "I1"
```

Example: Receiver signals to destination mapping in direction finding scenario

```
ASSignment:DESTination:LIST
// "Destination 1"
ASSignment:DESTination:SELect "Destination 1"
ASSignment:DESTination:PATH:LIST?
// "Plugin 1"
ASSignment:DESTination:PATH:SELect "Plugin 1"
ASSignment:ANTennas:SELect "Antenna 4 - TestEmitter"
ASSignment:DESTination:PATH:ANTenna:ADD
ASSignment:DESTination:PATH:ANTenna:LIST?
// "Antenna 4 - TestEmitter"
ASSignment:DESTination:PATH:ANTenna:SELect "Antenna 4 - TestEmitter"
ASSignment:DESTination:PATH:ANTenna:DELete
ASSignment:DESTination:PATH:ANTenna:LIST?
// ""
```


ASSignment:EMITters:LIST?

Queries the alias names of the unassigned emitters.

Return values:

<List> "<Emitter/Inter#1>","<Emitter/Inter#2>",...

Example: See Example"Performing signal to destination mapping in 2D scenario" on page 465

Usage: Query only

Manual operation: See "Emitters" on page 352

ASSignment:ANTennas:LIST?

Queries the alias names of the unassigned receiver signals.

Return values:

<List> "<ReceiverSignal#1>","<ReceiverSignal#2>",...

Example: See Example"Receiver signals to destination mapping in direction finding scenario" on page 465.

Usage: Query only

Manual operation: See "Receiver Signals" on page 353

ASSignment:DESTination:LIST?

Queries a list of the available destinations.

Return values:

<List> "<GenName#1>","<GenName2>",...

Example: See Example"Performing signal to destination mapping in 2D scenario" on page 465

Usage: Query only

Manual operation: See "Available for Assignment" on page 354

ASSignment:DESTination:PATH:LIST?

Queries the available paths.

Return values:

<List> "<Path#1>","<Path#2>",...

List of available paths.

Example: See Example"Performing signal to destination mapping in 2D scenario" on page 465

Usage: Query only

Manual operation: See "Available for Assignment" on page 354

ASSignment:DESTination:PATH:ANTenna:LIST?

Queries the list of assigned receiver signals to the selected plugin.

Return values:

<List>	"<ReceiverSignal#1>","<ReceiverSignal#2>",...

Example:	See Example"Receiver signals to destination mapping in direction finding scenario" on page 465.
Usage:	Query only
Manual operation:	See "Available for Assignment" on page 354

ASSignment:DESTination:PATH:EMITter:LIST?

Queries the list of assigned emitters to the selected path.

Return values:

<List>	"<Emitter/Inter#1>","<Emitter/Inter#2>",...

Example:	See Example"Performing signal to destination mapping in 2D scenario" on page 465.
Usage:	Query only
Manual operation:	See "Available for Assignment" on page 354

ASSignment:GROup:LIST?

If interleaving groups are defined, queries the alias names of the unassigned interleaving groups.

Return values:

<List>	string

Example:	See `ASSignment:GROup:SELect` on page 467.
Usage:	Query only
Manual operation:	See "Interleaving Groups" on page 353

ASSignment:GROup:SELect <Select>

Assigns the selected group to the plugin and path selected with the commands `ASSignment:DESTination:SELect` on page 431 and `ASSignment:DESTination:PATH:SELect` on page 431.

Parameters:

<Select>	string

Example:

```
ASSignment:DESTintation:LIST?
// "Destination 1"
ASSignment:DESTination:SELect "Destination 1"
ASSignment:DESTination:PATH:LIST?
// "RF A"
ASSignment:DESTination:PATH:SELect "Plugin 1"
ASSignment:GROup:LIST?
// "Default","Group 2"
ASSignment:GROup:SELect "Default"
```

Manual operation: See "Interleaving Groups" on page 353

24.9 Emitter commands

Example: Creating emitters

```
EMITter:CREate "My_EmitterGuidance"
EMITter:COMMent "guidance&survaillance"
EMITter:EIRP 120
EMITter:FREQuency 3e+09
EMITter:MODE:ADD
EMITter:MODE:SELect 1
EMITter:MODE:NAME "Guidance"
ANTenna:CATalog?
// "Isotropic","My_PencilBeam","Test CSC","My_PlanarAntenna","Test","Custom"
EMITter:MODE:ANTenna "My_PencilBeam"
SCAN:CATalog?
// "Test Antenna Scan","My_RasterScan"
EMITter:MODE:SCAN "My_RasterScan"
EMITter:MODE:BEAM:ADD
EMITter:MODE:BEAM:SELect 1
EMITter:MODE:BEAM:STATe 1
SEQuence:CATalog?
// "Test Sequence","My_S2_S1-F1_S1-F2","PT"
EMITter:MODE:BEAM:SEQuence "My_S2_S1-F1_S1-F2"
EMITter:MODE:BEAM:OFFSet:FREQuency 0
EMITter:MODE:BEAM:OFFSet:ELEVation 0
EMITter:MODE:BEAM:OFFSet:AZIMuth 0
```

EMITter:EIRP <Eirp>

Sets the EIRP of the emitter.

Parameters:

<Eirp> float

Range: -100 to 200
Default unit: dBW

Example: See Example"Creating emitters" on page 468

Manual operation: See "EIRP" on page 221

EMITter:FREQuency <Frequency>

Sets the operating frequency.

Parameters:

<Frequency> float

Range: 1000 to 1e+11
Default unit: Hz

Example: Example"Creating emitters" on page 468

Manual operation: See "Frequency" on page 221

EMITter:MODE:ANTenna <Antenna>

Assigns an existing antenna pattern, see `ANTenna:CATalog?`.

Parameters:

<Antenna> string

Example: Example"Creating emitters" on page 468

Manual operation: See "Emitter Modes" on page 221

EMITter:MODE:BEAM:OFFSet:AZIMuth <Azimuth>

Sets the Azimuth value for the beam offset.

Parameters:

<Azimuth> float

Range: 0 to 360

Example: Example"Creating emitters" on page 468

Manual operation: See "Emitter Beams Definition" on page 222

EMITter:MODE:BEAM:OFFSet:ELEVation <Elevation>

Offsets the position of the beam in both the azimuth or elevation.

Parameters:

<Elevation> float

Range: -90 to 90

Example: Example"Creating emitters" on page 468

Manual operation: See "Emitter Beams Definition" on page 222

EMITter:MODE:BEAM:OFFSet:FREQuency <Frequency>

Offsets the frequency of the beam.

Parameters:

<Frequency> float

Range: -1e+09 to 1e+09

Example: Example"Creating emitters" on page 468

Manual operation: See "Emitter Beams Definition" on page 222

EMITter:MODE:BEAM:SEQuence <Sequence>

Assigns a pulse sequence, see `SEQuence:CREate`.

Parameters:

<Sequence> string

Example: Example"Creating emitters" on page 468

Manual operation: See "Emitter Beams Definition" on page 222

EMITter:MODE:BEAM:STATe <State>

Activates a beam.

Parameters:

<State> ON | OFF | 1 | 0

Example: Example"Creating emitters" on page 468

Manual operation: See "Emitter Beams Definition" on page 222

EMITter:MODE:SCAN <Scan>

Assigns an antenna scan, see `SCAN:CREate`.

Parameters:

<Scan> string

Example: Example"Creating emitters" on page 468

Manual operation: See "Emitter Modes" on page 221

EMITter:MODE:ID <Id>

Parameters:

<Id> float

Range: 1 to 65536

Manual operation: See "ID" on page 222

24.10 Destination commands

Example: Creating destinations

```
SCPI
SCENario:STOP
DESTination:COUNt?
// 4
DESTination:ADD "Destination 5"
DESTination:NAME "Destination 5"
DESTination:SELect 5
DESTination:PLUGin:NAME "Plugin 2"
PLUGin:COMMent "PDW export_plugin"
PLUGin:LOAD "C:/Users/R&S/Pulse Sequencer SW/SDK_x64/Export_Excel/bin/Export-PDW.dll"
DESTination:PLUGin:VARiable:SELect:ID 1
DESTination:PLUGin:VARiable:VALue "pdw_out.csv"
DESTination:PLUGin:VARiable:SELect:ID 2
DESTination:PLUGin:VARiable:SELect "Header"
DESTination:PLUGin:VARiable:VALue "1"
DESTination:COUNt?
// 5
DESTination:SELect 3
DESTination:DELete 3
```

DESTination:COUNt?

Queries the number of available destinations.

Return values:

<Count> integer

Example: See Example"Creating destinations" on page 471

Usage: Query only

DESTination:ADD <Add>

Adds a destination to the list.

Setting parameters:
<Add> string

Example: See Example"Creating destinations" on page 471

Usage: Setting only

Manual operation: See "Add, delete, delete all" on page 57

DESTination:PLUGin:NAME <Name>

Sets the name of the export plug-in.

Parameters:
<Name> string

Example: See Example"Creating destinations" on page 471

Manual operation: See "Plugin" on page 58

DESTination:CLEar

Deletes all destinations from the current list.

Example: See Example"Creating destinations" on page 471

Usage: Event

Manual operation: See "Add, delete, delete all" on page 57

DESTination:DELete <Delete>

Deletes the selected destination from the list.

Setting parameters:
<Delete> float

Example: See Example"Creating destinations" on page 471

Usage: Setting only

Manual operation: See "Add, delete, delete all" on page 57

24.11 Inter-pulse modulation commands

Example: Using list and waveform type IPM profiles

```
SCPI
IPM:CREate "My_PRI-Stagger"
IPM:COMMent "List: 600 us, 750 us, 910 us"
IPM:UNIT SEConds
IPM:TYPE LIST
IPM:LIST:ITEM:ADD
IPM:LIST:ITEM:SELect 1
IPM:LIST:ITEM:BASE PULSe
IPM:LIST:ITEM:VALue 0.0006
IPM:LIST:ITEM:REPetitions 1
IPM:LIST:ITEM:COUNt 1
IPM:LIST:ITEM:ADD
IPM:LIST:ITEM:SELect 2
IPM:LIST:ITEM:VALue 0.00075
IPM:LIST:ITEM:COUNt 2
...
IPM:LIST:SAVE "C:\_PS_files\IPM_PRI_Stagger.txt"
IPM:LIST:CLEar
// load an existing ASCII file
IPM:LIST:LOAD "C:\_PS_files\IPM_PRList.txt"

IPM:CREate "Stagger PRI"
IPM:UNIT SEConds
IPM:TYPE WAVeform
IPM:WAVeform:TYPE SINE
IPM:WAVeform:OFFSet 0.0004
IPM:WAVeform:PHASe 0
IPM:WAVeform:PKPK 0.00014
IPM:WAVeform:BASE PULSe
IPM:WAVeform:COUNt 30
```

Example: Assigning an IPM profile to a sequence

```
SCPI
SEQuence:SELect "Test Sequence"
SEQuence:ITEM:SELect 1
SEQuence:ITEM:IPM:ADD
SEQuence:ITEM:IPM:SOURce:TYPE PROFile
SEQuence:ITEM:IPM:SOURce "My_PRI-Stagger"
SEQuence:ITEM:IPM:TARGet:TYPE PARameter
SEQuence:ITEM:IPM:TARGet:PARameter PRI
SEQuence:ITEM:IPM:EQUation "0.5*x+1"
SEQuence:ITEM:IPM:MODE INDividual
SEQuence:ITEM:IPM:RESTart 0
SEQuence:ITEM:IPM:RANDom:RESet 0
```

Example: Using the step IPM profile

```
SCPI
IPM:CREate "FHOP"
IPM:COMMent "Frequency Hops"
IPM:UNIT HERTz
IPM:TYPE STEPs
IPM:STEP:STARt 0
IPM:STEP:INCRement 1e+06
IPM:STEP:STEPs 5
IPM:STEP:BASE LENGth
IPM:STEP:BURSt 2
```

Example: Using PRI profiles of random type

```
SCPI
IPM:CREate "My_PRI-Jitter"
IPM:UNIT SEConds
IPM:TYPE RANDom
IPM:RANDom:DISTribution UNIForm
IPM:RANDom:UNIForm:MINimum 0.0011
IPM:RANDom:UNIForm:MAXimum 0.00115
IPM:RANDom:UNIForm:STEP 2.5e-06
DIALog:IPMPlot:VIEW TIMeseries
DIALog:IPMPlot:SAMPles 30

IPM:CREate "Test"
IPM:UNIT SEConds
IPM:TYPE RSTEp
IPM:RSTep:MINimum 0.0005
IPM:RSTep:MAXimum 0.005
IPM:RSTep:STEP:MINimum 0.0001
IPM:RSTep:STEP:MAXimum 0.0005
IPM:RSTep:PERiod 10

IPM:CREate "Test2"
IPM:UNIT HERTz
IPM:TYPE RLISt
// create a List (see the "My_PRI-Stagger" example)
IPM:RLISt:BASE LENGth
IPM:RLISt:BURSt 3
IPM:RLISt:REUSe 0
```

Example: Using the interpolated shape IPM profile

```
SCPI
IPM:CREate "Custom IPM"
IPM:UNIT DB
IPM:TYPE SHAPe
// create a List (see the "My_PRI-Stagger" example)
IPM:SHAPe:BASE PULSE
IPM:SHAPe:COUNt 10
// IPM:SHAPe:BASE TIME
// IPM:SHAPe:PERiod 0.5
IPM:SHAPe:INTerpol NONE
```

SEQuence:ITEM:IPM:SOURce:TYPE <Type>

Sets whether the variation is defined as a profile or as a variable.

Parameters:

<Type> PROFile | VARiable

Example: See Example"Assigning an IPM profile to a sequence" on page 473

Manual operation: See "Source" on page 158

SEQuence:ITEM:IPM:SOURce <Source>

Selects the profile source.

Use the command `IPM:CATalog?` to querry the existing profiles.

Parameters:

<Source> string

Example: See Assigning an IPM profile to a sequence

Manual operation: See "Source" on page 158

SEQuence:ITEM:IPM:SOURce:VARiable <Variable>

Sets the variable that defines the variation.

Parameters:
<Variable> string

Manual operation: See "Source" on page 158

SEQuence:ITEM:IPM:TARGet:TYPE <Type>

Sets whether the profile is assigned to a parameter or to a variable.

Parameters:
<Type> PARameter | VARiable

Example: See Assigning an IPM profile to a sequence

Manual operation: See "Target" on page 158

SEQuence:ITEM:IPM:TARGet:PARameter <Parameter>

Sets the pulse parameter to that the IPM variation is assigned.

Parameters:
<Parameter> LEVel | RLEVel | SRATe | FREQuency | PRI | WIDTh | FALL | AMFRequency | FMDeviation | DELay | FSKDeviation | PRF | FMFRequency | CDEViation | PHASe | RISE | AMDepth

Example: See Assigning an IPM profile to a sequence

Manual operation: See "Target" on page 158

SEQuence:ITEM:IPM:TARGet:VARiable <Variable>

Sets the variable to that the variation is assigned.

Parameters:
<Variable> string

Manual operation: See "Target" on page 158

SEQuence:ITEM:IPM:EQUation <Equation>

Defines output value of the IPM mathematically.

Parameters:
<Equation> string

Example: See Assigning an IPM profile to a sequence

Manual operation: See "Transformation f(x)" on page 159

SEQuence:ITEM:IPM:MODE <Mode>

Defines the way the variations are applied on repeating pulses.

Parameters:
<Mode> INDividual | SAME

Example: See Assigning an IPM profile to a sequence

Manual operation: See "Repetition" on page 159

SEQuence:ITEM:IPM:RESTart <Restart>

Restarts the IPM for this sequence line item.

Parameters:
<Restart> ON | OFF | 1 | 0

Example: See Assigning an IPM profile to a sequence

Manual operation: See "Restart" on page 159

SEQuence:ITEM:IPM:RANDom:RESet <Reset>

Resets the start seed of random generator.

Parameters:
<Reset> ON | OFF | 1 | 0

Example: See Assigning an IPM profile to a sequence

Manual operation: See "Restart" on page 159

IPM:TYPE <Type>

Sets the shape of the profile.

Parameters:
<Type> STEPs | WAVeform | RLISt | LIST | SHAPe | RANDom | EQUation | PLUGin | RSTep | BINomial

Example: See Using list and waveform type IPM profiles

Manual operation: See "Profile" on page 143

IPM:UNIT <Unit>

Sets the units of the profile.

Parameters:
<Unit> NONE | SEConds | HERTz | DB | DEGRees | PERCent

Example: See Using list and waveform type IPM profiles

Manual operation: See "Unit of Affected Parameter" on page 143

IPM:EQUation <Equation>

Defines the IPM shape as a function.

Parameters:
<Equation> string

Example: `IPM:EQUation "(i<10) ? 5 : (i*0.5)"`

Manual operation: See "Equation" on page 149

IPM:LIST:BASE <Base>

Sets the IPM profile base and defines how the steps repetition is defined.

Parameters:
<Base> PULSe | TIME

PULSe
Steps are repeated several times, as set with the command `IPM:LIST:ITEM:REPetition`.

TIME
Steps are repeated for the defined time duration, as set with the command `IPM:LIST:ITEM:TIME`.

Example: See Using list and waveform type IPM profiles

Manual operation: See "List" on page 145

IPM:LIST:ITEM:VALue <Value>

Sets the value of the selected list item.

Parameters:
<Value> float

Range: -1e+11 to 1e+11

Example: See Example"Using list and waveform type IPM profiles" on page 473

Manual operation: See "List" on page 145

IPM:LIST:ITEM:REPetition <Repetition>

Sets the number of times a list item is repeated.

Parameters:
<Repetition> float

Range: 1 to 1e+09

Example: See Using list and waveform type IPM profiles

Manual operation: See "List" on page 145

IPM:LIST:ITEM:TIME <Time>

Sets how long a list item is repeated.

Parameters:
<Time> float

Range: 0 to 1e+09

Example:
```
IPM:TYPE LIST
IPM:LIST:ITEM:SELect 3
IPM:LIST:ITEM:BASE TIME
IPM:LIST:ITEM:VALue 0.0006
IPM:LIST:ITEM:TIME 0.1
```

Manual operation: See "List" on page 145

IPM:LIST:FIRing:ENABle <Enable>

Enables using firing order for list-based IPM profiles.

Parameters:
<Enable> ON | OFF | 1 | 0

Example: See `IPM:LIST:FIRing:SEQuence` on page 480.

Manual operation: See "List" on page 145

IPM:LIST:FIRing:SEQuence <Sequence>

Sets the firing order sequence.

Parameters:
<Sequence> string

Example:
```
IPM:LIST:FIRing:ENABle 1
IPM:LIST:FIRing:SEQuence "1x1,2x2,3x3,4x4"
```

Manual operation: See "List" on page 145

IPM:LIST:LOAD <Load>

Loads an IPM profile form an ASCII file.

Parameters:
<Load> string

File path, file name, and file extension

Example: See Using list and waveform type IPM profiles

Manual operation: See "Edit List" on page 155

IPM:LIST:SAVE <Save>

Stores the IPM profile as a file.

Parameters:
<Save> string

File path incl. file name and extension.

Example: See Using list and waveform type IPM profiles

Manual operation: See "Edit List" on page 155

IPM:RANDom:DISTribution <Distribution>

Sets the distribution function.

Parameters:
<Distribution> UNIForm | NORMal | U

Example: See Example"Using PRI profiles of random type" on page 474

Manual operation: See "Random" on page 152

IPM:RANDom:NORMal:LIMit <Limit>

Sets the limit parameter of the normal distribution function.

Parameters:
<Limit> float

Range: -1e+09 to 1e+09

Example: See `IPM:RANDom:NORMal:MEAN` on page 481

Manual operation: See "Random" on page 152

IPM:RANDom:NORMal:MEAN <Mean>

Sets the mean parameter of the normal distribution function.

Parameters:
<Mean> float

Range: -1e+09 to 1e+09

Example:
```
IPM:RANDom:DISTribution NORMal
IPM:RANDom:NORMal:MEAN 0
IPM:RANDom:NORMal:STD 1
IPM:RANDom:NORMal:LIMit 3
```

Manual operation: See "Random" on page 152

IPM:RANDom:NORMal:STD <Std>

Sets the standard deviation parameter of the normal distribution function.

Parameters:
<Std> float

Range: 1e-09 to 1e+06

Example: See `IPM:RANDom:NORMal:MEAN` on page 481

Manual operation: See "Random" on page 152

IPM:RANDom:U:CENTer <Center>

Sets the center parameter of the U distribution.

Parameters:

<Center> float

Range: -1e+09 to 1e+09

Manual operation: See "Random" on page 152

IPM:RANDom:U:RANGe <Range>

Sets the range parameter of the U distribution.

Parameters:

<Range> float

Range: 1e-09 to 1e+09

Manual operation: See "Random" on page 152

IPM:RANDom:UNIForm:MINimum <Minimum>
IPM:RANDom:UNIForm:MAXimum <Maximum>

Sets the range of the uniform distribution function.

Parameters:

<Maximum> float

Range: -1e+09 to 1e+09

Example: See Example"Using PRI profiles of random type" on page 474

Manual operation: See "Random" on page 152

IPM:RANDom:UNIForm:STEP <Step>

Sets the granularity of the uniform distribution function.

Parameters:

<Step> float

Range: 1e-09 to 1e+09

Example: See Example"Using PRI profiles of random type" on page 474

Manual operation: See "Random" on page 152

IPM:RLISt:BASE <Base>

Sets the IPM profile base and defines how the increments repetition is defined.

Parameters:

<Base> LENGth | TIME

LENGth
Increments are repeated several times, as set with the command `IPM:RLISt:BURSt`.

TIME
Increments are repeated for the defined time duration, as set with the command `IPM:RLISt:PERiod`.

Example: See Example"Using PRI profiles of random type" on page 474.

Manual operation: See "Random List" on page 150

IPM:RLISt:BURSt <Burst>

Defines how many times an increment is repeated.

Parameters:

<Burst> float

Range: 1 to 8192

Example: See Example"Using PRI profiles of random type" on page 474.

Manual operation: See "Random List" on page 150

IPM:RLISt:PERiod <Period>

Sets how long an increment is repeated.

Parameters:

<Period> float

Range: 1e-09 to 1e+09

Example:

```
IPM:RLISt:BASE TIME
IPM:RLISt:PERiod 0.01
```

Manual operation: See "Random List" on page 150

IPM:RLISt:REUSe <Reuse>

If disabled, each value is used only once.

Parameters:

<Reuse> ON | OFF | 1 | 0

Example: `IPM:RLISt:REUSe 0`

Manual operation: See "Random List" on page 150

IPM:SHAPe:BASE <Base>

Defines the way the list items are processed.

Parameters:

<Base> PULSe | TIME

Example: See Using the interpolated shape IPM profile

Manual operation: See "Interpolated Shape" on page 148

IPM:SHAPe:COUNt <Count>

Sets the number of pulses for that the data from the list is used.

Parameters:

<Count> integer

Range: 1 to 1e+09

Example: See Using the interpolated shape IPM profile

Manual operation: See "Interpolated Shape" on page 148

IPM:SHAPe:INTerpol <Interpol>

Enables a linear transition between the increments.

Parameters:

<Interpol> LINear | NONE

Example: See Using the interpolated shape IPM profile

Manual operation: See "Interpolated Shape" on page 148

IPM:SHAPe:PERiod <Period>

Sets the period of time over that the list items are equally distributed.

Parameters:

<Period> float

Range: 1e-09 to 1e+09

Example: See Using the interpolated shape IPM profile

Manual operation: See "Interpolated Shape" on page 148

IPM:STEP:BASE <Base>

Sets the IPM profile base and defines how the increments repetition is defined.

Parameters:

<Base> LENGth | TIME

LENGth
Steps are repeated several times, as set with the command `IPM:STEP:BURSt`.

TIME
Steps are repeated for the defined time duration, as set with the command `IPM:STEP:PERiod`.

Example: See Example"Using the step IPM profile" on page 474

Manual operation: See "Steps" on page 144

IPM:STEP:BURSt <Burst>

Sets the number of times an increment is repeated.

Parameters:

<Burst> float

Range: 1 to 1000

Example: See Example"Using the step IPM profile" on page 474

Manual operation: See "Steps" on page 144

IPM:STEP:PERiod <Period>

Sets how long an increment is repeated.

Parameters:

<Period> float

Range: 1e-09 to 1e+09

Example:

```
IPM:STEP:BASE TIME
IPM:STEP:PERiod 0.01
```

Manual operation: See "Steps" on page 144

IPM:STEP:INCRement <Increment>

Sets the step size.

Parameters:

<Increment> float

Range: -1e+09 to 1e+09

Example: See Example"Using the step IPM profile" on page 474

Manual operation: See "Steps" on page 144

IPM:STEP:STARt <Start>

Sets the start value.

Parameters:

<Start>	float Range: -1e+09 to 1e+09

Example: See Example"Using the step IPM profile" on page 474

Manual operation: See "Steps" on page 144

IPM:STEP:STEPs <Steps>

Sets the number of steps.

Parameters:

<Steps>	float Range: 1 to 10000

Example: See Example"Using the step IPM profile" on page 474

Manual operation: See "Steps" on page 144

IPM:RSTep:MINimum <Minimum>
IPM:RSTep:MAXimum <Maximum>

Sets the value range.

Parameters:

<Maximum>	float Range: 0 to 1e+11

Example: See Example"Using PRI profiles of random type" on page 474

Manual operation: See "Random Steps" on page 151

IPM:RSTep:STEP:MINimum <Minimum>
IPM:RSTep:STEP:MAXimum <Maximum>

Sets the step size range.

Parameters:

<Maximum>	float Range: 0.1 to 0.5

Example: See Example"Using PRI profiles of random type" on page 474

Manual operation: See "Random Steps" on page 151

IPM:RSTep:PERiod <Period>

Sets the pattern length.

Parameters:

<Period>	float Range: 0 to 4096

Example: See Example"Using PRI profiles of random type" on page 474

Manual operation: See "Random Steps" on page 151

IPM:WAVeform:TYPE <Type>

Sets the profile shape.

Parameters:
<Type> RAMP | SINE | TRIangular

Example: See Using list and waveform type IPM profiles

Manual operation: See "Waveform" on page 147

IPM:WAVeform:BASE <Base>

Defines how the waveform period is defined, as a time duration or as a number of pulses.

Parameters:
<Base> PULSe | TIME

Example: See Using list and waveform type IPM profiles

Manual operation: See "Waveform" on page 147

IPM:WAVeform:COUNt <Count>

Sets the waveform period as number of pulses.

Parameters:
<Count> integer
Range: 1 to 1e+09

Example: See Using list and waveform type IPM profiles

Manual operation: See "Waveform" on page 147

IPM:WAVeform:OFFSet <Offset>

Shifts the profile by the selected offset.

Parameters:
<Offset> float
Range: -1e+09 to 1e+09

Example: See Using list and waveform type IPM profiles

Manual operation: See "Waveform" on page 147

IPM:WAVeform:PHASe <Phase>

Enables a phase offset to change the start phase of the sine wave.

Parameters:

<Phase> float

Range: -1e+09 to 1e+09
Default unit: sec

Example: See Using list and waveform type IPM profiles

Manual operation: See "Waveform" on page 147

IPM:WAVeform:PERiod <Period>

Sets the waveform period.

Parameters:

<Period> float

Range: 1e-09 to 1e+09
Default unit: sec

Example: See Using list and waveform type IPM profiles

Manual operation: See "Waveform" on page 147

IPM:WAVeform:PKPK <Pkpk>

Sets the value range of the linear ramp profile or the period of the sine profile.

Parameters:

<Pkpk> float

Range: 1e-09 to 1e+09
Default unit: sec

Example: See Using list and waveform type IPM profiles

Manual operation: See "Waveform" on page 147

IPM:BINomial:VAL1 <Val1>
IPM:BINomial:VAL2 <Val2>

Sets the values of the binomial distribution function.

Parameters:

<Val2> float

Range: -1e+09 to 1e+09

Example:

```
IPM:TYPE BINOMIAL
IPM:BINomial:VAL1 100
IPM:BINomial:PVAL1 80
IPM:BINomial:VAL2 0
```

Manual operation: See "Binomial" on page 154

IPM:BINomial:PVAL1 <Pval1>

Sets the probability of occurrence of value 1 in the binomial distribution function.

Parameters:

<Pval1> float

Range: 0 to 100
Default unit: PCT

Example: See IPM:BINomial:VAL2 on page 488

Manual operation: See "Binomial" on page 154

DIALog:IPMPlot:VIEW <View>

Defines what kind of information is represented in the IPM profile diagram.

Setting parameters:

<View> TIMeseries | HISTogram

TIMeseries
Visualization of the profile variation over time

HISTogram
Statistical representation of the relative frequency density

Example: See Example"Using PRI profiles of random type" on page 474.

Usage: Setting only

Manual operation: See "2D" on page 143

DIALog:IPMPlot:SAMPles <Samples>

Sets the number of values to be displayed in the preview diagram of the IPM profile.

Setting parameters:

<Samples> float

Example: See Example"Using PRI profiles of random type" on page 474.

Usage: Setting only

Manual operation: See "2D" on page 143

24.12 Import interface commands

Example: PDW import (unmodulated pulses)

```
IMPort:PDW:FILE:PDW:LOAD "C:/_PS_files/PDW import/PDW01_unmodulated.txt"
IMPort:PDW:FILE:PDW?
// C:/_PS_files/PDW import/PDW01_unmodulated.txt
IMPort:PDW:FILE:TEMPlate:LOAD "C:/_PS_files/PDW import/PDW01_unmodulated.pdwt"
IMPort:PDW:FILE:TEMPlate?
// C:/_PS_files/PDW import/PDW01_unmodulated.pdwt

IMPort:PDW:NORM 1
IMPort:PDW:EXEcute
IMPort:PDW:STATus?
// 1
IMPort:PDW:DATA:SEL 1
IMPort:PDW:DATA:TOA?
// 0
IMPort:PDW:DATA:WIDTh?
// 3e-05
IMPort:PDW:DATA:FREQuency?
// 3e+09
IMPort:PDW:DATA:OFFSet?
// 0
IMPort:PDW:DATA:PHASe?
// 0
IMPort:PDW:DATA:LEVel?
// 0
IMPort:PDW:DATA:MOP?
// CW
```

IMPort:PDW:FILE:PDW <Pdw>

Sets or queries the name of the used PDW list file.

Parameters:

<Pdw> absolute file path and filename, incl. file extension

Example: See Example"PDW import (unmodulated pulses)" on page 490.

Manual operation: See "PDW File" on page 366

IMPort:PDW:FILE:TEMPlate <Template>

Sets or queries the name of the used import template file.

Parameters:

<Template> absolute file path and filename, incl. file extension

Example: See Example"PDW import (unmodulated pulses)" on page 490.

Manual operation: See "Template" on page 367

IMPort:PDW:FILE:PDW:LOAD
IMPort:PDW:FILE:TEMPlate:LOAD

Loads the selected file.

Example: See Example"PDW import (unmodulated pulses)" on page 490.

Usage: Event

Manual operation: See "Select import template file, Create new import template, Save/Save as" on page 367

IMPort:PDW:FILE:PDW:SAVE
IMPort:PDW:FILE:TEMPlate:SAVE

Stores the selected file.

Example: See Example"PDW import (unmodulated pulses)" on page 490.

Usage: Event

Manual operation: See "Select import template file, Create new import template, Save/Save as" on page 367

IMPort:PDW:NORM <Norm>

Normalizes the TOA (time of arrival) of the first pulse to 0. Subsequent TOAs are relative.

Parameters:
<Norm> ON | OFF | 1 | 0

Example: See Example"PDW import (unmodulated pulses)" on page 490.

Manual operation: See "Normalize TOA" on page 367

IMPort:PDW:EXECute

Starts parsing the PDW list file.

Example: See Example"PDW import (unmodulated pulses)" on page 490.

Usage: Event

Manual operation: See "Process PDW File" on page 367

IMPort:PDW:STATus?

Queries the parsing status.

Return values:

<Status> ON | OFF | 1 | 0

1
Import completed

Example: See Example"PDW import (unmodulated pulses)" on page 490.

Usage: Query only

Manual operation: See "Import Status" on page 367

IMPort:PDW:DATA:SEL <Sel>

Selects the pulse for that the further queries apply.

Parameters:

<Sel> float

Range: 1 to max

Example: See Example"PDW import (unmodulated pulses)" on page 490.

Manual operation: See "Imported Data" on page 367

IMPort:PDW:DATA:MOP?

Queries the used modulation on pulse (MOP).

Use the corresponding command to query further pulse and modulation parameter for the respective MOP.

Return values:

<Mop> CW | AM | FM | ASK | FSK | PSK | LFM | NLFM | TFM | BKR2a | BKR2b | BKR3 | BKR4a | BKR4b | BKR5 | BKR7 | BKR11 | BKR13 | CPH | PLFM

Example: See Example"PDW import (unmodulated pulses)" on page 490.

Usage: Query only

Manual operation: See "Imported Data" on page 367

IMPort:PDW:DATA:AM:DEPTh?
IMPort:PDW:DATA:AM:MODFreq?
IMPort:PDW:DATA:ASK:CHIPcount?
IMPort:PDW:DATA:ASK:RATE?
IMPort:PDW:DATA:ASK:STATes?
IMPort:PDW:DATA:ASK:STEP?
IMPort:PDW:DATA:CPH:CHIPcount?
IMPort:PDW:DATA:CPH:VALues <Values>
IMPort:PDW:DATA:FM:DEViation?
IMPort:PDW:DATA:FM:MODFreq?
IMPort:PDW:DATA:FREQuency?
IMPort:PDW:DATA:FSK:CHIPcount?

IMPort:PDW:DATA:FSK:RATE?
IMPort:PDW:DATA:FSK:STATes?
IMPort:PDW:DATA:FSK:STEP?
IMPort:PDW:DATA:LEVel?
IMPort:PDW:DATA:OFFSet?
IMPort:PDW:DATA:PHASe?
IMPort:PDW:DATA:PLFM:VALues <Values>
IMPort:PDW:DATA:PSK:CHIPcount?
IMPort:PDW:DATA:PSK:RATE?
IMPort:PDW:DATA:PSK:STATes?
IMPort:PDW:DATA:PSK:STEP?
IMPort:PDW:DATA:LFM:RATE?
IMPort:PDW:DATA:NLFM:CUBic?
IMPort:PDW:DATA:NLFM:LINear?
IMPort:PDW:DATA:NLFM:QUADratic?
IMPort:PDW:DATA:TOA?
IMPort:PDW:DATA:WIDTh?

Queries the pulse parameter.

Return values:
<Width> float

Example: See Example"PDW import (unmodulated pulses)" on page 490.

Usage: Query only

Manual operation: See "Imported Data" on page 367

IMPort:PDW:DATA:ASK:PATTern?
IMPort:PDW:DATA:FSK:PATTern?
IMPort:PDW:DATA:PSK:PATTern?

Queries the pulse parameter.

Return values:
<Pattern> string

Example: See Example"PDW import (unmodulated pulses)" on page 490.

Usage: Query only

IMPort:PDW:STORe

Stores the imported PDW list file as waveform element in the repository.

Example: See Example"PDW import (unmodulated pulses)" on page 490.

Usage: Event

Manual operation: See "Import into Repository" on page 368

IMPort:VIEW:COUNt <Count>

Sets the entries per page to be displayed.

Parameters:
<Count> 50 | 100 | 500 | 1000 | 5000 | 10000 | 50000 | 100000

Manual operation: See "View > PDW Data View" on page 364

IMPort:VIEW:TIME:STARt <Start>

Sets the start line displayed on the page.

Parameters:
<Start> float

Manual operation: See "View > PDW Data View" on page 364

IMPort:VIEW:COUNt <Count>

Sets the entries per page to be displayed.

Parameters:
<Count> 50 | 100 | 500 | 1000 | 5000 | 10000 | 50000 | 100000

Manual operation: See "View > PDW Data View" on page 364

IMPort:VIEW:MOVE:BACKwards
IMPort:VIEW:MOVE:FORWard
IMPort:VIEW:MOVE:STARt
IMPort:VIEW:MOVE:END

Goes to the first/next/previous/last page.

Usage: Event

Manual operation: See "View > PDW Data View" on page 364

24.13 License server commands

LSERver:APPLy

Applies the changes.

Usage: Event

LSERver:HOST <Host>

Sets the license server host.

Parameters:

<Host> string

LSERver:OPTions?

Queries the available options.

Return values:

<Options> string

Usage: Query only

LSERver:PORT <Port>

Sets the license server port.

Parameters:

<Port> float

LSERver:READy?

Queries the status of the license server.

Return values:

<Ready> ON | OFF | 1 | 0

Usage: Query only

LSERver:STATus?

Queries the status of the license server.

Return values:

<Status> string

Usage: Query only

24.14 Marker commands

Marker commands enable or up to four markers in any combination at once.

Markers signals are binary signals. Each marker signal is represented by a single bit within a marker byte. The Table 24-1 explains the assailment.

Table 24-1: Setting parameter as function of the marker states

Marker	Marker byte	Value of the settings parameter in the remote commands
M1	0000 0001	0x01
M2	0000 0010	0x02

Marker	Marker byte	Value of the settings parameter in the remote commands
M3	0000 0100	0x04
M4	0000 1000	0x08

In the following examples, we assume that a pulse-base sequence "Test Sequence" and the required Pulses have been created.

See also:

- Example"Handling items" on page 420
- Example"Creating an unmodulated pulse" on page 507

Example: Defining pulse and sequence markers and enabling the global markers

```
SCPI
PULSe:SELect "P1"

// enable Marker 1 als gate marker
PULSe:MARKer:GATE 1

SEQuence:SELect "Test Sequence"
SEQuence:ITEM:SELect 1
SEQuence:ITEM:MARKer:FIRSt 2
SEQuence:ITEM:MARKer:LAST 4
SEQuence:ITEM:MARKer:ALL 1

// enable Marker 1 to mark the fifth loop run, i.e.
// enable the condition $loop_i = 5 for Marker 1
// the used loop variable $loop must exist
SEQuence:ITEM:MARKer:CONDition 1
SEQuence:ITEM:MARKer:CONDition:VARiable "$loop_i"
SEQuence:ITEM:MARKer:CONDition:TYPE EQUAL
SEQuence:ITEM:MARKer:CONDition:VALue "5"

SCENario:OUTPut:MARKer:ENABle 1
SCENario:OUTPut:MARKer:FLAGs 14
```

Example: Defining emitter markers

```
SCPI
Scenario:CEMit:Select 1
SCENario:CEMit:MARKer:FORCe 1
SCENario:CEMit:MARKer:GATE 1
```

PULSe:MARKer:GATE <Gate>

Enables up to four gate markers.

Parameters:

<Gate> float

see Table 24-1.

Range: 0 to 65535

Example: See Example"Defining pulse and sequence markers and enabling the global markers" on page 497

Manual operation: See "Pulse Markers" on page 380

SEQuence:ITEM:MARKer:CONDition <Condition>
SEQuence:ITEM:MARKer:FIRSt <First>
SEQuence:ITEM:MARKer:LAST <Last>
SEQuence:ITEM:MARKer:ALL <All>

Enables up to four markers of the corresponding type.

Parameters:

<All> float

See Table 24-1.

Range: 0 to 65535

Example: See Example"Defining pulse and sequence markers and enabling the global markers" on page 497.

SEQuence:ITEM:MARKer:CONDition:TYPE <Type>

Sets the sign in the logical condition.

Parameters:

<Type> SMALler | GREater | EQUal | NOTequal

Example: See Example"Defining pulse and sequence markers and enabling the global markers" on page 497

SEQuence:ITEM:MARKer:CONDition:VARiable <Variable>

Defines the value that is compared with the fixed values set with the command `SEQuence:ITEM:MARKer:CONDition:VALue` on page 499.

Parameters:

<Variable> string

Example: See Example"Defining pulse and sequence markers and enabling the global markers" on page 497

SEQuence:ITEM:MARKer:CONDition:VALue <Value>

Sets the numerical value used with the comparison.

Parameters:

<Value> string

Example: See Example"Defining pulse and sequence markers and enabling the global markers" on page 497

SCENario:OUTPut:MARKer:ENABle <Enable>

Enables that markers are considered by the generation of the output waveform file.

Parameters:

<Enable> ON | OFF | 1 | 0

Example: See Example"Defining pulse and sequence markers and enabling the global markers" on page 497

Manual operation: See "Marker Signals" on page 350

SCENario:OUTPut:MARKer:FLAGs <Flags>

Enables up to four markers.

Parameters:

<Flags> int

Binary value, where:
M1 = 1
M1 = 2
M1 = 4
M1 = 8

Range: 0 to 15
*RST: 15

Example: See Example"Defining pulse and sequence markers and enabling the global markers" on page 497

Manual operation: See "Marker Signals" on page 350

SCENario:CEMit:MARKer:FORCe <Force>
SCENario:DF:MARKer:FORCe <Force>
SCENario:LOCalized:MARKer:FORCe <Force>

Determines how the marker is handled.

Parameters:

<Force> ON | OFF | 1 | 0

ON | 1
Forces the selected marker type for every pulse of the selected emitter

OFF | 0
Leaves the marker unchanged, as defined in the pulses and sequences of this emitter.

Example: See Example"Defining emitter markers" on page 497.

Manual operation: See "Emitter Marker" on page 315

SCENario:CEMit:MARKer:GATE <Gate>
SCENario:DF:MARKer:GATE <Gate>
SCENario:LOCalized:MARKer:GATE <Gate>

Enables marker for gate.

Parameters:

<Gate> float

Binary value, where:
M1 = 1
M1 = 2
M1 = 4
M1 = 8

Range: 0 to 15

Example: See Example"Defining emitter markers" on page 497.

Manual operation: See "Emitter Marker" on page 315

24.15 Plugin and reporting commands

Example: Loading plugins in the repository

```
PLUGin:CREate "Plugin 5"
PLUGin:SELect "Plugin 5"
PLUGin:LOAD "C:\_PS_files\APatImport.dll"
PLUGin:MODule:AUTHor?
// Rohde&Schwarz
PLUGin:MODule:NAME?
// Multiple Patterns
PLUGin:MODule:TYPE?
// Pattern Import
PLUGin:MODule:VERSion?
// 1.0.0
PLUGin:MODule:COMMent?
// This plugin creates custom antenna patterns.
PLUGin:MODule:DATA?
// 0
```

Example: Using a plugin as an IPM profile

```
SCPI
IPM:CREate "Custom IPM"
IPM:UNIT PERCent
IPM:TYPE PLUGin
PLUGin:CATalog?
// "My_QAM_PlugIn","CustomIPM"
IPM:PLUGin:NAME "CustomIPM"
IPM:PLUGin:VARiable:CATalog?
// "$step","$pw"
IPM:PLUGin:VARiable:SELect "$pw"
IPM:PLUGin:VARiable:VALue 0.00001
```

Example: Generating reports

```
SCPI
SCENario:PDW:ENABle 1
SCENario:PDW:PATH "C:\_My_Report_Files"
SCENario:PDW:TYPE PLUGin
PLUGin:CATalog?
// "My_QAM_PlugIn","CustomIPM","Report PDW","Reporting DFS"
SCENario:PDW:PLUGin:NAME "Report PDW"
SCENario:PDW:PLUGin:VARiable:CATalog?
// "File","Header"
SCENario:PDW:PLUGin:VARiable:SELect "File"
SCENario:PDW:PLUGin:VARiable:VALue?
// "pdw_out.txt"
```

PLUGin:LOAD <Load>

Loads the selected DLL file, see also Chapter C, "Plug-in programming API", on page 650.

Setting parameters:

<Load> string

File path incl. file name and extension

Example: See Example"Loading plugins in the repository" on page 501

Usage: Setting only

PLUGin:MODule:AUTHor?
PLUGin:MODule:COMMent?
PLUGin:MODule:VERSion?
PLUGin:MODule:TYPE?

Queries information on the loaded file.

The query returns information as specified in the description of the corresponding function in Chapter C, "Plug-in programming API", on page 650.

The following are the possible values for the type query.

Return values:

<Type> REPort | IPM

IPM
Plugin for IPM

REPort
Plugin for reports created during the waveform generation

Example: See Example"Loading plugins in the repository" on page 501

Usage: Query only

PLUGin:MODule:DATA?

Queries whether the plugin requires data from a data source.

Return values:

<Data> 0 | 1

0
Data source is not required

1
Data source is required

*RST: 0

Example: See Example"Loading plugins in the repository" on page 501

Usage: Query only

DESTination:PLUGin:VARiable:CATalog <Catalog>
SCENario:PDW:PLUGin:VARiable:CATalog?
IPM:PLUGin:VARiable:CATalog?

Queries the variables used in the plugin.

Return values:

<Catalog> string

Example: See Example"Using a plugin as an IPM profile" on page 501

Usage: Query only

Manual operation: See "Plug-in" on page 153

DESTination:PLUGin:VARiable:SELect <Select>
SCENario:PDW:PLUGin:VARiable:SELect <Select>
IPM:PLUGin:VARiable:SELect <Select>

Selects a plugin variable.

Parameters:

<Select> string

Example: See Example"Using a plugin as an IPM profile" on page 501

Manual operation: See "Plug-in" on page 153

DESTination:PLUGin:VARiable:VALue <Value>
SCENario:PDW:PLUGin:VARiable:VALue <Value>
IPM:PLUGin:VARiable:VALue <Value>

Sets the values of the selected variable.

Parameters:

<Value> string

Example: See Example"Using a plugin as an IPM profile" on page 501

Manual operation: See "Plug-in" on page 153

DESTination:PLUGin:VARiable:SELect:ID <Id>
SCENario:PDW:PLUGin:VARiable:SELect:ID <Id>

Selects a plugin variable ID.

Parameters:

<Id> float

*RST: 0

DESTination:PLUGin:VARiable:RESet
SCENario:PDW:PLUGin:VARiable:RESet

Resets the variable values to the defaults.

Usage: Event

SCENario:PDW:ENABle <Enable>

Enables generation of Pulse Descripter Word (PDW) reports.

Parameters:

<Enable> ON | OFF | 1 | 0

Example: See Example"Generating reports" on page 501

Manual operation: See "Reporting Enable" on page 388

SCENario:PDW:PATH <Path>

Sets the target directory in that the generated report files are stored.

Parameters:

<Path> string

Example: See Example"Generating reports" on page 501

Manual operation: See "Target Path, Set Path" on page 388

SCENario:PDW:TYPE <Type>

Sets the template used be the reporting function.

Parameters:

<Type> DEFault | TEMPlate | PLUGin | AMMos

Manual operation: See "Type" on page 389

SCENario:PDW:PLUGin:NAME <Name>

Selects and loads a reporting template. This template must exist in the "Plugin" library.

To query a list of available plugins, use the command `PLUGin:CATalog?`.

Parameters:

<Name> string

Example: See Example"Generating reports" on page 501

Manual operation: See "Plugin and plugin variables" on page 391

SCENario:PDW:AMMos:PPDW <Ppdw>

If enabled, the format of the AMMOS file is set to PPDW. Otherwise PDW is assumed.

Parameters:

<Ppdw> ON | OFF | 1 | 0

Example:

```
SCENario:PDW:TYPE AMMos
SCENario:PDW:AMMos:PPDW 1
SCENario:PDW:AMMos:FRAMe 200
SCENario:PDW:AMMos:UTIMe:ENABle 1
SCENario:PDW:AMMos:UTIMe:ISO
SCENario:PDW:AMMos:UTIMe:ISO "2017-05-24T23:46:00"
SCENario:PDW:AMMos:AZIMuth RX
```

Manual operation: See "Format" on page 390

SCENario:PDW:AMMos:FRAMe <Frame>

Sets the frame length.

Parameters:

<Frame> float

Range: 50 to 500

Example: See `SCENario:PDW:AMMos:PPDW` on page 505.

Manual operation: See "Frame Length" on page 391

SCENario:PDW:AMMos:UTIMe:ENABle <Enable>

Defines how the report start time is set.

Parameters:

<Enable> ON | OFF | 1 | 0

0
The reporting start time is time at that the scenario calculation starts.

1
The reporting starts at user-defined moment, set with the command `SCENario:PDW:AMMos:UTIMe:ISO`.

Example: See `SCENario:PDW:AMMos:PPDW` on page 505.

Manual operation: See "Start Time, User Set Value" on page 391

SCENario:PDW:AMMos:UTIMe:ISO <Iso>

Sets the reporting start time, if `SCENario:PDW:AMMos:UTIMe:ENABle1`.

Parameters:

<Iso> "<YYYY>-<Month>-<DD>T<HH:MM:SS>"

Example: See `SCENario:PDW:AMMos:PPDW` on page 505.

Manual operation: See "Start Time, User Set Value" on page 391

SCENario:PDW:AMMos:AZIMuth <Azimuth>

For `SCENario:PDW:TYPEAMMos`, defines whether the angle of the Rx antenna or the bearing is reported.

Parameters:

<Azimuth> RX | BEARing

Manual operation: See "Azimuth" on page 391

SCENario:PDW:TEMPlate <Template>

Edits the selected template.

Parameters:

<Template> string

Example:

```
SCPI
SCENario:PDW:TEMPlate?
# Pulse Sequencer PDW Report
#
Date:        <ISODATE>
Repository: <REPOSITORY>
Scenario:    <SCENARIO>

          TOA |          RF |           PW |        PA | MF |   MOP |       BW |
           ns |         GHz |           us |       dBm |    |       |      kHz |
----------------------------------------------------------------------
.HDR
 <TOA> | <RF> | <PW> | <PA> |  <MF> | <MOP> | <BW> |
# END OF REPORT
.OPT <TOA:12.0f:-9>
.OPT <RF:9.6f:9>
.OPT <PW:9.3f:-6>
.OPT <PA:6.1f:0>
.OPT <BW:6.0f:3>
```

Manual operation: See "Edit Template" on page 389

24.16 Pulse commands

The following are examples on how to create and configure pulses in remote environment.

Example: Creating an unmodulated pulse

```
SCPI
PULSe:CREate "P_1"
PULSe:CATalog?
// "P_1"
PULSe:SELect "P_1"
PULSe:COMMent "PW=100us"
PULSe:TIME:REFerence FULL
PULse:TIME:WIDTh 100 us
PULSe:TIME:RISE 10 us
PULSe:TIME:FALL 10 us
PULSe:TYPE:RISE LINear
PULSe:TYPE:FALL LINear
PULSe:MOP:ENABle 0
PULSe:MOP:EXCLude:ENABle 0
// remane the pulse
PULSe:NAME "P1"
```

Example: Creating a linear chirp pulse

```
SCPI
PULSe:CREate "LinearChirp"
PULSe:CUSTom 0
PULSe:TIME:RISE 1e-05
PULSe:TIME:WIDTh 0.0001
PULSe:TIME:FALL 1e-05

PULSe:MOP:ENABle 1
PULSe:MOP:TYPE CHIRP
PULSe:MOP:CHIRp:TYPE UP
PULSe:MOP:CHIRp:DEViation 5e+06

// PULSe:MOP:TYPE PWISechirp
// PULSe:MOP:PIECewise:ADD
// PULSe:MOP:PIECewise:SELect 1
// PULSe:MOP:PIECewise:DURation 25
// PULSe:MOP:PIECewise:FREQuency 200
// PULSe:MOP:PIECewise:OFFSet 0
// PULSe:MOP:PIECewise:ADD
// PULSe:MOP:PIECewise:SELect 2
// PULSe:MOP:PIECewise:DURation 25
// PULSe:MOP:PIECewise:FREQuency -400
// PULSe:MOP:PIECewise:OFFSet 10E6
```

Example: Creating an FM step pulse

```
SCPI
PULSe:CREate "FM_Step"
PULSe:TIME:RISE 0
PULSe:TIME:WIDTh 0.000125
PULSe:TIME:FALL 0

PULSe:MOP:ENABle 1
PULSe:MOP:COMMent "fstart = -50 MHz, fend = 50 MHz, Df = 25 MHz (N = 5)"
PULSe:MOP:TYPE FMSTep
PULSe:MOP:FMSTep:ADD
PULSe:MOP:FMSTep:SELect 1
PULSe:MOP:FMSTep:DURation 2.49999993684469e-05
PULSe:MOP:FMSTep:FREQuency -50000000
PULSe:MOP:FMSTep:ADD
PULSe:MOP:FMSTep:SELect 2
PULSe:MOP:FMSTep:DURation 2.49999993684469e-05
PULSe:MOP:FMSTep:FREQuency -25000000
...
```

Example: Creating a BPSK pulse

```
SCPI
PULSe:CREate "Test"
PULSe:MOP:ENABle 1
PULSe:MOP:TYPE BPSK
PULSe:MOP:BPSK:TYPE NORMal
PULSe:MOP:BPSK:SRATe:AUTO 0
PULSe:MOP:BPSK:SRATe 1e+06
PULSe:MOP:BPSK:PHASe 180
PULSe:MOP:BPSK:TTYPe COSine
PULSe:MOP:BPSK:TTIMe 5
```

PULSe:MOP:AM:FREQuency <Frequency>

Sets modulation frequency.

Parameters:

<Frequency> float

Range: 0.001 to 1e+09

Example: See PULSe:MOP:AM:TYPE on page 510

Manual operation: See "AM" on page 97

PULSe:MOP:AM:MDEPth <Mdepth>

Sets the modulation depth.

Parameters:

<Mdepth> float

Range: 0 to 100
Default unit: percent

Example: See PULSe:MOP:AM:TYPE on page 510

Manual operation: See "AM" on page 97

PULSe:MOP:AM:TYPE <Type>

Selects the modulation type.

Parameters:

<Type> STD | LSB | USB | SB

Example:
```
PULSe:MOP:TYPE AM
PULSe:MOP:AM:TYPE STD
PULSe:MOP:AM:FREQuency 50000
PULSe:MOP:AM:MDEPth 50
```

Manual operation: See "AM" on page 97

PULSe:MOP:AMSTep:DURation <Duration>
PULSe:MOP:FMSTep:DURation <Duration>

Sets the step time.

Parameters:

<Duration> float

Range: 0 to 3600
Default unit: s

Example: See Example"Creating an FM step pulse" on page 508

Manual operation: See "FM Step" on page 100

PULSe:MOP:AMSTep:LEVel <Level>

Sets the step level.

Parameters:

<Level> float

Range: -100 to 0

Example: See Example"Creating an FM step pulse" on page 508

Manual operation: See "AM Step" on page 98

PULSe:MOP:FMSTep:FREQuency <Frequency>

Sets the step frequency.

Parameters:

<Frequency> float

Range: -1e+09 to 1e+09

Example: See Example"Creating an FM step pulse" on page 508.

Manual operation: See "FM Step" on page 100

PULSe:MOP:MSK:INVert <Invert>
PULSe:MOP:FSK:INVert <Invert>
PULSe:MOP:ASK:INVert <Invert>

Inverts the modulation.

Parameters:

<Invert> ON | OFF | 1 | 0

Example:

```
PULSe:MOP:TYPE ASK
PULSe:MOP:ASK:MDEPth 100
PULSe:MOP:ASK:INVert 1
PULSe:MOP:ASK:SRATe 1e+06
```

Manual operation: See "ASK" on page 98

PULSe:MOP:ASK:MDEPth <Mdepth>

Sets the modulation depth.

Parameters:

<Mdepth> float

Range: 0 to 100
Default unit: percent

Example: See PULSe:MOP:ASK:INVert on page 511

Manual operation: See "ASK" on page 98

PULSe:MOP:MSK:SRATe <Srate>
PULSe:MOP:ASK:SRATe <Srate>

Sets the symbol rate.

Parameters:

<Srate> float

Range: 1 to 1e+09

Example: See PULSe:MOP:ASK:INVert on page 511

Manual operation: See "ASK" on page 98

PULSe:MOP:BARKer:CODE <Code>

Selects the code sequence.

Parameters:

<Code> R2A | R2B | R3 | R4A | R4B | R5 | R7 | R11 | R13

Example:

```
PULSe:MOP:TYPE BARKer
PULSe:MOP:BARKer:CODE R13
PULSe:MOP:BARKer:TTIMe 10
PULSe:MOP:BARKer:BLANk 1
```

Manual operation: See "Barker" on page 107

PULSe:MOP:BARKer:TTIMe <Ttime>

Sets the transition time.

Parameters:

<Ttime>	float Range: 0 to 50 Default unit: percent
Example:	See PULSe:MOP:BARKer:CODE on page 512
Manual operation:	See "Barker" on page 107

PULSe:MOP:BARKer:BLANk <Blank>

Blanks out the signal during the transition time.

Parameters:

<Blank>	ON | OFF | 1 | 0
Example:	See PULSe:MOP:BARKer:CODE on page 512
Manual operation:	See "Barker" on page 107

PULSe:MOP:BPSK:PHASe <Phase>

Sets the phase.

Parameters:

<Phase>	float Range: 0 to 180 Default unit: degree
Example:	See Example"Creating a BPSK pulse" on page 509
Manual operation:	See "BPSK" on page 109

PULSe:MOP:BPSK:SRATe <Srate>

Sets the symbol rate.

Parameters:

<Srate>	float Range: 1 to 1e+09
Example:	See Example"Creating a BPSK pulse" on page 509
Manual operation:	See "BPSK" on page 109

PULSe:MOP:BPSK:SRATe:AUTO <Auto>

Enables automatic adjusting of the bits in the pulse width.

Parameters:

<Auto>	ON | OFF | 1 | 0
Example:	See Example"Creating a BPSK pulse" on page 509

Manual operation: See "BPSK" on page 109

PULSe:MOP:BPSK:TTIMe <Ttime>

Sets the transition time.

Parameters:

<Ttime> float

Range: 0 to 50
Default unit: percent

Example: See Example"Creating a BPSK pulse" on page 509

Manual operation: See "BPSK" on page 109

PULSe:MOP:BPSK:TTYPe <Ttype>

Selects the transition type.

Parameters:

<Ttype> LINear | COSine

Example: See Example"Creating a BPSK pulse" on page 509

Manual operation: See "BPSK" on page 109

PULSe:MOP:BPSK:TYPE <Type>

Sets the modulation type.

Parameters:

<Type> STANdard | CONStant

Example: See Example"Creating a BPSK pulse" on page 509

Manual operation: See "BPSK" on page 109

PULSe:MOP:CHIRp:TYPE <Type>

Selects the modulation type.

Parameters:

<Type> UP | DOWN | SINE | TRIangular | PIECewise

Example: See Example"Creating a linear chirp pulse" on page 508

Manual operation: See "Linear Chirp" on page 102

PULSe:MOP:CHIRp:DEViation <Deviation>

Sets the modulation deviation.

Parameters:

<Deviation> float

Range: 1 to 1e+09

Example: See Example"Creating a linear chirp pulse" on page 508

Manual operation: See "Linear Chirp" on page 102

PULSe:MOP:FSK:TYPE <Type>

Selects the FSK modulation type.

Parameters:

<Type> FS2 | FS4 | FS8 | FS16 | FS32 | FS64

Example:

```
PULSe:MOP:TYPE FSK
PULSe:MOP:FSK:TYPE FS8
```

Manual operation: See "FSK, 4FSK, 8FSK, 16FSK, 32FSK, 64FSK" on page 101

PULSe:MOP:PIECewise:DURation <Duration>

Set the length of the chirp interval as a percentage of the duration the MOP is applied on.

Parameters:

<Duration> float

Range: 0 to 100

Example: See Example"Creating a linear chirp pulse" on page 508

Manual operation: See "Piecewise Linear Chirp" on page 103

PULSe:MOP:PIECewise:RATE <Rate>

Set the chirp rate.

Parameters:

<Rate> float

Range: -1e+15 to 1e+15
Default unit: Hz/s

Example: See Example"Creating a linear chirp pulse" on page 508

Manual operation: See "Piecewise Linear Chirp" on page 103

PULSe:MOP:PIECewise:OFFSet <Offset>

Offsets the start frequency of the chirp.

Parameters:

<Offset> float

Range: -1e+09 to 1e+09

Example: See Example"Creating a linear chirp pulse" on page 508

Manual operation: See "Piecewise Linear Chirp" on page 103

PULSe:MOP:CCHirp:FREQuency <Frequency>

Set the frequency of the custom chirp.

Parameters:

<Frequency> float

Range: -1e+09 to 1e+09

Example:

```
// custom chirp with 4 frequencies
PULSe:MOP:TYPE CCHIPRP
PULSe:MOP:CCHirp:ADD
PULSe:MOP:CCHirp:COUNt 4
PULSe:MOP:CCHirp:SEL 1
PULSe:MOP:CCHirp:FREQuency 1e+07
PULSe:MOP:CCHirp:SEL 2
PULSe:MOP:CCHirp:FREQuency 5e+06
PULSe:MOP:CCHirp:SEL 3
PULSe:MOP:CCHirp:FREQuency -1e+07
PULSe:MOP:CCHirp:SEL 4
PULSe:MOP:CCHirp:FREQuency 0
```

Manual operation: See "Custom Chirp" on page 104

PULSe:MOP:QAM:TYPE <Type>

Selects the QAM type.

Parameters:

<Type> Q16 | Q32 | Q64 | Q128 | Q256

Example:

```
PULSe:MOP:TYPE QAM
PULSe:MOP:QAM:TYPE Q256
```

Manual operation: See "16QAM, 32QAM, 64QAM, 128QAM, 256QAM" on page 111

PULSe:MOP:ENABle <Enable>

Defines whether a MOP is applied.

Parameters:

<Enable> ON | OFF | 1 | 0

Example: See Example"Creating a linear chirp pulse" on page 508

Manual operation: See "Enable" on page 96

PULSe:MOP:TYPE <Type>

Select the modulation scheme.

Parameters:

<Type> AM | ASK | AMSTep | FM | FSK | FMSTep | CHIRp | PCHirp | BARKer | POLYphase | PLISt | BPSK | QPSK | NOISe | PWISechirp | CCHiprp | PSK8 | QAM | MSK

Example: See Example"Creating a linear chirp pulse" on page 508.

Manual operation: See "MOP Type" on page 97

Manual operation: See Chapter 6.2.3, "Modulation on pulse (MOP) settings", on page 96.

PULSe:MOP:FM:DEViation <Deviation>

Sets the modulation deviation.

Parameters:

<Deviation> float

Range: 0.1 to 1e+09
Default unit: Hz

Manual operation: See "FM" on page 99

PULSe:MOP:FM:FREQuency <Frequency>

Sets the modulation frequency.

Parameters:

<Frequency> float

Range: 0.002 to 1e+09

Manual operation: See "FM" on page 99

PULSe:MOP:8FSK:DEViation <Deviation>
PULSe:MOP:4FSK:DEViation <Deviation>
PULSe:MOP:FSK:DEViation <Deviation>

Sets the modulation deviation.

Parameters:

<Deviation> float

Range: 0.001 to 1e+09
Default unit: Hz

Manual operation: See "FSK, 4FSK, 8FSK, 16FSK, 32FSK, 64FSK" on page 101

PULSe:MOP:8PSK:SRATe <Srate>
PULSe:MOP:8FSK:SRATe <Srate>

PULSe:MOP:4FSK:SRATe <Srate>
PULSe:MOP:QAM:SRATe <Srate>
PULSe:MOP:FSK:SRATe <Srate>

Sets the symbol rate of the modulated signal.

Parameters:

<Srate> float

Range: 1 to 1e+09

Manual operation: See "FSK, 4FSK, 8FSK, 16FSK, 32FSK, 64FSK" on page 101

PULSe:MOP:NOISe:BWIDth <Bwidth>

Sets the bandwidth.

Parameters:

<Bwidth> float

Range: 1 to 1e+09
Default unit: Hz

Manual operation: see Chapter 6.2.3.7, "Noise", on page 112

PULSe:MOP:PCHirp:COEFficient <Coefficient>

Sets the coefficient of the chirp polynomial.

Parameters:

<Coefficient> float

Range: -1e+32 to 1e+32

Manual operation: See "Polynomial Chirp" on page 106

PULSe:MOP:PCHirp:TERM <Term>

Sets the term of the chirp polynomial.

Parameters:

<Term> float

Range: 0 to 32

Manual operation: See "Polynomial Chirp" on page 106

PULSe:MOP:PLISt:VALue <Value>

Sets the phase.

Parameters:

<Value> float

Range: -180 to 180
Default unit: degree

Manual operation: See "Custom Phase" on page 109

PULSe:MOP:POLY:LENGth <Length>

Sets the polyphase length (code order).

Parameters:

<Length> integer

Range: 1 to 100

Manual operation: See "Poly Phase" on page 108

PULSe:MOP:POLY:TYPE <Type>

Selects the modulation type.

Parameters:

<Type> FRANk | P1 | P2 | P3 | P4

Manual operation: See "Poly Phase" on page 108

PULSe:MOP:QPSK:SRATe <Srate>

Sets the symbol rate.

Parameters:

<Srate> float

Range: 1 to 1e+09

Manual operation: See "QPSK" on page 110

PULSe:MOP:QPSK:TYPE <Type>

Selects the modulation type.

Parameters:

<Type> NORMal | OQPSk | DQPSk | ASOQpsk | BSOQpsk | TGSoqpsk

Manual operation: See "QPSK" on page 110

PULSe:MOP:QPSK:SOQPsk:IRIG <Irig>

Enables differential encoding according to the telemetry standard IRIG 106-04.

Parameters:

<Irig> ON | OFF | 1 | 0

Example:

```
PULSe:MOP:QPSK:TYPE TGSOQPSK
PULSe:MOP:QPSK:SRATe 1e+06
PULSe:MOP:QPSK:SOQPsk:IRIG 1
```

Manual operation: See "QPSK" on page 110

PULSe:TIME:FALL <Fall>
PULSe:TIME:RISE <Rise>

Sets the transition time of the rising and falling edges.

Parameters:

<Rise> float

Range: 0 to 3600

Example: See Example"Creating a linear chirp pulse" on page 508

Manual operation: See "Pulse Shape Settings" on page 95

PULSe:TIME:WIDTh <Width>

Sets the time during that the pulse is on top power.

Parameters:

<Width> float

Range: 0 to 3600
Default unit: s

Example: See Example"Creating a linear chirp pulse" on page 508

Manual operation: See "Pulse Shape Settings" on page 95

PULSe:SETTings <Settings>

Switches between the displayed settings.

Setting parameters:

<Settings> TIMing | MOP | MKR | GENeral

Example:

```
PULSe:SETTings MOP
PULSe:PREView:MODE MOP
PULSe:PREView:MOP IQ
```

Usage: Setting only

PULSe:PREView:MODE <Mode>

Switches between the envelope and modulation graphs.

Setting parameters:

<Mode> ENVelope | MOP

Example: See `PULSe:SETTings` on page 520

Usage: Setting only

Manual operation: See "Envelope graph" on page 112

PULSe:PREView:MOP <Mop>

Sets the displayed modulation characteristics.

Setting parameters:
<Mop> IQ | PHASe | FREQuency

Example: See `PULSe:SETTings` on page 520

Usage: Setting only

Manual operation: See "Time domain display" on page 114

24.17 Receiver commands

Example: Creating receivers

```
RECeiver:CREate "Receiver DF"
RECeiver:CATalog?
// "Receiver1", "Receiver DF"
RECeiver:SELect?
// Receiver DF
RECeiver:NAME?
// Receiver DF
RECeiver:MODel?
// INTerfero

RECeiver:ANTenna:ADD
RECeiver:ANTenna:SELect 1
RECeiver:ANTenna:ALIas "Antenna 1"
RECeiver:ANTenna:POSition:X 0.14
RECeiver:ANTenna:POSition:Y 0
RECeiver:ANTenna:POSition:RADius 0.14
RECeiver:ANTenna:POSition:ANGLe 0
RECeiver:ANTenna:POSition:HEIGht 0
ANTenna:CATalog?
// "Isotropic","My_PencilBeam","My_Cosecant","My_PlanarAntenna","CustomAntennPattern"
RECeiver:ANTenna:PATTern "My_PlanarAntenna"
RECeiver:ANTenna:SCAN "NONE"
RECeiver:ANTenna:GAIN 2
RECeiver:ANTenna:DIRection:AWAY 1
RECeiver:ANTenna:DIRection:ELEVation 0
RECeiver:ANTenna:DIRection:AZIMuth?
// 0

RECeiver:ANTenna:ADD
RECeiver:ANTenna:SELect 2

RECeiver:ANTenna:ALIas "Antenna 2"
RECeiver:ANTenna:POSition:X 0
RECeiver:ANTenna:POSition:Y 0.14
RECeiver:ANTenna:POSition:RADius 0.14
RECeiver:ANTenna:POSition:ANGLe 90
RECeiver:ANTenna:POSition:HEIGht 0
RECeiver:ANTenna:PATTern "My_PlanarAntenna"
RECeiver:ANTenna:SCAN "NONE"
RECeiver:ANTenna:GAIN 0
RECeiver:ANTenna:DIRection:AWAY 1
RECeiver:ANTenna:DIRection:ELEVation 0
RECeiver:ANTenna:DIRection:AZIMuth?
// 90
```

See also Example"Moving emitters and receivers" on page 565.

RECeiver:ANTenna:ALIas <Alias>

Sets an alias name for the selected antenna element.

Parameters:

<Alias> string

Example: See Example"Creating receivers" on page 522.

Manual operation: See "Antenna Configuration" on page 266

RECeiver:MODel <Model>

Sets the receiver model.

Parameters:

<Model> INTerfero | TDOA | COMBined

For details, see "Model" on page 265.

INTerfero
Interferometer
Calculates the relative phase difference between the single antenna ports.

TDOA
Time difference of arrival
Calculates the absolute time of arrival (TOA) of the incoming signal for each antenna.

COMBined
Calculates the relative phases between the antenna ports and calculates the the individual TOAs for each antenna port.

Example: See Example"Creating receivers" on page 522.

Manual operation: See "Model" on page 265

RECeiver:ANTenna:POSition:X <X>
RECeiver:ANTenna:POSition:Y <Y>

Sets the antenna element position as X and Y values, relative to the receiver origin.

Parameters:

<Y> float

Range: -1e+06 to 1e+06

Example: See Example"Creating receivers" on page 522.

Manual operation: See "Position" on page 268

RECeiver:ANTenna:POSition:RADius <Radius>

Sets the distance from the antenna element to the receiver origin.

Parameters:

<Radius> float

Range: 0 to 1e+06

Example: See Example"Creating receivers" on page 522.

Manual operation: See "Position" on page 268

RECeiver:ANTenna:POSition:ANGLe <Angle>

Sets the antenna element position as an angle offset from the X-axis.

Parameters:

<Angle> float

Range: 0 to 360

Example: See Example"Creating receivers" on page 522.

Manual operation: See "Position" on page 268

RECeiver:ANTenna:POSition:HEIGht <Height>

Sets the antenna element height, relative to the receiver origin.

Parameters:

<Height> float

Range: -1e+06 to 1e+06

Example: See Example"Creating receivers" on page 522.

Manual operation: See "Position" on page 268

RECeiver:ANTenna:PATTern <Pattern>

Assigns an existing antenna pattern, see `ANTenna:CATalog?`.

Parameters:
<Pattern> string

Example: See Example"Creating receivers" on page 522.

Manual operation: See "Properties" on page 267

RECeiver:ANTenna:SCAN <Scan>

Sets the antenna scan.

Parameters:
<Scan> string

Example: See Example"Creating receivers" on page 522.

Manual operation: See "Properties" on page 267

RECeiver:ANTenna:GAIN <Gain>

Sets the gain of the individual antenna element.

Parameters:
<Gain> float

Range: -120 to 120

Example: See Example"Creating receivers" on page 522.

Manual operation: See "Properties" on page 267

RECeiver:ANTenna:DIRection:AWAY <Away>

Sets the azimuth automatically, so that the beam axis is radial to the receiver origin.

Parameters:
<Away> ON | OFF | 1 | 0

Example: See Example"Creating receivers" on page 522.

Manual operation: See "Properties" on page 267

RECeiver:ANTenna:DIRection:AZIMuth <Azimuth>

Turns the antenna beam axis.

Parameters:
<Azimuth> float

Range: 0 to 360

Example: See Example"Creating receivers" on page 522.

Manual operation: See "Properties" on page 267

RECeiver:ANTenna:DIRection:ELEVation <Elevation>

Turns the antenna beam axis.

Parameters:

<Elevation> float

Range: -90 to 90

Example: See Example"Creating receivers" on page 522.

Manual operation: See "Properties" on page 267

24.18 Repository commands

Do not change the default folder structure and the repository file names.

Changing file paths and file names may lead to data loss and irreparable faults in the data sources.

Example: Working with repositories

```
SCPI
// create new empty repository
REPository:CREate "Repository for tests"
REPository:CATalog?
// line brakes added to improve the readability
//"Repository for tests",
//"C:\Users\Public\Documents\Rohde-Schwarz\Pulse Sequencer\Repositories",
//"K32 and K39 Tests",
//"C:\Users\Public\Documents\Rohde-Schwarz\Pulse Sequencer\Repositories",
//"Rep","C:\Users\Public\Documents\Rohde-Schwarz\Pulse Sequencer\Repositories"
REPository:SAVE
REPository:SELect "Repository for tests"
REPository:AUTHor "My Company Name"
REPository:DATE?
REPository:SECurity LEV1
REPository:VERSion "1.1"
REPository:COMPlexity EMITter
REPository:XPOL:ATTenuation 30
REPository:SAVE
REPository:PATH?
// C:\Users\Public\Documents\Rohde-Schwarz\PulseSequencer\Repositories\20140722_113521
REPository:FILename?
// C:\Users\Public\Documents\Rohde-Schwarz\PulseSequencer\Repositories
   \20140722_113521\Config.ps_rep
REPository:ACCess?
// RW,Login=no,Pass=no,Uname=testuser
REPository:REMove "Rep"
```

```
REPManager:PATH:LIST?
// "C:\Users\Public\Documents\Rohde-Schwarz\PulseSequencer\Repositories"
REPManager:PATH:ADD "C:\_ps_files"
REPManager:PATH:LIST?
// "C:\Users\Public\Documents\Rohde-Schwarz\PulseSequencer\Repositories","C:\_ps_files"
REPManager:CATalog?
// "Repository for tests",
   "C:\Users\Public\Documents\Rohde-Schwarz\PulseSequencer\Repositories",
// "K32 and K39 Tests",
   "C:\Users\Public\Documents\Rohde-Schwarz\PulseSequencer\Repositories",
// "My_Tests","C:\_ps_files"
REPManager:LOAD "K32 and K39 Tests"
REPManager:EXPort "K32 and K39 Tests","c:\_ps_files\my.psarch"
REPManager:DELete "My_Tests"
REPManager:CATalog?
// "Repository for tests",
   "C:\Users\Public\Documents\Rohde-Schwarz\PulseSequencer\Repositories",
// "K32 and K39 Tests",
   "C:\Users\Public\Documents\Rohde-Schwarz\PulseSequencer\Repositories"
```


REPository:ACCess?

Queries information on the access rights of the current user.

Return values:

<Access> <permission>,<login>,<pass>,<Uname>

<permission>
Permission of the current user, for example `RW` (read-write)

<login>,<pass>
`Login/Pass=No`: Password not required
`Login/Pass=Yes`: Password required

<Uname>
User name of the current user

Example: See Example"Working with repositories" on page 526

Usage: Query only

Manual operation: See "Users" on page 400

REPository:AUTHor <Author>

Enters information on the author.

Parameters:
<Author> string

Example: See Example"Working with repositories" on page 526

Manual operation: See "Info" on page 61

REPository:DATE <Date>

Queries the creation data.

Parameters:
<Date> string

Example: See Example"Working with repositories" on page 526

Manual operation: See "Info" on page 61

REPository:FILename?

Queries the file name of the repository archive.

Return values:
<Filename> string
File path, incl. file name, and extension

Example: See Example"Working with repositories" on page 526

Usage: Query only

Manual operation: See "Storage" on page 62

REPository:PATH?

Queries the directory in that the repository archive is stored.

Return values:
<Path> string

Example: See Example"Working with repositories" on page 526

Usage: Query only

Manual operation: See "Storage" on page 62

REPository:SAVE

Stores the repository archive.

To query the storage location, use the command `REPository:PATH?`.

Example: See Example"Working with repositories" on page 526

Usage: Event

Manual operation: See "Storage" on page 62

REPository:SECurity <Security>

Sets the security level.

Parameters:

<Security> LEV0 | LEV1 | LEV2 | LEV3 | LEV4

Example: See Example"Working with repositories" on page 526

Manual operation: See "Classification" on page 61

REPository:VERSion <Version>

Sets the repository version.

Parameters:

<Version> string

Example: See Example"Working with repositories" on page 526

Manual operation: See "Info" on page 61

REPository:XPOL:ATTenuation <Attenuation>

Sets the attenuation used to calculate the cross-polarized antenna patterns.

Parameters:

<Attenuation> float

*RST: 0

Example: See Example"Working with repositories" on page 526

Manual operation: See "Antenna Cross Polarization > Attenuation" on page 62

REPository:UUID?

Queries the repository's Universally Unique Identifier (UUID).

Return values:

<Uuid> string

Example: `REPository:UUID?`
`// 89d00568-e9d7-4470-8b8d-2179336da541`

Usage: Query only

REPManager:CATalog?

Queries available repository elements in the database.

Return values:

<Catalog> "<RepositryName>","<path>"

`<RepositryName>` is the name of the repository as defined with the command `REPository:CREate`
`<Path>` is the compete file path

Example: See Example"Working with repositories" on page 526

Usage: Query only

Manual operation: See "Discovered Repositories on the Mass Storage" on page 65

REPManager:LOAD <RepName>[,<Path>[,<Username>,<Passwd>]]

Loads the selected repository to the workspace.

If more than one repository with the same name exist, loaded is the first repository with a name match.

To query the available repository elements in the database, use the command `REPository:CATalog?`.

Setting parameters:

<RepName> string

Repository name, as configured in the workspace.

<Path> string

Compete file path, as queried with the command `REPManager:PATH:LIST?`.
The `<Path>` must be specified, if `<Username>` and `<Passwd>` are used.

<Username> string

Required if the repository is password protected

<Passwd> string

Required if the repository is password protected

Example: See Example"Working with repositories" on page 526

Usage: Setting only

Manual operation: See "Load" on page 66

REPManager:DELete <RepName>[,<Path>[,<Username>,<Passwd>]]

Deletes the entire repository from the permanent mass storage.

Setting parameters:

<RepName> string

Repository name, as configured in the workspace.
If more than one repository with the same name exists, the `<Path>` must be specified.

<Path> string

Compete file path, as queried with the command `REPManager:PATH:LIST?`.
The `<Path>` must be specified, if the `<RepName>` is not unique and if `<Username>` and `<Passwd>` are used.

<Username> string

Required if the repository is password protected

<Passwd> string

Required if the repository is password protected

Example: See Example"Working with repositories" on page 526

Usage: Setting only

Manual operation: See "Delete" on page 66

REPManager:EXPort <RepName>[,<Path>],<PSArchiveFile>

Exports the selected repository file to an archive file.

Setting parameters:

<RepName> string

Repository name, as configured in the workspace.

<Path> string

Compete file path, as queried with the command `REPManager:PATH:LIST?`.

<PSArchiveFile> Complete file path, incl. file name, and extension (`*.psarch`).

Example: See Example"Working with repositories" on page 526.

Usage: Setting only

Manual operation: See "Export" on page 66

REPManager:IMPort <Import>

Imports a repository from a directory.

Setting parameters:

<Import> string

Usage: Setting only

Manual operation: See "Import" on page 66

REPManager:PATH:ADD <Add>

Add the selected directory.

Setting parameters:

<Add> string

Complete file path

Example: See Example"Working with repositories" on page 526

Usage: Setting only

Manual operation: See "Add Path" on page 66

REPManager:PATH:DELete <Delete>

Removes the selected file path.

Setting parameters:

<Delete> string

File path

Example: See Example"Working with repositories" on page 526

Usage: Setting only

Manual operation: See "Remove Path" on page 67

REPManager:PATH:LIST?

Queries the directory in that the repository files are stored.

Return values:

<List> string

Compete file path

Example: See Example"Working with repositories" on page 526

Usage: Query only

24.19 Scenario commands

Example: Creating simple pulse train scenario

```
SCPI
SCENario:CREate "SimplePulseTrain"
SCENario:TYPE SEQuence
SCENario:ID?
// 4
SCENario:OUTPut:FREQuency 6e+09
SCENario:OUTPut:LEVel -10
SCENario:OUTPut:RESet:ENABle 1
SCENario:OUTPut:RUNMode CONTinious
SEQuence:CATalog?
// "My_S1_PT","My_S2_S1-F1_S1-F2","My_PT_PRI-Stagger","My_PT_F-Hops",
// "My_WV_Seq","My_TestSequence"
SCENario:SEQuence "My_S1_PT"
PLUGin:CREate
PLUGin:LOAD "C:/Export-PDW.dll"
DESTination:ADD "Destination 1"
DESTination:PLUGin:NAME "Plugin 1"
SCENario:DESTination "Destination 1"
SCENario:OUTPut:DURation:MODE AUTO
// SCENario:OUTPut:DURation:TIME 0.10
SCENario:OUTPut:THReshold -100
SCENario:OUTPut:RECall:ENABle 1
SCENario:CALCulate
SCENario:STATe?
// RUN
SYSTem:PROGress?
// 74
SYSTem:PROGress?
// 100
SCENario:CACHe:VOLatile:VALid?
// 1
SCENario:CACHe:VOLatile:RELease
// SCENario:CACHe:VOLatile:CLEar
SCENario:STARt
```

Example: Creating sequence collection scenario

```
SCPI
SCENario:CREate "My_PT_Collection"
SCENario:TYPE CSEQuence
SCENario:CSEQuence:ADD
SCENario:CSEQuence:SELect 1
SCENario:CSEQuence:ALIas "PT"
SCENario:CSEQuence "My_S2_S1-F1_S1-F2"
SCENario:CSEQuence:ADD
SCENario:CSEQuence:SELect 2
SCENario:CSEQuence:ALIas "F-Hops"
SCENario:CSEQuence "My_PT_F-Hops"
SCENario:CSEQuence:CURRent 1
...
SCENario:CALCulate
```

Example: Creating a simple emitter scenario

```
SCPI
SCENario:CREate "SimpleEmitterScenario"
SCENario:TYPE EMITter
EMITter:CATalog?
// "My_EmitterGuidance","TestEmitter"
SCENario:EMITter "My_EmitterGuidance"
SCENario:EMITter:MODE 2
SCENario:EMITter:MODE:BEAM 1
SCENario:EMITter:DIRection:YAW -10
SCENario:EMITter:DIRection:PITCh -5
...
SCENario:DESTination "My Destination"
SCENario:CALCulate
```

Example: Creating a scenario with multiple emitters and background emitters

```
SCPI
SCENario:CREate "LocalizedEmitters"
SCENario:TYPE LOCalized
EMITter:CATalog?
// "My_EmitterGuidance","TestEmitter"
ANTenna:CATalog?
// "Isotropic","My_PencilBeam","My_Cosecant","My_PlanarAntenna","Testantenna"
SCAN:CATalog?
// "My_RasterScan","My_Circular","Test Antenna Scan"

SCENario:LOCalized:RECeiver:ANTenna "My_Cosecant"
SCENario:LOCalized:RECeiver:SCAN "My_Circular"
SCENario:LOCalized:RECeiver:GAIN 5
SCENario:LOCalized:RECeiver:HEIGht 1
SCENario:LOCalized:RECeiver:DIRection:YAW 0
SCENario:LOCalized:RECeiver:DIRection:PITCh -5

SCENario:LOCalized:ADD
SCENario:LOCalized:SELect?

SCENario:LOCalized:ADD
SCENario:LOCalized:SELect 6
SCENario:LOCalized:TYPE EMITter
SCENario:LOCalized:ALIas "E4"
SCENario:LOCalized:EMITter "My_EmitterGuidance"
SCENario:LOCalized:EMITter:MODE 1
SCENario:LOCalized:EMITter:MODE:BEAM 2
SCENario:LOCalized:DIRection:TRACk 1
SCENario:LOCalized:DIRection:PITCh 0
SCENario:LOCalized:DIRection:YAW -87.8901
SCENario:LOCalized:LOCation:EAST 253.448
SCENario:LOCalized:LOCation:NORTh -6879.31
SCENario:LOCalized:DISTance?
// 6883.98


SCENario:LOCalized:ADD
SCENario:LOCalized:SELect 7
SCENario:LOCalized:TYPE Emitter
SCENario:LOCalized:ALIas "PT"
SEQuence:CATalog?
// "My_S2_S1-F1_S1-F2","My_PT_PRI-Stagger","My_PT_F-Hops",
// "My_TestSequence","My_PulseTrain_Seq"
SCENario:LOCalized:SEQuence "My_PulseTrain_Seq"

...
SCENario:CALCulate
```

Example: Direction finding scenario

```
SCENario:NAME "DF"
SCENario:DF:RECeiver:MOVement:VEHicle STATIONARY
SCENario:DF:RECeiver "Receiver DF"
SCENario:DF:LOCation:REC:PMODe STATIC
SCENario:DF:RECeiver:LATitude -3.781666183E+01
SCENario:DF:RECeiver:LONGitude 1.4496665955E+02
SCENario:DF:RECeiver:HEIGht 1.E+02
SCENario:DF:RECeiver:DIRection:YAW 0.E+00
SCENario:DF:RECeiver:DIRection:PITCh 0.E+00
SCENario:DF:RECeiver:DIRection:ROLL 0.E+00

SCENario:DF:TYPE EMITTER
SCENario:DF:ALIas "TestEmitter"
SCENario:DF:EMITter "TestEmitter"
SCENario:DF:MOVement:VEHicle
SCENario:DF:MCHG:STATe 0
SCENario:DF:EMITter:MODE 1
SCENario:DF:EMITter:MODE:BEAM 1
SCENario:DF:EMITter:MODE:TRACkrec 1
SCENario:DF:LOCation:PMODe MOVING
SCENario:DF:LOCation:EAST 3.998600097656E+03
SCENario:DF:LOCation:NORTh 6.902350097656E+03
SCENario:DF:LOCation:HEIGht 0.E+00
SCENario:DF:DISTance 7.977545898438E+03
SCENario:DF:LOCation:AZIMuth 3.008413852012E+01
SCENario:DF:LOCation:ELEVation 0.E+00
SCENario:DF:MOVement:SPEed 1.E-01
SCENario:DF:MOVement:ACCeleration 0.E+00
SCENario:DF:MOVement:EAST 0.E+00
SCENario:DF:MOVement:NORTh 0.E+00
SCENario:DF:MOVement:HEIGht 0.E+00
SCENario:DF:MOVement:RMODe ONEWAY
...
```

Example: Enabling mode changing

The following is a simple example on how to enable 3 modes for the emitter with alias E4 in the example Example"Creating a scenario with multiple emitters and background emitters" on page 535.

```
SCENario:LOCalized:SELect 6
SCENario:LOCalized:ALIas?
// "E4"
SCENario:LOCalized:TYPE?
// EMITter
SCENario:LOCalized:EMITter?
// "My_EmitterGuidance"
// enable mode change and add 3 entries
SCENario:LOCalized:MCHG:STATe 1
SCENario:LOCalized:MCHG:ADD
SCENario:LOCalized:MCHG:ADD
SCENario:LOCalized:MCHG:COUNt?
// 3
// start configuration of the last (entry#3)
SCENario:LOCalized:MCHG:SELect 3
// configure the stop time first
SCENario:LOCalized:MCHG:STOP 10
SCENario:LOCalized:MCHG:STARt 5
SCENario:LOCalized:EMITter:MODE 2
SCENario:LOCalized:EMITter:MODE:BEAM 2
// configure entry#2
SCENario:LOCalized:MCHG:SELect 2
// set stop time of entry#2 = start time of entry#3
SCENario:LOCalized:MCHG:STOP 5
SCENario:LOCalized:MCHG:STARt 3
SCENario:LOCalized:EMITter:MODE 2
SCENario:LOCalized:EMITter:MODE:BEAM 1
// configure entry#1
SCENario:LOCalized:MCHG:SELect 1
// set stop time of entry#1 = start time of entry#2
SCENario:LOCalized:MCHG:STOP 3
SCENario:LOCalized:MCHG:STARt 0
SCENario:LOCalized:EMITter:MODE 1
SCENario:LOCalized:EMITter:MODE:BEAM 1

// SCENario:LOCalized:MCHG:CLEar
```

Example: Configuring position steps

The following is a simple example on how to enable 3 positions for the emitter with alias E4.

```
SCENario:LOCalized:SELect 6
SCENario:LOCalized:ALIas?
// "E4"
SCENario:LOCalized:TYPE?
// EMITter
SCENario:LOCalized:EMITter?
// "My_EmitterGuidance"
// enable position steps and add 2 entries
SCENario:LOCalized:PSTep:STATe 1
SCENario:LOCalized:PSTep:ADD
SCENario:LOCalized:PSTep:ADD
SCENario:LOCalized:PSTep:COUNt?
// 2
// position step#1 is already defiend

SCENario:LOCalized:PSTep:SELect 2
SCENario:LOCalized:DIRection:YAW -150
SCENario:LOCalized:LOCation:EAST 4.439169921875E+03
SCENario:LOCalized:LOCation:NORTh 7.702899902344E+03

SCENario:LOCalized:PSTep:SELect 3
SCENario:LOCalized:DIRection:TRACk 1
SCENario:LOCalized:LOCation:EAST 2.323706054688E+03
SCENario:LOCalized:LOCation:NORTh 4.177126464844E+03
```

Example: Configuring different modes

The following is a simple example on how to enable emitter with different modes.

```
SCENario:LOCalized:EMITter "TestEmitter"
SCENario:LOCalized:EMITter:STATe:ENABle 1
SCENario:LOCalized:EMITter:STATe:ADD
SCENario:LOCalized:EMITter:STATe:SELect 1
SCENario:LOCalized:EMITter:STATe:DURation 1
SCENario:LOCalized:EMITter:STATe:VALue 1
SCENario:LOCalized:EMITter:STATe:ADD
SCENario:LOCalized:EMITter:STATe:COUNt?
// 2
SCENario:LOCalized:EMITter:STATe:SELect 2
SCENario:LOCalized:EMITter:STATe:DURation 2
SCENario:LOCalized:EMITter:STATe:VALue 0

SCENario:LOCalized:EMITter:STATe:LOOP 1
```

Example: Creating and configuring PDW list scenario

The following is a simple example on how to create a PDW list scenario.

```
// create scenario
SCENario:TYPE PDW
SCENario:CREate
SCENario:NAME "PDW List"
```

See Example"PDW import (unmodulated pulses)" on page 490.

```
// repeat the same steps for all your PDW lists

// configure the PDW list scenario
SCENario:CPDW:ADD
SCENario:CPDW:SELect 1
SCENario:CPDW:ALIas "PDW Custom"
SCENario:CPDW:NAME "PDW"
SCENario:CPDW:LVABs 0
SCENario:CPDW:FREQ 3e+9

SCENario:CPDW:ADD
SCENario:CPDW:SELect 2
SCENario:CPDW:ALIas "My PDW"
SCENario:CPDW:NAME "PDW_2"
SCENario:CPDW:LVABs -3
SCENario:CPDW:FREQ 3e+9

SCENario:CPDW:INTerleaving 0

// calculate the multiplexed signal of the two PDW lists
SCENario:CPDW:INTerleaving 1
SCENario:OUTPut:THReshold -1.E+02
// set the priority of each PSW list
// the higher the number the higher the priority
SCENario:CPDW:SELect 1
SCENario:CPDW:ENABle 1
SCENario:CPDW:DELay 0
SCENario:CPDW:PRIority 10
SCENario:CPDW:LEVel 0
SCENario:CPDW:LDELay 10
SCENario:CPDW:GROup:CATalog?
// "Default","Group 2"
SCENario:CPDW:GROup "Default"

SCENario:CPDW:SELect 2
SCENario:CPDW:ENABle 1
SCENario:CPDW:DELay 0.01
SCENario:CPDW:PRIority 5
SCENario:CPDW:LEVel -5
SCENario:CPDW:LDELay 0
SCENario:CPDW:GROup "Default"
```

```
// convert and calculate the multiplexed signal
SCENario:CPDW:CURRent 1
SCENario:STARt
SCENario:INTerleave
SCENario:CACHe:REPository:ENABle:INTerleave
```

```
// remove the second file from the PDW list
SCENario:CPDW:SELect 2
SCENario:CPDW:DELete
// remove all entries in the PDW list scenario
// SCENario:CPDW:CLEar
```

Example: Creating and configuring PDW list scenario

The following is a simple example on how to configure interleaving groups. It uses as an example a PDW list scenario, for example as configured in Example"Creating and configuring PDW list scenario" on page 539.

```
// create scenario
SCENario:TYPE PDW
SCENario:CREate
SCENario:NAME "PDW List"

SCENario:INTerleaving:GROup:COUNt?
// 1
SCENario:CPDW:GROup:SELect 1
SCENario:CPDW:GROup:ALIas?
// "Default"
SCENario:CPDW:GROup:ADD
SCENario:CPDW:GROup:COUNt?
// 2
SCENario:CPDW:GROup:SELect 2
SCENario:CPDW:GROup:ALIas "Group 2"
SCENario:CPDW:GROup:CATalog?
// "Default","Group 2"

SCENario:CPDW:INTerleaving 1
SCENario:CPDW:INTerleaving:MODE MERGE
SCENario:CPDW:SELect 1
SCENario:CPDW:ENABle 1
SCENario:CPDW:GROup "Default"
SCENario:CPDW:SELect 2
SCENario:CPDW:ENABle 1
SCENario:CPDW:GROup "Default"
SCENario:CPDW:SELect 3
SCENario:CPDW:ENABle 1
SCENario:CPDW:GROup "Group 2"

// the first and the second PDW lists are interleaved;
// the third one belongs to a different interleaving group
```

SCENario:TYPE <Type>

Sets the scenario type.

Parameters:

<Type> SEQuence | CSEQuence | EMITter | CEMitter | LOCalized | DF | PDW

Example: See Example"Creating simple pulse train scenario" on page 533

Manual operation: See "Scenario Type" on page 76

SCENario:ID?

Queries the database identifier of the selected scenario.

Return values:

<Id> float

*RST: 0

Example: See Example"Creating simple pulse train scenario" on page 533

Usage: Query only

SCENario:LOCalized:INTerleaving <Interleaving>
SCENario:DF:INTerleaving <Interleaving>
SCENario:CEMit:INTerleaving <Interleaving>
SCENario:CPDW:INTerleaving <Interleaving>

If enabled, multiple PDW lists are interleaved into a single output file using a priority-based dropping algorithm.

Set the priority with the command `SCENario:CPDW:PRIority`.

Parameters:

<Interleaving> ON | OFF | 1 | 0

Example: See Example"Creating and configuring PDW list scenario" on page 539.

Manual operation: See "Interleaving" on page 77

SCENario:LOCalized:INTerleaving:MODE <Mode>
SCENario:DF:INTerleaving:MODE <Mode>
SCENario:CEMit:INTerleaving:MODE <Mode>

Select the mode for interleaving.

Parameters:

<Mode> DROP | MERGe

DROP
Interleaving uses a priority-based dropping algorithm.

MERGE
Emitters or PDW lists are merged into multiple output files using groups.

Example: See Example"Creating and configuring PDW list scenario" on page 540.

Manual operation: See "Mode" on page 77

SCENario:CALCulate

Starts the signal calculation.

Example: See Example"Creating simple pulse train scenario" on page 533.

Usage: Event

Manual operation: See "Calculate" on page 81

SCENario:STARt

Starts the signal generation.

Example: See Example"Creating simple pulse train scenario" on page 533.

Usage: Event

Manual operation: See "Config" on page 80

SCENario:STOP

Stops the signal calculation.

Example: See Example"Creating simple pulse train scenario" on page 533.

Usage: Event

Manual operation: See "Config" on page 80

SCENario:INTerleave

If `SCENario:CEMit:INTerleaving`|`SCENario:CPDW:INTerleaving`|`SCENario:LOCalized:INTerleaving`|`SCENario:DF:INTerleaving 1`, triggers the calculation of a single output file.

The output file comprises the individual PDWs or pulses, where overlapping PDWs or pulses within an interleaving group are dropped, based on a defined priority.

Example: See Example"Creating and configuring PDW list scenario" on page 539.

Usage: Event

Manual operation: See "Interleave" on page 83

SCENario:STATe?

Queries the current scenario status.

Return values:

<State> IDLE | RUN

Example: See Example"Creating simple pulse train scenario" on page 533

Usage: Query only

Manual operation: See "Config" on page 80

SCENario:ILCache:VOLatile:VALid?
SCENario:CACHe:VOLatile:VALid?

Queries whether the volatile/repository memory contains a valid signal file.

Return values:
<Valid> ON | OFF | 1 | 0

Example: See Example"Creating simple pulse train scenario" on page 533

Usage: Query only

SCENario:ILCache:VOLatile:CLEar
SCENario:CACHe:VOLatile:CLEar

Deletes the files from the volatile/repository memory.

Example: See Example"Creating simple pulse train scenario" on page 533

Usage: Event

Manual operation: See "Results" on page 81

SCENario:DESTination <Destination>

Sets the destination for the signal.

Parameters:
<Destination> string

Use the command `DESTination:PLUGin:VARiable:CATalog` on page 503 to query a list of available export plugins.

SCENario:SEQuence <Sequence>

Assigns a pulse sequence, see `SEQuence:CATalog?` on page 426.

Parameters:
<Sequence> string

Example: See Example"Creating simple pulse train scenario" on page 533

SCENario:SEQuence:CLEar

Usage: Event

SCENario:DF:ALIas <Alias>
SCENario:CEMit:ALIas <Alias>
SCENario:CSEQuence:ALIas <Alias>
SCENario:LOCalized:ALIas <Alias>

Enters an alias name.

Parameters:

<Alias> string

Example: See Example"Creating a scenario with multiple emitters and background emitters" on page 535

Manual operation: See "Alias Name" on page 312

SCENario:CPDW:CURRent <Current>
SCENario:DF:CURRent <Current>
SCENario:LOCalized:CURRent <Current>
SCENario:CSEQuence:CURRent <Current>

Sets the sequence/emitter that is used by the scenario.

Parameters:

<Current> float

Number of the sequence/emitter in the list with multiple sequences

*RST: 1

Example: See Example"Creating sequence collection scenario" on page 534

Manual operation: See "Sequence (Collection) Scenario Settings" on page 85

SCENario:DF:DIRection:YAW <Yaw>
SCENario:EMITter:DIRection:YAW <Yaw>
SCENario:CEMit:DIRection:YAW <Yaw>
SCENario:LOCalized:DIRection:YAW <Yaw>
SCENario:DF:RECeiver:DIRection:YAW <Yaw>
SCENario:LOCalized:RECeiver:DIRection:YAW <Yaw>

Sets the yaw.

Parameters:

<Yaw> float

Range: 0 to 360

Example: See Example"Creating a scenario with multiple emitters and background emitters" on page 535

Manual operation: See "Attitude > Pitch, Yaw, Roll" on page 273

SCENario:DF:DIRection:PITCh <Pitch>
SCENario:DF:RECeiver:DIRection:PITCh <Pitch>
SCENario:EMITter:DIRection:PITCh <Pitch>
SCENario:CEMit:DIRection:PITCh <Pitch>
SCENario:LOCalized:RECeiver:DIRection:PITCh <Pitch>
SCENario:LOCalized:DIRection:PITCh <Pitch>

Sets the pitch.

Parameters:

<Pitch> float

Range: -90 to 90
Default unit: grad

Example: See Example"Creating a scenario with multiple emitters and background emitters" on page 535

Manual operation: See "Attitude" on page 318

SCENario:DF:DIRection:ROLL <Roll>
SCENario:DF:RECeiver:DIRection:ROLL <Roll>
SCENario:EMITter:DIRection:ROLL <Roll>
SCENario:CEMit:DIRection:ROLL <Roll>
SCENario:LOCalized:RECeiver:DIRection:ROLL <Roll>
SCENario:LOCalized:DIRection:ROLL <Roll>

Sets the roll.

Parameters:

<Roll> float

Range: 0 to 360

Manual operation: See "Attitude" on page 318

SCENario:DF:EMITter <Emitter>
SCENario:EMITter <Emitter>
SCENario:CEMit:EMITter <Emitter>
SCENario:LOCalized:EMITter <Emitter>

Assigns an existing emitter or an existing waveform, see `WAVeform:CATalog?` and `EMITter:CATalog?`.

Parameters:

<Emitter> string

Example: See Example"Creating a scenario with multiple emitters and background emitters" on page 535

Manual operation: See "Emitter Name" on page 312

SCENario:DF:EMITter:ENABle <Enable>
SCENario:LOCalized:EMITter:ENABle <Enable>
SCENario:CEMit:EMITter:ENABle <Enable>

In a map-based sceanrio, enable selected item for calculation.

Parameters:

<Enable> ON | OFF | 1 | 0

SCENario:DF:EMITter:MODE <Mode>
SCENario:EMITter:MODE <Mode>
SCENario:CEMit:EMITter:MODE <Mode>
SCENario:LOCalized:EMITter:MODE <Mode>

Set the emitter mode.

Parameters:

<Mode> float

Range: 1 to 32

Example: See Example"Creating a scenario with multiple emitters and background emitters" on page 535

Manual operation: See "Mode" on page 314

SCENario:DF:EMITter:MODE:BEAM <Beam>
SCENario:EMITter:MODE:BEAM <Beam>
SCENario:CEMit:EMITter:MODE:BEAM <Beam>
SCENario:LOCalized:EMITter:MODE:BEAM <Beam>

Sets the used beam of the current mode.

Parameters:

<Beam> float

Range: 1 to 32

Example: See Example"Creating a scenario with multiple emitters and background emitters" on page 535

Manual operation: See "Beam" on page 315

SCENario:DF:DIRection:TRACk <Track>
SCENario:LOCalized:DIRection:TRACk <Track>

Turns the antenna in the direction of the receiver.

Parameters:

<Track> ON | OFF | 1 | 0

Example: See Example"Creating a scenario with multiple emitters and background emitters" on page 535

Manual operation: See "Attitude" on page 318

SCENario:DF:EMITter:MODE:TRACkrec <Trackrec>
SCENario:CEMit:EMITter:MODE:TRACkrec <Trackrec>
SCENario:LOCalized:EMITter:MODE:TRACkrec <Trackrec>

If enabled, the scan follows the receiver automatically.

Parameters:

<Trackrec> ON | OFF | 1 | 0

Manual operation: See "Track Receiver" on page 315

SCENario:DF:DISTance <Distance>
SCENario:LOCalized:DISTance <Distance>

Sets the distance to the receiver.

Parameters:

<Distance> float

Range: 0 to 1e+09
Default unit: m

Example: See Example"Creating a scenario with multiple emitters and background emitters" on page 535

Manual operation: See "East, North, Height/Distance, Azimuth, Elevation" on page 317

SCENario:DF:LOCation:EAST <East>
SCENario:DF:LOCation:NORTh <North>
SCENario:LOCalized:LOCation:EAST <East>
SCENario:LOCalized:LOCation:NORTh <North>

Sets the emitter coordinates.

Parameters:

<North> float

Range: -1e+09 to 1e+09
Default unit: m

Example: See Example"Creating a scenario with multiple emitters and background emitters" on page 535

Manual operation: See "East, North, Height/Distance, Azimuth, Elevation" on page 317

SCENario:DF:LOCation:AZIMuth <Azimuth>
SCENario:LOCalized:LOCation:AZIMuth <Azimuth>

Sets the azimuth.

Parameters:

<Azimuth> float

Range: 0 to 360

Example:

```
SCENario:LOCalized:LOCation:NORTh 7000
SCENario:LOCalized:LOCation:EAST 0
SCENario:LOCalized:LOCation:ALTitude 0

SCENario:LOCalized:DISTance?
// 7000
SCENario:LOCalized:LOCation:AZIMuth?
// 0
SCENario:LOCalized:LOCation:ELEVation?
// 0

SCENario:LOCalized:LOCation:AZIMuth 90
SCENario:LOCalized:LOCation:NORTh?
// 4.28626e-13
SCENario:LOCalized:LOCation:EAST?
// 7000

SCENario:LOCalized:LOCation:ALTitude 10
SCENario:LOCalized:LOCation:ELEVation?
// 0.0818511
SCENario:LOCalized:DISTance?
// 7000.01
```

Manual operation: See "East, North, Height/Distance, Azimuth, Elevation" on page 317

SCENario:DF:LOCation:ELEVation <Elevation>
SCENario:LOCalized:LOCation:ELEVation <Elevation>

Sets the elevation.

Parameters:

<Elevation> float

Range: -90 to 90

Example: See `SCENario:LOCalized:LOCation:AZIMuth` on page 551

Manual operation: See "East, North, Height/Distance, Azimuth, Elevation" on page 317

SCENario:DF:RECeiver:LATitude <Latitude>
SCENario:DF:RECeiver:LONGitude <Longitude>
SCENario:LOCalized:RECeiver:LATitude <Latitude>
SCENario:LOCalized:RECeiver:LONGitude <Longitude>

Sets the latitude/longitude coordinates of the static receiver.

Parameters:

<Longitude> float

Range: -180 to 180

Example: See Example"Direction finding scenario" on page 536.

Manual operation: See "Position > Latitude, Longitude, Altitude, Yaw, Pitch, Roll" on page 263

SCENario:DF:RECeiver:HEIGht <Height>
SCENario:DF:LOCation:HEIGht <Height>
SCENario:LOCalized:RECeiver:HEIGht <Height>
SCENario:LOCalized:LOCation:HEIGht <Height>

Sets the height of the antenna.

Parameters:
<Height> float

Range: -1e+09 to 1e+09

Example: See Example"Creating a scenario with multiple emitters and background emitters" on page 535

Manual operation: See "East, North, Height/Distance, Azimuth, Elevation" on page 317

SCENario:DF:WAVeform:ANTenna <Antenna>
SCENario:LOCalized:RECeiver:ANTenna <Antenna>
SCENario:LOCalized:WAVeform:ANTenna <Antenna>

Assigns an existing antenna pattern, see `ANTenna:CATalog?` on page 425.

Parameters:
<Antenna> string

Example: See Example"Creating a scenario with multiple emitters and background emitters" on page 535

SCENario:DF:WAVeform:SCAN <Scan>
SCENario:LOCalized:RECeiver:SCAN <Scan>
SCENario:LOCalized:WAVeform:SCAN <Scan>

Assigns an existing antenna scan, see `SCAN:CATalog?` on page 426.

Parameters:
<Scan> string

Example: See Example"Creating a scenario with multiple emitters and background emitters" on page 535

SCENario:DF:TYPE <Type>
SCENario:LOCalized:TYPE <Type>

Defines whether an emitter is configured.

Parameters:
<Type> EMITter

Example: See Example"Creating a scenario with multiple emitters and background emitters" on page 535

Manual operation: See "Type" on page 312

SCENario:LOCalized:RECeiver:GAIN <Gain>

Sets the antenna Gain.

Parameters:

<Gain> float

Range: -120 to 120

Example: See Example"Creating a scenario with multiple emitters and background emitters" on page 535

Manual operation: See "Gain" on page 262

SCENario:DF:SEQuence <Sequence>
SCENario:LOCalized:SEQuence <Sequence>

Assigns a sequence to the background signal.

Parameters:

<Sequence> string

Example: See Example"Creating a scenario with multiple emitters and background emitters" on page 535

Manual operation: See "Background Signals" on page 302

SCENario:DF:WAVeform <Waveform>
SCENario:LOCalized:WAVeform <Waveform>

Assigns an existing emitter, see `WAVeform:CATalog?` and `EMITter:CATalog?`.

Parameters:

<Waveform> string

Example: See Example"Creating a scenario with multiple emitters and background emitters" on page 535

Manual operation: See "Signal Source" on page 337

SCENario:DF:WAVeform:LEVel <Level>
SCENario:DF:WAVeform:EIRP <Eirp>
SCENario:LOCalized:WAVeform:LEVel <Level>
SCENario:LOCalized:WAVeform:EIRP <Eirp>

Sets the EIRP of the interferer.

Parameters:

<Eirp> float

Range: -200 to 200

Example: See Example"Creating a scenario with multiple emitters and background emitters" on page 535

SCENario:DF:WAVeform:FREQuency <Frequency>
SCENario:LOCalized:WAVeform:FREQuency <Frequency>

Sets the frequency of the emitter.

Parameters:

<Frequency> float

Range: 1000 to 1e+11

Example: See Example"Creating a scenario with multiple emitters and background emitters" on page 535.

Manual operation: See "Available Background Emitters" on page 336

SCENario:DF:FREQuency?
SCENario:LOCalized:FREQuency?
SCENario:CEMit:FREQuency?
SCENario:CPDW:FREQ <Freq>

Sets the frequency for the selected emitter.

Parameters:

<Freq> float

Range: -1000 to 1e+11

Example: See Example"Creating and configuring PDW list scenario" on page 539.

Manual operation: See "Frequency" on page 369

SCENario:CEMit:FQOFfset <Fqoffset>

Sets the frequency offset for the selected emitter.

Parameters:

<Fqoffset> float

Range: -2e+09 to 2e+09

Manual operation: See "Frequency offset" on page 228

SCENario:CEMit:SCNDelay <Scndelay>

Sets the scan delay for the selected emitter.

Parameters:
<Scndelay> float

Range: -3600 to 3600

Example: `SCENario:CEMit:SCNDelay 100`

Manual operation: See "Scan delay" on page 228

SCENario:CEMit:LVABs <Lvabs>
SCENario:CPDW:LVABs <Lvabs>

Sets the absolute level for the selected PDW list.

Parameters:
<Lvabs> float

Range: -130 to 30

Example: See Example"Creating and configuring PDW list scenario" on page 539.

Manual operation: See "Absolute Level" on page 369

SCENario:DF:SYNChronize:ENABle <Enable>
SCENario:LOCalized:SYNChronize:ENABle <Enable>

Enables synchronized setup.

Parameters:
<Enable> ON | OFF | 1 | 0

SCENario:DF:RECeiver <Receiver>

Selects an existing receiver, see `RECeiver:CATalog?` on page 426.

Parameters:
<Receiver> string

Example:
```
RECeiver:CATalog?
// "Receiver DF","DF"
SCENario:DF:RECeiver "Receiver DF"
```

Manual operation: See "Receiver" on page 270

SCENario:CSEQuence <Csequence>

Select an existing sequence, see `SEQuence:CATalog?` on page 426.

Parameters:
<Csequence> string

Example: See Example"Creating sequence collection scenario" on page 534

Manual operation: See "Sequence (Collection) Scenario Settings" on page 85

SCENario:CSEQuence:VARiable <Variable>

Sets the collection variable.

Parameters:

<Variable> string

Example: `SCENario:CSEQuence:VARiable?`

Manual operation: See "Variables" on page 129

SCENario:DF:MAPS:ENABle <Enable>
SCENario:LOCalized:MAPS:ENABle <Enable>

Enable maps for the selected scenario. This operation cannot be undone.

Parameters:

<Enable> ON | OFF | 1 | 0

SCENario:DF:MAPS:LOAD <Load>
SCENario:LOCalized:MAPS:LOAD <Load>

This command loads a georeferenced map for the selected scenario.

Supported formats:

- `.tif`
- `.tiff`

Setting parameters:

<Load>

Usage: Setting only

SCENario:DF:MCHG:STATe <State>
SCENario:CEMit:MCHG:STATe <State>
SCENario:LOCalized:MCHG:STATe <State>

Enables mode changes.

Parameters:

<State> ON | OFF | 1 | 0

Example: See Example"Enabling mode changing" on page 537.

Manual operation: See "Single Mode" on page 313

SCENario:DF:MCHG:STARt <Start>
SCENario:DF:MCHG:STOP <Stop>
SCENario:CEMit:MCHG:STARt <Start>
SCENario:CEMit:MCHG:STOP <Stop>

SCENario:LOCalized:MCHG:STARt <Start>
SCENario:LOCalized:MCHG:STOP <Stop>

Sets the start and end time per mode entry.

Parameters:

<Stop> float

*RST: 0

Example: See Example"Enabling mode changing" on page 537.

Manual operation: See "Timing" on page 314

SCENario:CEMit:MCHG:CLEar
SCENario:DF:MCHG:CLEar
SCENario:LOCalized:MCHG:CLEar

Removes all defined modes.

Example: See Example"Enabling mode changing" on page 537.

Usage: Event

Manual operation: See " Append, Remove Current, Remove All" on page 314

SCENario:DF:EMITter:STATe:ENABle <Enable>
SCENario:LOCalized:EMITter:STATe:ENABle <Enable>

Enables that an emitter can use on and off states.

Parameters:

<Enable> ON | OFF | 1 | 0

Example: See Example"Configuring different modes" on page 538.

Manual operation: See "Use on/off states with emitter mode" on page 322

SCENario:DF:EMITter:STATe:DURation <Duration>
SCENario:LOCalized:EMITter:STATe:DURation <Duration>

Sets the duration during that the emitter remains in the current state.

Parameters:

<Duration> float

Range: -1e+06 to 1e+06

Example: See Example"Configuring different modes" on page 538.

Manual operation: See "Duration" on page 323

SCENario:DF:EMITter:STATe:VALue <Value>
SCENario:LOCalized:EMITter:STATe:VALue <Value>

Sets the emitter state during the selected period.

Parameters:

<Value> ON | OFF | 1 | 0

Example: See Example"Configuring different modes" on page 538.

Manual operation: See "State" on page 323

SCENario:DF:EMITter:STATe:LOOP <Loop>
SCENario:LOCalized:EMITter:STATe:LOOP <Loop>

Repeats the states definition cyclically.

Parameters:

<Loop> ON | OFF | 1 | 0

Example: See Example"Configuring different modes" on page 538.

Manual operation: See "Execute States in Loop" on page 323

SCENario:CPDW:NAME <Name>

Selects the waveform element, used to import the PDW list.

Query the list of waveform elements with the command `WAVeform:CATalog?`.

Parameters:

<Name> string

Example: See Example"Creating and configuring PDW list scenario" on page 539.

Manual operation: See "PDW List" on page 369

SCENario:CPDW:ALIas <Alias>

Enters an alias name.

Parameters:

<Alias> string

Example: See Example"Creating and configuring PDW list scenario" on page 539.

Manual operation: See "Alias Name" on page 369

SCENario:DF:ENABle <Enable>
SCENario:LOCalized:ENABle <Enable>
SCENario:CEMit:ENABle <Enable>
SCENario:CPDW:ENABle <Enable>

If enabled, the PDW list is included in the output file.

Parameters:

<Enable> ON | OFF | 1 | 0

Example: See Example"Creating and configuring PDW list scenario" on page 539.

Manual operation: See "Enable" on page 371

SCENario:DF:PRIority <Priority>
SCENario:LOCalized:PRIority <Priority>
SCENario:CEMit:PRIority <Priority>
SCENario:CPDW:PRIority <Priority>

Sets the priority of the selected PDW list , where the higher the value the higher the priority.

Parameters:

<Priority> float

Range: 1 to 100

Example: See Example"Creating and configuring PDW list scenario" on page 539.

Manual operation: See "Priority [0 = Highest]" on page 371

SCENario:DF:LDELay <Ldelay>
SCENario:LOCalized:LDELay <Ldelay>
SCENario:CEMit:LDELay <Ldelay>
SCENario:CPDW:LDELay <Ldelay>

If interleaving is enabled, shifts the processing of the selected PDW list in time.

Parameters:

<Ldelay> float

Range: -1e+09 to 1e+09

Example: See Example"Creating and configuring PDW list scenario" on page 539.

Manual operation: See "Time Offset" on page 371

SCENario:DF:LEVel <Level>
SCENario:LOCalized:LEVel <Level>
SCENario:CEMit:LEVel <Level>
SCENario:CPDW:LEVel <Level>

Adds a level offset.

Parameters:

<Level> float

Range: -200 to 0

Example: See Example"Creating and configuring PDW list scenario" on page 539.

Manual operation: See "Level Offset" on page 372

SCENario:DF:GROup <Group>
SCENario:LOCalized:GROup <Group>
SCENario:CEMit:GROup <Group>
SCENario:CPDW:GROup <Group>

Assigns the emitter to one of the available interleaving groups.

Parameters:

<Group> string

Query a list of the alias names of the existing interleaving groups with the command `SCENario:CPDW:GROup:CATalog?`.

Example: See Example"Creating and configuring PDW list scenario" on page 539.

Manual operation: See "Group" on page 372

SCENario:CEMit:GROup:CATalog?
SCENario:DF:GROup:CATalog?
SCENario:LOCalized:GROup:CATalog?
SCENario:CPDW:GROup:CATalog?

Queries the alias names of the configured interleaving groups.

Return values:

<Catalog> string

A list of coma-separated alias names.

Example: See Example"Creating and configuring PDW list scenario" on page 540.

Usage: Query only

Manual operation: See "Select, Insert, Append/Prepend, Remove, Clear" on page 372

SCENario:CEMit:GROup:ALIas <Alias>
SCENario:DF:GROup:ALIas <Alias>
SCENario:LOCalized:GROup:ALIas <Alias>
SCENario:CPDW:GROup:ALIas <Alias>

Sets an alias name for the selected interleaving group.

See also `ASSignment:GROup:SELect` on page 467.

Parameters:

<Alias> string

Example: See Example"Creating and configuring PDW list scenario" on page 540.

Manual operation: See "Alias Name" on page 373

SCENario:OUTPut:CLIPping <Clipping>

Sets a maximum level to limit the dynamic range of the signal. Pulses at levels above this threshold are reduced (clipped) to the configured level.

Manual operation: See "Clipping Level" on page 349

SCENario:OUTPut:DURation:MODE <Mode>

Sets how the waveform duration is defined.

Parameters:

<Mode> AUTO | MANual

AUTO
Sets the simulation time to maximum of sequence, scan or movement duration.

MANual
Sets the simulation time to a fixed value.

Example: See Example"Creating simple pulse train scenario" on page 533

Manual operation: See "Duration" on page 77

SCENario:OUTPut:DURation:TIME <Time>

Sets the duration of the generated waveform.

Parameters:

<Time> float

Range: 1e-06 to 1.8432e+06

Example: See Example"Creating simple pulse train scenario" on page 533.

Manual operation: See "Duration" on page 77

SCENario:OUTPut:DURation:AUTO?

Requires `SCENario:OUTPut:DURation:MODE AUTO`.

Queries the value of the automatically determined signal duration.

Return values:

<Auto> float

Range: 1e-06 to 1.8432e+06

Usage: Query only

Manual operation: See "Duration" on page 77

SCENario:OUTPut:FREQuency <Frequency>

Sets the carrier RF frequency of the generated signal.

Parameters:

<Frequency> float

Range: 1000 to 1e+11

Example: See Example"Creating simple pulse train scenario" on page 533

Manual operation: See "Frequency" on page 78

SCENario:OUTPut:LEVel <Level>

Sets the reference level used by the calculation of the pulse envelope.

Parameters:

<Level> float

Range: -130 to 30

Example: See Example"Creating simple pulse train scenario" on page 533

Manual operation: See "Level" on page 78

SCENario:OUTPut:SUPRess:ENABle <Enable>

Enable to prevent waveform recalculation if the RF frequency is changed.

Parameters:

<Enable> ON | OFF | 1 | 0

Example: `SCENario:OUTPut:SUPRess:ENABle 1`

SCENario:OUTPut:RESet:ENABle <Enable>

Restarts the connected instrument on scenario start.

Parameters:

<Enable> ON | OFF | 1 | 0

Example: See Example"Creating simple pulse train scenario" on page 533

SCENario:OUTPut:RUNMode <Runmode>

Defines the way the generated signal is processed.

Parameters:

<Runmode> CONTinuous | SINGle

Example: See Example"Creating simple pulse train scenario" on page 533

SCENario:CEMit:THReshold <Threshold>
SCENario:CPDW:THReshold <Threshold>

SCENario:LOCalized:THReshold <Threshold>
SCENario:DF:THReshold <Threshold>
SCENario:OUTPut:THReshold <Threshold>

Sets a threshold. Pulses at levels below this threshold are omitted.

Parameters:

<Threshold> float

Range: -100 to 0

Example: See Example"Creating simple pulse train scenario" on page 533

Manual operation: See "Threshold" on page 240

SCENario:OUTPut:MULTithread <Multithread>

Enable to optmize the calculation speed.

Parameters:

<Multithread> ON | OFF | 1 | 0

Example: `SCENario:OUTPut:MULTithread ON`

Manual operation: See "Optimize calculation speed" on page 349

SCENario:OUTPut:MTMode <Mtmode>

If multithreading is enabled with `SCENario:OUTPut:MULTithread` on page 564, sets the mode to use for multithreading.

Parameters:

<Mtmode> AUTO | MANual

Example:

```
SCENario:OUTPut:MTMode?
// SCENario:OUTPut:MTMode AUTO
```

Manual operation: See "Optimize calculation speed" on page 349

SCENario:OUTPut:MTTHreads <Mtthreads>

In manual mode, sets the required number of threads for the signal calculation.

Parameters:

<Mtthreads> float

Range: 0 to 1000

Example:

```
SCENario:OUTPut:MTMode
MANUAL SCENario:OUTPut:MTTHreads?
// SCENario:OUTPut:MTTHreads 16
```

Manual operation: See "Optimize calculation speed" on page 349

SCENario:VOLatile:SEL <Sel>

If several files are created, select the one to be visualized.

Parameters:

<Sel> float

Subsequent number, indicating the files in the volatile memory.

Manual operation: See "Results" on page 81

SCENario:VOLatile:VIEW

If a waveform exists in the volatile memory, opens the "Waveform Viewer" and displays this waveform.

Usage: Event

Manual operation: See "Results" on page 81

24.20 Moving emitters and receivers commands

Example: Moving emitters and receivers

The following is a simple example on how to enable a moving emitter. The same configuration steps apply for the receivers in the localized emitters and direction finding scenarios.

```
SCENario:NAME "Dynamic Scenario"
SCENario:LOCalized:TYPE EMITTER
SCENario:LOCalized:ALIas "Plane"
SCENario:LOCalized:MOVement:VEHicle AIRPLANE
SCENario:LOCalized:LOCation:PMODe MOVING
SCENario:LOCalized:MOVement:TYPE WAYPOINT
SCENario:LOCalized:MOVement:WAYPoint "C:/Users/Public/Documents/Rohde-Schwarz/
Pulse Sequencer/Example Files/Movements/Waypoints/Munich_Flight.xtd"
SCENario:LOCalized:MOVement:RFRame WGS
SCENario:LOCalized:MOVement:RMODe ONEWAY

SCENario:LOCalized:MOVement:VFIle "Big_Aircraft.xvd"
SCENario:LOCalized:MOVement:SMOothening 1

SCENario:LOCalized:MOVement:ATTitude MOTION
SCENario:LOCalized:MOVement:YAW?
// 0
SCENario:LOCalized:MOVement:PITCh?
// 0
SCENario:LOCalized:MOVement:ROLL 1.E-01
SCENario:LOCalized:MOVement:IMPort
```

```
SCENario:LOCalized:ADD
SCENario:LOCalized:TYPE EMITTER
SCENario:LOCalized:ALIas "Ship"
SCENario:LOCalized:MOVement:VEHicle SHIP
SCENario:LOCalized:LOCation:PMODe MOVING

SCENario:LOCalized:MOVement:TYPE ARC
// start position
SCENario:LOCalized:LOCation:EAST 2.386050048828E+03
SCENario:LOCalized:LOCation:NORTh 1.016520019531E+03
SCENario:LOCalized:LOCation:HEIGht 0.E+00
SCENario:LOCalized:DISTance 4.767236816406E+03
SCENario:LOCalized:LOCation:AZIMuth 6.692472065778E+01
SCENario:LOCalized:LOCation:ELEVation 0.E+00
// center position
SCENario:LOCalized:MOVement:SPEed 4.E+01
SCENario:LOCalized:MOVement:ANGLe 1.6988E+02
SCENario:LOCalized:MOVement:EAST 2.3103E+03
SCENario:LOCalized:MOVement:NORTh 1.59021E+03
SCENario:LOCalized:MOVement:RMODe ROUNDTRIP

SCENario:LOCalized:RECeiver:MOVement:VEHicle AIRPLANE
SCENario:LOCalized:LOCation:REC:PMODe MOVING
SCENario:LOCalized:RECeiver:MOVement:TYPE LINE
// start position
SCENario:LOCalized:RECeiver:LATitude 0.E+00
SCENario:LOCalized:RECeiver:LONGitude 0.E+00
SCENario:LOCalized:RECeiver:HEIGht 4.E+03

SCENario:LOCalized:RECeiver:MOVement:SPEed 1.8E+02
SCENario:LOCalized:RECeiver:MOVement:ACCeleration 0.E+00
// end position
SCENario:LOCalized:RECeiver:MOVement:EAST -1.360382727273E+03
SCENario:LOCalized:RECeiver:MOVement:NORTh 5.4037425E+03
SCENario:LOCalized:RECeiver:MOVement:HEIGht 4.E+03
SCENario:LOCalized:RECeiver:MOVement:RMODe CYCLic
```

SCENario:DF:LOCation:PMODe <Pmode>
SCENario:LOCalized:LOCation:PMODe <Pmode>

Sets if the emitter is static or moving.

Parameters:

<Pmode> STATic | STEPs | MOVing

Example: See Example"Moving emitters and receivers" on page 565.

Manual operation: See "Mode" on page 316

SCENario:DF:LOCation:REC:PMODe <Pmode>
SCENario:LOCalized:LOCation:REC:PMODe <Pmode>

Sets if the receiver is static or moving.

Parameters:

<Pmode> STATic | MOVing

Example: See Example"Moving emitters and receivers" on page 565.

Manual operation: See "Mode" on page 272

SCENario:DF:MOVement:TYPE <Type>
SCENario:DF:RECeiver:MOVement:TYPE <Type>
SCENario:LOCalized:RECeiver:MOVement:TYPE <Type>
SCENario:LOCalized:MOVement:TYPE <Type>

Defines the trajectory shape.

Setting the parameter to "NONE" sets the positioning mode to "static".

Parameters:

<Type> LINE | ARC | WAYPoint | TRACe | NONE

Example: See Example"Moving emitters and receivers" on page 565.

Manual operation: See "Trajectory" on page 325

SCENario:DF:LOCation:ALTitude <Altitude>

Use for defining the altitude of a fixed emitter (no movement) on a georeferenced map.

Parameters:

<Altitude> float

Range: -1e+09 to 1e+09

Example:

```
SCEN:DF:LOC:ALT 42
SCEN:DF:LOC:LAT 48.1351
SCEN:DF:LOC:LONG 11.5820
```

Manual operation: See "Longitude, Latitude, Altitude" on page 317

SCENario:LOCalized:LOCation:ALTitude <Altitude>

Sets the altitude of the antenna.

Parameters:

<Altitude> float

Range: -1e+09 to 1e+09

Example: see Example"Creating a scenario with multiple emitters and background emitters" on page 535

Manual operation: See "Longitude, Latitude, Altitude" on page 317

SCENario:DF:LOCation:LATitude <Latitude>
SCENario:DF:LOCation:LONGitude <Longitude>
SCENario:LOCalized:LOCation:LATitude <Latitude>
SCENario:LOCalized:LOCation:LONGitude <Longitude>

Use for defining the position of a fixed emitter (no movement) on a georeferenced map.

Positive values represent °East.

Negative values represent °West.

Parameters:

<Longitude> float

Range: -180 to 180

Example:

```
SCEN:LOC:LOC:ALT 42
SCEN:LOC:LOC:LAT 48.1351
SCEN:LOC:LOC:LONG 11.5820
```

Manual operation: See "Longitude, Latitude, Altitude" on page 317

SCENario:DF:MOVement:ALTitude <Altitude>
SCENario:LOCalized:MOVement:ALTitude <Altitude>

Use for defining the altitude of a moving emitter (line trajectory) on a georeferenced map. Use to define the altitude of the end-points of the line.

Parameters:

<Altitude> float

Range: -1e+09 to 1e+09

Example:

```
SCEN:LOC:MOV:ALT 42
SCEN:LOC:MOV:LAT 48.1351
SCEN:LOC:MOV:LONG 11.5820
```

Manual operation: See "End Position" on page 326

SCENario:DF:MOVement:LATitude <Latitude>
SCENario:DF:MOVement:LONGitude <Longitude>

SCENario:LOCalized:MOVement:LATitude <Latitude>
SCENario:LOCalized:MOVement:LONGitude <Longitude>

Use for defining the movement of an emitter (line trajectory) on a georeferenced map. Use to define the end-points of the line.

Positive values represent °East.

Negative values represent °West.

Parameters:

<Longitude> float

Range: -180 to 180

Example:
```
SCEN:LOC:MOV:ALT 42
SCEN:LOC:MOV:LAT 48.1351
SCEN:LOC:MOV:LONG 11.5820
```

Manual operation: See "End Position" on page 326

SCENario:DF:MOVement:CLATitude <Clatitude>
SCENario:DF:MOVement:CLONgitude <Clongitude>
SCENario:LOCalized:MOVement:CLATitude <Clatitude>
SCENario:LOCalized:MOVement:CLONgitude <Clongitude>

Use for defining the movement of an emitter (arc trajectory) on a georeferenced map. Use to define the center-point of the arc.

Positive values represent °East.

Negative values represent °West.

Parameters:

<Clongitude> float

Range: -180 to 180

Example:
```
SCEN:LOC:MOV:CLAT 48.1351
SCEN:LOC:MOV:CLONG 11.5820
```

Manual operation: See "Center Position" on page 327

SCENario:DF:RECeiver:MOVement:PSTep:SELect <Select>
SCENario:LOCalized:RECeiver:MOVement:PSTep:SELect <Select>

Selects the specified point on a trace trajectory.

Parameters:

<Select> float

*RST: 1

Example:
```
SCEN:LOC:REC:MOV:PST:SEL 7
```

Manual operation: See "Trace Points" on page 279

SCENario:DF:MOVement:SPEed <Speed>
SCENario:DF:RECeiver:MOVement:SPEed <Speed>
SCENario:LOCalized:RECeiver:MOVement:SPEed <Speed>
SCENario:LOCalized:MOVement:SPEed <Speed>

Sets the speed of the moving emitter.

Parameters:

<Speed> float

Range: 0 to 5999

Example: See Example"Moving emitters and receivers" on page 565.

Manual operation: See "Speed" on page 326

SCENario:DF:MOVement:ACCeleration <Acceleration>
SCENario:DF:RECeiver:MOVement:ACCeleration <Acceleration>
SCENario:LOCalized:RECeiver:MOVement:ACCeleration <Acceleration>
SCENario:LOCalized:MOVement:ACCeleration <Acceleration>

Sets the acceleration of the moving emitter.

Parameters:

<Acceleration> float

Range: -100 to 100

Example: See Example"Moving emitters and receivers" on page 565.

Manual operation: See "Acceleration" on page 326

SCENario:DF:MOVement:ANGLe <Angle>
SCENario:DF:RECeiver:MOVement:ANGLe <Angle>
SCENario:LOCalized:RECeiver:MOVement:ANGLe <Angle>
SCENario:LOCalized:MOVement:ANGLe <Angle>

Sets the arc angle and thus defines the arc length.

Parameters:

<Angle> float

Range: -360 to 360

Example: See Example"Moving emitters and receivers" on page 565.

Manual operation: See "Angle" on page 327

SCENario:DF:MOVement:EAST <East>
SCENario:DF:RECeiver:MOVement:EAST <East>
SCENario:LOCalized:RECeiver:MOVement:EAST <East>
SCENario:LOCalized:MOVement:EAST <East>
SCENario:DF:MOVement:NORTh <North>
SCENario:DF:RECeiver:MOVement:NORTh <North>

SCENario:LOCalized:RECeiver:MOVement:NORTh <North>
SCENario:LOCalized:MOVement:NORTh <North>

Sets the East/North coordinates of the emitter at the end of the movement.

Parameters:

<North> float

Range: -1e+09 to 1e+09

Example: See Example"Moving emitters and receivers" on page 565.

Manual operation: See "End Position" on page 326

SCENario:DF:MOVement:HEIGht <Height>
SCENario:DF:RECeiver:MOVement:HEIGht <Height>
SCENario:LOCalized:RECeiver:MOVement:HEIGht <Height>
SCENario:LOCalized:MOVement:HEIGht <Height>

Sets the height of the emitter at the end of the movement.

Parameters:

<Height> float

Range: -1e+09 to 1e+09

Example: See Example"Moving emitters and receivers" on page 565.

Manual operation: See "End Position" on page 326

SCENario:DF:MOVement:WAYPoint <Waypoint>
SCENario:DF:RECeiver:MOVement:WAYPoint <Waypoint>
SCENario:LOCalized:RECeiver:MOVement:WAYPoint <Waypoint>
SCENario:LOCalized:MOVement:WAYPoint <Waypoint>

Loads the selected waypoint file.

To import and apply the files, send the command `SCENario:LOCalized:MOVement:IMPort`.

Parameters:

<Waypoint> string

Filename or complete file path, incl. file extension.
Waypoint files must have the extension `*.txt`, `*.kml` or `*.xtd`.
Example files are provided with the software.
For description, see Chapter A.3, "Movement files", on page 638.

Example: See Example"Moving emitters and receivers" on page 565.

Manual operation: See "Load, Delete" on page 331

SCENario:DF:RECeiver:MOVement:WAYPoint:CLEar
SCENario:DF:LOCation:WAYPoint:CLEar

SCENario:LOCalized:LOCation:WAYPoint:CLEar
SCENario:LOCalized:RECeiver:MOVement:WAYPoint:CLEar

Discards the selected file.

Example: See Example"Moving emitters and receivers" on page 565.

Usage: Event

Manual operation: See "Load, Delete" on page 276

SCENario:DF:MOVement:ATTitude <Attitude>
SCENario:DF:RECeiver:MOVement:ATTitude <Attitude>
SCENario:LOCalized:RECeiver:MOVement:ATTitude <Attitude>
SCENario:LOCalized:MOVement:ATTitude <Attitude>

Defines how the attitude information is defined.

Parameters:

<Attitude> WAYPoint | MOTion | CONStant

WAYPoint
The attitude parameters are extracted from the selected way-point file.

MOTion
Enables a constant rate of change of the roll.
See `SCENario:LOCalized:MOVement:ROLL` on page 575

Constant
The attitude is constant values.

Example: See Example"Moving emitters and receivers" on page 565.

Manual operation: See "Attitude" on page 333

SCENario:DF:RECeiver:MOVement:VEHicle <Vehicle>
SCENario:LOCalized:RECeiver:MOVement:VEHicle <Vehicle>

Assigns the selected icon.

Parameters:

<Vehicle> LVEHicle | SHIP | AIRPlane | STATionary | RECeiver

Example: See Example"Moving emitters and receivers" on page 565.

Manual operation: See "Vehicle Icon" on page 262

SCENario:DF:MOVement:VEHicle <Vehicle>
SCENario:LOCalized:MOVement:VEHicle <Vehicle>

Assigns the selected icon.

Parameters:

<Vehicle> LVEHicle | SHIP | AIRPlane | STATionary | DEFault | CAR

Example: See Example"Moving emitters and receivers" on page 565.

Manual operation: See "Vehicle Icon" on page 312

SCENario:DF:MOVement:VFILe <Vfile>
SCENario:DF:RECeiver:MOVement:VFILe <Vfile>
SCENario:LOCalized:RECeiver:MOVement:VFILe <Vfile>
SCENario:LOCalized:MOVement:VFILe <Vfile>

Loads the selected vehicle description file (`*.xvd`).

To import and apply the files, send the command `SCENario:LOCalized:MOVement:IMPort`.

Parameters:

<Vfile> string

Filename or complete file path, incl. file extension.
Example files are provided with the software.
For description, see Chapter A.4, "Vehicle description files (Used for smoothening)", on page 645.

Example: See Example"Moving emitters and receivers" on page 565.

Manual operation: See "Load, Delete" on page 333

SCENario:DF:MOVement:VFILe:CLEar
SCENario:DF:RECeiver:MOVement:VFILe:CLEar
SCENario:LOCalized:RECeiver:MOVement:VFILe:CLEar
SCENario:LOCalized:MOVement:VFILe:CLEar

Discards the selected vehicle description file.

Example: See Example"Moving emitters and receivers" on page 565.

Usage: Event

Manual operation: See "Load, Delete" on page 333

SCENario:DF:MOVement:YAW <Yaw>
SCENario:DF:RECeiver:MOVement:YAW <Yaw>
SCENario:LOCalized:RECeiver:MOVement:YAW <Yaw>
SCENario:LOCalized:MOVement:YAW <Yaw>
SCENario:DF:MOVement:PITCh <Pitch>
SCENario:DF:RECeiver:MOVement:PITCh <Pitch>
SCENario:LOCalized:RECeiver:MOVement:PITCh <Pitch>
SCENario:LOCalized:MOVement:PITCh <Pitch>
SCENario:DF:MOVement:ROLL <Roll>
SCENario:DF:RECeiver:MOVement:ROLL <Roll>
SCENario:LOCalized:RECeiver:MOVement:ROLL <Roll>
SCENario:LOCalized:MOVement:ROLL <Roll>

Sets the angles of rotation in the corresponding direction.

Parameters:
<Roll> float

Range: -180 to 180

Example: See Example"Moving emitters and receivers" on page 565.

Manual operation: See "Yaw, Pitch, Roll" on page 333

SCENario:DF:MOVement:RFRame <Rframe>
SCENario:DF:RECeiver:MOVement:RFRame <Rframe>
SCENario:LOCalized:RECeiver:MOVement:RFRame <Rframe>
SCENario:LOCalized:MOVement:RFRame <Rframe>

Select the reference frame used to define the emitters coordinates.

Parameters:
<Rframe> WGS | PZ

Example: See Example"Moving emitters and receivers" on page 565.

Manual operation: See "Reference Frame" on page 332

SCENario:DF:MOVement:RMODe <Rmode>
SCENario:DF:RECeiver:MOVement:RMODe <Rmode>
SCENario:LOCalized:RECeiver:MOVement:RMODe <Rmode>
SCENario:LOCalized:MOVement:RMODe <Rmode>

Defines the behavior of the moving object when the end of the trajectory is reached.

Parameters:
<Rmode> CYCLic | ROUNdtrip | ONEWay

Example: See Example"Moving emitters and receivers" on page 565.

Manual operation: See "Mode" on page 327

SCENario:DF:MOVement:SMOothening <Smoothening>
SCENario:DF:RECeiver:MOVement:SMOothening <Smoothening>
SCENario:LOCalized:RECeiver:MOVement:SMOothening <Smoothening>
SCENario:LOCalized:MOVement:SMOothening <Smoothening>

If a vehicle description file is loaded, activates smoothening.

See `SCENario:LOCalized:MOVement:VFILe` on page 575.

Parameters:
<Smoothening> ON | OFF | 1 | 0

Example: See Example"Moving emitters and receivers" on page 565.

Manual operation: See "Smoothing for waypoints > State" on page 332

SCENario:DF:MOVement:IMPort
SCENario:DF:RECeiver:MOVement:IMPort

SCENario:LOCalized:RECeiver:MOVement:IMPort
SCENario:LOCalized:MOVement:IMPort

Imports the selected waypoint and vehicle description files into the repository and **applies** them.

Example: See Example"Moving emitters and receivers" on page 565.

Usage: Event

Manual operation: See "Calculate" on page 334

SCENario:DF:MOVement:CLEar
SCENario:DF:RECeiver:MOVement:CLEar
SCENario:LOCalized:RECeiver:MOVement:CLEar
SCENario:LOCalized:MOVement:CLEar

Discards the waypoint and vehicle description file.

Example: See Example"Moving emitters and receivers" on page 565.

Usage: Event

Manual operation: See "Clear" on page 334

SCENario:DF:SUBitem:CURRent <Current>
SCENario:LOCalized:SUBitem:CURRent <Current>

Parameters:

<Current> float

Range: 1 to 4096

SCENario:DF:SUBitem:SELect <Select>
SCENario:LOCalized:SUBitem:SELect <Select>

Parameters:

<Select> float

Range: 1 to 4096

PREView:POSition?

If movement is enabled, queries the current positions of the TX items and RX items.

Return values:

<Position> string

Semicolon-separated string with the format:

```
TIME=<time_from_simulation_start>;
ID=<Item ID>;NAME=<Item alias
name>;DIST=<distance>m;LEVATT=<Level at Rx
origin>dBm;
AZI=<Azimuth>deg;ELEV=<Elevation>deg;N=<North>m;
E=<East>m;H=<Height>m;LON=<Longitude>deg;
LAT=<Latitude>deg;ALT=<Altitude>m;VEL=<Velocity>m/s
```

Example:
```
PREView:POSition?
// line breaks added for better readability
// TIME=00:00:24.012;
// ID=1;NAME=Ground;DIST=2323.635km;LEVATT=8dBm;AZI=15.2deg;
ELEV=0.0deg;N=2241.90m;E=610.87m;H=0.00km;LON=0.00549deg;
LAT=0.02028deg;ALT=0.43m;VEL=0.00m/s
// ID=2;NAME=Plane;DIST=3812.188m;LEVATT=-4dBm;AZI=291.9deg;
ELEV=13.9deg;N=1383.20m;E=-3432.21m;H=916.23m;LON=-0.03083deg;
LAT=0.01251deg;ALT=917.30m;VEL=100.00m/s
// ID=3;NAME=Ship;DIST=1055.920m;LEVATT=12dBm;AZI=324.0deg;
ELEV=-11.9deg;N=836.01m;E=-606.84m;H=-218.61m;LON=-0.00545deg;
LAT=0.00756deg;ALT=-218.57m;VEL=100.00m/s
// ID=4;NAME=Receiver;DIST=---;LEVATT=---;AZI=---;ELEV=---;
N=0.00m;E=0.00m;H=0.00m;LON=0.00000deg;LAT=0.00000deg;ALT=0.00m;
VEL=0.00m/s
```

Usage: Query only

Manual operation: See "2D Live Preview Stats" on page 299

24.21 Platform command

PLATform:NAME <Name>

Queries the name of the platform.

Parameters:

<Name> string

Manual operation: See "Name" on page 248

PLATform:ID <Id>

Platform identifier.

Parameters:

<Id> float

Range: 1 to 65536

Manual operation: See "ID" on page 249

PLATform:EMITter <Emitter>

The string must be unique within the repository.

Letters, numbers, spaces and some special characters can be used.

Examples of special characters:

- Supported: !$%()=?-+_.
- Not supported: §&/:äöü

Parameters:

<Emitter> string

Manual operation: See "Emitter" on page 253

PLATform:EMITter:ID <Id>

Queries the ID number of the emitter.

Parameters:

<Id> float

*RST: 1

PLATform:EMITter:ALIas <Alias>

Sets an alias name for the selected platform emitter element.

Parameters:

<Alias> string

Manual operation: See "List of emitters" on page 250

PLATform:EMITter:ANGLe <Angle>

You can set the position of the selected emitter relative to the platform's origin, using this command combined with `PLATform:EMITter:RADius`.

- `PLATform:EMITter:ANGLe` sets the angle of the emitter element on the azimuth plane, relative to the platform's heading.
- `PLATform:EMITter:RADius` sets the distance of the emitter element on the azimuth plane, relative to the platform's origin.

Parameters:

<Angle> float

Range: 0 to 360
Increment: 0.01

Manual operation: See "Angle" on page 255

PLATform:EMITter:AZIMuth <Azimuth>

Angle of the emitter element's pointing direction relative to the platform's heading.

Parameters:

<Azimuth> float

Range: 0 to 360
Increment: 0.01

Manual operation: See "Azimuth" on page 253

PLATform:EMITter:BLANkranges <Blankranges>

Creates a set of blank ranges for the currently selected platform emitter.

This approach is more efficient than using several blank range start/stop commands.

The ranges are defined as a set of comma-separated values.

Parameters:

<Blankranges>

Manual operation: See "Set of ranges" on page 256

PLATform:EMITter:BLANkranges:SELect <Select>

Selects a particular blank range for the currently selected platform emitter.

Use with the following commands to configure a blank range:

- `PLATform:EMITter:BLANkranges:STARt` on page 580
- `PLATform:EMITter:BLANkranges:STOP` on page 581

To configure several blank ranges with a single command, you can use `PLATform:EMITter:BLANkranges` on page 580 . This approach is more efficient than using several blank range start/stop commands.

Parameters:

<Select> float

*RST: 1

Manual operation: See "No" on page 256

PLATform:EMITter:BLANkranges:STARt <Start>

Sets the start angle for the selected blank range.

The reference value (i.e. 0°) is the configured "Azimuth" value for the selected emitter.

Use together with `PLATform:EMITter:BLANkranges:SELect` on page 430.

To configure several blank ranges with a single command, you can use `PLATform:EMITter:BLANkranges` on page 580 . This approach is more efficient than using several blank range start/stop commands.

Parameters:

<Start> float

Range: 0 to 360

Example: `PLATform:EMITter:BLANkranges:SELect 1 ; PLATform:EMITter:BLANkranges:STARt 160`

Manual operation: See "Start" on page 256

PLATform:EMITter:BLANkranges:STOP <Stop>

Sets the stop angle for the selected "Blank Range".

The reference value (i.e. 0°) is the configured "Azimuth" value for the selected emitter.

Use together with `PLATform:EMITter:BLANkranges:SELect` on page 430.

To configure several blank ranges with a single command, you can use `PLATform:EMITter:BLANkranges` on page 580 . This approach is more efficient than using several blank range start/stop commands.

Parameters:

<Stop> float

Range: 0 to 360

Example: `PLATform:EMITter:BLANkranges:SELect 1 ; PLATform:EMITter:BLANkranges:STOP 200`

Manual operation: See "Stop" on page 256

PLATform:EMITter:DIRection:AWAY <Away>

This command automatically configures the transmission direction of the selected emitter.

No effect if emitter is at origin.

Affects emitters whose position (relative to the origin) has been defined by one of the following methods:

- Set X and Y values
- Set Angle and Radius values

The transmission direction is configured so that it is directly away from the origin.

Parameters:
<Away> ON | OFF | 1 | 0

Manual operation: See "Attitude" on page 254

PLATform:EMITter:ELEVation <Elevation>

Elevation of the emitter item's pointing direction, relative to the azimuth plane.

Parameters:
<Elevation> float
Range: -90 to 90

Manual operation: See "Elevation" on page 253

PLATform:EMITter:HEIGht <Height>

Height of the selected emitter element relative to the platform's origin.

Can be used, for example, to differentiate between:

- Radars mounted on different parts of a ship or aircraft.
- Various radars situated across a land-based radar installation.

Parameters:
<Height> float
Range: -500 to 500

Manual operation: See "Height" on page 254

PLATform:EMITter:RADius <Radius>

You can set the position of the selected emitter relative to the platform's origin, using this command combined with `PLATform:EMITter:ANGLe`.

- `PLATform:EMITter:ANGLe` sets the angle of the emitter element on the azimuth plane, relative to the platform's heading.
- `PLATform:EMITter:RADius` sets the distance of the emitter element on the azimuth plane, relative to the platform's origin.

Parameters:
<Radius> float
Range: 0 to 2000

Manual operation: See "Radius" on page 255

PLATform:EMITter:ROLL <Roll>

Roll of the emitter item's pointing direction relative to the platform's up direction.

Can be used, for example, to simulate the emissions from a mast-mounted radar on a marine platform affected by wind.

Parameters:

<Roll>	float	
	Range:	-180 to 180

Manual operation: See "Roll" on page 253

PLATform:EMITter:X <X>

Set the position of the selected emitter relative to the platform's origin, using this command combined with `PLATform:EMITter:Y`.

X and Y represent the two principle axis of the platform.

- The Y-axis represents the axis along the center-line of the platform.
  This axis:
  - Corresponds to its heading.
  - Passes through the origin.
- The X-axis:
  - Is at right-angles to the Y-axis.
  - Passes through the origin.
- `PLATform:EMITter:Y` sets the distance of the emitter element from the origin, along the Y-axis.
  Positive values are towards the heading.
- `PLATform:EMITter:X` sets the distance of the emitter element from the origin, along the X-axis.

Parameters:

<X>	float	
	Range:	-2000 to 2000

Manual operation: See "X" on page 254

PLATform:EMITter:Y <Y>

Set the position of the selected emitter relative to the platform's origin, using this command combined with `PLATform:EMITter:X`.

X and Y represent the two principle axis of the platform.

- The Y-axis represents the axis along the center-line of the platform.
  This axis:
  - Corresponds to its heading.
  - Passes through the origin.
- The X-axis:
  - Is at right-angles to the Y-axis.
  - Passes through the origin.
- `PLATform:EMITter:Y` sets the distance of the emitter element from the origin, along the Y-axis.

Positive values are towards the heading.
Step = 0.01 m

- `PLATform:EMITter:X` sets the distance of the emitter element from the origin, along the X-axis.
  Step = 0.01 m

Parameters:

<Y> float

Range: -2000 to 2000

Manual operation: See "Y (Heading)" on page 254

24.22 Sequence commands

In the following examples, we assume that a pulse-base sequence "Test Sequence" and the required Pulses have been created.

See also:

- Example"Handling items" on page 420
- Example"Creating an unmodulated pulse" on page 507

Example: Creating a filler segment

```
SCPI
SEQuence:CREate "Test Sequence"
SEQuence:ITEM:ADD
SEQuence:ITEM COUNt?
// 2
SEQuence:ITEM:SELect 2
SEQuence:ITEM:TYPE FILLer
SEQuence:ITEM:FILLer:SIGNal BLANk
SEQuence:ITEM:FILLer:MODE DURation
SEQuence:ITEM:FILLer:TIME FIXed
SEQuence:ITEM:FILLer:TIME:FIXed 5e-3
```

Example: Creating a simple sequence with two pulses repeated different number of times

```
SCPI
SEQuence:SELect "Test Sequence"
SEQuence:TYPE?
SEQuence:PHASe:MODE ABSolute
SEQuence:TIME:MODE PRI

SEQuence:ITEM:ADD
SEQuence:ITEM:SELect 1; SEQeunce:ITEM:TYPE PULSe
SEQunece:ITEM:PULSe "P1"
SEQuence:ITEM:PRI 2.5E-03
// SEQuence:TIME:MODE PRF
// SEQuence:ITEM:PRF?
// 4.E+02
SEQuence:ITEM:PDELay 2E-03
SEQuence:ITEM:REP:TYPE FIXed
SEQuence:ITEM:REP:COUNt:FIXed 2

SEQuence:ITEM:ADD
SEQuence:ITEM:SELect 2; SEQeunce:ITEM:TYPE LOOP
SEQuence:ITEM:LOOP:TYPE VARiable
SEQuence:ITEM:LOOP:COUNt:MINimum 2
SEQuence:ITEM:LOOP:COUNt:MAXimun 10
SEQuence:ITEM:LOOP:COUNt:STEP 2

SEQuence:ITEM:ADD
SEQuence:ITEM:SELect 3; SEQeunce:ITEM:TYPE PULSe
SEQunece:ITEM:PULSe "P1"
SEQuence:ITEM:PRI 5 ms
SEQuence:ITEM:FREQuency:OFFSet 1000000
SEQuence:ITEM:LEVel:OFFSet -10
SEQuence:ITEM:REP:TYPE DURation
SEQuence:ITEM:REP:COUNt:DURation 0.001
SEQuence:ITEM:REP:COUNt:ROUNding UP
SEQuence:ITEM:INDent 1
```

SEQuence:TYPE <Type>

Sets the sequence type.

Parameters:

<Type>	PULSe
Example:	See Example"Handling items" on page 420
Manual operation:	See "Sequence Name, Comment, Type, Sequence Description Table, Block Diagram" on page 121

SEQuence:PHASe:MODE <Mode>

Defines how the phase is set at each pulse start.

Parameters:

<Mode>	ABSolute | CONTinuous | MEMory
Example:	See Example"Creating a filler segment" on page 584
Manual operation:	See "Phase Mode" on page 121

SEQuence:TIME:MODE <Mode>

Switches between time-based (PRI) and frequency-based (PRF) pulse repetition definition.

Parameters:

<Mode>	PRI | PRF
Example:	See Example"Creating a simple sequence with two pulses repeated different number of times" on page 585
Manual operation:	See "Timing Mode" on page 123

SEQuence:ITEM:FILLer:MODE <Mode>

Sets how the filler duration is determined.

Parameters:
<Mode> DURation | TSYNc

Example: See Example"Creating a filler segment" on page 584

Manual operation: See "Mode" on page 128

SEQuence:ITEM:FILLer:SIGNal <Signal>

Sets the signal type.

Parameters:
<Signal> BLANk

Example: See Example"Creating a filler segment" on page 584

Manual operation: See "Signal" on page 128

SEQuence:ITEM:FILLer:TIME <Time>

Defines the way the duration is defined.

Parameters:
<Time> FIXed | EQUation

Example: See Example"Creating a filler segment" on page 584

Manual operation: See "Time" on page 129

SEQuence:ITEM:FILLer:TIME:EQUation <Equation>

Sets the filler duration as an equation.

Parameters:
<Equation> string

Example: `SEQuence:ITEM:FILLer:TIME:EQUation?`

Manual operation: See "Time" on page 129

SEQuence:ITEM:FILLer:TIME:FIXed <Fixed>

Sets the duration of the filler.

Parameters:
<Fixed> float

Range: 0 to 1e+09
Default unit: sec

Example: See Example"Creating a filler segment" on page 584

Manual operation: See "Time" on page 129

SEQuence:ITEM:FREQuency:OFFSet <Offset>

Enables a frequency offset.

Parameters:

<Offset> float

Range: -1e+09 to 1e+09
Default unit: Hz

Example: See Example"Creating a simple sequence with two pulses repeated different number of times" on page 585

Manual operation: See "Δ Freq." on page 124

SEQuence:ITEM:INDent <Indent>

Indents the selected item rows to include it, for example, in a loop.

Parameters:

<Indent> float

Range: 0 to 5

Example: See Example"Creating a simple sequence with two pulses repeated different number of times" on page 585

Manual operation: See "Nesting" on page 123

SEQuence:ITEM:LEVel:OFFSet <Offset>

Sets a level offset.

Parameters:

<Offset> float

Range: -100 to 0
Default unit: dB

Example: See Example"Creating a simple sequence with two pulses repeated different number of times" on page 585

Manual operation: See "Δ Level" on page 124

SEQuence:ITEM:LOOP:COUNt:FIXed <Fixed>

Sets the repetition number as a numeric value.

Parameters:

<Fixed> float

Range: 1 to 65535

Example: See Example"Creating a simple sequence with two pulses repeated different number of times" on page 585

Manual operation: See "Repetition" on page 124

SEQuence:ITEM:LOOP:COUNt:MINimum <Minimum>
SEQuence:ITEM:LOOP:COUNt:MAXimum <Maximum>

Sets the value range of the loop count.

Parameters:

<Maximum> float

Range: 1 to 65535

Example: See Example"Creating a simple sequence with two pulses repeated different number of times" on page 585

Manual operation: See "Loop Repetition" on page 127

SEQuence:ITEM:LOOP:COUNt:STEP <Step>

Sets the loop count granularity.

Parameters:

<Step> float

Range: 1 to 65535

Example: See Example"Creating a simple sequence with two pulses repeated different number of times" on page 585

Manual operation: See "Loop Repetition" on page 127

SEQuence:ITEM:LOOP:TYPE <Type>

Sets how the loop repetition is defined.

Parameters:

<Type> FIXed | VARiable

Example: See Example"Creating a simple sequence with two pulses repeated different number of times" on page 585

Manual operation: See "Loop Repetition" on page 127

SEQuence:ITEM:LOOP:VARiable <Variable>

Sets a loop variable.

Parameters:

<Variable> string

Example: See Example"Creating a simple sequence with two pulses repeated different number of times" on page 585

Manual operation: See "Loop Variables" on page 127

SEQuence:ITEM:PDELay <Pdelay>

Enables a start delay.

Parameters:

<Pdelay> float

Range: 0 to 1e+09
Default unit: sec

Example: See Example"Creating a simple sequence with two pulses repeated different number of times" on page 585

Manual operation: See "Delay" on page 125

SEQuence:ITEM:PHASe:OFFSet <Offset>

Sets a phase offset.

Parameters:

<Offset> float

Range: -180 to 180

Example: `SEQuence:ITEM:PHASe:OFFSet 30`

Manual operation: See "Phase" on page 125

SEQuence:ITEM:PRF <Prf>
SEQuence:ITEM:PRI <Pri>

Sets the pulse repetition interval (PRI) or the pulse repetition frequency (PRF).

Parameters:

<Pri> float

Range: 0 to 1e+09

Example: See Example"Creating a simple sequence with two pulses repeated different number of times" on page 585

Manual operation: See "PRI/PRF" on page 125

SEQuence:ITEM:PULSe <Pulse>

Assigns a pulse to the selected item.

Use the command `PULSe:CATalog?` to querry the available pulses.

Parameters:

<Pulse> string

Pulse name

Example: See Example"Creating a simple sequence with two pulses repeated different number of times" on page 585

Manual operation: See "Pulse" on page 124

SEQuence:ITEM:REP:COUNt:DURation <Duration>

Sets a time duration.

Parameters:

<Duration> float

Range: 0 to 1e+09
Default unit: sec

Example: See Example"Creating a simple sequence with two pulses repeated different number of times" on page 585

Manual operation: See "Repetition Number" on page 126

SEQuence:ITEM:REP:COUNt:FIXed <Fixed>

Sets the repetition number as a numeric value.

Parameters:

<Fixed> float

Range: 1 to 65535

Example: See Example"Creating a simple sequence with two pulses repeated different number of times" on page 585

Manual operation: See "Repetition" on page 124

SEQuence:ITEM:REP:COUNt:MINimum <Minimum>
SEQuence:ITEM:REP:COUNt:MAXimum <Maximum>

Sets the value range of the repetition count.

Parameters:

<Maximum> float

Range: 1 to 65535

Example: See Example"Creating a simple sequence with two pulses repeated different number of times" on page 585

Manual operation: See "Repetition Number" on page 126

SEQuence:ITEM:REP:COUNt:ROUNding <Rounding>

Sets how the repetition number is rounded.

Parameters:

<Rounding> DN | UP

Example: See Example"Creating a simple sequence with two pulses repeated different number of times" on page 585

Manual operation: See "Repetition Number" on page 126

SEQuence:ITEM:REP:COUNt:STEP <Step>

Sets the repetition count granularity.

Parameters:

<Step> float

Range: 1 to 65535

Example: See Example"Creating a simple sequence with two pulses repeated different number of times" on page 585

Manual operation: See "Repetition Number" on page 126

SEQuence:ITEM:REP:TYPE <Type>

Sets how the repetition number is defined.

Parameters:

<Type> FIXed | VARiable | DURation

Example: See Example"Creating a simple sequence with two pulses repeated different number of times" on page 585

Manual operation: See "Repetition Number" on page 126

SEQuence:ITEM:REP:VARiable <Variable>

Seta a repetition variable.

Parameters:

<Variable> string

Manual operation: See "Repetition Variables" on page 126

SEQuence:ITEM:TYPE <Type>

Sets the content type of the selected item.

Parameters:

<Type> PULSe | FILLer | LOOP

Example: See Example"Creating a filler segment" on page 584

Manual operation: See "Element Type" on page 124

24.23 Status commands

This system contains the commands for the status reporting system.

Value ranges

- **Return parameters**
  Queries return the current value of the respective register, which permits a check of the device status.
  Range: A decimal value in the range 0 to 32767 (=2^{15}-1)
- **Setting parameters**
  The configuration commands set the respective register thus determining which status changes of the R&S Pulse Sequencer Digital cause the status registers to be changed.
  Range: A decimal value in the range 0 to 32767 (=2^{15}-1)

STATus:OPERation:CONDition?

Queries the content of the CONDition part of the STATus:OPERation register. This part contains information on the action currently being performed in the instrument. The content is not deleted after being read out because it indicates the current status.

Return values:
<Condition> float

Example: `:STATus:OPERation:CONDition?`
Queries the Status:Operation:Condition register.

Usage: Query only

24.24 System, message log and program commands

The `SYSTem`, `MSGLog` and `PROGram` subsystems contain a series of commands for general functions which do not directly affect signal generation.

SYSTem:ERRor?

Queries the error/event queue for the oldest item and removes it from the queue. The response consists of an error number and a short description of the error.

Positive error numbers are instrument-dependent. Negative error numbers are reserved by the SCPI standard.

Return values:
<Error> string
`Error/event_number, "Error/event_description>[;Device-dependent info]"`
If the queue is empty, the response is `0,"No error"`

Example: See `SYSTem:ERRor:ALL?` on page 594

Usage: Query only

SYSTem:ERRor:ALL?

Queries the error/event queue for all unread items and removes them from the queue. The response is a comma-separated list of error number and a short description of the error in FIFO order.

Positive error numbers are instrument-dependent. Negative error numbers are reserved by the SCPI standard.

Return values:

<All> string

List of: `Error/event_number, "Error/event_description>[;Device-dependent info]"`

If the queue is empty, the response is `0,"No error"`

Example: `SYSTem:ERRor:ALL?`

Queries all entries in the error queue

`Response: 0, 'no error'`

No errors have occurred since the error queue was last read out

Usage: Query only

SYSTem:ERRor:COUNt?

Queries the number of entries in the error queue. If the error queue is empty, '0' is returned.

Return values:

<Count> string

Example: `SYSTem:ERRor:COUNt?`

Queries the number of entries in the error queue

`Response: 1`

One error has occurred since the error queue was last read out

Usage: Query only

SYSTem:PROGress?

Queries the signal generation progress status.

Return values:

<Progress> float

Example: See Example"Creating simple pulse train scenario" on page 533

Usage: Query only

Manual operation: See "Config" on page 80

MSGLog:ERRor?

Queries the last error listed in the "Message Log" dialog.

Return values:

<Error> string

Example:

```
MSGLog:POPup 1
MSGLog:ERRor?
// 09:38:41:837    Cannot open input file.
```

Usage: Query only

Manual operation: See Chapter 25, "Troubleshooting", on page 619.

MSGLog:POPup <Popup>

Opens/closes the "Message Log" dialog.

Parameters:

<Popup> ON | OFF | 1 | 0

Manual operation: See Chapter 25, "Troubleshooting", on page 619.

PROGram:HIDE
PROGram:SHOW

Minimizes/maximizes the R&S Pulse Sequencer Digital workspace.

Usage: Event

24.25 Signal preview commands

SCENario:VOLatile:VIEW:XMODe <Xmode>

Sets the units (time or samples) used on the x axis.

Setting parameters:

<Xmode> SAMPles | TIME

Usage: Setting only

SCENario:VOLatile:VIEW:YMODe <Ymode>

Sets the view mode.

Setting parameters:
<Ymode> IQ | MAGDb | MAGW | MAGV | PHASe | FREQuency | PAV

Usage: Setting only

SCENario:VOLatile:VIEW:ZOOM:POINt <Point>

Sets center point of the displayed area.

Setting parameters:
<Point> float

Always related to time

Default unit: s

Example:
```
SCENario:VOLatile:VIEW:ZOOM:POINt 300 us
SCENario:VOLatile:VIEW:ZOOM:RANGe 100u
// 100 us around the 300 us point, i.e.
// displayed is the time span of 200 us to 400 us

SCENario:VOLatile:VIEW:ZOOM:POINt 0.5 ms
SCENario:VOLatile:VIEW:ZOOM:RANGe 2 m
// displayed is the time span of 0 s to 2 ms
```

Usage: Setting only

SCENario:VOLatile:VIEW:ZOOM:RANGe <Range>

Sets the displayed waveform part as a range around the selected center point, set with the command `SCENario:VOLatile:VIEW:ZOOM:POINt`.

Setting parameters:
<Range> float

Expressed as a time span (units can be omitted) or as number of samples

Example: See `SCENario:VOLatile:VIEW:ZOOM:POINt`

Usage: Setting only

24.26 List of remote commands

25 Troubleshooting

The R&S Pulse Sequencer Digital monitors the functions performed and automatically detects errors and irregularities. The software logs status and error messages, warnings, SCPI commands send to the software, or additional information in a log file. Some of the messages are also entered in the error/event queue of the status reporting system.

25.1 Querying error messages

The following information sources help you determine the cause for an error or an unexpected program behavior:

- **Message log**
  The "Message Log" dialog lists information about all messages in a history list. If the software detects an error, the "Message Log" dialog pops up automatically and displays the error message.
- **Debug information**
  Additionally to the log information, the software provides a dialog with debug information. It lists information like the version of the used VISA library, the list of all loaded modules, or the report and data paths.
- The **psstartup.log** file
  If the software stops functioning properly, a log file is created automatically and stored in the `%HOMEPATH%` directory when the software is started again, see Table 2-2.

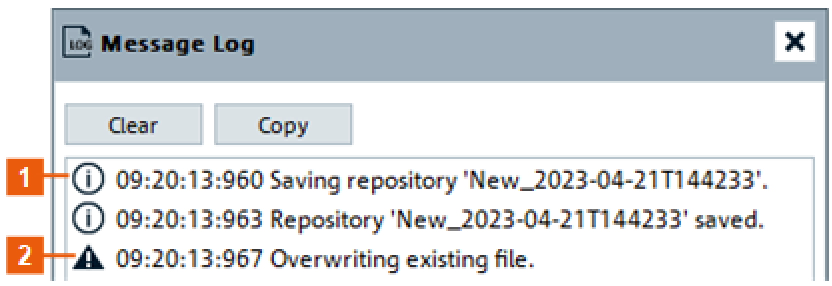
See:

- "To access the "Message Log" dialog" on page 619
- "To open the "Message Log" dialog in debug mode" on page 620
- "To access the "Debug Information" dialog" on page 621
- "To remove write lock that results from a previously crashed session" on page 622

To access the "Message Log" dialog

► Perform one of the following:

a) In the toolbar, select the "Log" icon.
b) In the menu bar, select "Window > Message Log".



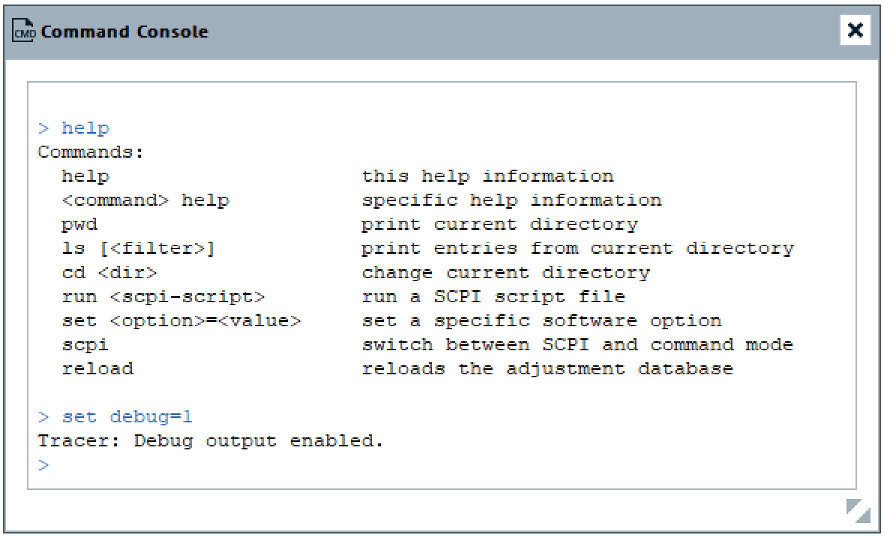
1 = Info message
2 = Warning

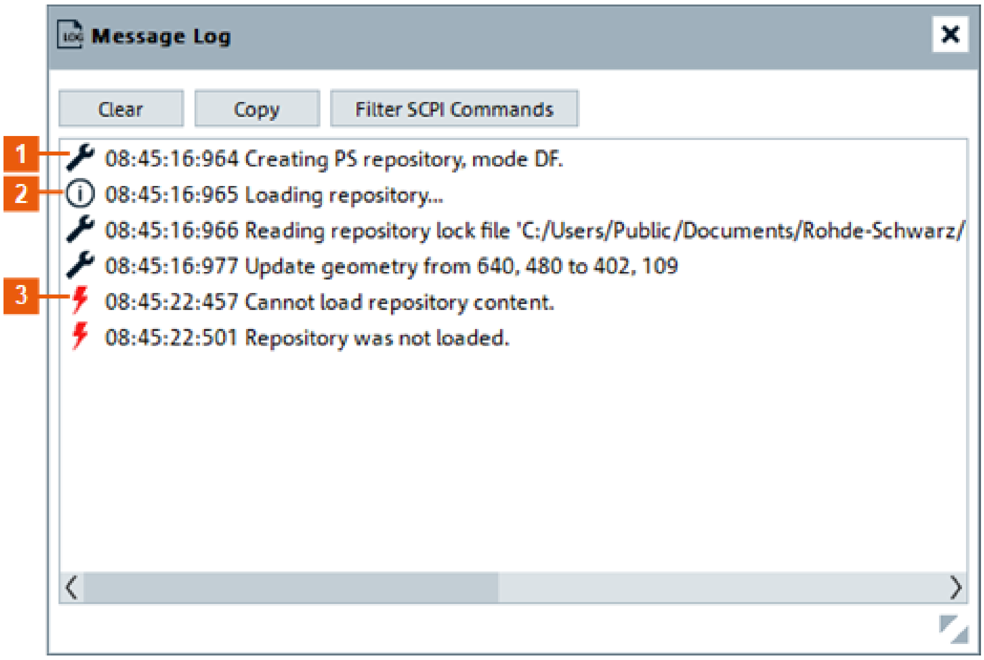
The displayed information is read-only but you can mark and copy it.
Remote command:
`MSGLog:POPup` on page 595
`MSGLog:ERRor?` on page 595

To open the "Message Log" dialog in debug mode

1. In the toolbar, select the "Console Window" icon.
2. In the "Command Console" window, type *help*.

   The dialog lists all available commands.
3. Enter *set debug = 1* to enable the debug messages to be output in the "Message Log" view.

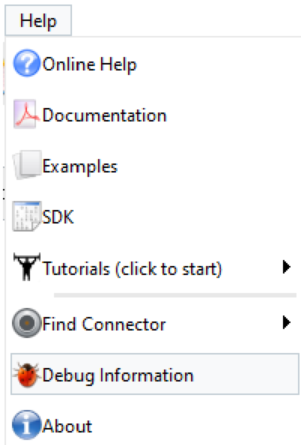




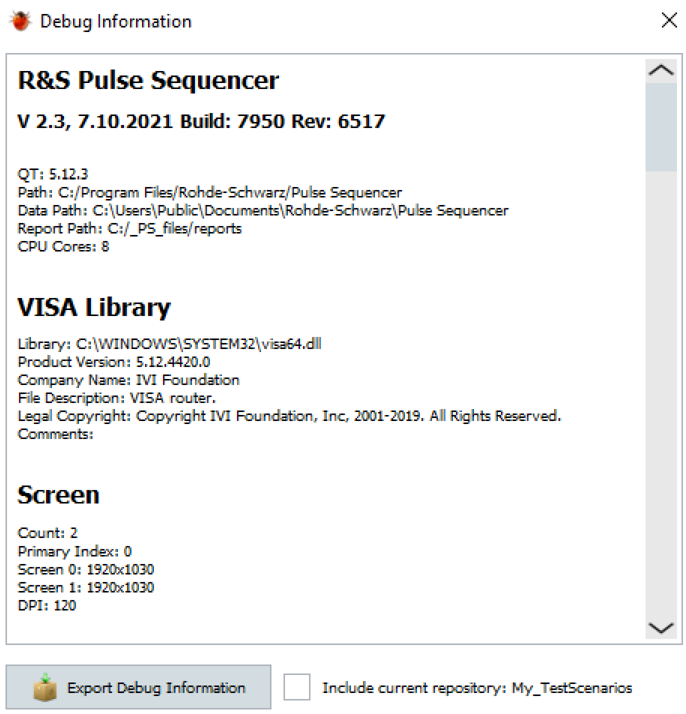
1 = Debug message
2 = Info message
3 = Error message

To access the "Debug Information" dialog

► In the menu bar, select "Help > Debug Information".



A dialog opens and displays information that helps the debugging.



To remove write lock that results from a previously crashed session

A red lock symbol on the database icon in the "Repository Tree" indicates that a session was terminated improperly. A locked repository can be unlocked only from the same PC and by the same user with write permission that had opened the repository before.

1. In the project tree, double-click the repository name.
2. In the "Repository" dialog, select "Storage > Remove Write Lock".

   The repository is unlocked; you can change or store repository settings.
   See also Chapter 4.1, "Repository settings", on page 60.

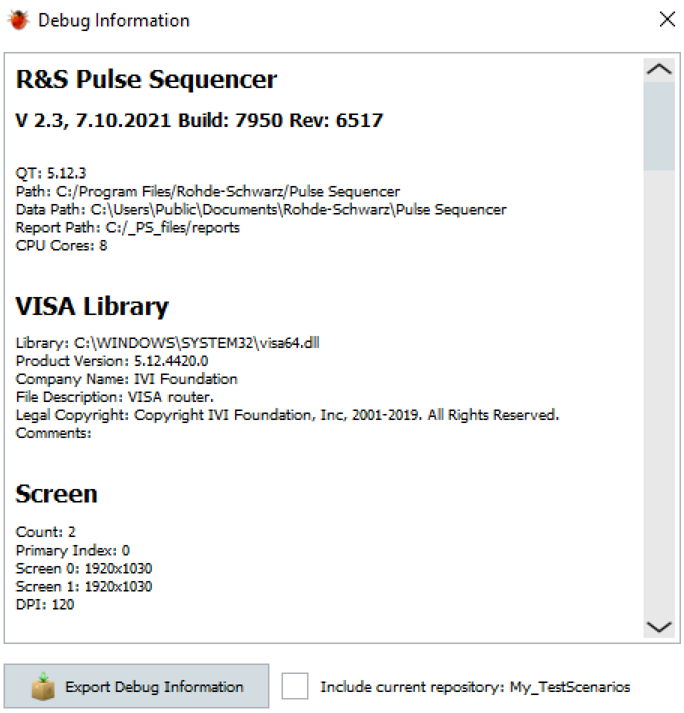
25.2 Collecting information for technical support

If you encounter problems that you cannot solve yourself, contact your Rohde & Schwarz support center as listed at http://www.customersupport.rohde-schwarz.com. Our support center staff is optimally trained to assist you in solving problems.

The support center finds solutions more quickly and efficiently if you provide them with information on the software and an error description.

To collect error information in a debug file

1. Select "Help > Debug Information".



2. You can include or not the selected repository in the debug information file.
   a) Select "Include current repository" and in the "Export Debug Information" dialog confirm the export with "Yes" .
   b) If your repository contains sensitive data, deselect "Include current repository".
3. Select "Export Debug Information".
4. Navigate to the directory you want to save the file in.
   The filename is assigned automatically.

   The debug information and further required data is collected automatically.
   The debug file `DebugInfo.psdbarch` is created and saved in the selected directory.

The file contains:

- `*.ini` files, collected from the `%HOMEPATH%\AppData\Roaming\Rohde-Schwarz\Pulse Sequencer Digital` directory.
- Text as it is displayed in the "Debug Information" dialog (`debug_info.txt`).
- If enabled, the repository (`*.psarch`).

Name	Size	Packed	Type	Modified
..			File folder	
colors.ini	1.319	1.319	Configuration setti...	17.09.2021 09:44
config.ini	2.745	2.745	Configuration setti...	17.09.2021 10:12
crash.ini	20	20	Configuration setti...	17.09.2021 09:54
debug_info.txt	7.571	7.571	Text Document	17.09.2021 10:34
help.psarch	107.520	107.520	PSARCH File	17.09.2021 10:34
mapping.ini	4.973	4.973	Configuration setti...	28.07.2021 08:54
settings_2.ini	44.204	44.204	Configuration setti...	17.09.2021 09:44

Attach the `DebugInfo.psdbarch` file to an email in which you describe the problem. Send the email to the customer support address for your region as listed on the Internet (http://www.customersupport.rohde-schwarz.com).

25.3 Contacting customer support

Technical support – where and when you need it

For quick, expert help with any Rohde & Schwarz product, contact our customer support center. A team of highly qualified engineers provides support and works with you to find a solution to your query on any aspect of the operation, programming or applications of Rohde & Schwarz products.

Contact information

Contact our customer support center at www.rohde-schwarz.com/support, or follow this QR code:

Figure 25-1: QR code to the Rohde & Schwarz support page

Annex

A Supported file types and file formats

This section summarizes the file types supported by the application and provides information about the default file locations.

List of supported file types and file extensions

Table A-1: Supported file types

Extension	Description

`*.ant_pat`
`*.ffe`
`*.tsv`
`*.ffd`
`*.csv`
	Antenna pattern

`*.dll`	Plug-in, for example with user modulation, IPM profile
`*.ini`	Initialization file Contains information on default location for temporary files
`psstartup.log`	Log file created automatically if the software stops functioning properly
`*.riq`	Rohde & Schwarz proprietary I/Q data file format used by the R&S®PR100 portable receiver.
`*.wav`	Rohde & Schwarz proprietary audio-like file format that contains I/Q samples instead of the usual FM stereo signal. Although the file extension suggests that this file is a standard auto file, stereo players do not recognize it.
`*.iq.tar`	Rohde & Schwarz proprietary I/Q data file format used by some signal and spectrum analyzers, e.g. R&S®FSW. The `*.iq.tar` file contains I/Q data in binary format together with meta information that describes the nature and the source of data, e.g. the sample rate.
`*.mat`	MATLAB file that contains complex 1xN or Mx1 vectors.
`*.txt`	Standard ASCII files with comma-separated values. The `*.txt` files can be used to: ● Describe settings that are based on lists, like for example the custom pulse envelope shapes, the list for IPM profiles, or data lists. ● Describe custom I/Q data, with I and Q vectors in one file or in separated `*_i.txt` and `*_q.txt` files; If file pairs are stored in the same directory, they are recognized automatically by the "Import Wizard"

Extension	Description
*.bin	Files with custom I/Q data in binary format
*.aif *.dat *.pdw and *.ppdw	AMMOS IF files AMREC recording files AMMOS PDW Rohde & Schwarz proprietary data formats used with Rohde & Schwarz monitoring equipment. These files typically contain multiple data streams. The R&S Pulse Sequencer Digital extracts IF (I/Q) or PDW data from these files.

A.1 File format of the reporting template

The template uses an ASCII text. Text that is not a token, a format (.FMT), or an option (.OPT) description is directly transferred to the final report.

Example: Example of a template that generates report in ASCII format

```
# R&S Pulse Sequencer, PDW Report
#
Date:        <DATE>
Repository:  <REPOSITORY>
Scenario:    <SCENARIO>

TOA      RF      PW      PA      MOP
         GHz     us      dBm
-----------------------------------
.HDR
<TOA> ; <RF> ; <PW> ; <PA> ; <MOP>
# END OF REPORT
.FMT TOA:12.0f:-9
.FMT RF:9.6f:9
.FMT PW:7.3f:-6
.FMT PA:4.0f:0
```

Text

<Token>

.HDR
<Col#1> ; <Col#2> ; ...

.FMT Token:Format:Exponent

Figure A-1: PDW template explanation

General template text and available tokens

The general template text is copied to the final report. All tokens are replaced with values or text provided by the software.

See Table A-2 for an overview of the available tokens.

Table A-2: List of available tokens

Token name	Description
<USER>	Current user name
<TPATH>	Target file path of the directory in that the generated report file is stored
<ISODATE>	Date & time in ISO format

Token name	Description
<DATE>	Date & time in local format
<REPOSITORY>	Repository name
<RCOMM>	Repository comment
<VERSION>	Repository version
<AUTHOR>	Repository author
<RCDATE>	Repository creation data
<RPATH>	Root path of repository
<SCENARIO>	Scenario name
<SCOMM>	Scenario comment

Adding and formatting data with the .HDR keyword

Each report contains one or more data lines. The position of these data lines in the report is indicated with the keyword `.HDR`. Usually, the reported data is formatted in table form, where the column names are defined with the information right after the `.HDR` line.

Each report template must have a `.HDR` line and subsequent second line defining the individual columns information and following the syntax:

```
.HDR
<Col#1>;<Col#2>;...
```

See Table A-3 for an overview of the available tokens.

Table A-3: List of available .HDR tokens

Token name	Description
Values related to the generated pulse data	
<TOA>	Time of arrival
<PRI>	PRI of current pulse (0% - 0%)
<PRF>	PRF of current pulse (1/PRI or 0)
<PW>	Pulse width (0% - 0%)
<RT>	Rise time
<FT>	Fall time
<RF>	Center frequency of pulse (absolute)
<BW>	RF bandwidth (FMOP) incl. BB filter
<PA>	Pulse amplitude (at pulse start)
<MF>	MOP flag, where 1 indicates that MOP is active
<MOP>	MOP type, AMOP, PMOD, FMOP
<DF>	Frequency offset from carrier

Token name	Description
<RFC>	Center frequency of emitter / sequence
<DP>	Phase offset (deg)
<PHS>	Absolute start phase at the beginning of pulse (deg)
<RX_AZI>	Azimuth angle of receive antenna at the beginning of pulse (deg)
<RX_ELE>	Elevation angle of receive antenna at the beginning of pulse (deg)
<BEAR_AZI>	Emitter bearing (e.g. simulate direction finder AoA output in PDW)
<BEAR_ELE>	Emitter elevation (e.g. simulate direction finder output in PDW)
<TX_NORTH>	Emitter distance in north direction from origin
<TX_EAST>	Emitter distance in east direction from origin
<TX_UP>	Emitter height above origin
<TX_PITCH>	Emitter attitude pitch angle
<TX_YAW>	Emitter attitude yaw angle
<TX_ROLL>	Emitter attitude roll angle
<RX_NORTH>	Receiver distance in north direction from origin
<RX_EAST>	Receiver in east direction from origin
<RX_UP>	Receiver height above origin
<RX_PITCH>	Receiver attitude pitch angle
<RX_YAW>	Receiver attitude yaw angle
<RX_ROLL>	Receiver attitude roll angle
Global variables from IPM profiles or loops, etc.	
<VarName>	Variable from the internal variable pool
Automatically generated count values related to the sequencing	
<@ITEM>	Sequence line item, 1 to N
<@REP>	Repetition, 1 to M
<@COLL>	Collection line item, 1 to L

Defining format with the .FMT keyword

Each format line must begin with `.FMT` and follow the syntax:

```
.FMT Token:Format:Exponent
```

The `.FMT` keyword formats the numeric data output, see Table A-4.

Table A-4: Format keyword syntax description

Parameter	Description
`Token`	Name of the token
`Format`	The format of the numeric data is compared to a regular expression: `[0-9]*.?[0-9]*[-+diufFeExX]` To prevent e.g. application crash, data that does not match this expression is ignored.
`Exponent`	Sets an exponent to convert the numeric value; e.g. -9 converts the reported data into nano.

See Figure A-1.

Defining options with the .OPT keyword

This keyword defines one or multiple options, that further control the output of the PDW report generator. Each option line must begin with `.OPT` and follow the syntax:

```
.OPT TAG=<CURLY>,EXT=<html>
```

Table A-5: Option keyword syntax description

Parameter	Description
`TAG`	`TAG=CURLY|ANGLE` Sets the brackets type
`EXT`	Sets the extension of the reporting file

Defining conditions with the .COND keyword

This keyword defines conditions, so that PDWs are generated only, if the condition is fulfilled. The condition line must begin with `.COND` keyword, can contain one or multiple comma-separated conditions and follow the syntax:

```
.COND <token name> <condition> <value>,[<token name> <condition> <value>],
```

Where the following <condition> are supported:

- Equal to: `==` or `=`
- Different that: `!=` or `<>`
- Greater/smaller than: `>` and `<`
- Greater/smaller or equal: `>=` and `<=`

Example:

The condition `.COND PA > -100.0` reports only the PDWs with pulse amplitude > -100 dBm.

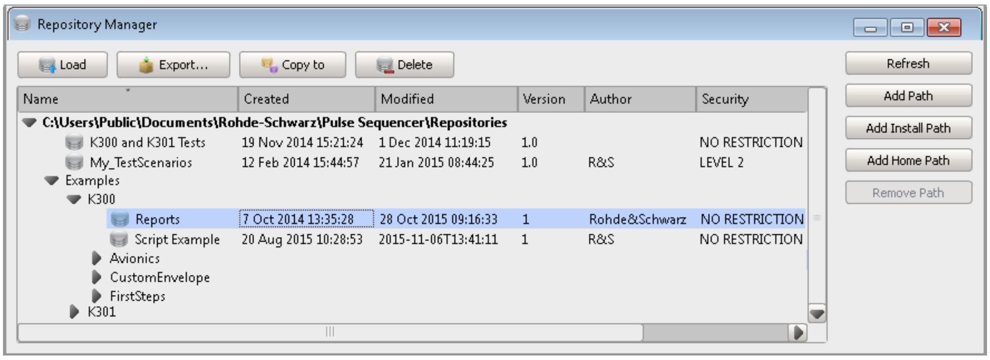
Template examples

Template examples are provided in the repositories supplied with the software.

You can load the example repository or copy&paste the templates content in your own repository, try out them, and further develop them.

Access:

1. Select "File > Load and Manager Repositories".
2. In the "Repository Manager", double click the "Examples > K300 > Reports".



   Created is a new repository with several scenarios using different kind of reporting templates.

3. Open a scenario.
   In the "Scenario" dialog, select "Waveform Generation > Config > Reporting > Template > Edit".

Example: Example of a template that generates report in HTML format

The following is the content of the template that creates a report in HTML format. The report is the same as in the ASCII file example shown on Figure 21-3, but the output is HTML formatted.

```
# Pulse Sequencer Report Template
# The following section defines the HTML report. All values
# in curly brackets are replaced by the actual values.
.OPT TAG=CURLY,EXT=html

<HTML> <HEAD> <TITLE>PDW Report</TITLE> <HEAD>
<BODY> <H2>PDW Report Generated with PS Example</H2>
  <P><TABLE style="width:100%;text-align:left;background-color:gold;">
    <colgroup> <col style="width:30%"> <col style="width:70%"> </colgroup>
    <TR style="background-color:blue;color:white;" >
    <TD>User</TD> <TD>{USER}</TD> </TR>
    <TR> <TD>Repository Path</TD> <TD>{TPATH}</TD> </TR>
    <TR> <TD>Date</TD> <TD>{DATE}</TD> </TR>
    <TR> <TD>ISO Date</TD> <TD>{ISODATE}</TD> </TR>
  </TABLE></P>
  <P><TABLE style="width:100%;text-align:left;background-color:gold;">
    <colgroup> <col style="width:30%"> <col style="width:70%"> </colgroup>
    <TR style="background-color:blue;color:white;" >
    <TD>Repository</TD> <TD>{REPOSITORY}</TD> </TR>
```

```
    <TR> <TD>Comment</TD> <TD>{RCOMM}</TD> </TR>
    <TR> <TD>Version</TD> <TD>{VERSION}</TD> </TR>
    <TR> <TD>Author</TD> <TD>{AUTHOR}</TD> </TR>
    <TR> <TD>Created</TD> <TD>{RCDATE}</TD> </TR>
    <TR> <TD>Path</TD> <TD>{RPATH}</TD> </TR>
  </TABLE></P>
  <H3>{SCENARIO}</H3>
  <P>{SCOMM}</P>

  <P><TABLE style="width:100%;text-align:left;background-color:lightgrey;">
   <TR style="background-color:blue;color:white;" >
   <TD>TOA</TD> <TD>RF</TD> <TD>PW</TD> <TD>PA</TD> <TD>MF</TD> <TD>MOP</TD>
   <TD>BW</TD> <TD>Rep</TD> </TR>
   <TR style="background-color:blue;color:white;" >
   <TD>ns</TD> <TD>GHz</TD> <TD>us</TD> <TD>dBm</TD> <TD> </TD> <TD> </TD>
   <TD>kHz</TD> <TD> </TD> </TR>
.HDR
   <TR> <TD>{TOA}</TD> <TD>{RF}</TD> <TD>{PW}</TD> <TD>{PA}</TD> <TD>{MF}</TD>
   <TD>{MOP}</TD> <TD>{BW}</TD> <TD>{@REP}</TD> </TR>
  </TABLE>
</BODY>
# The following .FMT lines define the data format
#
.FMT TOA:12.0f:-9
.FMT RF:9.1f:9
.FMT PW:9.3f:-6
.FMT PA:6.1f:0
.FMT BW:6.0f:3
.FMT @REP:3i:0
```

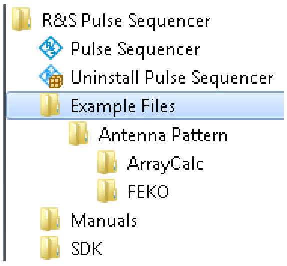
A.2 Antenna pattern file formats

R&S Pulse Sequencer Digital supports:

- "FEKO far field file format *.ffe" on page 632
- "Antenna Magus file format *.tsv" on page 632
- "ANSYS HFSS file format *.ffd" on page 632
- "Antenna patterns in *.csv file format" on page 633
- "Rohde&Schwarz proprietary antenna pattern *.ant_pat file format" on page 634

Examples of antenna pattern files

A subset of antenna pattern examples is included in the software.



Access:

► On your PC, open "Start > All Programs > R&S Pulse Sequencer Digital > Example Files > Antenna Pattern".

You can load and import the patterns, try out them, and further develop them. See "To import a user-defined antenna pattern from file in one of the predefined file formats" on page 212.

FEKO far field file format *.ffe

FEKO is an electromagnetic simulation software tool distributed by the EM Software & Systems-S.A. (Pty) Ltd.

The R&S Pulse Sequencer Digital supports the `*.ffe` files with version V1, V2, and V3.

For information on the file format, see the official product page http://www.feko.info/.

Antenna Magus file format *.tsv

Antenna Magus is an antenna design software tool.

The `*.tsv` files are text files that describe the antenna pattern in a tabular structure. These files contain a header section and data part with tab-separated values.

For information, see http://www.antennamagus.com/.

Example: Antenna Magus *.tsv file (extract)

```
# Far-field exported from Antenna Magus 5.3.0.1686
# On Friday, March 06, 2015 at 1:18:13 PM
# Frequency: 3.0E+9 Hz
# Power: 0.171146799151093 W
# Theta Samples: 181
# Phi Samples: 361
#Theta     Phi     Re(E_Theta)     Im(E_Theta)     Re(E_Phi)     Im(E_Phi)
0.000000000000E+000     0.000000000000E+000     4.303687774512E-016     0.000000000000E+000     -7.148424634567E+000     -1.436268526906E+000
1.000000000000E+000     0.000000000000E+000     4.425352450194E-016     0.000000000000E+000     -7.363209895869E+000     -1.542584165100E+000
...
```

ANSYS HFSS file format *.ffd

ANSYS HFSS software is the industry standard for simulating 3-D full-wave electromagnetic fields.

The R&S Pulse Sequencer Digital supports the `*.ffd` multi-frequency far field pattern files but imports only the first frequency.

For information, see http://www.ansys.com/.

Example: ANSYS HFSS *.ffd file (extract)

The values in the columns are the complex values of the electromagnetic field in theta and phi direction.

```
0 180 19
0 360 37
Frequencies 1
Frequency 8.680000000000000e+008
-3.029304169961667e+000 -3.592562203897908e+000 6.108290737961166e-001 1.376043194687679e+000
-2.877212877279787e+000 -3.299035718428788e+000 1.127582356352358e+000 1.978979886470077e+000
```

Antenna patterns in *.csv file format

Several of the popular antenna design software tools export antenna patterns also as `*.csv` files.

The `*.csv` files are text files that describe the antenna pattern in a tabular structure. These files contain a header section and data part with up to six columns. The values can be separated by commas, semicolons, or withe spaces.

When a file is selected, the "Import Wizard" tries to determine the used column separator, decimal delimiter and to data content of the columns. You can change the values afterwards.

Following parameters describe an antenna (see Figure A-2):

- Theta
  Elevation angle with value range 0° to 180°, where Theta = 0° means negative Z-axis direction
- Phi
  Azimuth angle with value range 0° to 360°, where Phi = 0° means positive Y-axis direction
- Use constant resolution step for both antenna parameters.

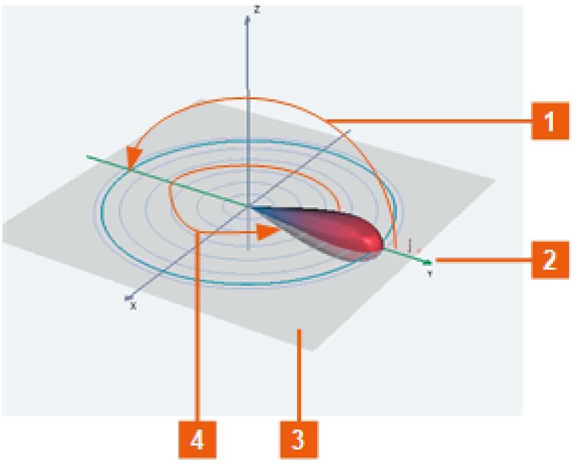


Figure A-2: Coordinate system

1 = Elevation angle with value range 0° to 180°
2 = Phi = 0°
3 = Theta = 0°
4 = Azimuth angle with value range 0° to 360°

Example: Antenna Magus file in *.csv format (extract)

```
FRQ=3.0E+9
PIN=0.171146799151093
THETA_SYM_0=0
THETA_SYM_90=0
PHI_SYM_0=0
PHI_SYM_90=0
INTERP_DBI=0
Theta(,deg) Phi(,deg) E_Theta(Re,) E_Theta(Im,) E_Phi(Re,) E_Phi(Im,)

# Far-field exported from Antenna Magus 5.3.0.1686
# On Friday, March 06, 2015 at 1:17:55 PM
# Theta Samples: 181
# Phi Samples: 361
0.000000000000E+000,0.000000000000E+000,4.303687774512E-016,0.000000000000E+000,-7.148424634567E+000,-1.436268526906E+000
1.000000000000E+000,0.000000000000E+000,4.425352450194E-016,0.000000000000E+000,-7.363209895869E+000,-1.542584165100E+000
...
```

Example: ANSYS HFSS files in *.csv format (extract)

The following are two examples of the ANSYS HFSS file in `*.csv` file format. The first file describes the electromagnetic fields, whereas the second one the gain in dBi.

```
Phi[deg],Theta[deg],re(rEPhi)[mV],im(rEPhi)[mV]
0,0,610.829,1376.04
10,0,1127.58,1978.98
...
```

```
Phi[deg],Theta[deg],dB(GainTotal)
0,0,-2.79687
10,0,-2.79687
...
```

Rohde&Schwarz proprietary antenna pattern *.ant_pat file format

The Table A-6 describes the used tags and parameters.

*Table A-6: Format of *.ant_pat file*

Element	Description
<antenna_pattern>	Root element of the antenna pattern file
<az_res>	Resolution of the columns in the `<data>` section Value in degrees integer divider of 360

Element	Description
<elev_res>	Resolution of the rows in the <data> section Value in degrees integer divider of 180
<data>	The file has to contain up to: ● [1 + 360/<az_res>] columns ● [1 + 180/<elev_res>] rows If the column resolution exceeds the resolution specified with the <az_res> element, an interpolation is applied. You can also define a subset of values, for example to define a beam antenna pattern. Missing values are internally set to zero, see Example"Beam antenna pattern" on page 635.

The following are two examples of the file format: the description of a beam-like antenna pattern and the description of an antenna pattern with back lobes and medium resolution.

Example: Beam antenna pattern

This example shows that partially defined antenna patterns are also allowed.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<antenna_pattern>
 <az_res> 5.00000000e+00 </az_res>
 <elev_res> 5.00000000e+00 </elev_res>
 <data>
        -30,-20,-10,0,  10, 20, 30
    -30,100,100,100,100,100,100,100,
    -20,100,100,100,100,100,100,100,
    -10,100,100,10, 100,10, 100,100,
      0,100,100,100,0,  100,100,100,
     10,100,100,10, 100,10,1 00,100,
     20,100,100,100,100,100,100,100,
     30,100,100,100,100,100,100,100,
 </data>
</antenna_pattern>
```

In this example, defined are only the subset of values that describe the beam pattern; all other values are assumed to be zero, see Figure A-3.

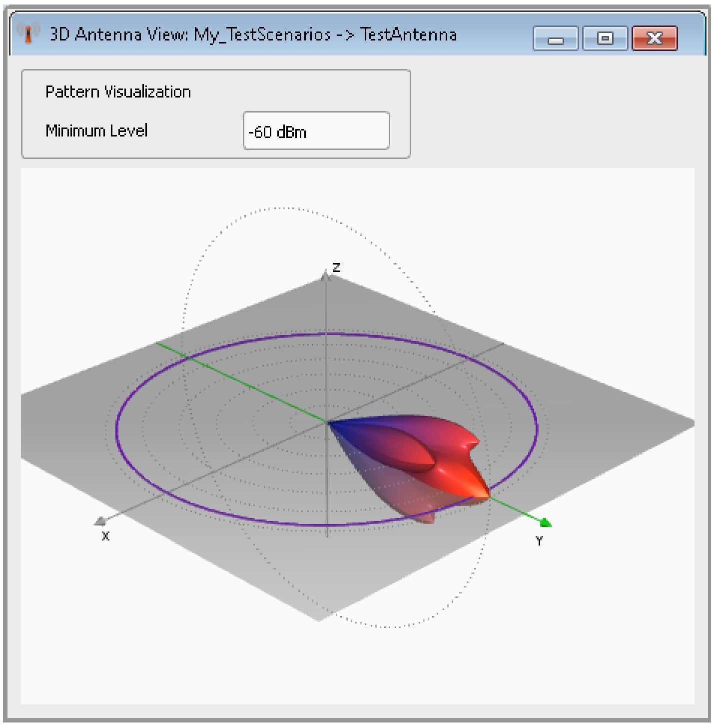


Figure A-3: 3D view of the imported partially defined antenna pattern

Example: Antenna pattern with back lobes

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<antenna_pattern>
<az_res> 5.00000000e+00 </az_res>
<elev_res> 5.00000000e+00 </elev_res>
<data>
 -180,-160,-140,-120,-100,-80,-60,-40,-20,0,20, 40, 60, 80, 100,120,140,160,180
 -90,100,100,100,100,100,100,100,100,100,30,100,100,100,100,100,100,100,100,100
 -80,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100
 -70,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100
 -60,100,100,100,100,100,100,100,100,100,20, 100,100,100,100,100,100,100,100,100
 -50,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100
 -40,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100
 -30,100,100,100,100,100,100,100,100,100,10, 100,100,100,100,100,100,100,100,100
 -20,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100
 -10,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100
 0, 30, 100,100,20 ,100,100,10, 100,100,0,  100,100,10, 100,100,20, 100,100,30
 10,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100
 20,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100
 30,100,100,100,100,100,100,100,100,100,10, 100,100,100,100,100,100,100,100,100
 40,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100
 50,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100
 60,100,100,100,100,100,100,100,100,100,20, 100,100,100,100,100,100,100,100,100
 70,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100
 80,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100
 90,100,100,100,100,100,100,100,100,100,30, 100,100,100,100,100,100,100,100,100
</data>
</antenna_pattern>
```

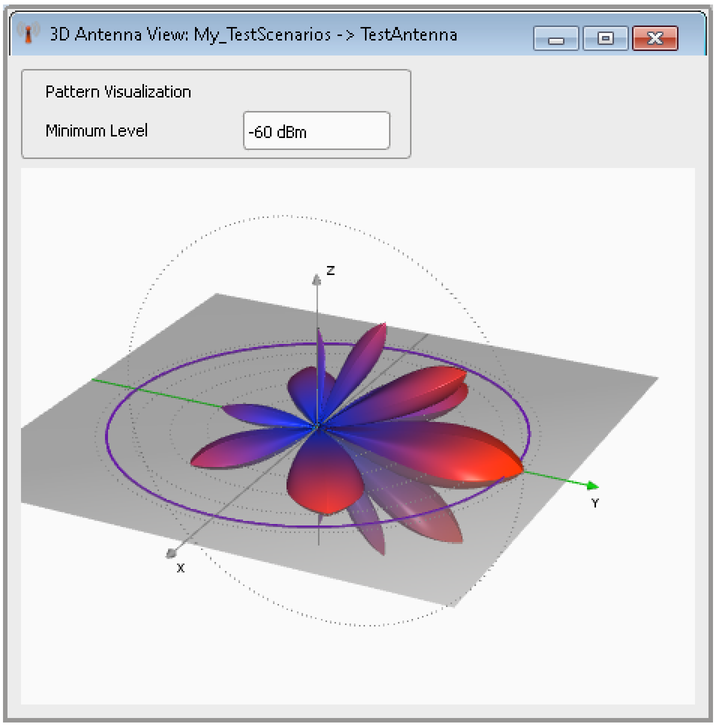


Figure A-4: 3D view of the imported antenna pattern with back lobes

A.3 Movement files

A.3.1 Waypoint file format

A waypoint file is a simple text file (`*.txt`) that describes a trajectory as a sequence of positions called waypoints. The file contains a resolution [ms] that give the time between two consecutive waypoints and a list of waypoint coordinates, one row per waypoint.

Each waypoint is specified with its longitude [° (decimal format)], latitude [° (decimal format)], altitude [m].

Example:

The resolution command at the beginning of the file specifies the time (in ms) between two consecutive waypoints. In this example, it is 50 ms. The value is different than 10 ms; the waypoint file is resampled.

```
RESOLUTION: 50
144.966666334601,-37.8166633061788,100.000000009313
144.966662392613,-37.8166632247233,100.000000039116
144.966658453002,-37.8166630889914,100.00000008475
144.966654516955,-37.8166628990241,100.000000149943
144.966650585658,-37.8166626548785,100.000000235625
144.966646660296,-37.8166623566284,100.000000339001
...
etc.
```

A.3.2 Vector trajectory file format

In the vector trajectory file format, motion is defined in an East North plane which is tangent to earth at the specified reference point. The curvature of the Earth is not considered in the interpretation of the `Up (U)` coordinate elements when moving east or north.

The vector trajectory file format is suitable for description of movements where the curvature of the earth is negligible. This file format is not intended for long-distance simulations where the curvature of the earth is significant.

This file format uses a script containing the commands defined in the following table:

Command	Description
`REFERENCE: lon, Lat, alt`	Specifies the Cartesian Reference of the ENU coordinates system. Given as a geodetic WGS84 point (longitude, Latitude, Altitude).
`START: E ,N, U, Velocity`	Start location is the initial "current location" in the ENU Cartesian coordinate of center `REFERENCE`. East, North and UP (ENU) coordinates are then provided (m). The last argument is the start velocity in (m/s).
`ARC: E, N, Angle`	Specifies a 2-Dimensional ARC (East, North) with the first two arguments representing the center of the ARC (m) in the Cartesian basis. The last argument specifies the angle in degrees (°) of the Arc Starting the "current location". Angle sign is significant because it indicates positive direction (counterclockwise) or negative direction (clockwise). The end edge of the arc represents the new "current location". Velocity does not change when using an ARC command.
`LINE: ΔE, ΔN, Acceleration`	Specifies a 2-dimensional line form the current location or first edge `(Current loc E, Current loc N)` to the next location or second edge `(Current loc E + ΔE, Current loc N + ΔN)` (m). The argument `Acceleration` specifies a constant acceleration/deceleration, where `Acceleration = 0` indicates constant velocity. Unit for acceleration is (m/s2). The second edge is used as the current location for the next command. The speed at this second edge is also used as the start speed for the next command.

Command	Description

`LINE3D: ΔE, ΔN, ΔU, Acceleration`	Describes a 3-dimensional straight line in ENU coordinates, where the first edge is described as `(E, N, U)` and the second edge as `(E + ΔE, N + ΔN, U + ΔU)` (m).
	The argument `Acceleration` specifies a constant acceleration/deceleration, where `Acceleration = 0` indicates constant velocity. Unit for acceleration is (m/s2).
	`*********** MOVEMENT FILE *****************`
	`RESOLUTION: 10`
	`REFERENCE: 0,0,0`
	`START: 0, 0, 10000, 20`
	`LINE3D: 0, 0, -10000, 0`

`STAY: Time`	Stay at the current location for `Time` period (ms).

Example: Example of a waypoint file

This example explains a waypoint file, describing a moving emitter on a rectangular trajectory of 940m by 1440m with rounded corners defined in Table A-7 and Figure A-5.

The initial reference is first defined followed by acceleration to final speed of 100 km/h in 250 m. The emitter then maintains the speed for 400 m. The speed then decreases to 25 km/h in 250 m. The UE then turn 90 degrees with turning radius of 20 m at 25 km/h. The speed increases to 100 km/h in 250 m.

The sequence is repeated to complete the rectangle.

Table A-7: 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

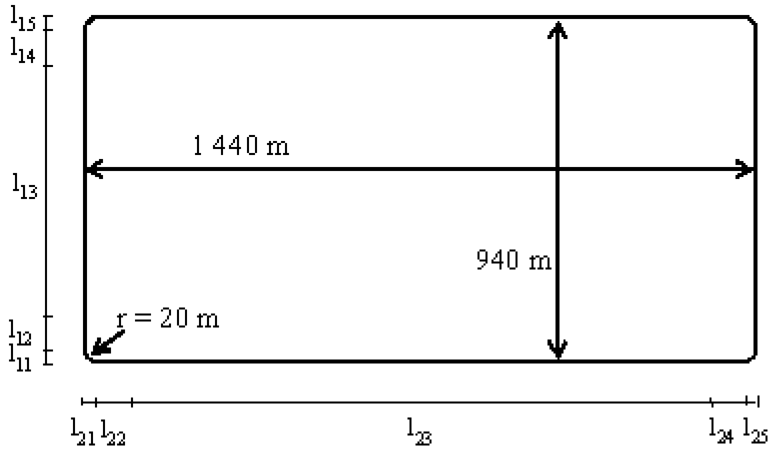


Figure A-5: Trajectory

The following is the content of the waypoint file.

The start tag `MOVEMENT FILE` is a mandatory element for each movement description. If this tag is omitted, the software interprets the coordinates described in the waypoint file as static one.

A.3.3 Trajectory description files

The trajectory description files use the file extension `*.xtd`. See Example"Trajectory description files in <positiononly> waypoint format" on page 643 for a simple example of the file format.

The Table A-8 describes the used tags and parameters.

*Table A-8: Format of *.xtd file*

Container	Tag name	Parameter	Description
<general>			
	<property>	<waypointformat>	Defines the format of the waypoint data. The possible values are: ● "positiononly" Only waypoint data ● "position_attitude" Waypoint and attitude data ● "position_velocitymagnitude" Waypoints and velocity magnitude information per point ● "position_velocitymagnitude_attitude" Waypoints, velocity magnitude and attitude data ● "position_velocityvector" Waypoints and velocity vector per location ● "position_velocityvector_attitude" Waypoints, velocity vector and attitude per point Smoothening is only possible with `<property waypointformat="positiononly">` and waypoints and `*.kml`. The description of the waypoints has to follow the selected format. For example, `<waypointvector data="0.05,144.966662392613,-37.8166632247233, 100.000000039116"/>` with `<property waypointformat="positiononly"/>` and `<datavectorhastimestamp="yes">`.
		<datavectorhastimestamp>	Determines the way the time stamp (elapsed time starting simulation time) is defined. ● With `<datavectorhastimestamp="yes">` The time stamp of a waypoint is retrieved from the `<waypointvector data>`, as the first data vector parameter. ● With `<datavectorhastimestamp="no">` the time stamp is internally calculated based on the following rule: – The time stamp of first waypoint is $T_1 = 0$ The time stamp of the waypoint$_N$ is T_N = (N–1)*`<timeresolution>`
		<coordsystem>	Defines the coordinate system used by the definition of the `<waypointvector data>`. Possible values: "enu", "cart ecef", "geod wgs84" or "geod pz90"
		<timeresolution>	If `<datavectorhastimestamp="no">`, applies implicitly time stamps to waypoints. Otherwise, the time stamp is retrieved automatically, see `<datavectorhastimestamp>`.
		<enurefpoint>	Coordinates of the reference waypoint in ENU format
		<endbehaviour>	Determines the behavior of the moving object at the end of the trajectory. Possible values: "jump", "return", "stop"
		<attitudeunit>	Defines the unit ("rad" or "deg") the attitude is expressed in
		<noofwaypoints>	Number of the used waypoints vectors. Use the parameter to minimize the size of the waypoint without actually deleting the waypoints

Container	Tag name	Parameter	Description
		<attitudecontinu-ous>	If `<attitudecontinuous="no">`, the attitude angles to be read are bounded by 2*PI. The `<attitudecontinuous="yes">` means that they are not boun-ded.
<waypoints>			Description of the trajectory as a list of waypoint vectors.
	<waypointvec-tor>	<data>	Coordinates of the waypoint in the format selected by `<coordsystem>` ● [time,]longitude,latitude,altitude (geod wgs84) ● [time,]X,Y,Z (cart ecef) ● [time,]EAST,NORTH,UPPER (enu) Depending on the `<datavectorhastimestamp>`, 3 or 4 values are evaluated with `<property waypointformat="positiononly"/>` Depending on the `<property waypointformat>`, more information can be provided. The following is a list of the information appended to the coordinates of the waypoint mentioned above. ● `<property waypointformat="positiononly">` No additional information ● `<property waypointformat="position_attitude">` **yaw/heading,pitch/elevation,roll/bank**, e.g. `[time,]longitude,latitude,altitude,yaw,pitch,roll` The vehicle attitude (yaw, pitch and roll) parameters are configured relative to the local horizon ● `<property waypointformat="position_velocitymagnitude">` **vel**, e.g. `[time,]X,Y,Z,vel` ● `<property waypointformat="position_velocitymagnitude_attitude">` **vel,yaw/heading,pitch/elevation,roll/bank**, e.g. `[time,]EAST,NORTH,UPPER,vel,yaw,pitch,roll` ● `<property waypointformat="position_velocityvector">` Velocity vector **Vx,Vy,Vz**, e.g. `[time,]X,Y,Z,Vx,Vy,Vz` ● `<property waypointformat="position_velocityvector_attitude">` **yaw/heading,pitch/elevation,roll/bank,Vx,Vy,Vz**, e.g. `[time,]longitude,latitude,altitude,yaw,pitch,roll, Vx,Vy,Vz.` **Note:** If the waypoints are in geod wgs84, velocity vector is assumed in Cartesian ECEF coordinate system.

Example: Trajectory description files in <positiononly> waypoint format

The following are two examples in `<property waypointformat="positiononly">` format. The examples emphasize on the different formats and explain the used time stamp principle.

- The `<waypointvector data>` parameters in the following example are vectors with size of 3, because of the tag `<property datavectorhastimestamp="no">`. Even if a fourth value has been defined, it would have been ignored.
  The time stamp is defined with the tag `<property timeresolution="0.05">`: T_1 = 0, T_2 = (2-1)*0.05 = 0.05, and so on, T_{12} = (12-1)*0.05 = 0.55

```
<trajectory>
    <general>
        <property waypointformat="positiononly"/>
        <property datavectorhastimestamp="no"/>
```

```
        <property coordsystem="enu"/>
        <property timeresolution="0.05"/>
        <property enurefpoint="54.0,10.0,12"/>
        <property endbehaviour="return"/>
        <property noofwaypoints="12"/>
    </general>
    <waypoints>
        <waypointvector data="0,0,0"/>
        <waypointvector data="0,1,0"/>
        <waypointvector data="0,2,0"/>
        <waypointvector data="0,3,0"/>
        <waypointvector data="0,4,0"/>
        <waypointvector data="0,5,0"/>
        <waypointvector data="0,6,0"/>
        <waypointvector data="0,7,0"/>
        <waypointvector data="0,8,0"/>
        <waypointvector data="0,9,0"/>
        <waypointvector data="0,10,0"/>
        <waypointvector data="0,11,0"/>
</waypoints>
</trajectory>
```

- In the following example, the tag `<property datavectorhastimestamp="yes">` and all four values in the `<waypointvector data>` parameters are evaluated. The time stamp is retrieved automatically; the time stamp of a waypoint is the first value in the corresponding `<waypointvector data>` parameter:
  $T_1 = 0$, $T_2 = 0.05$, and so on, $T_{12} = 0.55$.

```
<trajectory>
    <general>
        <property waypointformat="positiononly"/>
        <property datavectorhastimestamp="yes"/>
        <property coordsystem="geod wgs84" />
        <property endbehaviour="return"/>
        <property duration="2.0"/>
    </general>
    <waypoints>
        <waypointvector data="0,144.966666334601,-37.8166633061788,100.000000009313"/>
        <waypointvector data="0.05,144.966662392613,-37.8166632247233,100.000000039116"/>
        <waypointvector data="0.10,144.966658453002,-37.8166630889914,100.00000008475"/>
        <waypointvector data="0.15,144.966654516955,-37.8166628990241,100.000000149943"/>
        <waypointvector data="0.20,144.966650585658,-37.8166626548785,100.000000235625"/>
        <waypointvector data="0.25,144.966646660296,-37.8166623566284,100.000000339001"/>
        <waypointvector data="0.30,144.966642742053,-37.8166620043635,100.000000461936"/>
        <waypointvector data="0.35,144.966638832109,-37.81666159819,100.000000603497"/>
        <waypointvector data="0.40,144.966634931642,-37.8166611382304,100.000000762753"/>
        <waypointvector data="0.45,144.96663104183,-37.8166606246233,100.000000941567"/>
        <waypointvector data="0.50,144.966627163843,-37.8166600575235,100.000001138076"/>
        <waypointvector data="0.55,144.96662329885,-37.8166594371019,100.000001354143"/>
```

```
</waypoints>
</trajectory>
```

A.4 Vehicle description files (Used for smoothening)

The vehicle description files use the file extension `*.xvd`. The following is a simple example of the file format.

```
<vehicle>
    <info name="Car"/>
    <limits>
        <property maxspeed="100.0"/>
        <property maxg="0.6"/>
        <property maxg_lateral="0.6"/>
        <property maxjerk="15"/>
        <property maxjerk_lateral="15"/>
        <property maxyawrate="0"/>
        <property maxyawacceleration="0"/>
        <property maxyawjerk="0"/>
        <property maxpitchangle="0"/>
        <property maxpitchrate="0"/>
        <property maxpitchacceleration="0"/>
        <property maxpitchjerk="0"/>
        <property maxrollangle="0"/>
        <property maxrollrate="0"/>
        <property maxrollacceleration="0"/>
        <property maxrolljerk="0"/>
        <property proximity="0"/>
    </limits>
</vehicle>
```

The Table A-9 describes the used tags and parameters.

*Table A-9: Format of *.xvd file*

Container	Tag name	Parameter	Description
<info>			
	<name>		Vehicle name
<limits>			
	<property>	<maxspeed>	Maximal Speed (m/s)
		<maxg>	Maximum tangential acceleration in g unit
		<maxg_lateral>	Maximum radial acceleration (due to centrifugal force) in g unit
		<maxjerk>	Maximum tangential Jerk in m/s2
		<maxjerk_lateral>	Maximum radial Jerk in m/s2
		<maxyawrate>	Maximum rate of change of the yaw/heading (rad/s)

Container	Tag name	Parameter	Description
		<maxyawacceleration>	Maximum second rate of change (acceleration) of the yaw/heading (rad/s2)
		<maxyawjerk>	Maximum third rate of change (jerk) of the yaw/heading (rad/s3)
		<maxpitchangle>	Maximum pitch/elevation angle (rad)
		<maxpitchrate>	Maximum rate of change of the pitch/elevation (rad/s)
		<maxpitchacceleration>	Maximum second rate of change (acceleration) of the pitch/elevation (rad/s2)
		<maxpitchjerk>	Maximum third rate of change (jerk) of the pitch/elevation (rad/s3)
		<maxrollangle>	Maximum roll/bank angle (rad)
		<maxrollrate>	Maximum rate of change of the roll/bank (rad/s)
		<maxrollacceleration>	Maximum second rate of change (acceleration) of the roll/bank (rad/s2)
		<maxrolljerk>	Maximum third rate of change (jerk) of the roll/bank (rad/s3)
		<proximity>	Specifies the maximum allowed deviation from the original waypoints (m)

B Formula syntax

You can use mathematical expression to define custom modulation types or envelope shapes.

The R&S Pulse Sequencer Digital uses the fast math parser library muParser, that is an extensible high performance math expression parser library written in C++.

This section list some of the default features supported by the parser. For detailed information, see the muParser product home page http://beltoforion.de/article.php?a=muparser&p=features.

Basic syntax elements

Table B-1: Mathematical operations

Element	Description
+	Addition
-	Subtraction
*	Multiplication
/	Division
^	By the power of

Table B-2: Built-in functions (Extract)

Element	Description
sin	sine function
cos	cosine function
exp	raised to the power of x
etc.	

Table B-3: Other operators

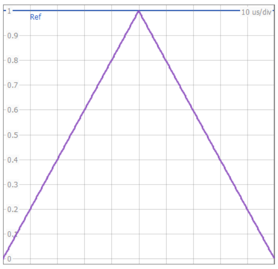
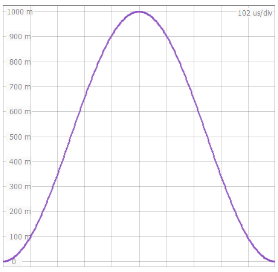
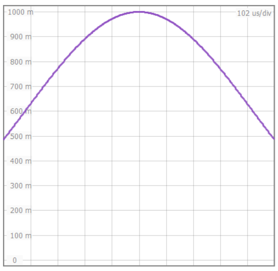
Element	Description	Syntax
?:	if then else operator	<if_condition>**?**<then_expression>**:**<else_expression>

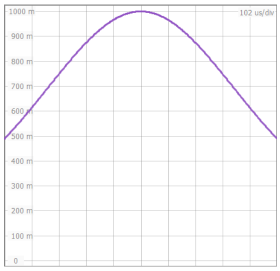
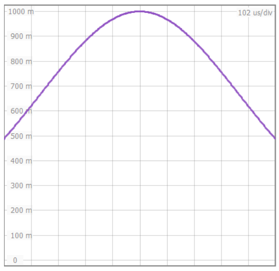
Table B-4: Constant and variables

Element		Description
Constants:	PI	π
	T_r	Pulse rise time
	T_w	Pulse width time
	T_f	Pulse fall time

Element		Description
	T $T = T_r+T_w+T_f$	Total pulse time
Variables:	ip<n>	Inter-pulse modulation value <n>
	t t = 0 .. T	Time

Equations and parameters used to define custom pulse envelopes

Envelope name	Equation	Envelope shape
"Rectangular Pulse"	1	
"Triangular Pulse"	(t<T/2) ? t/(T/2) :1-(t-T/2)/(T/2)	
"Raised Cosine Pulse"	0.5 * (1 - cos(2 * PI * t/T))	
"Gaussian Pulse"	exp(-4 * ln(2) * ((t-Tr-Tw/2) / Tw)^2)	

Envelope name	Equation	Envelope shape
"Lorentzian Pulse"	1 / ((1 + 4 * (sqrt(2)-1) * ((t-Tr-Tw/2) / Tw)^2))^2	
"Sech Pulse"	1 / (cosh((2 * (t-Tr-Tw/2) * ln(1+sqrt(2))) / Tw))^2	

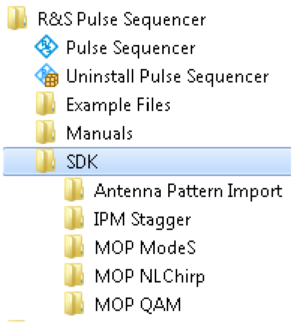
Where Amplitude value range is 0 V to 1 V.

C Plug-in programming API

The functionality of the R&S Pulse Sequencer Digital can be extended by plug-ins. A plug-in is a Microsoft Windows DLL. This DLL can be loaded into a repository and then remains there.

You can load plug-ins to:

- Define a custom inter-pulse modulation profile (IPM)
- Create a report file
- Define the file format of the custom antenna pattern files.



Access:

A subset of example plug-ins is included in the software.

▶ On your PC, open "Start > All Programs > R&S Pulse Sequencer Digital > SDK".

You can load the plug-ins in your own repository, try out them, and further develop them.
See "To import a plugin" on page 377.

The following is a list of all functions that the plug-in has to provide:

C.1 Common functions

getType (char szModType[1024])

(mandatory)

Queries the plug-in type.

The type defines the purpose of the plug-in. It is the most important parameter returned to the main application.

Do not change the type of the plug-in after it is used within a repository. The type also defines which functions have to be provided by the plug-in (API). It also affects in which R&S Pulse Sequencer Digital dialogs the plug-in becomes visible.

Parameters:

szModType string

(return parameter)
A keyword that denominates the plug-in type.

"modulation" | "mop"
MOP plug-in

"ipm"
IPM plug-in

"report"
Reports plug-in for reports created during the waveform generation

"export"
Customer written plugin which receives PDW data from the R&S Pulse Sequencer Digital.

getVersion (char szModVer[1024])

(mandatory)

Queries the version string of the plug-in.

The version string is not further evaluated in the software. It only serves as an additional piece of information. The returned version string is visible on the plug-in dialog panel.

Parameters:

szModVer "<Major>.<Minor>.<Revision>"

(return parameter)
`<Major>`, `<Minor>`, and `<Revision>` are numerical values.

getComment (char szModComment[4096])

(mandatory)

Queries the comment string of the plug-in.

The comment string is not further evaluated in the software. It only serves as an additional piece of information. The returned comment string is visible on the plug-in dialog panel.

Parameters:

szModComment string

(return parameter)
Line breaks are permissible. To generate new line, insert `'\n'` into the string.

---

getAuthor (char szModAuthor[1024])

(mandatory)

Queries the author string of the plug-in.

The author string is not further evaluated in the software. It only serves as an additional piece of information. The returned author string is visible on the plug-in dialog panel.

Parameters:

szModAuthor string

(return parameter)
Empty string is allowed

---

getName (char szModName[1024])

(mandatory)

Queries the plug-in name.

The name string is not further evaluated in the software. It only serves as an additional piece of information. However, be careful changing names once the plug-in is in use. The returned name string is visible on the plug-in dialog panel.

Return values:

szModName string

(return parameter)

---

getError (char szModError[1024])

(mandatory)

Queries the plug-ins for errors.

This function is called if a plug-in function returns an error. In addition, the function is called repeatedly to poll the plug-in for errors that are not related to a specific function call. This function must return an empty string if there are no pending error messages. Returning an error stops the R&S Pulse Sequencer Digital calculations.

Error messages should only be returned once.

Parameters:

szModError string

(return parameter)
Error message as zero-terminated ASCII string.

int getNextMsg (char szInfoMsg[4096])

(mandatory)

Sends messages to the application. The messages are passed to the "Message Log" of the application, merely for user information.

Parameters:

szInfoMsg "<Prefix>:<Text>"

(return parameter)
The message can contain a prefix that denominates the type of information sent.
`D` = Debug
`I` = Information
`W` = Warning
`E` = Error
If no prefix is present, the information type (`I`) is assumed.

Return values:

boolean The function is called repeatedly until it returns false (0).

int initPlugin (void)

(mandatory)

This function initializes the plug-in. It is called once after the plug-in is loaded into memory and can set internal variables.

Return values:

boolean **true (1)**
The initialization completed successfully

false (0)
Error during the initialization. The plug-in is unloaded from memory

shutdownPlugin (void)

(optional)

This function is called before the plug-in gets unloaded from memory. This function can be used to release allocated resources. A plug-in is unloaded from memory if it gets removed from the repository or the main application closes.

int getPrivateMemSize (void)

(optional)

This function sets the required amount of user data.

Avoid storing parameter settings as global variables because global variables are shared among multiple instances of the plugin. Register a user data block in the application and pass the pointer `pcPrivateMem` as argument instead.

Return values:

PrivateMemSize	Required size of user memory in bytes

---

int getVariable (int iIndex, char szConfig[4096])

(mandatory)

Plug-ins can register variables in the R&S Pulse Sequencer Digital user interface. These variables are initialized with a default value but you can change the value later. The modified variable value is stored as part of the repository. This mechanism allows the reuse of a plug-in in different configurations.

This function is called repeatedly by the main application and reads information about the variables to be registered. On each function call, the zero-based variable index value `iIndex` increases by one. The index uniquely references a variable.

Parameters:

iIndex
: **(input parameter)**
Zero-based variable index

szConfig
: **(return parameter)**
The string of variables to be registered is composed as follows:

DBL,<name>,<unit>,<default>,<min>,<max>,<dec. digits>
Real Number

INT,<name>,<unit>,<default>,<min>,<max>
Integer

BOOL,<name>,<default>
Boolean

STR,<name>,<default>[,<validator-regexp>]
String

SEL,<name>,<opt>|<opt>|...,<default index>
Selection
Value range of `<default index>`: 0 to (number of <opt>-1)

Return values:

boolean
: **true**
A variable is registered under this index number

false
No variable was registered; the function is not called anymore

Example: `SEL,Type,16-QAM|32-QAM|64-QAM|256-QAM,2`
This field is named "Type" and lists the options:
16-QAM, 32-QAM, 64-QAM, and 128-QAM
The default value is 64-QAM

---

int setVariable (int iIndex, const char szValue[4096])
int setVariableEx (void *pcPrivateMem, int iIndex, const char szValue[4096])

(mandatory) - for R&S Pulse Sequencer Digital export plug-in.

(optional)

Sets the value of a variable. The variable is referenced by the same zero-based index `iIndex` used in `getVariable()`. The variable value is transferred as string. This string must be interpreted according to the variable data type.

Parameters:

pcPrivateMem
(input parameter)
User data memory pointer, see `int getPrivateMemSize`.

iIndex
(input parameter)
The zero-based index value references the variable

szValue
Zero-terminated ASCII string
(input parameter)
Variable value as formatted ASCII string. Use functions such as `atof()` or `atoi()` for conversion.

C.2 Export plugin

---

int initExport (void)

(mandatory)

This function is used to initialize the plugin after it is loaded. The function is called with each PDW calculation.

Return values:

boolean
true (1)
The initialization completed successfully
false (0)
Error during the initialization or the parameter is not used. The plug-in is unloaded from memory

---

bool requiresSCPIRequest (char szRequest[1024], int *ID)

(optional)

Returns requested SCPI string.

Use this function for accessing the whole simulation database if information is required that is not part of the PDW export format. Parameters could include emitter names, antenna types, IPM settings and so on.

Parameters:

szRequest	**(output parameter)** Plug-in SCPI string. Max 1023 characters. The function is called to retrieve SCPI queries from the plugin
ID	**(output parameter)** Each query needs an ID. This ID can be used in the `SCPIReply` function to identify the answer.
boolean	**true (1)** Indicates to the R&S Pulse Sequencer Digital that the plug-in will query additional information **false (0)** Indicates that no more information is needed

---

bool SCPIReply (const char *szReply, int ID, bool bOK)

(optional)

Returns SCPI query result.

Parameters:

szReply	**(input parameter)** SCPI string. Max 1023 characters. The function is called by the R&S Pulse Sequencer Digital to answer SCPI queries from the plug-in.
ID	**(input parameter)** Each query has an ID. This ID can be used inside this function to identify the answer.
boolean	**true (1)** Indicates to the R&S Pulse Sequencer Digital that the SCPI reply was received

---

bool start

(mandatory)

Signals that the calculation starts.

The R&S Pulse Sequencer Digital calls this function at the start of the calculation before the first PDWs are exported in the `exportPDW` function.

Use this function for performing tasks before the simulation data transfer, such as opening a file or a socket.

Parameters:

boolean	true (1) Successful

bool ready

(mandatory)

Signals that the calculation is ready.

The R&S Pulse Sequencer Digital calls this function after the calculation has finished and the last PDW has been exported in the `exportPDW` function.

Use this function for performing tasks after the simulation data was transferred, such as closing a file, cleaning up temporary files or closing a socket.

Parameters:

boolean	true (1) Successful

bool exportPDW (const char *szPDW)

(mandatory)

Exports PDW parameters.

This function receives the PDW data in a predefined data format.

Use this function to extract all required information for your use case, adapt the data to your data formats, write it to a file, stream it to a socket or transfer it as required.

Parameters:

szPDW	**(input parameter)** PDW simulation data in a defined format, typecast the data pointer accordingly.
boolean	true (1) Successful

bool run

(mandatory)

Signals run.

Corresponds to selecting "Run" in the R&S Pulse Sequencer Digital application or using the corresponding SCPI command.

Use this function to perform tasks after the simulation data has been received, such as streaming it to a socket or performing postprocessing.

Parameters:

boolean	true (1) Successful

bool is running

(mandatory)

Signals run state.

Parameters:

boolean true (1)
Indicates to the R&S Pulse Sequencer Digital that the plug-in is currently busy.

bool stop

(mandatory)

Signals stop.

Use this function to perform tasks the same as if you select "Stop" in the R&S Pulse Sequencer Digital application or using the corresponding SCPI command, for example aborting file writes, streaming or postprocessing tasks.

Parameters:

boolean true (1)
Successful

C.3 IPM functions

setRndFnPtr (void *pRndFn, void *pThis)

(optional)

Sets a pointer to a random function `pRndFn` and the associated `pThis` pointer. Both pointers are needed for calling the built-in random number function.

Plug-ins must use the random generator provided by the application. This generator produces a random sequence with a defined start seed. If necessary, the application can reproduce a particular calculation with the same values. This mechanism is required because the main application performs an initial dry run to determine parameters such as sequence duration and clock rate. This random generator ensures that the same values are used during dry run and the final calculation.

The prototype of the callback function is defined as follows:

```
typedef int (*RndFnCallback)( void*, int );
```

The random generator is called using the provided pointers:

```
int iRnd = pRndFn( pThis, iMaxVal );
```

The random generator returns a 32-bit random unsigned integer value in the range 0 to 65535.

restart ( )
restartEx (void *pcPrivateMem)

(optional)

Restarts the IPM pattern generation.

This function is called if the IPM pattern generation is requested to restart. The sequence "IPM Configuration" dialog contains the option "Restart IPM for this line item". Setting this option calls this function at the beginning of the line item calculation.

This option is useful if IPM patterns follow a certain rule or have memory, e.g. lists where items shall not be reused.

Parameters:

pcPrivateMem	**(input parameter)** User data memory pointer, see `int getPrivateMemSize`

---

int calculateNextIpmValue (double dTime, uint64 iCount, double *pdValue)
int calculateNextIpmValueEx (void *pcPrivateMem, double dTime, uint64 iCount, double *pdValue)

(mandatory)

Requests the next IPM value from the IPM plug-in.

IPM values can either be calculated based on a timestamp or an index. The time is the start time of the current sequence line item, e.g. the pulse start time. The index is an integer number that increases by one each time an IPM value is requested from the plug-in.

Parameters:

pcPrivateMem	**(input parameter)** User data memory pointer, see `int getPrivateMemSize`
dTime	**(input parameter)** Timestamp of the next IPM value
iCount	**(input parameter)** 64-bit unsigned integer value Zero-based index of the next IPM value
pdValue	**(return parameter)** Calculated IPM value.

Return values:

boolean	**true (1)** Calculation successful

C.4 Reporting functions

For information on the report generation function, see:

- Chapter 21, "Creating reports and documenting measurement results", on page 386

The following is a list of plug-in functions for creating reports:

---

setParam (const char *szType, const char szValue[1024])

(mandatory)

Sets various report parameters.

This function is repeatedly called. It sets different parameters on each call.

Because pulses can have different parameters, some pulse parameters have to be set before the calculation of the particular pulse starts. User variables that are defined and used within a sequence are set with their names (<user_var>) and values.

Parameters:

szType

Zero-terminated ASCII string

(input parameter)
Sets the type of data that is transmitted.

"REPOSITORY"
String
Repository name

„SCENARIO“
String
Scenario name

„ISODATE“
String
Date & time in ISO format

„VERSION“
String
Repository version

„AUTHOR“
String
Repository author

„TPATH“
String
Target file path of the directory in that the generated report file is stored

	„TOA“ Float Time of arrival (timestamp of pulse)
	„PW“ Float Pulse width (0% - 0%)
	„PRI“ Float PRI of current pulse (0% - 0%)
	„BW“ Float RF bandwidth (FMOP) incl. BB filter
	„RFC“ Float Center frequency of emitter / sequence
	„@ITEM“ Integer Sequence line item, 1 to N
	„@COLL“ Integer Collection line item, 1 to L
	„@CNT“ Integer Repetition count
	„@REP“ Integer Current repetition, 1...M
	„<user_var>“ Variable type, depending on the variable User variable name
szValue	Zero-terminated ASCII string **(input parameter)**

---

reportBegin ( )

(mandatory)

A new report begins.

This function is called once at the beginning of a calculation run. In case of collection scenarios, this function is called once before the first collection item gets processed.

The function can be used to open the report file and to set the header information.

---

reportEnd ( )

(mandatory)

A report ends.

This function is called once when all calculations are completed. In case of collection scenarios, this function is called once when the last collection item is calculated.

---

reportEntryComplete ( )

(mandatory)

A pulse entry is generated.

This function is called after each pulse. It creates the report entry for each pulse.

C.5 Custom antenna pattern import functions

The following is a list of functions for importing antenna patterns in custom file format.

---

bool importPattern (const char *szFile)

(mandatory)

Run the pattern import procedure.

This function imports the antenna pattern into the internal pattern memory. Since this import is done in the foreground thread, it should not take too much time.

Return values:

boolean	**true** Success

---

getConversionInfo (char szInfo[4096])

(optional)

Get the result from the antenna pattern import.

This function is called after the `importPattern()` call. It returns general information and error messages from the import procedure.

Parameters:

szInfo	"<Prefix>:<Text>" **(return parameter)**

The text can contain multiple lines. Each line can start with a prefix that denominates the information type.
`D` = Debug
`I` = Information
`W` = Warning
`E` = Error

---

int getPatternCount ( )

(mandatory)

Read the number of imported patterns.

This function is the first one that gets called by the main application after a successful import.

---

bool getFrequency (int iPatIndex, double *pdFMin, double *pdFMax)

(mandatory)

Get the frequency for a specific pattern.

This function is called once for each imported pattern. It returns the *frequency range* assigned to an antenna pattern.

Parameters:

iPatIndex	Zero-based antenna index

Return values:

boolean	**true** Success

---

bool getThetaRange (int iPatIndex, double *pdTMin, double *pdTMax, int *piValueCnt)

(mandatory)

Get the theta range of a specific pattern.

This function is called once for each imported pattern. It returns the *theta range* assigned to an antenna pattern.

Parameters:

iPatIndex	Zero-based antenna index
pdTMin	Minimum theta angle
pdTMax	Maximum theta angle
piValueCnt	Number of theta values

Return values:

boolean	**true** Success

bool getPhiRange (int iPatIndex, double *pdPMin, double *pdPMax, int *piValueCnt)

(mandatory)

Get the phi range of a specific pattern.

This function is called once for each imported pattern. It returns the *phi range* assigned to an antenna pattern.

Parameters:

iPatIndex	Zero-based antenna index
pdPMin	Minimum phi angle
pdPMax	Maximum phi angle
piValueCnt	Number of phi values

Return values:

boolean	**true** Success

bool getPatValue (int iPatIndex, int iPhiIndex, int iThetaIndex, float *pfEHorizontal, float *pfEVertical, float *pfPhaseVertical)

(mandatory)

Get one data point from a pattern.

This function is called once for each imported pattern and once for each data point. It returns the *pattern data* to the main application.

Parameters:

iPatIndex	Zero-based antenna index
iPhiIndex	Zero-based phi index
iThetaIndex	Zero-based theta index
pfEHorizontal	Voltage-based gain, horizontal
pfEVertical	Voltage-based gain, vertical
pfPhaseVertical	Vertical phase with respect to horizontal phase

Return values:

boolean	**true** Success

cleanUp (void)

Clean up all data.

This function is called after all data has been read from the plug-in. All memory can now be freed.

C.6 List of Plugin Functions

D Scripting API

The detailed description of the script syntax is out of scope of this user manual. Refer to the script standard, that can be found under the following link:

http://www.ecma-international.org/publications/standards/Ecma-262.htm

Alternatively, any tutorial or documentation on JavaScript could be used.

In addition to the script standard, the R&S Pulse Sequencer Digital introduces 2 proprietary objects: the global object `ps` and the newable object `QVisa`.

The objects `QDir, QFile, QMessageBox, QFileDialog` are `Qt` classes, where R&S Pulse Sequencer Digital supports only a subset of the methods.

In this section, you find a list of the supported methods per class and the description of the proprietary methods and objects.

The standard `Qt` classes and methods are not described here. For information, refer to the `Qt` documentation:

http://doc.qt.io/qt-5/index.html

D.1 Global script extensions

The global object `ps` extents the script functionality. It provides the following methods within the script context:

```
QString         scpi            ( const QString &sOut )

bool            sync            ( int iTimeout )

void            traceError      ( const QString &sMsg )
void            traceDebug      ( const QString &sMsg )
void            traceInfo       ( const QString &sMsg )

void            print           ( const QString &sMsg )

void            sleep           ( unsigned int iMs )

void            setGlobalVar    ( const QString &sToken, const QString &sValue )
QString         getGlobalVar    ( const QString &sToken )
QString         getScriptPath   ( )
```

ps.scpi (const QString &sOut)

Remote control of the application itself.

This method can be used as command or query. In the latter case, the returned data is a string value.

Example:

```
ps.scpi( '*RST' )
var sIdent = ps.scpi( '*IDN?' )
```

ps.sync (int iTimeout)

Suspends script execution until the R&S Pulse Sequencer Digital has finished a calculation task.

This is done by polling the operation condition register of the SCPI parser and wait for the command `STATus:OPERation:CONDition?` to return zero.

The method returns true if the register value becomes zero within the timeout period `iTimeout`. If there is a timeout, the method returns false.

Example:

```
var iTimoMs = 5000
ps.sync( iTimoMs )
```

ps.traceError (const QString &sMsg)
ps.traceDebug (const QString &sMsg)
ps.traceInfo (const QString &sMsg)

Traces log messages and prints messages to the "Message Log" dialog.

Three methods exist to output text messages to the application message log. By default, the message log shows only information and error messages.

Perform one of the following to enable debug output:

- Start the application with the command-line argument `--debug`
- In the command console window, enter *Set Debug = 1*.
  See also "To open the "Message Log" dialog in debug mode" on page 620.

Example:

```
ps.info( 'This is an info text.' )
ps.debug( 'This is a debug message.' )
ps.error( 'This is an error message.' )
```

ps.print (const QString &sMsg)

Printing any text or string to the "Script Manager" dialog.

Convert non-string variables into strings before using them with this method.

Example:

```
ps.print( 'My Text' )
var dVal = 1e-3
ps.print( 'Value = ' + dVal.toString() )
```

---

ps.sleep (unsigned int iMs)

Sets a waiting period specified in milliseconds and suspends script execution for at least this amount of time.

Example:

```
var iTimeMs = 500
ps.sleep( iTimeMs )
```

---

ps.setGlobalVar (const QString &sToken, const QString &sValue)
ps.getGlobalVar (const QString &sToken)

Exchanging Data Between Scripts.

The `ps` script extension provides two methods that access a global variable pool. These variables are of the data type string and remain valid for as long as the application runs. The variables are not saved when the application terminates. Each variable is identified by a name token.

The method `setGlobalVar` registers a new variable or updates an existing variable. The method `getGlobalVar` reads data from a variable. If the variable does not exist, an empty string is returned.

Example:

```
ps.setGlobalVar( 'Var1', 'This is some text.' )
ps.setGlobalVar( 'Var2', dValue.toString() )

dValue = parseFloat( ps.getGlobalVar( 'Var2' ) )
```

---

ps.getScriptPath ( )

Returns the absolute file path of the directory where the script file is stored.

D.2 Newable script extensions

The term newable describes objects that can be created dynamically using the `new` operator. A script can create multiple instances of these objects. The script engine automatically destroys all dynamically created objects at the end of the script.

D.2.1 QDir

For description of the `QDir` class, see http://doc.qt.io/qt-5/qdir.html.

The following `QDir` methods are available within script context:

```
QString         absoluteFilePath    ( const QString & fileName ) const
QString         absolutePath        ( ) const
bool            cd                  ( const QString & dirName )
bool            cdUp                ( )
QString         dirName             ( ) const
bool            exists              ( ) const
QString         filePath            ( const QString & fileName ) const
bool            isReadable          ( ) const
bool            isRoot              ( ) const
bool            mkdir               ( const QString & dirName ) const
bool            mkpath              ( const QString & dirPath ) const
QString         path                ( ) const
bool            remove              ( const QString & fileName )
bool            rmdir               ( const QString & dirName ) const
void            setPath             ( const QString & path )
QStringList     entryList           ( quint32 filters,
                                      quint32 sort ) const
// static
QString         currentPath         ( )
QString         homePath            ( )
QString         rootPath            ( )
QString         tempPath            ( )
```

D.2.2 QFile

For description of the `QFile` class, see http://doc.qt.io/qt-5/qfile.html.

The two methods `writeString` and `readLine` are Rohde & Schwarz proprietary methods. They work on the `QTextStream` method.

The following `QFile` methods are available within script context:

```
bool                exists      ( ) const
bool                open        ( quint32 mode )
bool                remove      ( )
void                setFileName ( const QString & name )
bool                copy        ( const QString & newName )
qint64              size        ( ) const
void                close       ( )
quint32             error       ( ) const

// reimplemented on QTextStream
bool                seek        ( qint64 pos )
void                flush       ( )
bool                atEnd       ( ) const

// custom methods (not part of QFile, work on QTextStream)
```

```
bool              writeString ( const QString &sStr )
QString           readLine    ( int iMaxLen )
```

file.writeString (const QString &sStr)

Writes the string to the selected file. Strings are appended as separate lines.

file.readLine (int iMaxLen)

Reads the next line of text from the file.

Setting parameters:
iMaxLen Maximum number of characters to be read.

D.2.3 QFileDialog

For description of the `QFileDialogFile` class, see http://doc.qt.io/qt-5/qfiledialog.html.

The following `QFileDialogFile` methods are available within script context:

```
void          setAcceptMode       ( quint32 mode )
void          setDefaultSuffix    ( const QString & suffix )
void          setDirectory        ( const QString & directory )
void          setFileMode         ( quint32 mode )
void          setNameFilter       ( const QString & filter )
void          setOption           ( quint32 option, bool on )
void          setViewMode         ( quint32 mode )
QStringList   selectedFiles       ( ) const
int           exec                ( )

// static public method
QString       getOpenFileName     ( const QString & caption,
                                    const QString & dir,
                                    const QString & filter,
                                    quint32 options )

QString       getExistingDirectory( const QString & caption,
                                    const QString & dir,
                                    quint32 options )

QString       getSaveFileName     ( const QString & caption,
                                    const QString & dir,
                                    const QString & filter,
                                    quint32 options )
```

D.2.4 QMessageBox

For description of the `QMessageBox` class, see http://doc.qt.io/qt-5/qmessagebox.html.

The following `QMessageBox` methods are available within script context:

```
void          setDetailedText      ( const QString & text )
void          setIcon              ( quint32 Icon )
void          setInformativeText   ( const QString & text )
void          setStandardButtons   ( quint32 buttons )
void          setText              ( const QString & text )
void          setTextFormat        ( quint32 format )
void          setWindowTitle       ( const QString & title )

// static methods
int           critical             ( const QString & title,
                                       const QString & text,
                                       quint32 buttons = QMessageBox::Ok,
                                       quint32 defaultButton = QMessageBox::NoButton )

quint32       information          ( const QString & title,
                                       const QString & text,
                                       quint32 buttons = QMessageBox::Ok,
                                       quint32 defaultButton = QMessageBox::NoButton )

int           question             ( const QString & title,
                                       const QString & text,
                                       quint32 buttons = QMessageBox::StandardButtons
                                       ( QMessageBox::Yes | QMessageBox::No ),
                                       quint32 defaultButton = QMessageBox::NoButton )

int           warning              ( const QString & title,
                                       const QString & text,
                                       quint32 buttons = QMessageBox::Ok,
                                       quint32 defaultButton = QMessageBox::NoButton )
```

D.3 Script examples

A subset of script examples is provided in the repositories supplied with the software.

Access:

1. Select "File > Load and Manager Repositories".
2. In the "Repository Manager", double click on the "Examples > K32 > Script Example".

3. Select "Script > Manage".

   The script examples are loaded in the "Script Manager".
   You can try out script examples and further develop them.

The following is the content of one of the scripts.

IPM Diagnostics

```
// This script reads 100 IPM values and writes
// them to a text file. the data can be used for an analysis
// of the IPM data.

// get the temporary path
dir = new QDir();
temppath = dir.tempPath();
fd = new QFileDialog();

// select file
filename = fd.getSaveFileName( 'Save File As', temppath, '', 0 );
if( filename.length > 0 ) {

  // create new IPM profile
  ps.scpi( 'IPM:CRE "ipm_test"' );
  ps.scpi( 'IPM:SEL "ipm_test"' );

  // switch to normal distribution
  ps.scpi( 'IPM:TYPE RAND' );
  ps.scpi( 'IPM:RAND:DIST NORM' );

  // configure distribution
  ps.scpi( 'IPM:RAND:NORM:MEAN 0' );
  ps.scpi( 'IPM:RAND:NORM:STD 1' );
  ps.scpi( 'IPM:RAND:NORM:LIM 100' );

  // start internal diagnostic system
  ps.scpi( 'IPM:DIAG:INIT' );

  file = new QFile();
  file.setFileName( filename );
  file.open( 0x0002 );

  ps.print( 'Writing file...' );

  for( var i=1; i<100; i++ ) {
    value = ps.scpi( 'IPM:DIAG:READ?' );
    file.writeString( value + '\n' );
  }

  file.close();
  ps.print( 'Done' );
```

```
  // remove IPM profile
  ps.scpi( 'IPM:REM "ipm_test"' );
}
```

D.4 List of Script Objects and Methods

E Common coding algorithms

In the notation used below a_n denotes the n^{th} input symbol and b_n denotes the correspondingly coded output symbol. Individual bits in the symbols from the LSB (least significant bit) to the MSB (most significant bit) are denoted by a_{0n}, a_{1n}, etc. The same applies to the output symbols.

Common coding types are listed in Table E-1.

Table E-1: Common coding algorithms

Coding	Coding algorithm	Applicable for K bit/symbol
"None"	$b_n = a_n$	k = 1...8
"Differential"	$b_n = (a_n + b_{-1})$ modulo 2^k	k = 1...7
"Differential + Gray"	Gray coding with differential coding	k = 1...7

The following example illustrates how one of common coding schemes in combination with a modulation method influence the signal.

Example: Differential coding for QPSK modulation with K = 2 bit/symbol

Decimal display; the value range for modulation symbols is $a_n = \{0, 1, 2, 3\}$

According to Table E-1 and for $k = 2$, the recursive coding is defined as $b_n = (a_n + b_{-1})$ *modulo 4*

Depending on the state of a preceding modulation symbol b_{-1}, the coded modulation symbol b_n is obtained for example from modulation symbol $\mathbf{a_n = 2}$ as follows:

b_{-1}	0	1	2	3
b_n	2	3	0	1

With differential coding, the assignment of modulation symbols a_n (binary indication: MSB, LSB) to the phase differences shown in the following table is generated:

Table E-2: Phase difference for QPSK

Modulation symbol a_n	00	01	10	11
Phase difference	0°	90°	180°	270°

Glossary: Terms and abbreviations

A

AM: Amplitude modulation

Antenna: The antenna characterizes the radiation pattern of the Emitter

ARB: Arbitrary Waveform Generator
An I/Q modulation source forming a part of the supported signal generators. The ARB allows the playback and output of any externally calculated modulation signal in the form of waveform file. It also allows the generation of multi-carrier or multi-segment signals from waveform files.

ASK: Amplitude-shift keying modulation

ASR: Airport Surveillance Radar

AWGN: Additive white gaussian noise

B

Barker: Special sequences (codes) that assures phase modulated signals with low autocorrelation properties.

Boresight: Antenna boresight is the direction to which an antenna shows the maximum gain.

BPSK: Binary phase shift keying modulation

BW: Bandwidth

C

C-BPSK: BPSK modulation with a constant envelope

Chirp: Signal in which the frequency varies over the time.

CW: Continuous wave signal, that is an unmodulated signal.

D

DQPSK: Differential QPSK

DUT: Device under test

E

EIRP: Equivalent isotopically radiated power

ELINT: Electronic intelligence, meaning information that comes from signals that typically do not contain speech or text.

Emitter: In the context of this software, emitters emulate radar systems.

EW: Electronic warfare

F

FFT: Fast Fourier transform

FM: Frequency modulation

FSK: Frequency-shift keying modulation

G

Gain: Antenna gain is a measure of the antenna's ability to concentrate electromagnetic energy in a narrow beam.

GUI: Graphical User Interface

H

HPBW: Half-Power Beam Width

I

Inter-Pulse Modulation: IPM

Intra-Pulse Modulation: MOP

IPM: Inter-Pulse Modulation

M

MOP: Modulation on Pulse

MSW: Multi-segment waveform

O

OQPSK: Offset QPSK

P

PAPR: Peak to average power ratio

PDW: Pulse Descriptor Word
Format of the reported data.

Plugin: A loadable Microsoft Windows DDL module that describes custom modulation schemes or envelope shapes.

Poly Phase: polyphase

polyphase: Code used by pulse modulation and suitable for pulse compression.

PRF: Pulse repetition frequency

PRI: Pulse repetition interval
Defines the overall time of a pulse cycle.

PRT: Pulse repetition time

Pulse sequence: Pulse train

Pulse train: A sequence of repetitive pulses

Pulse-to-Pulse Modulation: IPM

PW: Pulse width

Q

QPSK: Quadrature phase shift keying modulation

R

RADAR: Radio Detecting and Ranging

RCS: Radar cross section, RCS, or σ
The RCS is a measure of the energy that an object intercepts and scatters back towards the radar.

Repository: In the context of this software, a repository is a file-based database located on the local hard drive or on a network storage drive.

S

Scenario: In the context of this software, a scenario is the top-level description of the signals to be generated.

Sequence: In the context of this software, a sequence describes how pulses are arranged to form a waveform.

Stagger PRI: Pulse train composed of two or more pulse subtrains that use the same PRI.

V

VSG: Vector Signal Generator

Index

Symbols

A

B

C