Discover the Experience

EB Propsim F8
EB Propsim F32
EB Propsim FS8
User Reference

EB Propsim

Radio Channel Emulator

User Reference

Release 3

Document version 1.5

This product is covered by at least by one of the following patents and patent applications:

US 7054781, US 7184720, US 7394880, US 11/266,599, EP 1371160, EP 1393476, EP 1449316, EP 04733006.3, FI 111206, FI 113313, FI 113427, FI 116432, JP 3730961, JP 4046609, JP 4317874, KR 10-0851723, KR 10-0857678, ZL 02800418.3, CN 02805107.6

Published by:
Elektrobit System Test Ltd.
Tutkijantie 8, FI-90590 OULU
FINLAND

Telephone: Int. +358 40 344 2000
Fax: Int. +358 8 343 032

Email: wct.support@elektrobit.com
http://www.elektrobit.com

EB website at www.elektrobit.com

The EB website holds the latest news and information on a wide range of subjects related to the EB Propsim product family and test equipment for wireless communications.

In EB Extranet and support site, you can find information on your product and other EB Propsim radio channel emulators. The site also offers you download possibilities and the most convenient way to contact your EB Propsim support. EB Extranet and support site can be accessed at: **http://www.elektrobit.com/ebpropsimextranet.**

Notice

Elektrobit System Test Ltd. reserves the right to change its products and the information in this document without notice due to continuous improvement and development.

Elektrobit System Test Ltd. makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability or fitness for a particular purpose. Elektrobit System Test Ltd. shall not be liable for errors contained herein or for damages in connection with the furnishing, performance, or use of this material.

Warranty

Elektrobit System Test Ltd. warrants that this product will be free from defects in materials and workmanship for a period of one year from the date of shipment. During the warranty period, Elektrobit System Test Ltd. will, at its option, either repair or replace products that prove to be defective.

For obtaining service under this warranty, the Buyer must notify Elektrobit System Test Ltd. of the defect before the expiration of the warranty period and make suitable arrangements for the performance of service. The expenses associated to shipping the product to the service center and back to Buyer will be paid by Elektrobit System Test Ltd. Elektrobit System Test Ltd. warrants that its software and firmware designated for use with an instrument will execute its programming instructions when properly installed on that instrument. Elektrobit System Test Ltd. does not warrant that the operation of the instrument, or software, or firmware will be uninterrupted or error-free.

Limitation of Warranty

This warranty shall not apply to defects, failures or damages resulting from improper or inadequate maintenance and care, Buyer-supplied software or interfacing, unauthorized modification or misuse, operation outside of the environmental specifications for the product, or improper site preparation or maintenance.

NO OTHER WARRANTY IS EXPRESSED OR IMPLIED. ELEKTROBIT SYSTEM TEST LTD DISCLAIMS ANY IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. ELEKTROBIT SYSTEM TEST LTD'S RESPONSIBILITY TO REPAIR OR REPLACE DEFECTIVE PRODUCTS IS THE SOLE AND EXCLUSIVE REMEDY PROVIDED TO THE BUYER. ELEKTROBIT SYSTEM TEST LTD SHALL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER BASED ON CONTRACT, TORT, OR ANY OTHER LEGAL THEORY.

For complete terms and conditions for sale and service, please contact EB.

Abbreviations

NAME	DESCRIPTION
2D	2-dimensional
AAS	Adaptive Antenna Solution
AD	Analog-to-digital
.ANT	Antenna file format
.ASC	Text file format for channel impulse response data
.ASO	Aerospace channel model file
ATE	Automatic Test Equipment
AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
BS	Base Station
BW	Bandwidth
C/I	Carrier-to-Interference
CIR	Channel Impulse Response
.COR	Correlation file
CSS	Channel Sequence Simulation
CW	Carrier Wave = Continuous Wave = Sine Wave
DA	Digital-to-analog
DoA	Direction of Arrival
DoD	Direction of Departure
DUT	Device Under Test
EVM	Error Vector Magnitude
GPIB	General Purpose Interface Bus, IEEE 488.2
GSM	Global System for Mobile communications
GUI	Graphical User Interface
.ICS	CSS generated model file format
IR	Impulse Response
.IR	File format for impulse response data
LOS	Line-of-sight
LTE	Long Term Evolution
.MAT	Matlab file
MIMO	Multiple Input Multiple Output
MISO	Multiple Input Single Output
MPC	Multipath component
MS	Mobile Station
PDF	Probability density function
RF	Radio Frequency

NAME	DESCRIPTION
RFLO	RF Local Oscillator signal
RTC	Run-Time Control (emulation type)
RX	Receiver
SCM	Spatial Channel Model
SCME	SCM Extended
SD	Sample density: number of samples per half-wave. Half of the wavelength of the carrier wave divided by the sample distance. When simulating a moving vehicle, channel impulse responses are updated every λ/SD, where λ is the wavelength of the carrier wave.
.SHD	Shadowing file
.SIM	Emulator hardware control file
SIMO	Single Input Multiple Output
SISO	Single Input Single Output
SM	Spatial multiplexing
.SMU	Emulation file containing CIRs and other parameters in emulator compatible format
SNR	Signal-to-noise ratio
SOS	Sum-of-sinusoids
SP	Spatial multiplexing
STC	Space Time Coding
.TAP	File containing channel parameters
TDD	Time Division Duplex
TDL	Tapped Delay Line
TX	Transmitter
UDP	User Datagram Protocol
WCDMA / W-CDMA	Wideband Code Division Multiple Access
WiMAX	Worldwide Interoperability for Microwave Access

Table of Contents

1 Introduction

EB Propsim Quick Guide is a use case oriented guide of the EB Propsim radio channel emulator. It covers the initial setup of the product and provides instructions on use of the views and applications for most typical applications. Quick Guide is included in the EB Propsim delivery as printed copy.

User Reference documentation includes more detailed information about the use, maintenance and troubleshooting as well as technical specifications and product configuration options of the EB Propsim. The User Reference is divided into sections listed in Table 1.

When mentioned in this user reference, term “EB Propsim” applies to all EB Propsim devices (F8, F32 and FS8) unless otherwise stated.

SECTION	NAME	CONTENTS
1	Introduction	Introduction of the EB Propsim and user documentation (this section)
2-3	Product Specification	Product technical data including e.g. product configuration items, electrical specifications, mechanical dimensions, environmental and safety specifications
4	Graphical User Interface	Description of the toolbars, menus and settings of different Graphical User Interface views
5	Shadowing (optional feature)	Description of Shadowing controls and fields in GUI
6	Internal Interference Generator (optional feature)	Description of Internal interference generator controls and fields in GUI
7	Aerospace and Satellite Option (optional feature, supported in EB Propsim F8 Release 1.4)	Description of ASO Editor controls and fields in GUI
8	Remote Control	Description of ATE commands supported for test automation through GPIB or LAN
9	File Formats	Description of file formats supported by EB Propsim for exporting and importing channel impulse response data
10	Maintenance	Instructions related to product maintenance
11	Troubleshooting	Explanation of error messages and advice on troubleshooting actions
12	Wireless Propagation Environment	Overview of wireless propagation phenomena and radio channel models

Table 1. Sections of User Reference Documentation

Separate documentation is available about EB Propsim Standard Channel Models and for OTA MIMO application, GCM application and 802.11 Channel Model Tool.

All the documentation is also available in EB Propsim Extranet http://www.elektrobit.com/ebpropsimextranet. The user name and password for the extranet account are included in the delivery content of your EB Propsim.

1.1 EB Propsim

The wireless environment imposes many constraints and limitations on the performance of wireless telecommunication systems. Radio channel propagation characteristics, such as attenuation, shadowing, fast fading, variable delays, Doppler effect, noise and interference cause severe degradation to all wireless transmission. Thus the development of wireless communication systems requires rigorous testing to ensure that the products are able to operate even under the most demanding propagation environments.

The traditional field-testing of wireless systems is generally labor intensive, time-consuming and expensive. Furthermore, even under the same test setups and test scenarios the test results are typically non-repeatable, since the propagation environment uncertainties plus external noise and interference affect the results.

The EB Propsim is a radio channel emulator that enables recreating the wireless channel propagation effects in a controlled laboratory environment. It is a one box solution for performing a realistic and accurate emulation of all typical radio channel propagation effects such as multipath propagation, fast fading, dynamic delays, attenuation, noise, interference and shadowing. The EB Propsim supports multiple channels, wide bandwidth, high dynamic range and channel emulation with very high accuracy. The physical radio channel characteristics can all be emulated independently on EB Propsim, supplementing or even replacing traditional field-testing. EB Propsim product family is shown in Figure 1.

Figure 1. EB Propsim F32, EB Propsim FS8 and EB Propsim F8

EB Propsim products have been developed to fulfill the requirements of both present and future wireless communication systems. With extremely high RF performance they provide superior performance and feature set for radio interface testing of current communication systems, such as WCDMA, GSM, TD-SCDMA, EV-DO / CDMA2000, 3GPP LTE, WiMAX, WiFi and 802.11n/ac. Additionally, the features of the EB Propsim products enable testing of the next generation communication systems, such as LTE-Advanced, 802.16m and UMB.

1.1.1 Channel emulation concept

The EB Propsim is a generic channel emulator. It emulates only the radio channel excluding transmitter and receiver, and is thus independent of system technology or modulation. The EB Propsim supports all major wireless standards and signal types in a broad frequency range. It supports the development of most demanding wireless applications, such as beamforming, 8×8 MIMO, software defined radios and aerospace satellite communications.

In a typical test scenario the transmitter and receiver to be tested are connected to the EB Propsim, which then emulates a wireless propagation environment, replacing the real radio channel, as illustrated in Figure 2. The EB Propsim uses real-world signals generated by the external test equipment as an input.

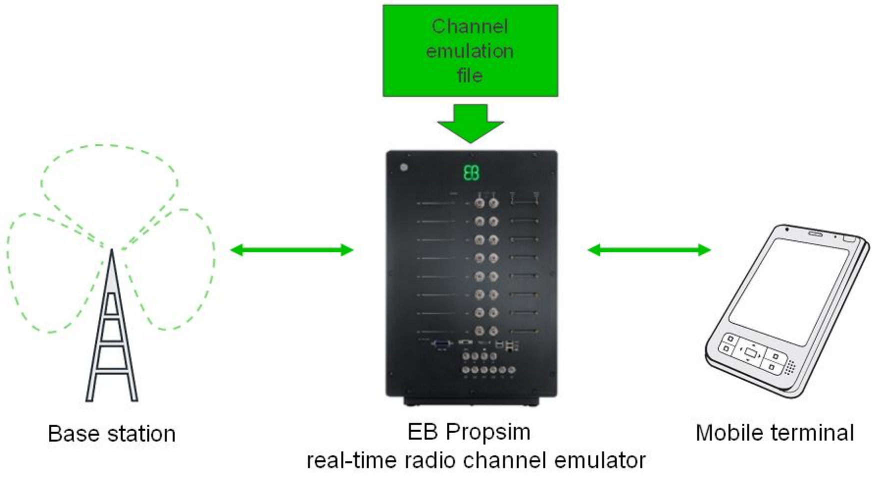


Figure 2. Replacing the radio channel with EB Propsim

The overall flexibility and extensive number of configuration options of the EB Propsim allow it to run a number of different types of tests and emulations. The emulations are based on preconfigured channel emulation files, stored in the EB Propsim. New emulation files can be added any time by the users. The file-based emulation approach ensures full repeatability and controllability of the emulations.

The EB Propsim includes pre-stored standard channel emulations (emulation connection diagram with channel models according to 3GPP and WiMAX standards). In basic use, the user only needs to load the emulation and adjust the power levels of the EB Propsim. For advanced use, all versatile channel modeling tools for creating user defined channel models and to modify pre-defined standard channel models are available.
The EB Propsim includes an easy to use GUI that guarantees quick emulation setup. The GUI can be used to control all aspects of the emulator functionality. It ensures that all emulation setup and configuration tasks require minimal amount of manual work.

The EB Propsim includes a “toolbox” of applications for creating channel models and emulations. Several channel models can also be combined for a single emulation which may use up to 128 fading channels, and thus up to 128 different channel models. The channel models are stored in the EB Propsim as pre-calculated files.

1.1.2 Physical connectors and LEDs

All external connectors and LEDs of the EB Propsim are found in the emulator front panel, as shown in Figure 3 and Figure 4.

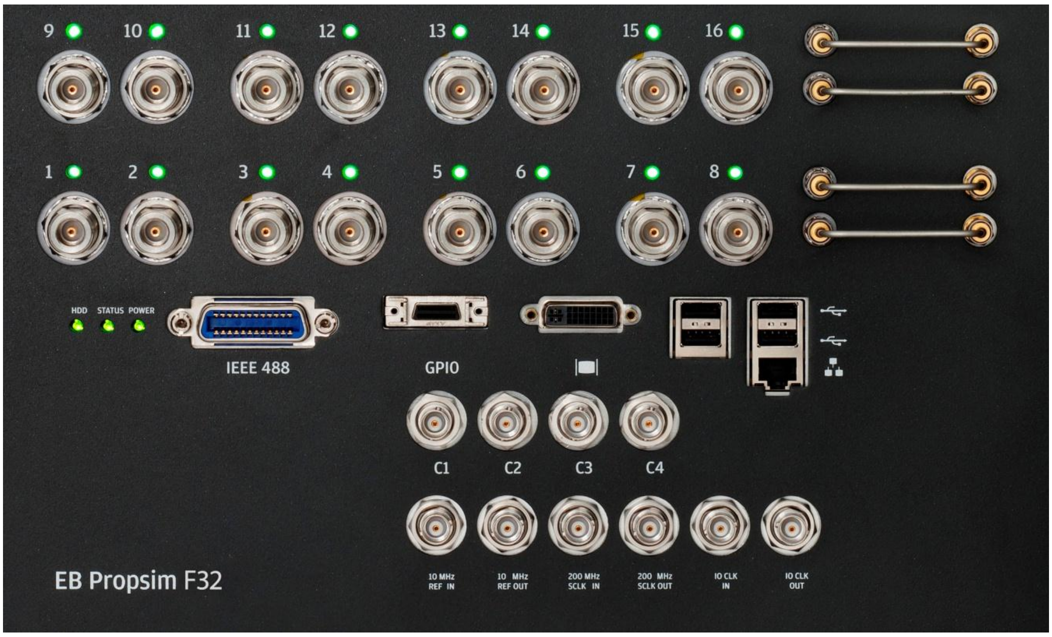


Figure 3. EB Propsim F32

The EB Propsim F32 is equipped with 8, 16, 24 or 32 RF channels. The EB Propsim FS8 is equipped with 4 or 8 RF channels. Each channel has input/output and output only RF connectors in the front panel. Additionally, each RF channel has two input and output connectors for an external local oscillator. Table 2 lists the physical connectors and LEDs found in the front panel of the EB Propsim F32 and FS8. Table 3 lists the physical connectors and LEDs found in the front panel of the EB Propsim F8.

NAME	DESCRIPTION
RF IN/OUT	RF input connector (for each physical emulator channel). Also RF output connector when using Propsim integrated signal duplexing
RF OUT	RF output connector (for each physical emulator channel)
RFLO IN	Local oscillator input (for physical emulator channels)
RFLO OUT	Local oscillator output (for physical emulator channels)
IEEE 488	GPIB connector for automated test control
GPIO	16-pin programmable general purpose input/output connector
DVI	DVI output for an external display (delivery package includes DVI - VGA adapter)
USB	USB connectors (4) for external keyboard, mouse and memory
Ethernet	Gigabit LAN, 10/100/1000 Base-T connector for LAN connectivity
C1	Sync in
C2	Sync out
C3 – C4	Programmable input/output connectors, reserved for future use
10 MHz REF IN	Input connector for 10 MHz reference signal
10 MHz REF OUT	Output connector for 10 MHz reference signal
200MHz SCLK IN	200 MHz sampling clock signal input

NAME	DESCRIPTION
200MHz SCLK OUT	200 MHz sampling clock signal output
IO CLK IN	0-200 MHz sample clock in (available in F32)
IO CLK OUT	0-200 MHz sample clock out (available in F32)
Power indicator	EB LOGO (F32), Stand-by switch (FS8)
HDD LED	Hard disk status: Blinking green HDD activity
STATUS LED	System status: • Green Status ok • Blinking Emulation running • Red Warning Warning is indicated in case of overheating, over voltage or self test failure. More detailed information about the reason of the warning is reported in the GUI.
POWER LED	Power status: • Blinking red Initializing • Green Status ok • Red Power failure
CH STATUS LED	Channel status (for each RF connector): • No indication Connector not in use • Green Status ok • Blinking Emulation running • Blue Connector selected in GUI • Red Warning Warning is indicated in case of overheating, over voltage, input cut-off, missing RF local or self test failure. More detailed information about the reason of the warning is reported in the GUI.

Table 2. EB Propsim F32 / FS8 connectors and LEDs

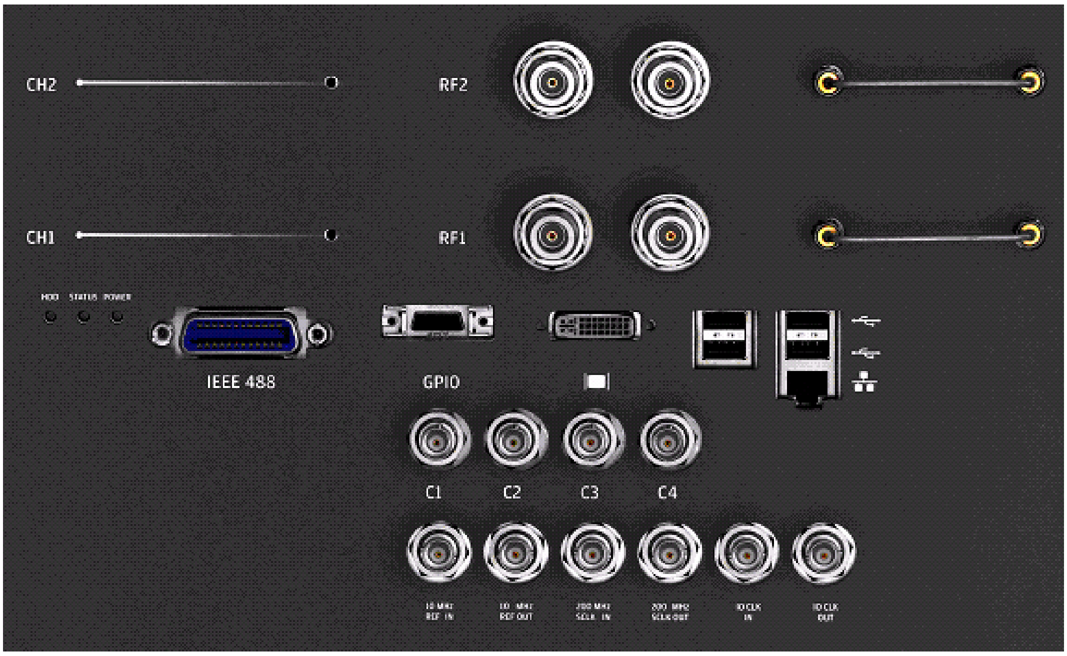


Figure 4. EB Propsim F8 front panel

The EB Propsim F8 is equipped with 2, 4, 6 or 8 RF channels. Each channel has input and output RF connectors in the front panel. Additionally, each RF channel has input and output connectors for an external local oscillator. Table 3 lists the physical connectors and LEDs found in the front panel of the EB Propsim F8.

NAME	DESCRIPTION
RF IN	RF input connector (for each physical emulator channel)
RF OUT	RF output connector (for each physical emulator channel)
RFLO IN	Local oscillator input (for each physical emulator channel)
RFLO OUT	Local oscillator output (for each physical emulator channel)
IEEE 488	GPIB connector for automated test control
GPIO	16-pin programmable general purpose input/output connector
DVI	DVI output for an external display (delivery package includes DVI - VGA adapter)
USB	USB connectors (4) for external keyboard, mouse and memory
Ethernet	Gigabit LAN, 10/100/1000 Base-T connector for LAN connectivity
C1	Sync in
C2	Sync out
C3 – C4	Programmable input/output connectors, reserved for future use
10 MHz REF IN	Input connector for 10 MHz reference signal
10 MHz REF OUT	Output connector for 10 MHz reference signal
200MHz SCLK IN	200 MHz sampling clock signal input
200MHz SCLK OUT	200 MHz sampling clock signal output
IO CLK IN	0-200 MHz sample clock in
IO CLK OUT	0-200 MHz sample clock out
EB LOGO	Power indicator
HDD LED	Hard disk status: Blinking green HDD activity
STATUS LED	System status: • Green Status ok • Blinking Emulation running • Red Warning Warning is indicated in case of overheating, over voltage or self test failure. More detailed information about the reason of the warning is reported in the GUI.
POWER LED	Power status: • Blinking red Initializing • Green Status ok • Red Power failure
CH STATUS LED	Channel status (for each channel): • No indication Connectors not in use • Green Status ok • Blinking Emulation running • Red Warning Warning is indicated in case of overheating, over voltage, input cut-off, missing RF local or self test failure. More detailed information about the reason of the warning is reported in the GUI.

Table 3. EB Propsim F8 connectors and LEDs

1.1.3 EB Propsim views and applications

The EB Propsim GUI has a navigation bar located on the left side. The navigation bar provides easy access to all views and applications in the EB Propsim. The GUI is shown in Figure 5 and Figure 6.

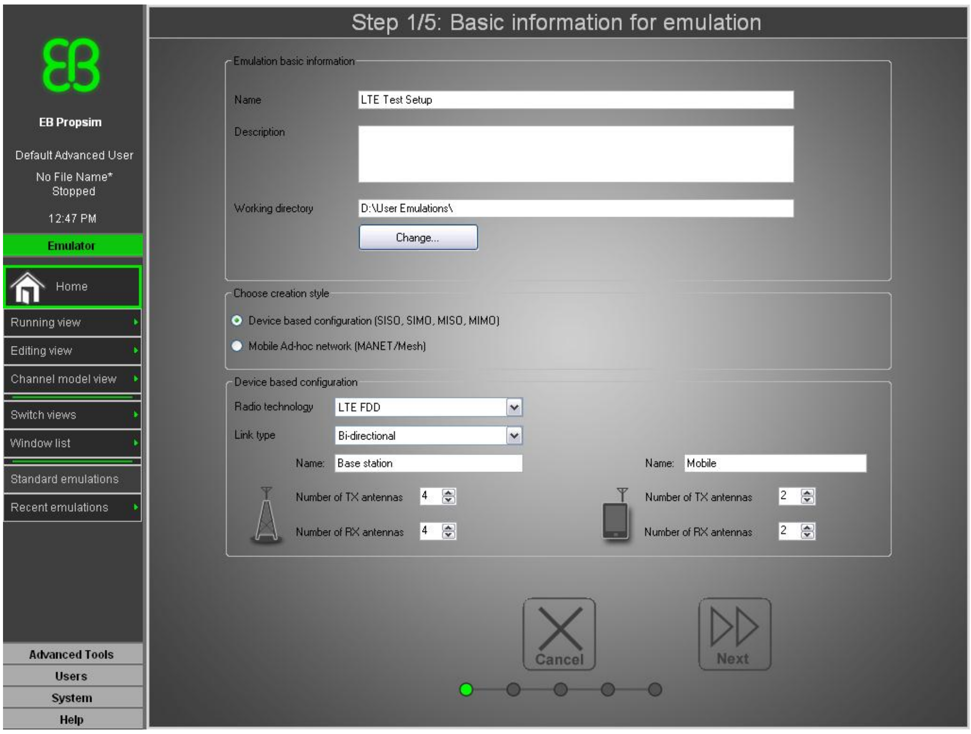


Figure 5. EB Propsim graphical user interface, Scenario Wizard

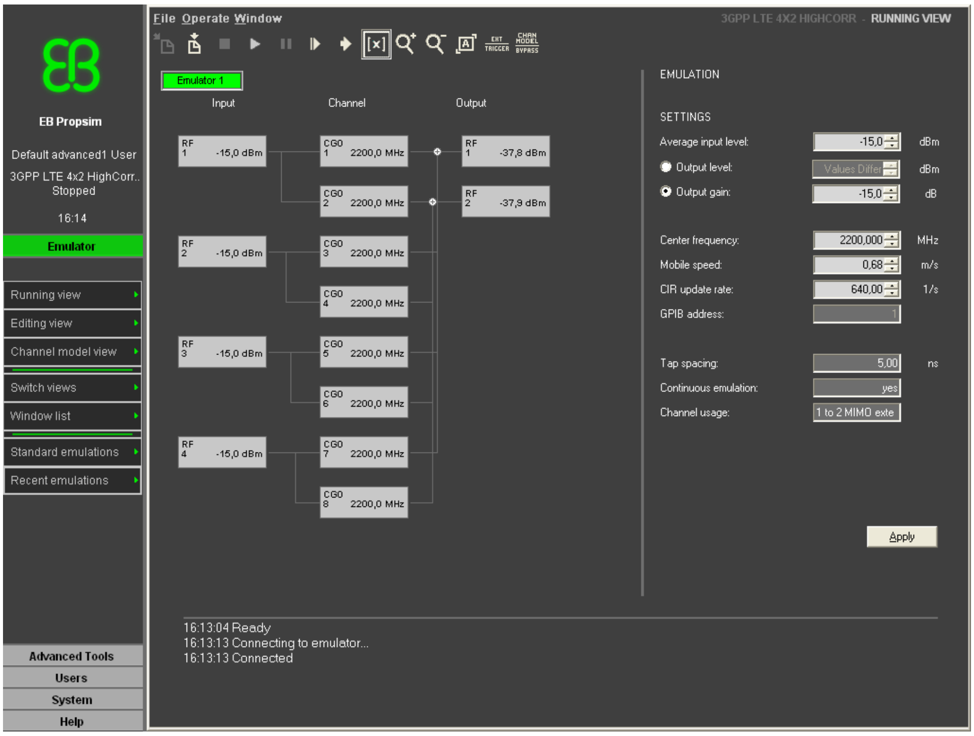


Figure 6. EB Propsim graphical user interface, Running View

The EB Propsim includes a “toolbox” of applications for creating channel models and emulations and running the emulations.

- The Home View - Scenario Wizard is used to easily create emulations for different test scenarios, technologies and lab environments.
- The Running view is used to run emulations pre-stored in the EB Propsim or created with the channel modeling tools. The Running view also allows you to control many of the hardware emulation operations of the EB Propsim.
- Editing view is used for creating and modifying emulation block diagrams and to build (generate) channel emulations.
- Channel model view is used to create statistical channel models.
- Correlation Editor is used to define statistical correlation between channels of correlative or MIMO models.
- Shadowing editor (optional feature) is used for defining shadowing profiles for channels. It can also be used to generate path loss triggered handovers / handoffs. The Shadowing editor can be launched from Editing view and Scenario Wizard.
- Interference generator (optional feature) is used to create interference sources for the emulations. Interferer can be added to an emulation in Editing view or Running view.
- Geometric Channel Modeling (GCM) application (optional feature) is a channel modeling tool for creating dynamic spatial channel models.
- MIMO OTA application (optional feature) includes the necessary channel modeling tools to create geometry-based channel models for testing MIMO terminals. It also provides automatic and accurate system calibration tools.
- 802.11 channel modeling tool (optional feature) is used to generate radio channel realizations according to IEEE 802.11n/ac TGn/ac Channel Models specification. Channel models of IEEE 802.11n/ac are designed for WLAN MIMO application.
- ASO editor (optional feature) is channel modeling tool for using the wireless communication channels in Aerospace and Satellite applications (supported in F8 release 1.4).
- WES application (optional feature) is a channel modeling tool for performing repeatable and accurate drive tests in a complex, multi-base station real-life like network either based on measured or simulated data.

The emulator tools are described in details in other sections of the User Reference documentation.

2 Product Specification F8

This document is product specification of EB Propsim F8. It states:

- most wireless applications and technologies supported by the EB Propsim F8 emulator
- standard product configuration items
- product high level block diagram and interfaces
- electrical specifications
- mechanical dimensions
- environmental and safety standards and specification
- supported emulation channel topologies
- list of the emulator tools
- details of the operating system
- requirements for the external devices and equipment

Figure 7. EB Propsim F8

Note! Elektrobit System Test Ltd. reserves the right to change the specifications without notice due to continuous improvement of the product.

2.1 Applications

EB Propsim F8 is a versatile radio channel emulator platform capable for several different test cases. In general the emulator is independent of input signal and channel model.

Typically the radio channel emulator is required in performing the product conformance test cases defined in related standards. In addition to standard test cases multiple other wireless applications in defense, research and satellite areas can be tested with the EB Propsim F8 channel emulator. The emulator is suitable for performance testing of the following example applications and technologies:

- 4G research
- CDMA, CDMA2000
- DVB-H, DVB-T
- GSM, EDGE, GPRS
- High acceleration and velocity platform testing (space and terrestrial)
- Jammers, communication and radar
- 3GPP LTE (TDD and FDD), Long Term Evolution
- MANET/Mesh, Mobile Add-hoc Networks
- Multi-user, client network level testing
- Radar
- Satellites
- SDR, software defined radios
- TETRA
- WCDMA
- WiBRO
- WiMAX, fixed and mobile
- WLAN, WiFi
- MIMO, multiple in multiple out transceivers
- Beamforming
- OTA, over the air testing
- Ray tracing
- and many others

Comprehensive package of standard channel models is included in every EB Propsim F8. See Standard Channel Models section in Appendix A for further information.

2.2 Product configuration items

Table 4 describes the standard EB Propsim F8 (standalone) product configuration items. Your product configuration can be found in EB Propsim extranet http://www.elektrobit.com/extranet. The user name and password are included in the product delivery contents.

CODE	DESCRIPTION
Product item codes included to baseline and EB recommended configurations	
F8-01.10	EB Propsim F8 cabinet platform
F8-12.10	RF interface channel. Frequency range support 350MHz …. 3000MHz. Signal bandwidth 40MHz
F8-12.30	RF interface channel. Frequency range support 350MHz …. 6000MHz. Signal bandwidth 40MHz
F8LO-1.N	1 Internal Local oscillator frequency for N (2, 4, 6 or 8) RF channels
F8LO-2.N	2 Internal Local oscillator frequencies for N (2, 4, 6 or 8) RF channels
Optional items for emulation channels	
F8-15.10	Frequency range extension 350…3000MHz --> 220MHz…3000MHz.
F8-15.11	Frequency range extension 350…3000MHz --> 350MHz…6000MHz.
F8-15.12	Frequency range extension 350…3000MHz --> 220MHz…6000MHz.
F8-15.13	Frequency range extension 350…6000MHz --> 220MHz…6000MHz.
F8-31.20	Upgrade of n RF channel F8-C40 baseline emulator to F8-C80 baseline emulator (n = 2,4,6,8)
F8-31.24	Upgrade of n RF channel F8-C40 baseline emulator to F8-C125 baseline emulator (n = 2,4,6,8)
F8-31.28	Upgrade of n RF channel F8-C80 baseline emulator to F8-C125 baseline emulator (n = 2,4,6,8)
F8-46.120	Firmware with maximum of 1.3s excess channel delay with four (4) independent fading paths per channel (supported in release 1.4)
Internal RF local oscillator sub-system extension	
F8-16.xNyN	F8LO-x.N ---> F8LO-y.N. Increases the number of RF LOs from x (1,2,3) to y (2,3,4) for up to N (2, 4, 6 or 8) RF channels in cabinet.
Frequency band licenses	
F8FB-A	Frequency band A. Enables frequency range 350…2200MHz.
F8FB-B	Frequency band B. Enables frequency range 2200…3000MHz.

CODE	DESCRIPTION
F8FB-C	Frequency band C. Enables frequency range 3000…4000MHz.
F8FB-D	Frequency band D. Enables frequency range 4000…6000MHz.
F8FB-E	Frequency band E. Enables frequency range 220…350MHz.
Firmware options	
F8-05.01	Shadow fading emulation
F8-06.01	MIMO extension 1 to 2
F8-06.02	MIMO extension 1 to 4
F8-06.03	MIMO extension 1 to 8
F8-10	Internal Interference Platform
F8-10.01	AWGN source. One (1) source per RF channel.
F8-10.10	CW source. One (1) source per RF channel.
Advanced emulator applications and tools	
F8-41	EB Propsim F8 Standard Channel Modeling toolbox for external PC
F8-42.01	MIMO channel modeling tool for WiFi (TGn 802.11n)
F8-43.02	Geometric Channel Modeling (GCM) Application
F8-44.100	EB MIMO OTA Application
F8-45.100	EB WES Application (CSS)
F8-46.100	Aerospace and Satellite channel modeling tool (supported in release 1.4)

Table 4. Emulator configuration items and codes

2.3 Definitions and requirements

2.3.1 Definitions

The performance values are verified with standalone product configuration described in Table 5 if not otherwise noted.

CODE	DESCRIPTION
F8-01.10	EB Propsim F8 cabinet platform
F8-12.31	RF interface channel, 8 channels. Frequency range support 350MHz …. 6000MHz. Signal bandwidth 80MHz
F8-15.12	Frequency range extension 350…3000MHz --> 220MHz…6000MHz.
F8-15.22	Signal bandwidth extension from 80MHz to 125MHz.
F8LO-2.8	2 Internal Local oscillator frequencies for 8 RF channels
F8-16.2848	F8LO-2.8 ---> F8LO-4.8. Increases the number of RF LOs from two (2) to four (4) for up to eight (8) RF channels in cabinet.
F8FB-A	Frequency band A. Enables frequency range 350…2200MHz.
F8FB-B	Frequency band B. Enables frequency range 2200…3000MHz.
F8FB-C	Frequency band C. Enables frequency range 3000…4000MHz.
F8FB-D	Frequency band D. Enables frequency range 4000…6000MHz.
F8FB-E	Frequency band E. Enables frequency range 220…350MHz.
F8-06.01	MIMO extension 1 to 2
F8-06.02	MIMO extension 1 to 4
F8-06.03	MIMO extension 1 to 8
F8-10	Internal Interference Platform
F8-10.01	AWGN source. One (1) source per RF channel.
F8-10.10	CW source. One (1) source per RF channel.
F8-05.01	Shadow fading emulation

Table 5. Emulator configuration for performance verification

2.3.2 Conditions required to meet specifications

The following conditions are required to meet the specifications:

- The emulator is within calibration cycle.
- At least 4 hours of storage at a constant temperature within the operating temperature range.
- The emulator has been switched on for a one hour warm-up period.
- No other view, application or 3rd party software (except F-secure virus protection, see chapter 2.6.1) is running simultaneously with EB Propsim F8 Running view.

2.3.3 Test setup

The system performance is verified with the settings and under the conditions listed in Table 6 if not otherwise noted.

	RF INTERFACE
Input signal	Sine wave (CW)
Input level	-15 dBm
Input level setting	-15 dBm average, crest factor 6 dB
Bandwidth	125 MHz
Fading profile	1-path Constant, 0 ns delay, no fading
Total channel gain	-1 dB (→ output power -16 dBm)
MIMO extension	not used
Ambient temperature	+24 ± 2 °C
Center frequency	2 GHz

Table 6. Emulator setup and conditions

The test setups for verifying performance values are presented in Figure 8 and Figure 9.

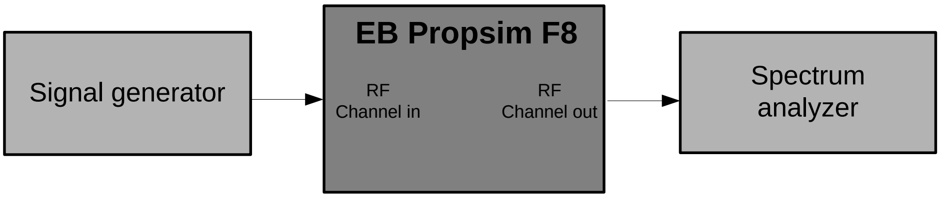


Figure 8. Test set up with signal generator and spectrum analyzer

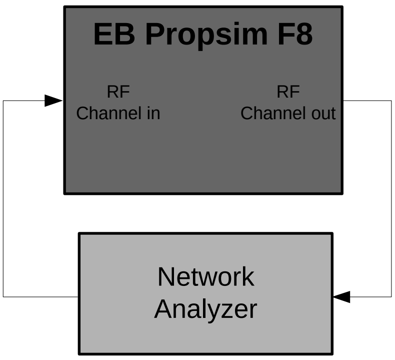


Figure 9. Test set up with network analyzer

2.4 Electrical specifications

2.4.1 Power supply

Voltage	100…240 VAC
Frequency	50/60 Hz
Power	1500W

2.4.2 Absolute maximum ratings

PORT / CONNECTOR	ABSOLUTE MAXIMUM RATING
RF 1 …8 INPUT	+10 dBm peak, AVG 0dBm
RF 1 …8 OUTPUT	+10 dBm peak reverse power, AVG 0 dBm
RF LO INPUT	+10 dBm CW forward power, AVG 0 dBm
RF LO OUTPUT	+10 dBm CW reverse power, AVG 0 dBm
C1 SYNC IN	0 / +3.8 VDC
C2 SYNC OUT	0 / +3.8 VDC
C3 – C4	reserved for future use
10 MHz REF IN	+20 dBm 50 ohm
10 MHz REF OUT	+20 dBm 50 ohm (reverse)
200MHz SCLK IN	+15 dBm 50 ohm
200MHz SCLK OUT	+15 dBm 50 ohm (reverse)
IO CLK IN	0 / +3.8 VDC
IO CLK OUT	0 / +3.8 VDC (reverse)
GPIO	0 / +3.8 VDC
IEEE 488	-0.3 VDC / 4.3 VDC
AC power plug	252 VAC 50/60 Hz

Table 7. Absolute maximum ratings

2.4.3 Emulator block diagram

The block diagram of the EB Propsim F8 is presented in Figure 10.

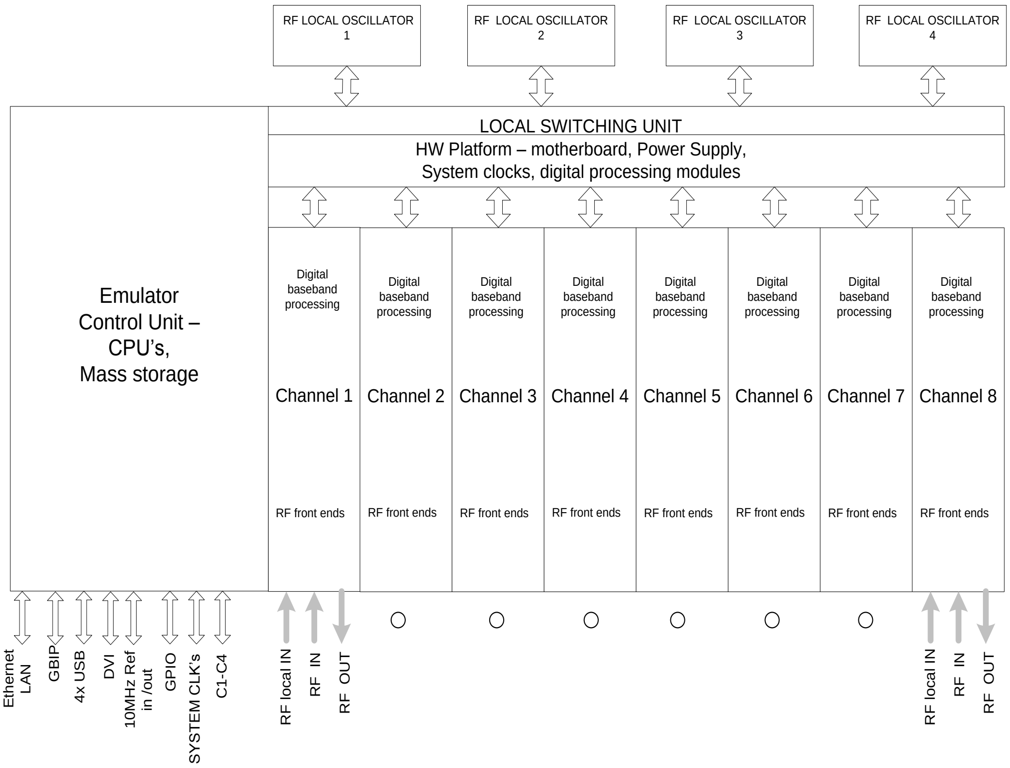


Figure 10. Emulator units, modules and interfaces

2.4.4 Physical connectors and LEDs

The EB Propsim F8 can be equipped with 2, 4, 6 or 8 independent physical channels. Each channel has input and output RF connectors in the front panel. Additionally, each physical channel has input connector for an external local oscillator.

With the 1 to 2 MIMO extension feature, a single EB Propsim F8 emulator can support emulations with a maximum of 16 channels and therefore e.g. up to four 2x2, one 2x4, one 4x2, one 2x8, one 8x2 and one 4x4 MIMO topologies.

With the 1 to 4 MIMO extension feature, a single EB Propsim F8 emulator can support emulations with a maximum of 32 channels. In this mode single 4x4, 4x8 and 8x4 MIMO topologies as well as two 4x4 MIMO topologies are supported.

With the 1 to 8 MIMO extension feature, a single EB Propsim F8 emulator can support emulations with a maximum of 64 channels. In this mode single 8x8 MIMO topology as well as 8 radios full Mesh network topology is supported.

See Chapter 2.7 for more details on supported topologies.

All external connectors and LEDs of the EB Propsim F8 are found in the emulator front panel, as shown in Figure 11, Figure 12 and Figure 13.

Figure 11. EB Propsim F8 front panel

Figure 12. EB Propsim F8 channel connectors

Figure 13. EB Propsim F8 control connectors

The LEDs and connectors are listed in Table 8.

NAME	DESCRIPTION	SPECIFICATION
RF IN	RF input connector (for each physical emulator channel)	N female, 50 ohm nominal Input VSWR < 1.5
RF OUT	RF output connector (for each physical emulator channel)	N female, 50 ohm nominal Output VSWR < 1.5
RFLO IN	Local oscillator input (for each physical emulator channel)	SMA female, 50 ohm nominal Input VSWR < 1.8
RFLO OUT	Local oscillator output (for each physical emulator channel)	SMA female, 50 ohm nominal Input VSWR < 1.8
GPIO	16-pin programmable general purpose input/output connector	LVDS. Reserved for future use.
IEEE 488	GPIB connector for automated test control	24 pin centronix, female
DVI	DVI output for an external display	DVI-I, female
USB	USB connectors (4) for external keyboard, mouse and memory	USB2.0 Standard
Ethernet	Gigabit LAN, 10/100/1000 Base-T connector for LAN connectivity.	RJ45
C1	Sync in	BNC, LVTTL
C2	Sync out	BNC, LVTTL
C3 – C4	Programmable input/output connectors, reserved for future use	BNC, LVTTL
10 MHz REF IN	Input connector for 10 MHz reference signal	Connector Type: BNC Impedance: 50 ohm. Signal level > 0 dBm, + 20dBm max. Max frequency deviation ±5 Hz
10 MHz REF OUT	Output connector for 10 MHz reference signal	Connector Type: BNC Impedance: 50 ohm. Signal level nominal 2dBm.
200MHz SCLK IN	200MHz sampling clock signal input	Connector Type: BNC Impedance: 50 ohm. Signal level > 3dBm, +15dBm max.
200MHz SCLK OUT	200MHz sampling clock signal output	Connector Type: BNC Impedance: 50 ohm. Signal level nominal level 7dBm
IO CLK IN	0-200MHz sample clock in	BNC Signal PECL
IO CLK OUT	0-200MHz sample clock out	BNC Signal LVTTL
EB LOGO	Power indicator	-
HDD LED	Hard disk staus: • Blinking green HDD activity	-

NAME	DESCRIPTION	SPECIFICATION
STATUS LED	System status: • Green Status ok • Blinking green Emulation running • Red Warning Warning is indicated in case of overheating, overvoltage or selftest failure. More detailed infromation about the reason of the warning is reported in the GUI.	-
POWER LED	Power status: • Blinking red Initializing • Green Status ok • Red Power failure In case of red power indication, shut down and restart the system.	-
CH STATUS LED	Channel status (for each channel): • No indication Connectors not in use • Green Status ok • Blinking green Emulation running • Red Warning Warning is indicated in case of overheating, overvoltage, input cut-off, missing RF local missing or selftest failure. More detailed infromation about the reason of the warning is reported in the GUI.	-

Table 8. EB Propsim F8 front panel connectors and LEDs

2.4.5 Rear panel

The rear panel of the EB Propsim F8 contains the AC power input with the mains power switch, see Figure 14 and Table 9.

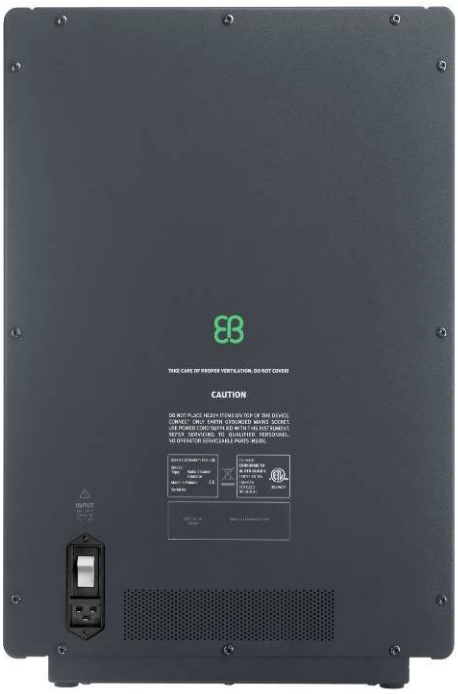


Figure 14. EB Propsim F8 rear panel

NAME	DESCRIPTION	SPECIFICATION
AC INPUT	AC input with mains power switch	IEC inlet filter

Table 9. EB Propsim F8 rear panel

2.4.6 User control interfaces

EB Propsim F8 can be controlled either through Graphical User Interface or with remote control commands.

- EB Propsim F8 Graphical User Interface (GUI) and other software tools are operating under Windows XP for embedded environment.
- Remote Control (i.e. Automatic Test Equipment Control, ATE Control) can be used to execute a large part of the actions available to the user in Running view application. See Remote Control section of user reference for details on commands.
- The Remote Control can be used via GPIB or LAN:
  - GPIB interface supports standard GPIB (IEEE 488.2) SCPI commands.
  - LAN interface supports standard SCPI commands.

2.4.7 RF characteristics

The key RF characteristics are listed in Table 10.

PARAMETER	VALUE
Platform frequency range	350 MHz to 3000 MHz **(F8-12.11)** 350 MHz to 6000 MHz **(F8-12.31)** 350 to 3000 MHZ -> 220 to 3000 MHz **(F8-15.10)** [1] 350 to 6000 MHZ -> 220 to 6000 MHz **(F8-15.13)** [1] Note: The channel f_c can be set independently.
Instantaneous channel bandwidth	40 MHz 80 MHz 125MHz
Fading tap mean amplitude range	0 … -60 dB (resolution 0.1 dB)
Input signal level range for full reference [2]	-30 ... +6 dBm
Input signal level range for > 35 dB SNR	-55 … +6 dBm
Input crest factor setting	0…24 dB (resolution 0.1 dB)
Output reference level, for full reference [2]	-116 ... -16 dBm (with 6 dB crest factor setting)
Input and output level resolution	0.1 dB
Output level setting	-116….-16 dBm
Output gain setting	0…-100 dB
Peak output level	0 dBm
Number of input bits (AD converter)	14
Number of output bits (DA converter)	16
Analog gain path attenuation range	100 dB
Path ON/OFF ratio	Typical > 100dB
Absolute max. input/output level without damage	+10 dBm

PARAMETER	VALUE
Output noise floor	< -160 dBm/Hz (output reference level ≤ -36 dBm) Typical value < -170 dBm/Hz (output reference ≤ -40 dBm)

[1] Operation outside specified frequency range is not specified.
[2] "Full reference" refers to such emulator setting that gives the maximum linearity performance for the user signal in RF domain.

Table 10. RF characteristic

2.4.7.1 Linearity and spurious signal performance

PARAMETER	VALUE
Error Vector Magnitude (EVM).	Typical WCDMA signal (3.84 Mchips/s) at 2000 MHz < -48 dB RMS Typical OFDMA signal 20 MHz, 64QAM at 2000 MHz < -45 dB RMS Typical OFDM signal 125 MHz, 16QAM at 2000 MHz < -40 dB RMS
Adjacent Channel Power (ACP) for WCDMA modulation (3.84 Mchips/s)	Typical -62 dBc @ 5 MHz offset at 2000MHz
Baseband harmonics, in band spurious	SFDR Typical value 60 dB
Leaking RFLO @ [(7100MHz + f_c) / 2]	max. < -40 dBm Typical 45 dBc max. < -60 dBm
Other out-of-band spurious signals	f_c +200MHz: max. < -40 dBm Typical 45 dBc max. < -60 dBm
Output level accuracy	±0.75 dB Typical value ±0.3 dB
Input and output phase linearity @ f_c	< 10 °
Output frequency response flatness	over 40 MHz BW < 1.5 dB over 80 MHz BW < 1.5 dB over 125 MHz BW < 3.0 dB Typical over 40 MHz BW < 1 dB over 80 MHz BW < 1 dB over 125 MHz BW < 2 dB
Group delay variation (peak-to-peak)	Typical < 2 ns over 30 MHz BW Typical 6 ns over 100 MHz BW Typical 10 ns over 125 MHz BW
Average gain variation when changing from one fading model to another (at band center)	< 0.5 dB
Gain variation due to variation in ambient temperature	< 0.2 dB / °C
Duration of signal path interrupt when changing emulation	Typical 0 µs for SISO configurations
Long term output power variation at band center in one year.	1 dB max

PARAMETER	VALUE
Classical LCR deviation from the theoretical LCR at levels -30 dB ... +3 dB from the mean power level (*SD* = 2)	± 10%
Classical CPDF deviation from the theoretical CPDF at levels -20 dB ... +10 dB from the mean power level (*SD* = 2)	± 1 dB
Classical CPDF deviation from the theoretical CPDF at levels -30 dB ... –20 dB from the mean power level (*SD* = 2)	± 5 dB

Table 11. Linearity and spurious signal performance

The multi channel performance is verified in terms of phase and amplitude variations between the emulator channels. One channel is selected as a reference against which the other channels are compared. All calibrations are done via GUI. The error i.e. channel-to-channel phase difference is measured with a network analyzer. The DoA performance is defined for uniform linear array where antenna element spacing is $\lambda/2$.

Normal angle is 90° while 0° means DoA directly from right.

	RF INTERFACE	
DoA (Direction Of Arrival) steps	< 0.1°	
DoA accuracy, calibration every 2…4 hours and typical lab environment	± 1° @ 2 GHz,	DoA 20°…160°
Average phase error between channels, daily calibration and typical lab environment	± 5° @ 2 GHz	
Average phase error between channels, calibration every 2…4 hours and typical lab environment	± 1° @ 2 GHz	
Amplitude error between channels, daily calibration and typical lab environment	± 0.5 dB @ 2 GHz	
Amplitude error between channels, calibration every 2…4 hours and typical lab environment	± 0.2 dB @ 2 GHz	

Table 12. Multi channel performance

2.4.8 Fading channel emulation

The digital calculation resources of EB Propsim F8 can be allocated in a flexible way. This chapter presents the maximum (or an example when mentioned) figures of the achievable performance of the emulator.

	RF INTERFACE
Maximum number of independent paths per fading channel.	Up to 48
Maximum number of independent paths per logical channel when all channels in use Paths within dynamic <15 µs delay spread block.	Up to 12
Tap Delay resolution[5]	Minimum 5 ns
Maximum delay spread	3000 us
Sample density SD (samples per half wavelength)	Typical 1024 when Doppler < 700 Hz

	RF INTERFACE
Max doppler 1 path/emulator	74 kHz (Effective SD 10)
Max doppler 2x2 MIMO 48 paths/channel	3.2 kHz (Effective SD > 64)
Max doppler 4x4 MIMO 12 paths/channel	2.0 kHz (Effective SD > 64)
Max doppler 4x4 MIMO 24 paths/channel	1.4 kHz (Effective SD > 64)
Number of impulse responses (CIRs) saved to memory	200 000 000 when 1 path / emulator 1 300 000 when 16 channels (4x4 MIMO) with 12 paths/channel
Maximum emulation repetition interval	60 minutes, 4x4 MIMO 6 paths/channels, vehicle speed 100 km/h, frequency 2 GHz
Minimum path insertion delay, input connector to output connector (SISO), 0 ns delay setting	< 4 μs Depends on channel topology
Fading profiles, standard	Delay Fixed Sinusoidal Linear 3GPP Birth-Death (random hopping) Average magnitude Phase Shift Fading distribution & Doppler spread Constant Constant – Pure Doppler (Pure Doppler) Rayleigh – Jakes (Classical) Rayleigh – Gaussian (Gaussian) Rayleigh – Flat (Flat) Rayleigh – Butterworth Rayleigh – Rounded Rice – Jakes (Rice) Rice – Gaussian Rice – Flat Rice – Butterworth Rice – Rounded Nakagami Lognormal Suzuki Geometrical model parameters DoA and DoD mean angle DoA and DoD spread DoA and DoD distribution Mobile heading Scatters Line-of-sight component parameters Correlation matrix

[5] User can apply custom delay interpolation methods utilizing more tap resources through IR-file in order to improve delay resolution

Table 13. Fading channel emulation performance

2.4.9 Multi-emulator synchronization

Synchronization of multiple emulators is supported from Release 1.1.1.

Test setup

The system performance is verified using settings listed in Table 14.

PARAMETER	VALUE
Number of emulators	5
Synchronization cables	Cable type: 5 ns / m (velocity factor 0.66) Connector type: BNC Cable length: 2.0 m
10 MHz clock source	Epsilon Clock® Series 2S, GPS-locked
10 MHz amplifier / switch	Epsilon SAS Switch and amplifier unit
Connection topology	1to8

Table 14. Test setup for multi-emulator performance testing

Performance, features & setup requirements

The performance, features and setup requirements of multi-emulator synchronization are listed in Table 15.

PARAMETER	VALUE
EB Propsim F8 HW	All the emulators of the setup shall be of equal firmware and HW version, and calibrated. Contact EB for further information.
EB Propsim F8 firmware	Release 1.1.1 or above
Synchronization accuracy	+/-3 ns typical
Number of emulators	2 – 6
Supported connection topologies	1to8, 2to8, 4to8 [6] The same connection topology shall be used in all emulators of multi-emulator setup
Synchronization cables (between cascaded emulators)	Cable type: 5 ns / m (velocity factor 0.66) Connector type: BNC All the synchronization cables used in the setup shall be of equal length. Supported lengths: 2.0 m ± 0.02 m 4.0 m ± 0.02 m 6.0 m ± 0.02 m
10 MHz clock cables	Connector type: BNC All the reference cables used in the setup shall be of equal type and length.
External 10 MHz ref clock	Connector type: BNC Impedance: 50 ohm Signal level: > 0 dBm, +20 dBm max, >10 dBm for optimal performance Point to point connection to each emulator Duty cycle: 50 ± 2 % Max frequency deviation: ±5 Hz Timing accuracy between emulators: < 200 ps Phase accuracy between emulators: <2 degrees

[6] Contact EB for information about support for other topologies

Table 15. Performance and features of simultaneous start

For more information, refer to EB Propsim F8 Multi-emulator Synchronization application note.

2.4.10 Optional feature: Shadowing

2.4.10.1 Test setup

The system performance is verified using the settings listed in Table 16.

	RF INTERFACE
Input signal	Sine wave (CW)
Input level	-15 dBm
Input level setting	-15 dBm average, crest factor 6 dB
Output gain	0 dB
Fading profile	1-path constant, 0 ns delay, no fading
Simulation configuration	One input, channel and output
Shadowing profile	Sinusoidal
Shadowing settings	Mean attenuation: 50 dB Period: 10 s Amplitude: 50 dB Start phase: 90

Table 16. Shadowing settings for testing

2.4.10.2 Test output

Measured output should show changing signal level:

- Maximum signal level: approximately -16 dBm
- Minimum signal level: approximately -116 dBm
- At the simulation start the signal level is approximately -116 dBm
- Signal level changes along sinusoidal pattern in which the period is 10 seconds

2.4.10.3 Shadowing performance

ITEM	CHARACTERISTICS
Maximum shadowing profile update rate	100 Hz
Output level accuracy for shadowing	+/- 0.75 dB
Output resolution	0.1 dB
Minimum shadowing attenuation	0 dB
Maximum shadowing attenuation	100 dB
Maximum period length of shadowing profile (computational)	> 10 years

Table 17. Shadowing performance

2.4.11 Optional feature: Internal Interference Generator

One interference source (AWGN or CW) is supported per channel at the time.

2.4.11.1 AWGN

Test setup

The system performance is verified using the parameters listed in Table 18.

	RF INTERFACE
Input signal	Sine wave (CW)
Center frequency	2 GHz
Input level	-15 dBm
Input level setting	-15 dBm average, crest factor 6 dB
Fading profile	1-path Constant, 0 ns delay, no fading
Output gain	0 dB
Noise level setting	-110 dBm/Hz
C/N settings	0 dB / 3.84 MHz

Table 18. Test setup for AWGN testing

AWGN interference specifications

	RF INTERFACE
Frequency offset range	Fixed 30 MHz AWGN ± 5 MHz, BW 40 MHz ± 25 MHz, BW 80 MHz ± 47.5 MHz, BW 125 MHz
Noise flatness at fixed f_0 [7]	Fixed 30MHz AWGN ± 0.25 dB, BW 30 MHz Full 40MHz BW ± 0.75 dB, BW 40 MHz Full 80MHz BW ± 0.9 dB, BW 80 MHz Full 125MHz BW ± 1.5 dB, BW 125 MHz
Noise crest factor	13 dB maximum / 10.5 dB with probability 0.001%
Absolute noise level range [8]	Fixed 30 MHz AWGN -90…-190 dBm/Hz Full 40 MHz -95…-190 dBm/Hz Full 80 MHz -95…-190 dBm/Hz Full 125 MHz -95…-190 dBm/Hz
Noise setting resolution	0.1 dB
NBW range	0.015 …125 MHz

	RF INTERFACE
SNR range	Fixed 30MHz AWGN NBW 1..30 MHz @20MHz → -30…30 dB Full 40MHz NBW 1...40 MHz (only with options F8-12.10, F8-12.30) @40MHz → -25…30 dB Full 80MHz NBW 1...80 MHz (only with options F8-12.11, F8-12.21, F8-12.31) @80MHz → -25…30 dB Full 125MHz NBW 1...125 MHz (only with option F8-12.12, F8-12.22, F8-12.32) @125MHz → -25…25 dB
SNR range, NBW 15 … 1000 kHz	0…30 dB
SNR resolution	0.1 dB
SNR Measurement update period	< 1 s

[7] Note: Noise performance unspecified for last 20 dBm/Hz of noise level setting range.
[8] Note: absolute noise level output gain setting must be –40 dB to achieve -190 dBm/Hz. Please note thermal noise when setting the noise level below -170 dBm/Hz

Table 19. AWGN specification

2.4.11.2 CW Interference

Test setup

The system performance is verified using the parameters listed in Table 20.

	RF INTERFACE
Input signal	Sine wave (CW)
Center frequency	2 GHz
Input level	-15 dBm
Input level setting	-15 dBm average, crest factor 6 dB
Fading profile	1-path Constant, 0 ns delay, no fading
Output gain	0 dB
Interference level setting	-15 dBm
C/I settings	0 dB

Table 20. Test setup for CW interference testing

CW interference specifications

	RF INTERFACE
Number of interfering signals per channel	1
Frequency range (band edges)	350 MHz to 3000 MHz **(F8-12.11)** 350 MHz to 6000 MHz **(F8-12.31)** 350 to 3000 MHZ -> 220 to 3000 MHz **(F8-15.10)** [12] 350 to 6000 MHZ -> 220 to 6000 MHz **(F8-15.13)** [12]

	RF INTERFACE
Frequency offset range	BW 40 MHz -20…+20 MHz BW 80 MHz -40…+40 MHz BW 125 MHz -62.5…+62.5 MHz
Frequency resolution	10 kHz
Frequency accuracy [9]	±1 kHz
Frequency accuracy [10]	±0.1 kHz
C/I resolution	0.1 dB
Interference power level range	-50…-10 dBm
Interference power level resolution	0.1 dB
Interference power level accuracy	±0.75 dB
Noise power [1Hz BW]	
> +/-500kHz	< -130 dBm/Hz
> +/-700kHz	< -132 dBm/Hz
> +/-1500kHz	< -135 dBm/Hz
Harmonics/spurious signals	
> +/-500kHz [11]	< -70 dBc
> +/-700kHz [11]	< -70 dBc
> +/-1500kHz [11]	< -70 dBc
Internal summing	
C/I range	-20 ... +20 dB
Accuracy, -20 < C/I ≤ +20	±0.5 dB
External summing	
C/I range	-60 ... +60 dB
Accuracy, |C/I| < 15	±0.5 dB
Accuracy, 15 ≤ |C/I| < 25	±1 dB
Accuracy, |C/I| ≥ 25	±2 dB

[9] Over setting bandwidth
[10] At point frequencies at fc, fc±200kHz, fc±400kHz, fc±800kHz, fc±1600kHz, fc±2700kHz, fc±2800kHz, fc±3600kHz, fc±5000kHz, fc±10MHz, fc±20MHz
[11] Not specified within 10 kHz offset

Table 21. CW interference specification

2.4.12 Optional feature: Aerospace and Satellite Option (supported release 1.4)

Product configuration

In addition to basic product configuration for performance verification listed in Table 5, the following platform extensions and options have been enabled for verification of the Aerospace and Satellite Option.

CODE	DESCRIPTION
F8-46.120	Firmware with maximum of 1.3s excess channel delay with four (4) independent fading paths per channel
F8-46.100	Aerospace and Satellite channel modeling tool

Table 22. Additional product configuration items for Aerospace and Satellite Option verification

Aerospace and Satellite Option specifications

Aerospace and Satellite Option is separate operational mode in EB Propsim F8. The performance of ASO mode is optimized for high velocities and Dopplers, smooth delay sliding and long delays. The following specifications apply for ASO models. Models for terrestrial fading emulation can not be used in ASO mode.

PARAMETER	VALUE
Number of fully independent and dynamic fading paths	1 (LOS)+3 (reflectors) paths / channel
Emulation channel topologies supported	SISO
User system frequency	up to 27GHz with external up/down conversion or IF freq. in use
Adjustable propagation delay	Excess delay from 4us …1300 ms
Delay spread	up to 10 ms
Maximum acceleration	100 G
Delay profiles	Constant, dynamic periodic or user defined
Channel model update rate [12]	50 kHz
Pre-defined fading profiles	Pure Doppler (3-D LoS), user defined
Doppler shift	up to +/-1.5 MHz
Doppler profiles	Sinusoidal, triangular or linear, user defined
Interference generator	AWGN, independent per channel
AWGN mode	Constant noise power density

[12] Model values are interpolated in HW to guarantee smooth and continuous phase rotation and delay sliding

Table 23. Aerospace and Satellite Option specification

2.4.13 Optional feature: Internal RF Local Oscillator

Internal local source performance is measured at RFLO output connectors when jumper cables are removed.

PARAMETER	REQUIREMENT
Number of RF local signal generators supported	1, 2, 3, 4

PARAMETER	REQUIREMENT
Frequency range independently	3.725 … 5.050 GHz (for f_c 350…3000 MHz) 3.725 … 6.550 GHz (for f_c 350…6000 MHz) 3.675 … 5.050 GHz (for f_c 220…3000 MHz) 3.675 … 6.550 GHz (for f_c 220…6000 MHz) RF LO frequency: (7100 MHz + f_c) / 2 E.g. Fc 2GHz → (7100.000+2000.000) MHz / 2 = 4550.000 MHz
Phase noise	< -95 dBc @ 1 kHz, < -105 dBc @10 kHz, < -135 dBc @1 MHz.
Harmonics	Typical < -30 dBc

Table 24. Internal RF local specification, one source

2.5 External equipment

The external equipment is not included in standard EB Propsim F8 delivery.

2.5.1 PC accessories

The following accessories are needed in order to operate the emulator locally via graphical user interface.

- Standard PC display (SVGA 1024 x 768 pixels, 65536 colours or higher), connector high density D15, DVI. DVI - VGA adapter is included in the EB Propsim F8 delivery.
- Standard USB mouse
- Standard USB keyboard

2.5.2 External RF local

If external RF local is used, the RF performance values listed for emulator in this document are valid only in case external RF local meets or exceeds the following minimum requirements.

PARAMETER	REQUIREMENT
Number of RF signal generators needed	1…8 depending on the number of center frequencies used in emulation channels. If all channels are used at the same center frequency, one RF generator is enough. Signal must be divided to each 8 RFLO inputs by an external RF splitter and power level shall be on specified level at each RF LO input connector.
Frequency range	3.725 … 5.050 GHz (for f_c 350…3000 MHz) 3.725 … 6.550 GHz (for f_c 350…6000 MHz) 3.675 … 5.050 GHz (for f_c 220…3000 MHz)[13] 3.675 … 6.550 GHz (for f_c 220…6000 MHz) [13] RF LO frequency: (7100 MHz + f_c) / 2 E.g. f_c 2GHz → (7100.000+2000.000) MHz / 2 = 4550.000 MHz
Signal level	1 dBm ±1 dB
Input impedance	50 ohms nominal
Input SWR	< 1.8
Phase noise	< -95 dBc @ 1 kHz, < -105 dBc @10 kHz, < -135 dBc @1 MHz.
Harmonics	< -30 dBc
Connector	SMA female
Maximum allowed power	+10 dBm

[13] check the RF LO frequency from GUI with fc under 280 MHz

Table 25. External RF LO requirements

2.6 OS and pre-installed 3rd party applications

The emulator includes pre-installed and integrated standard PC module. The operating system for the graphical user interface control applications is Windows XP for embedded.

ITEM	MODELS AND VERSION DETAILS
PC module	CPU 2.0 GHz RAM size: 1 GByte Mass storage size: 160 GByte
Operating system	Windows XP for embedded. Normal installation level.
Adobe Acrobat Reader	Adobe Acrobat reader version 9.0 or newer

Table 26. Pre-installed operating system modules and applications

Windows is a registered trademark of Microsoft Corporation in the United States and other countries.

Adobe Reader is a registered trademark of Adobe Systems Incorporated in the United States and/or other countries.

2.6.1 The security and virus protection of the operating system

The operating system of the emulator is using standard features and tools provided by the Windows XP for embedded.

It is highly recommended to install up to date 3rd party virus protection tool into the emulator and secure the network interface of the emulator (if used) with the firewall before taking the emulator in normal use.

WINDOWS LOGIN USERS	PASSWORD	DESCRIPTION, USER RIGHTS
Admin	[None]	Administrator rights

Table 27. OS user name and password

The product functionalities are tested and verified without 3rd party virus protection tools and with F-secure virus protection tools installed (version 7.0). The functionality with other 3rd party virus protection tools cannot be fully guaranteed. If other than F-secure virus protection tools are used and the emulator is not functioning normally please contact EB.

F-Secure is a registered trademark of F-Secure Corporation.

2.6.2 User installations

Installation of any additional software application or external equipment to the emulator is fully on user response.

EB cannot guarantee that other than defined windows applications and PC external equipment listed in Chapter 2.5 are fully functioning or that they do not cause any problems or errors to normal performance or use of the emulator.

2.7 Emulation channel topologies

Definition of Physical and Logical channel:

- **Physical fading channel**, single RF plug-in unit of up to 8 RF channels incorporates one (1) RF IN, one (1) RF OUT and one (1) RF Local input connector.
- **The logical fading channel** is formed in digital baseband domain with the baseband resource blocks.
- In Release 1.0 one physical fading channel supports none, one or two logical fading channel configurations. The configurations can be defined by the user. Each logical fading channel has logical “input” and “output” signal ports and independent fading paths.

Definition of basic channel topologies:

- SISO = Single In Single Out emulation channel
- MISO = Multiple In Single Out emulation channel
- SIMO = Single In Multiple Out emulation channel
- MIMO = Multiple In Multiple Out emulation channel

EB Propsim F8 supports a number of different connection topologies e.g. from 1-channel point-to-point simulation to antenna array reception test with multiple transmitters or MIMO transceiver testing. When standard mode or MIMO extension 1 to 2 is used, the channel topologies can be mixed for each emulation setup e.g. SISO, SIMO, MISO and MIMO (1x1, 1x2, 2x1, 2x2) or two SISO and two MIMO (1x1, 1x1, 2x2, 2x2). With MIMO extension 1 to 4, e.g. following single MIMO topologies are supported: 4x4, 4x8 and 8x4. Additionally a configuration with two MIMO 4x4 links can be created. 1 to 8 MIMO extension option increases the available fading channel count drastically. With 1 to 8 MIMO extension mode up to 8x8 MIMO topology can be emulated with single EB Propsim F8 (requires 8 RF channels).

Definition of channel groups:

- The channel group includes one or more physical fading channels.
- There can be one or up to eight (in one EB Propsim F8 frame) different channel groups in single emulation setup.
- Each channel group can have independent emulation parameters.

Channel usage options available depend on the EB Propsim F8 channel bandwidth configuration and installed MIMO Extension option license of EB Propsim F8. Table 28 below shows the maximum available logical channel count and maximum bandwidth (depends on platform bandwidth option) with different MIMO extension options.

MIMO extension Logical channels	Bandwidth 125 MHz	 80 MHz	 40 MHz
Normal channel setting 8 (Bi-di 2x2 MIMO)	48	48	48
1to2 MIMO ext 16 (Bi-di 4x2 MIMO)	24	48	48
1to4 MIMO ext 32 (Bi-di 4x4 MIMO)	12	24	48
1to8 MIMO ext 64 (Unidi 8x8 MIMO)	N/A	12	24

Table 28. Capacity of different MIMO extension options with 8 physical channels.

Note: the emulator supports up to four independent RF local oscillators. If more than four different center frequencies are used, user shall connect external RF local oscillators to the emulator. The Running view application will allocate and inform to which physical emulation channels the external RF local is to be connected. If same center frequency is used in more than one channel the same RF local oscillator is used for these channels.

PHYSICAL CHANNELS	NORMAL CHANNEL SETTING [14]	MIMO EXTENSION 1 TO 2 CHANNEL SETTING [15]	MIMO EXTENSION 1 TO 4 CHANNEL SETTING [16]	MIMO EXTENSION 1 TO 8 CHANNEL SETTING [17]
2	Number of Logical channels in use: **2** Maximum number of fading taps per channel in use: **48** Channel topologies supported up to: **SISO: 2 groups, 1x1** **SIMO: 1x2** **MISO: 2x1** **MIMO: --**	Number of Logical channels in use: **4** Maximum number of fading taps per channel in use: **24** Channel topologies supported up to: **SISO: 2 groups, 1x1** **SIMO: 1x2** **MISO: 2x1** **MIMO: 2x2**		
4	Number of Logical channels in use: **4** Maximum number of fading taps per channel in use: **48** Channel topologies supported up to: **SISO: 4 groups, 1x1** **SIMO: 1x4** **MISO: 4x1** **MIMO: 2x2**	Number of Logical channels in use: **8** Maximum number of fading taps per channel in use: **48** Channel topologies supported up to: **SISO: 4 groups, 1x1** **SIMO: 1x4** **MISO: 4x1** **MIMO: 2x4, 4x2**	Number of Logical channels in use: **16** Maximum number of fading taps per channel in use: **48** Channel topologies supported: **MIMO: 3x3, 4x4** **MANET/Mesh: 4 radios**	
6	Number of Logical channels in use: **6** Maximum number of fading taps per channel in use: **48** Channel topologies supported up to: **SISO: 6 groups, 1x1** **SIMO: 1x6** **MISO: 6x1** **MIMO: 2x3, 3x2**	Number of Logical channels in use: **12** Maximum number of fading taps per channel in use: **48** Channel topologies supported up to: **SISO: 6 groups, 1x1** **SIMO: 1x6** **MISO: 6x1** **MIMO: 2x6, 6x2**	Number of Logical channels in use: **24** Maximum number of fading taps per channel in use: **48** Channel topologies supported: **MIMO: 3x3, 4x4**	Number of Logical channels in use: **48** Maximum number of fading taps per channel in use: **24** Channel topologies supported: **MIMO: 6x6** **MANET/Mesh: 6 radios**

PHYSICAL CHANNELS	NORMAL CHANNEL SETTING [14]	MIMO EXTENSION 1 TO 2 CHANNEL SETTING [15]	MIMO EXTENSION 1 TO 4 CHANNEL SETTING [16]	MIMO EXTENSION 1 TO 8 CHANNEL SETTING [17]
8	Number of Logical channels in use: **8** Maximum number of fading taps per channel in use: **48** Channel topologies supported up to: **SISO: 8 groups, 1x1** **SIMO: 1x8** **MISO: 8x1** **MIMO: 2x4, 4x2**	Number of Logical channels in use: **16** Maximum number of fading taps per channel in use: **48** Channel topologies supported up to: **SISO: 8 groups, 1x1** **SIMO: 1x8** **MISO: 8x1** **MIMO: 2x8, 8x2, 4x4**	Number of Logical channels in use: **32** Maximum number of fading taps per channel in use: **48** Channel topologies supported: **MIMO: 4x4, 4x8, 8x4**	Number of Logical channels in use: **64** Maximum number of fading taps per channel in use: **24** Channel topologies supported: **MIMO: 4x4, 4x8, 8x8** **MANET/Mesh: 8 radios**

[14] One logical fading channel per one physical fading channel available for emulation configuration
[15] Up to two logical fading channels per one physical channel in use
[16] Up to four logical fading channels per one physical channel in use
[17] Up to eight logical fading channels per one physical channel in use

Table 29. Connection topologies vs. channel configurations

2.8 Mechanical specifications

The dimensions and weight of EB Propsim F8 are listed in Table 30 below.

ITEM	
Net weight	< 70 kg
Shipping weight	< 110 kg
Height	640 mm
Width	430 mm
Depth	535 mm

Table 30. Dimension and weight

2.9 Environmental and safety specifications

2.9.1 Environmental and climatic requirements (operating)

EB Propsim F8 has been designed for indoor use in a Pollution Degree 2 environment. The operating should be limited to stationary use at weather-protected and dust-free locations, such as:

- offices
- laboratories
- telecommunication centers
- storage rooms for valuable and sensitive products

Requirements

- Altitude up to 2000 m.
- Temperature 15 ºC to 30ºC.
- Maximum relative humidity 80 %b.
- Rate of change of temperature <0.5 °C/min
- Mains supply voltage fluctuations up to ±10 % of the nominal voltage.

After the transportation, the temperature of the device must be stabilized before switching the power on.

Cautions:

- During the use free, unrestricted air flow must be allowed around the device to ensure proper cooling. Figure 15 below depicts the air inflow and outflow vents on the sides and back of the emulator.
- Condensation must be prevented.

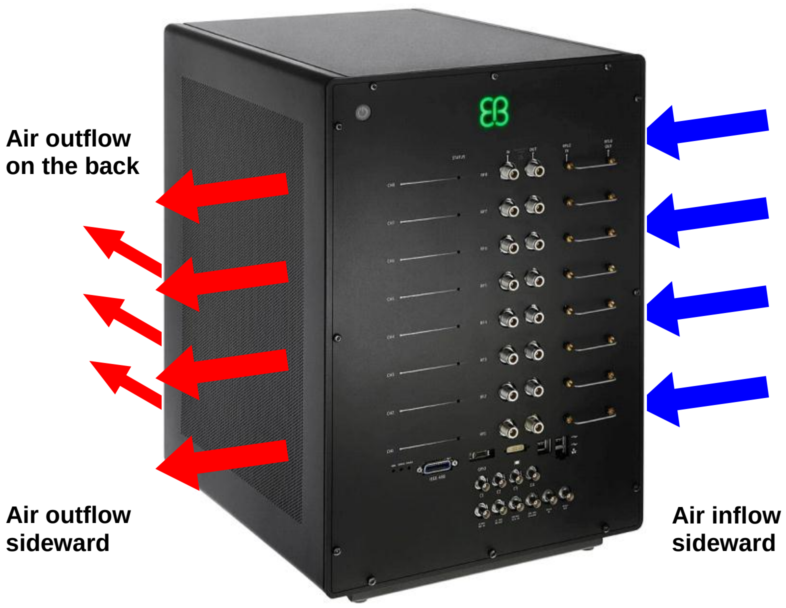


Figure 15. Air inflow and outflow on sides and back of the emulator

2.9.2 Transportation requirements

2.9.2.1 Transportation with package

ETS 300-019-1-2, class 2.2 careful transportation

- Air temperature
  - -25 ... +70°C (unventilated enclosures)
  - -25 ... +30°C, when direct transfer of the product between the two given temperatures and condensation is presumed.
- Relative humidity
  - up to 95% (not combined with rapid temperature changes at +40°C)
- Relative humidity
  - up to 95% (combined with rapid temperature changes at high relative humidity)

2.9.3 Electrical safety regulations

Tested according to IEC 61010-1 regulations.

2.9.4 Electromagnetic compatibility (EMC)

IEC 61326-1:

- Electrical equipment for measurement, control and laboratory use - EMC requirements – Part 1 General Requirements

IEC 61326-2-1:

- Electrical equipment for measurement, control and laboratory use - EMC requirements – Part 2-1.

EMC USA & Canada:

- FCC part 15.107 and 15.109 and IC standard RSS-210. ICES-003

2.9.5 Compatible standards and regulations

- WEEE according to EU regulations in case of product disposal, 2002/96/EC
- ETL Control number 3064827
- CB Certification number SE-56678, TRF 816104-1
- CE tested according to EMC: IEC 61326-1 & IEC 61326-2-1 Safety: IEC 61010-1

3 Product Specification F32 and FS8

This document is product specification of EB Propsim F32 and FS8. It states:

- most wireless applications and technologies supported by the EB Propsim F32 and FS8 emulators
- standard product configuration items
- product high level block diagram and interfaces
- electrical specifications
- mechanical dimensions
- environmental and safety standards and specification
- supported emulation channel topologies
- list of the emulator tools
- details of the operating system
- requirements for the external devices and equipment

Figure 16. EB Propsim F32 and EB Propsim FS8

Note! Elektrobit System Test Ltd. reserves the right to change the specifications without notice due to continuous improvement of the product.

3.1 Applications

EB Propsim is a versatile radio channel emulator platform capable for several different test cases. In general the emulator is independent of input signal and channel model.

Typically the radio channel emulator is required in performing the product conformance test cases defined in related standards. In addition to standard test cases multiple other wireless applications in defense, research and satellite areas can be tested with the EB Propsim channel emulator. The emulator is suitable for performance testing of the following example applications and technologies:

- 4G research
- CDMA, CDMA2000
- DVB-H, DVB-T
- GSM, EDGE, GPRS
- Jammers, communication and radar
- 3GPP LTE (TDD and FDD), Long Term Evolution
- MANET/Mesh, Mobile Add-hoc Networks
- Multi-user, client network level testing
- Radar
- SDR, software defined radios
- TETRA
- WCDMA
- WiBRO
- WiMAX, fixed and mobile
- WLAN, WiFi
- MIMO, multiple in multiple out transceivers
- Beamforming
- OTA, over the air testing
- Ray tracing
- and many others

Comprehensive package of standard channel models is included in every EB Propsim device. See Standard Channel Models section in Appendix A for further information.

3.2 Product configuration items

Table 31 describes the standard EB Propsim F32 (standalone) product configuration items. EB Propsim FS8 product configuration items are described in Table 32. Your product configuration can be found in EB Propsim extranet http://www.elektrobit.com/extranet. The user name and password are included in the product delivery contents.

CODE	DESCRIPTION
Product item codes included to baseline and EB recommended configurations	
F32-01.10	EB Propsim F32 cabinet platform
F32-12.30	RF interface channel. Frequency range support 350MHz …. 2700MHz. Signal bandwidth 40MHz
F32LO-1.N	1 Internal Local oscillator frequency for N (8, 16, 24 or 32) RF channels
F32-06.01	MIMO extension 1 to 2
F32-06.02	MIMO extension 1 to 4
F32-41	EB Propsim Standard Channel Modeling toolbox for external PC
Optional items for emulation channels	
F32-15.10	Frequency range extension 350…2700MHz --> 30MHz…2700MHz.
Internal RF local oscillator sub-system extension	
F32-16.12	Internal RF local oscillator sub-system extension - Increases the number of RF LO from one (1) to two (2)
Firmware options	
F32-05.01	Shadow fading emulation
F32-10	Internal Interference Platform
F32-10.01	AWGN source. One (1) source per RF channel.
F32-10.10	CW source. One (1) source per RF channel.
Advanced emulator applications and tools	
F32-42.01	MIMO channel modeling tool for WiFi (TGn 802.11n)
F32-43.02	Geometric Channel Modeling (GCM) Application
F32-44.100	EB MIMO OTA Application

CODE	DESCRIPTION
F32-45.100	EB WES Application (CSS)

Table 31. EB Propsim F32 Emulator configuration items and codes

CODE	DESCRIPTION
Product item codes included to baseline and EB recommended configurations	
FS8-01.10	EB Propsim FS8 cabinet platform
FS8-12.30	RF interface channel. Frequency range support 350MHz …. 2700MHz. Signal bandwidth 40MHz
FS8LO-1.N	1 Internal Local oscillator frequency for N (2, 4, 6 or 8) RF channels
FS8-06.01	MIMO extension 1 to 2
FS8-06.02	MIMO extension 1 to 4
FS8-41	EB Propsim Standard Channel Modeling toolbox for external PC
Optional items for emulation channels	
FS8-15.10	Frequency range extension 350…2700MHz --> 30MHz…2700MHz.
Internal RF local oscillator sub-system extension	
FS8-16.12	Internal RF local oscillator sub-system extension - Increases the number of RF LO from one (1) to two (2)
Firmware options	
FS8-05.01	Shadow fading emulation
FS8-10	Internal Interference Platform
FS8-10.01	AWGN source. One (1) source per RF channel.
FS8-10.10	CW source. One (1) source per RF channel.
Advanced emulator applications and tools	
FS8-42.01	MIMO channel modeling tool for WiFi (TGn 802.11n)
FS8-43.02	Geometric Channel Modeling (GCM) Application
FS8-44.100	EB MIMO OTA Application

CODE	DESCRIPTION
FS8-45.100	EB WES Application (CSS)

Table 32. EB Propsim FS8 Emulator configuration items and codes

3.3 Definitions and requirements

3.3.1 Definitions

The performance values are verified with standalone EB Propsim F32 product configuration described in Table 33 if not otherwise noted. Corresponding configuration for EB Propsim FS8 is described in Table 34.

CODE	DESCRIPTION
F32B32CH	F32 baseline configuration with thirty two (32) in/out interface channels RF range 350 …2700MHz. Signal bandwidth 40MHz.
F32-15.12	Frequency range extension 350…2700MHz --> 30MHz…2700MHz.
F32-16.12	Internal RF local oscillator sub-system extension - Increases the number of RF LO from one (1) to two (2)
F32-10	Internal Interference Platform
F32-10.01	AWGN source. One (1) source per RF channel.
F32-10.10	CW source. One (1) source per RF channel.
F32-05.01	Shadow fading emulation

Table 33. EB Propsim F32 configurations for performance verification

CODE	DESCRIPTION
FS8B8CH	FS8 baseline configuration with eight (8) in/out interface channels RF range 350 …2700MHz. Signal bandwidth 40MHz.
FS8-15.12	Frequency range extension 350…2700MHz --> 30MHz…2700MHz.
FS8-16.12	Internal RF local oscillator sub-system extension - Increases the number of RF LO from one (1) to two (2)
FS8-10	Internal Interference Platform
FS8-10.01	AWGN source. One (1) source per RF channel.
FS8-10.10	CW source. One (1) source per RF channel.
FS8-05.01	Shadow fading emulation

Table 34. EB Propsim FS8 configurations for performance verification

3.3.2 Conditions required to meet specifications

The following conditions are required to meet the specifications:

- The emulator is within calibration cycle.
- At least 4 hours of storage at a constant temperature within the operating temperature range.
- The emulator has been switched on for a one hour warm-up period.
- No other view, application or 3rd party software (except F-secure virus protection, see chapter 3.6.1) is running simultaneously with EB Propsim Running view.

3.3.3 Test setup

The system performance is verified with the settings and under the conditions listed in Table 35 if not otherwise noted.

	RF INTERFACE
Input signal	Sine wave (CW)
Input level	-15 dBm
Input level setting	-15 dBm average, crest factor 10 dB
Bandwidth	40 MHz
Fading profile	1-path Constant, 0 ns delay, no fading
Total channel gain	-5 dB (→ output power level -20 dBm)
MIMO extension	not used
Ambient temperature	+24 ± 2 °C
Center frequency	2 GHz

Table 35. Emulator setup and conditions

The test setups for verifying performance values are presented in Figure 17 and Figure 18.

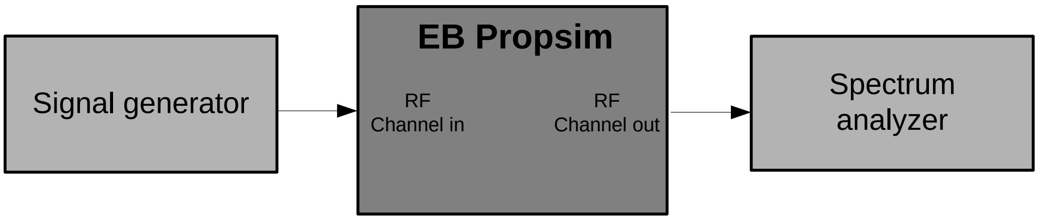


Figure 17. Test set up with signal generator and spectrum analyzer

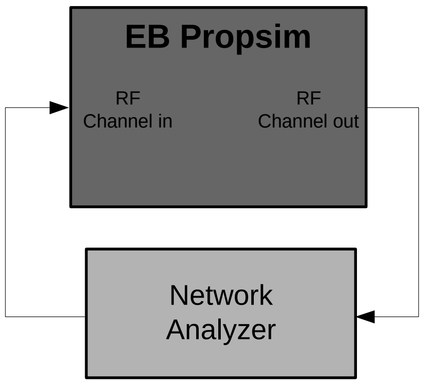


Figure 18. Test set up with network analyzer

3.4 Electrical specifications

3.4.1 Power supply

Voltage	100…240 VAC
Frequency	50/60 Hz
Power	1500W (F32)
	600W (FS8)

3.4.2 Absolute maximum ratings

PORT / CONNECTOR	ABSOLUTE MAXIMUM RATING
RF IN/OUT 1…32 (1..8 in FS8)	+33 dBm peak
RF OUT 1 …32 (1..8 in FS8)	+10 dBm peak reverse power
RF LO INPUT	+10 dBm peak
RF LO OUTPUT	+10 dBm peak
C1 SYNC IN	0 / +3.8 VDC
C2 SYNC OUT	0 / +3.8 VDC
C3 – C4	reserved for future use
10 MHz REF IN	+20 dBm 50 ohm
10 MHz REF OUT	+20 dBm 50 ohm (reverse)
200MHz SCLK IN	+15 dBm 50 ohm
200MHz SCLK OUT	+15 dBm 50 ohm (reverse)
IO CLK IN (F32)	0 / +3.8 VDC
IO CLK OUT (F32)	0 / +3.8 VDC (reverse)
GPIO	0 / +3.8 VDC
IEEE 488	-0.3 VDC / 4.3 VDC
AC power plug	250 VAC 50/60 Hz

Table 36. Absolute maximum ratings

3.4.3 Emulator block diagram

The block diagram of the EB Propsim F32 / FS8 is presented in Figure 19.

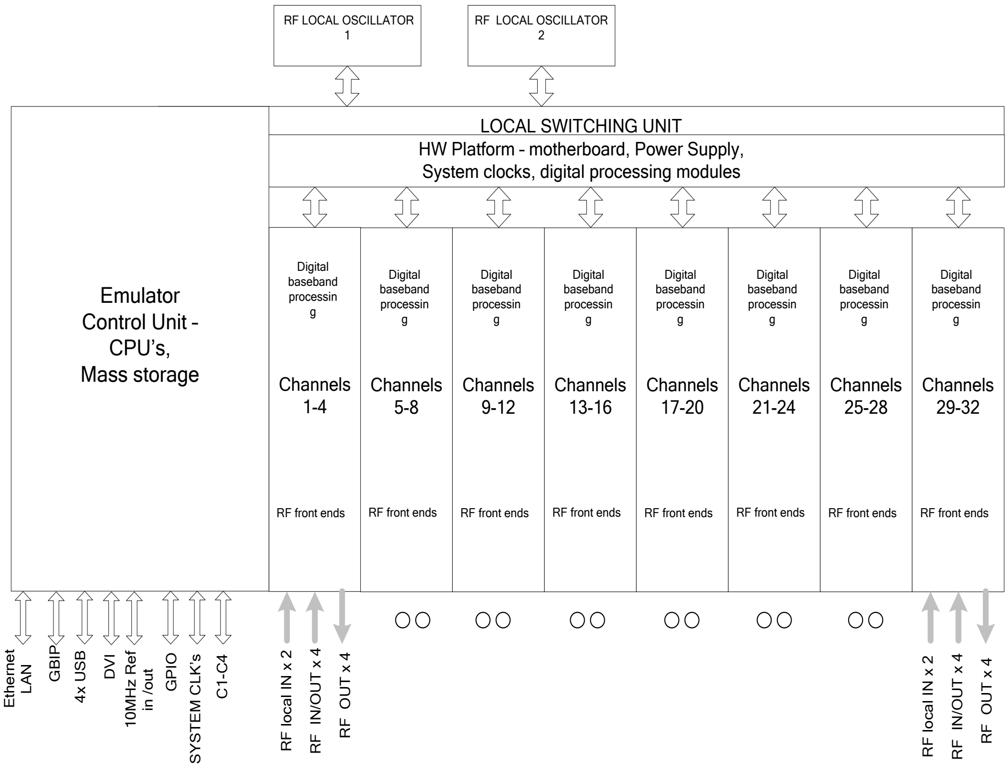


Figure 19. Emulator units, modules and interfaces

3.4.4 Physical connectors and LEDs

EB Propsim F32 can be equipped with 4, 8, 16 or 32 independent physical channels, EB Propsim FS8 correspondingly with 4 or 8. Each channel has input and output RF connectors in the front panel.

Single fully configured EB Propsim F32 can support following emulation topologies:
16 times 2x2 MIMO
8 times 2x2 MIMO Bi-Directional
8 times 4x4 MIMO
4 times 4x4 MIMO Bi-Directional
4 times 2x8 MIMO
2 times 2x8 MIMO Bi-Directional
4 times 4x8 MIMO
2 times 4x8 MIMO Bi-Directional
2 times 8x8 MIMO
8x8 MIMO Bi-Directional
4x32 MIMO
Manet/Mesh network up to 11 radios

Single fully configured EB Propsim FS8 can support following emulation topologies
4 times 2x2 MIMO
2 times 2x2 MIMO Bi-Directional
2 times 4x4 MIMO
4x4 MIMO Bi-Directional
2x8 MIMO
4x8 MIMO
Manet/Mesh network up to 5 radios

See Chapter 3.7 for more details on supported topologies.

All external connectors and LEDs of the EB Propsim are found in the emulator front panel, as shown in Figure 20, Figure 21, Figure 22 and Figure 23.

Figure 20. EB Propsim 32 front panel

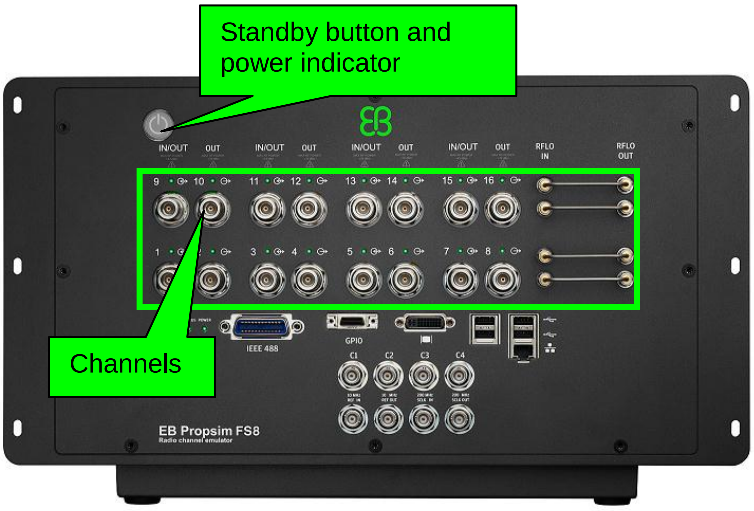


Figure 21. EB Propsim FS8 front panel, rack mountable

Figure 22. EB Propsim 32 / FS8 channel connectors

Figure 23. EB Propsim F32 and FS8 control connectors

The LEDs and connectors are listed in Table 37.

NAME	DESCRIPTION	SPECIFICATION
RF IN/OUT	RF input connector (for each physical emulator channel)	N female, 50 ohm nominal Input VSWR < 1.6
RF OUT	RF output connector (for each physical emulator channel)	N female, 50 ohm nominal Output VSWR < 1.6
RFLO IN	Local oscillator input (for each physical emulator channel)	SMA female, 50 ohm nominal Input VSWR < 1.8
RFLO OUT	Local oscillator output (for each physical emulator channel)	SMA female, 50 ohm nominal Input VSWR < 1.8
GPIO	16-pin programmable general purpose input/output connector	LVDS. Reserved for future use.
IEEE 488	GPIB connector for automated test control	24 pin centronix, female
DVI	DVI output for an external display	DVI-I, female
USB	USB connectors (4) for external keyboard, mouse and memory	USB2.0 Standard
Ethernet	Gigabit LAN, 10/100/1000 Base-T connector for LAN connectivity.	RJ45
C1	Sync in	BNC, LVTTL
C2	Sync out	BNC, LVTTL
C3 – C4	Programmable input/output connectors, reserved for future use	BNC, LVTTL
10 MHz REF IN	Input connector for 10 MHz reference signal	Connector Type: BNC Impedance: 50 ohm. Signal level > 0 dBm, + 20dBm max. Max frequency deviation ±5 Hz

NAME	DESCRIPTION	SPECIFICATION
10 MHz REF OUT	Output connector for 10 MHz reference signal	Connector Type: BNC Impedance: 50 ohm. Signal level nominal 2dBm.
200MHz SCLK IN	200MHz sampling clock signal input	Connector Type: BNC Impedance: 50 ohm. Signal level > 3dBm, +15dBm max.
200MHz SCLK OUT	200MHz sampling clock signal output	Connector Type: BNC Impedance: 50 ohm. Signal level nominal level 7dBm
IO CLK IN (F32)	0-200MHz sample clock in	BNC Signal PECL
IO CLK OUT (F32)	0-200MHz sample clock out	BNC Signal LVTTL
Power indicator	EB LOGO (F32) Standby switch (FS8)	-
HDD LED	Hard disk staus: • Blinking green HDD activity	-
STATUS LED	System status: • Green Status ok • Blinking Emulation running • Red Warning Warning is indicated in case of overheating, overvoltage or selftest failure. More detailed infromation about the reason of the warning is reported in the GUI.	-
POWER LED	Power status: • Blinking red Initializing • Green Status ok • Red Power failure In case of red power indication, shut down and restart the system.	-
CH STATUS LED	Channel status (for each RF connector): • No indication Connector not in use • Green Status ok • Blinking Emulation running • Red Warning • Blue Connector selected in GUI Warning is indicated in case of overheating, overvoltage, input cut-off, missing RF local missing or selftest failure. More detailed infromation about the reason of the warning is reported in the GUI.	-

Table 37. EB Propsim F32 and FS8 front panel connectors and LEDs

3.4.5 Rear panel

The rear panel of the EB Propsim F32 and FS8 contain the AC power input with the mains power switch, see Figure 24 and Table 38.

Figure 24. EB Propsim F32 and FS8 rear panels

NAME	DESCRIPTION	SPECIFICATION
AC INPUT	AC input with mains power switch	IEC inlet filter

Table 38. EB Propsim F32 and FS8 rear panels

3.4.6 User control interfaces

EB Propsim can be controlled either through Graphical User Interface or with remote control commands.

- EB Propsim Graphical User Interface (GUI) and other software tools are operating under Windows XP for embedded environment.
- Remote Control (i.e. Automatic Test Equipment Control, ATE Control) can be used to execute a large part of the actions available to the user in Running view application. See Remote Control section of user reference for details on commands.
- The Remote Control can be used via GPIB or LAN:
  - GPIB interface supports standard GPIB (IEEE 488.2) SCPI commands.
  - LAN interface supports standard SCPI commands.

3.4.7 RF characteristics

The key RF characteristics are listed in Table 39.

PARAMETER	VALUE
Platform frequency range	350 MHz to 2700 MHz **(FS8/F32-12.11)** 350 to 2700 MHZ -> 30 to 2700 MHz **(FS8/F32-15.10)** [1] Note: The channel f_c can be set independently.
Instantaneous channel bandwidth	40 MHz
Fading tap mean amplitude range	0 … -60 dB (resolution 0.1 dB)
Input signal level range for full reference [2]	-30 ... +20 dBm (with 10 dB crest factor setting)

PARAMETER	VALUE
Input signal level range for > 35 dB SNR	-50 … +20 dBm (with 10 dB crest factor setting)
Input crest factor setting	0…24 dB (resolution 0.1 dB)
Output reference level, for full reference [2]	-120 ... -20 dBm (RMS, with 10 dB crest factor setting)
Input and output level resolution	0.1 dB
Output level setting	-120….-20 dBm (with 10 dB crest factor setting)
Output gain setting	0…-100 dB
Peak output level	0 dBm
Number of input bits (AD converter)	16
Number of output bits (DA converter)	16
Analog gain path attenuation range	100 dB
Path ON/OFF ratio	Typical > 100dB
Absolute max. input/output level without damage	+33 dBm
Output noise floor	< -160 dBm/Hz (output reference level ≤ -40 dBm) Typical value < -165 dBm/Hz (output reference ≤ -40 dBm)

[1] Operation outside specified frequency range is not specified.

[2] “Full reference” refers to such emulator setting that gives the maximum linearity performance for the user signal in RF domain.

Table 39. RF characteristic

3.4.7.1 Linearity and spurious signal performance

PARAMETER	VALUE
Error Vector Magnitude (EVM).	Typical WCDMA signal (3.84 Mchips/s) at 2000 MHz < -45 dB RMS Typical OFDMA signal 20 MHz, 64QAM at 2000 MHz < -45 dB RMS
Adjacent Channel Power (ACP) for WCDMA modulation (3.84 Mchips/s)	Typical -57 dBc @ 5 MHz offset at 2000MHz
Baseband harmonics, in band spurious	SFDR, Typical value 60 dB
Leaking RFLO @ [(f_c)+3525MHz]	max. < -40 dBm Typical <-70dBm
Other out-band spurious 350MHz…7000MHz	max. < -40 dBm Typical <-70dBm
Output level accuracy	±1.5 dB[1] Typical value ±0.5 dB
Input and output phase linearity @ f_c	< 10 °
Output frequency response flatness	Over 40 MHz BW < 3 dB Typical Over 40 MHz BW < 1 dB
Group delay variation (peak-to-peak)	< 30 ns over 40 MHz BW Typical < 15 ns over 40 MHz BW

PARAMETER	VALUE
Average gain variation when changing from one fading model to another (at band center)	< 0.5 dB
Gain variation due to variation in ambient temperature	< 0.2 dB / °C
Duration of signal path interrupt when changing emulation	Typical 0 µs for SISO configurations[2]
Long term output power variation at band center in one year.	1 dB max
Classical LCR deviation from the theoretical LCR at levels -30 dB ... +3 dB from the mean power level (*SD* = 2)	± 10%
Classical CPDF deviation from the theoretical CPDF at levels -20 dB ... +10 dB from the mean power level (*SD* = 2)	± 1 dB
Classical CPDF deviation from the theoretical CPDF at levels -30 dB ... –20 dB from the mean power level (*SD* = 2)	± 5 dB

[1] Accuracy specified down to -115dBm signal level
[2] When using ports from same RF channel

Table 40. Linearity and spurious signal performance

The multi channel performance is verified in terms of phase and amplitude variations between the emulator channels. One channel is selected as a reference against which the other channels are compared. All calibrations are done via GUI. The error i.e. channel-to-channel phase difference is measured with a network analyzer. The DoA performance is defined for uniform linear array where antenna element spacing is $\lambda/2$.

Normal angle is 90° while 0° means DoA directly from right.

	RF INTERFACE	
DoA (Direction Of Arrival) steps	0.1°	
DoA accuracy, calibration every 2…4 hours and typical lab environment	± 1° @ 2 GHz,	DoA 20°…160°
Average phase error between channels, daily calibration and typical lab environment	± 5° @ 2 GHz	
Average phase error between channels, calibration every 2…4 hours and typical lab environment	± 1° @ 2 GHz	
Amplitude error between channels, daily calibration and typical lab environment	± 0.5 dB @ 2 GHz	
Amplitude error between channels, calibration every 2…4 hours and typical lab environment	± 0.2 dB @ 2 GHz	

Table 41. Multi channel performance

3.4.8 Fading channel emulation

The digital calculation resources of EB Propsim F32 and FS8 can be allocated in a flexible way. This chapter presents the maximum (or an example when mentioned) figures of the achievable performance of the emulator.

	RF INTERFACE
Maximum number of independent paths per fading channel.	Up to 48
Maximum number of independent paths per logical channel when all channels in use Paths within dynamic <15 µs delay spread block.	Up to 12
Tap Delay resolution[5]	Minimum 20 ns
Maximum delay spread	3000 us
Sample density SD (samples per half wavelength)	Typical 250 when Doppler < 700 Hz
Max doppler 1 path/emulator	18.5 kHz (Effective SD 10)
Max doppler 2x2 MIMO 48 paths/channel	2.0 kHz (Effective SD > 64)
Max doppler 4x4 MIMO 12 paths/channel	2.0 kHz (Effective SD > 64)
Max doppler 4x4 MIMO 24 paths/channel	2.0 kHz (Effective SD > 64)
Number of impulse responses (CIRs) saved to memory	200 000 000 when 1 path / emulator 1 300 000 when 16 channels (4x4 MIMO) with 12 paths/channel
Maximum emulation repetition interval	60 minutes, 4x4 MIMO 6 paths/channels, vehicle speed 100 km/h, frequency 2 GHz
Minimum path insertion delay, input connector to output connector (SISO), 0 ns delay setting	< 5 µs Depends on channel topology
Fading profiles, standard	Delay Fixed Sinusoidal Linear 3GPP Birth-Death (random hopping) Average magnitude Phase Shift Fading distribution & Doppler spread Constant Constant – Pure Doppler (Pure Doppler) Rayleigh – Jakes (Classical) Rayleigh – Gaussian (Gaussian) Rayleigh – Flat (Flat) Rayleigh – Butterworth Rayleigh – Rounded Rice – Jakes (Rice) Rice – Gaussian Rice – Flat Rice – Butterworth Rice – Rounded Nakagami Lognormal Suzuki Geometrical model parameters DoA and DoD mean angle DoA and DoD spread DoA and DoD distribution Mobile heading Scatters Line-of-sight component parameters Correlation matrix

[5] User can apply custom delay interpolation methods utilizing more tap resources through IR-file in order to improve delay resolution

Table 42. Fading channel emulation performance

3.4.9 Multi-emulator synchronization

Synchronization of multiple emulators is supported from Release 1.1.1.

Test setup

The system performance is verified using settings listed in Table 43.

PARAMETER	VALUE
Number of emulators	5
Synchronization cables	Cable type: 5 ns / m (velocity factor 0.66) Connector type: BNC Cable length: 2.0 m
10 MHz clock source	Epsilon Clock® Series 2S, GPS-locked
10 MHz amplifier / switch	Epsilon SAS Switch and amplifier unit
Connection topology	1to8

Table 43. Test setup for multi-emulator performance testing

Performance, features & setup requirements

The performance, features and setup requirements of multi-emulator synchronization are listed in Table 44.

PARAMETER	VALUE
EB Propsim HW	All the emulators of the setup shall be of equal firmware and HW version, and calibrated. Contact EB for further information.
EB Propsim firmware	Release 3.0.0 or above
Synchronization accuracy	+/-25 ns typical
Number of emulators	2 – 6
Supported connection topologies	1to8, 2to8, 4to8 [6] The same connection topology shall be used in all emulators of multi-emulator setup
Synchronization cables (between cascaded emulators)	Cable type: 5 ns / m (velocity factor 0.66) Connector type: BNC All the synchronization cables used in the setup shall be of equal length. Supported lengths: 2.0 m ± 0.02 m 4.0 m ± 0.02 m 6.0 m ± 0.02 m
10 MHz clock cables	Connector type: BNC All the reference cables used in the setup shall be of equal type and length.
External 10 MHz ref clock	Connector type: BNC Impedance: 50 ohm Signal level: > 0 dBm, +20 dBm max, >10 dBm for optimal performance Point to point connection to each emulator Duty cycle: 50 ± 2 % Max frequency deviation: ±5 Hz Timing accuracy between emulators: < 200 ps Phase accuracy between emulators: <2 degrees

[6] Contact EB for information about support for other topologies

Table 44. Performance and features of simultaneous start

For more information, refer to EB Propsim Multi-emulator Synchronization application note.

3.4.10 Optional feature: Shadowing

3.4.10.1 Test setup

The system performance is verified using the settings listed in Table 45.

	RF INTERFACE
Input signal	Sine wave (CW)
Input level	-15 dBm
Input level setting	-15 dBm average, crest factor 10 dB
Output gain	0 dB
Fading profile	1-path constant, 0 ns delay, no fading
Simulation configuration	One input, channel and output
Shadowing profile	Sinusoidal
Shadowing settings	Mean attenuation: 50 dB Period: 10 s Amplitude: 50 dB Start phase: 90

Table 45. Shadowing settings for testing

3.4.10.2 Test output

Measured output should show changing signal level:

- Maximum signal level: approximately -20 dBm
- Minimum signal level: approximately -120 dBm
- At the simulation start the signal level is approximately -120 dBm
- Signal level changes along sinusoidal pattern in which the period is 10 seconds

3.4.10.3 Shadowing performance

ITEM	CHARACTERISTICS
Maximum shadowing profile update rate	100 Hz
Output level accuracy for shadowing	+/- 1.5 dB
Output resolution	0.1 dB
Minimum shadowing attenuation	0 dB
Maximum shadowing attenuation	100 dB
Maximum period length of shadowing profile (computational)	> 10 years

Table 46. Shadowing performance

3.4.11 Optional feature: Internal Interference Generator

One interference source (AWGN or CW) is supported per channel at the time.

3.4.11.1 AWGN

Test setup

The system performance is verified using the parameters listed in Table 47.

	RF INTERFACE
Input signal	Sine wave (CW)
Center frequency	2 GHz
Input level	-15 dBm
Input level setting	-15 dBm average, crest factor 10 dB
Fading profile	1-path Constant, 0 ns delay, no fading
Output gain	0 dB
Noise level setting	-110 dBm/Hz
C/N settings	0 dB / 3.84 MHz

Table 47. Test setup for AWGN testing

AWGN interference specifications

	RF INTERFACE
Frequency offset range	Filtered 30 MHz AWGN ± 5 MHz
Noise flatness at fixed f_0 [7]	Filtered 30MHz AWGN ± 1.5 dB Full 40MHz BW ± 1.5 dB
Noise crest factor	13 dB maximum / 10.5 dB with probability 0.001%
Absolute noise level range [8]	Filtered 30 MHz AWGN -90…-190 dBm/Hz Full 40 MHz -95…-190 dBm/Hz
Noise setting resolution	0.1 dB
NBW range	0.015 …40 MHz
SNR range	Filtered 30MHz AWGN NBW 1…30 MHz @20MHz → -30…30 dB Full 40MHz NBW 1...40 MHz @40MHz → -25…30 dB
SNR range, NBW 15 … 1000 kHz	0…30 dB
SNR resolution	0.1 dB

	RF INTERFACE
SNR Measurement update period	< 1 s

[7] Note: Noise performance unspecified for last 20 dBm/Hz of noise level setting range.
[8] Note: absolute noise level output gain setting must be –40 dB to achieve -190 dBm/Hz. Please note thermal noise when setting the noise level below -170 dBm/Hz

Table 48. AWGN specification

3.4.11.2 CW Interference

Test setup

The system performance is verified using the parameters listed in Table 49.

	RF INTERFACE
Input signal	Sine wave (CW)
Center frequency	2 GHz
Input level	-15 dBm
Input level setting	-15 dBm average, crest factor 10 dB
Fading profile	1-path Constant, 0 ns delay, no fading
Output gain	0 dB
Interference level setting	-15 dBm
C/I settings	0 dB

Table 49. Test setup for CW interference testing

CW interference specifications

	RF INTERFACE
Number of interfering signals per channel	1
Frequency range (band edges)	350 MHz to 2700 MHz 350 to 2700 MHZ -> 30 to 2700 MHz
Frequency offset range	-20…+20 MHz
Frequency resolution	10 kHz
Frequency accuracy [9]	±1 kHz
Frequency accuracy [10]	±0.1 kHz
C/I resolution	0.1 dB
Interference power level range	-50…-10 dBm
Interference power level resolution	0.1 dB
Interference power level accuracy	±1.5 dB
Noise power [1Hz BW]	
> +/-500kHz	< -130 dBm/Hz
> +/-700kHz	< -132 dBm/Hz
> +/-1500kHz	< -135 dBm/Hz

RF INTERFACE	
Harmonics/spurious signals	
> +/-500kHz [11]	< -70 dBc
> +/-700kHz [11]	< -70 dBc
> +/-1500kHz [11]	< -70 dBc
Internal summing	
C/I range	-20 ... +20 dB
Accuracy, -20 < C/I ≤ +20	±0.5 dB
External summing	
C/I range	-60 ... +60 dB
Accuracy, |C/I| < 15	±0.5 dB
Accuracy, 15 ≤ |C/I| < 25	±1 dB
Accuracy, |C/I| ≥ 25	±2 dB

[9] Over setting bandwidth
[10] At point frequencies at fc, fc±200kHz, fc±400kHz, fc±800kHz, fc±1600kHz, fc±2700kHz, fc±2800kHz, fc±3600kHz, fc±5000kHz, fc±10MHz, fc±20MHz
[11] Not specified within 10 kHz offset

Table 50. CW interference specification

3.4.12 Internal RF Local Oscillator

Internal local source performance is measured at RFLO output connectors when jumper cables are removed.

PARAMETER	REQUIREMENT
Number of RF local signal generators supported	1, 2
Frequency range independently	3.875 … 6.225 GHz (for f_c 350…2700 MHz) 3.575 … 6.225 GHz (for f_c 30…2700 MHz) RFLO frequency: 3525 MHz + f_c E.g. f_c = 2GHz → 3525.000+2000.000 MHz = 5525.000 MHz
Phase noise	< -95 dBc @ 1 kHz, < -105 dBc @10 kHz, < -135 dBc @1 MHz.
Harmonics	Typical < -30 dBc

Table 51. Internal RF local specification, one source

3.5 External equipment

The external equipment is not included in standard EB Propsim delivery.

3.5.1 PC accessories

The following accessories are needed in order to operate the emulator locally via graphical user interface.

- Standard PC display (SVGA 1024 x 768 pixels, 65536 colours or higher), connector high density D15, DVI. DVI - VGA adapter is included in the EB Propsim delivery.
- Standard USB mouse
- Standard USB keyboard

3.6 OS and pre-installed 3rd party applications

The emulator includes pre-installed and integrated standard PC module. The operating system for the graphical user interface control applications is Windows XP for embedded.

ITEM	MODELS AND VERSION DETAILS
PC module	CPU 2.0 GHz RAM size: 1 GByte Mass storage size: 160 GByte
Operating system	Windows XP for embedded. Normal installation level.
Adobe Acrobat Reader	Adobe Acrobat reader version 9.0 or newer

Table 52. Pre-installed operating system modules and applications

Windows is a registered trademark of Microsoft Corporation in the United States and other countries.

Adobe Reader is a registered trademark of Adobe Systems Incorporated in the United States and/or other countries.

3.6.1 The security and virus protection of the operating system

The operating system of the emulator is using standard features and tools provided by the Windows XP for embedded.

It is highly recommended to install up to date 3rd party virus protection tool into the emulator and secure the network interface of the emulator (if used) with the firewall before taking the emulator in normal use.

WINDOWS LOGIN USERS	PASSWORD	DESCRIPTION, USER RIGHTS
Admin	[None]	Administrator rights

Table 53. OS user name and password

The product functionalities are tested and verified without 3rd party virus protection tools and with F-secure virus protection tools installed (version 7.0). The functionality with other

3rd party virus protection tools cannot be fully guaranteed. If other than F-secure virus protection tools are used and the emulator is not functioning normally please contact EB.

F-Secure is a registered trademark of F-Secure Corporation.

3.6.2 User installations

Installation of any additional software application or external equipment to the emulator is fully on user response.

EB cannot guarantee that other than defined windows applications and PC external equipment listed in Chapter 3.5 are fully functioning or that they do not cause any problems or errors to normal performance or use of the emulator.

3.7 Emulation channel topologies

Definition of Physical and Logical channel:

- **Physical fading channel**, single RF plug-in unit of up to 32 RF channels incorporates one (1) RF IN, one (1) RF OUT.
- **The logical fading channel** is formed in digital baseband domain with the baseband resource blocks.

Definition of basic channel topologies:

- SISO = Single In Single Out emulation channel
- MISO = Multiple In Single Out emulation channel
- SIMO = Single In Multiple Out emulation channel
- MIMO = Multiple In Multiple Out emulation channel
- MANET/Mesh = Group of radios, all links defined between radios

EB Propsim supports a number of different connection topologies e.g. from 1-channel point-to-point simulation to antenna array reception test with multiple transmitters or MIMO transceiver testing. The channel topologies can be mixed for each emulation setup e.g. SISO, SIMO, MISO, MIMO and MANET/Mesh, up to 8x8 bi-directional MIMO in single fully configured F32.

PHYSICAL CHANNELS	CONNECTION TOPOLOGIES
2 (FS8)	Channel topologies supported up to: **SISO: 2 groups, 1x1** **SIMO: 1x2** **MISO: 2x1** **MIMO: 2x2** Bi-directional channel topologies supported up to: **SISO 1x1 bi-directional**
4 (FS8)	Channel topologies supported up to: **SISO: 4 groups, 1x1** **SIMO: 1x4** **MISO: 4x1** **MIMO: 4x4** Bi-directional channel topologies supported up to: **MIMO: 2x2 bi-directional**
6 (FS8)	Channel topologies supported up to: **SISO: 6 groups, 1x1** **SIMO: 1x6** **MISO: 6x1** **MIMO: 6x4** **MIMO: 4x6** Bi-directional channel topologies supported up to: **MIMO: 4x2 bi-directional** **MANET/Mesh: 5 radios**
8 (FS8, F32)	Channel topologies supported up to: **SISO: 8 groups, 1x1** **SIMO: 1x8** **MISO: 8x1** **MIMO: 8x4** **MIMO: 4x8** Bi-directional channel topologies supported up to: **MIMO: 2 times 2x2 bi-directional** **MIMO: 4x4 bi-directional** **MANET/Mesh: 5 radios**

PHYSICAL CHANNELS	CONNECTION TOPOLOGIES
16 (F32)	Channel topologies supported up to: **SISO: 16 groups, 1x1** **SIMO: 1x16** **MISO: 16x1** **MIMO: 8x8** **MIMO 4x16** **MIMO 16x4** Bi-directional channel topologies supported up to: **MIMO: 4x8 bi-directional** **MIMO: 8x4 bi-directional** **MIMO: 4 times 2x2 bi-directional** **MIMO: 2 times 4x4 bi-directional** **MANET/Mesh: 8 radios**
24 (F32)	Channel topologies supported up to: **SISO: 24 groups, 1x1** **SIMO: 1x24** **MISO: 24x1** **MIMO: 10x8** **MIMO: 4x24** **MIMO: 24x4** Bi-directional channel topologies supported up to: **MIMO: 6 times 2x2 bi-directional** **MIMO: 3 times 4x4 bi-directional** **MIMO: 4x8 bi-directional** **MIMO: 8x4 bi-directional** **MANET/Mesh: 8 radios**
32 (F32)	Channel topologies supported up to: **SISO: 32 groups, 1x1** **SIMO: 1x32** **MISO: 32x1** **MIMO: 10x10** **MIMO: 4x32** **MIMO: 32x4** Bi-directional channel topologies supported up to: **MIMO: 8 times 2x2 bi-directional** **MIMO: 4 times 4x4 bi-directional** **MIMO: 8x8 bi-directional** **MANET/Mesh: 11 radios**

Table 54. Connection topologies vs. channel configurations

3.8 Mechanical specifications

The dimensions and weight of EB Propsim F32 are listed in Table 55 below.

ITEM	
Net weight	< 80 kg
Shipping weight	< 115 kg
Height	650 mm
Width	430 mm
Depth	550 mm

Table 55. EB Propsim F32 dimension and weight

The dimensions and weight of EB Propsim FS8 are listed in Table 56 below.

ITEM	
Net weight	< 30 kg
Shipping weight	< 50 kg
Height	280 mm
Width	430 mm 480 mm (rack mountable front panel)
Depth	550 mm

Table 56. EB Propsim FS8 dimension and weight

3.9 Environmental and safety specifications

3.9.1 Environmental and climatic requirements (operating)

EB Propsim has been designed for indoor use in a Pollution Degree 2 environment. The operating should be limited to stationary use at weather-protected and dust-free locations, such as:

- offices
- laboratories
- telecommunication centers
- storage rooms for valuable and sensitive products

Requirements

- Altitude up to 2000 m.
- Temperature 15 ºC to 30ºC.
- Maximum relative humidity 80 %b.
- Rate of change of temperature <0.5 °C/min
- Mains supply voltage fluctuations up to ±10 % of the nominal voltage.

After the transportation, the temperature of the device must be stabilized before switching the power on.

Cautions:

- During the use free, unrestricted air flow must be allowed around the device to ensure proper cooling. Figure 25 below depicts the air inflow and outflow vents on the sides and back of the emulator.
- Condensation must be prevented.

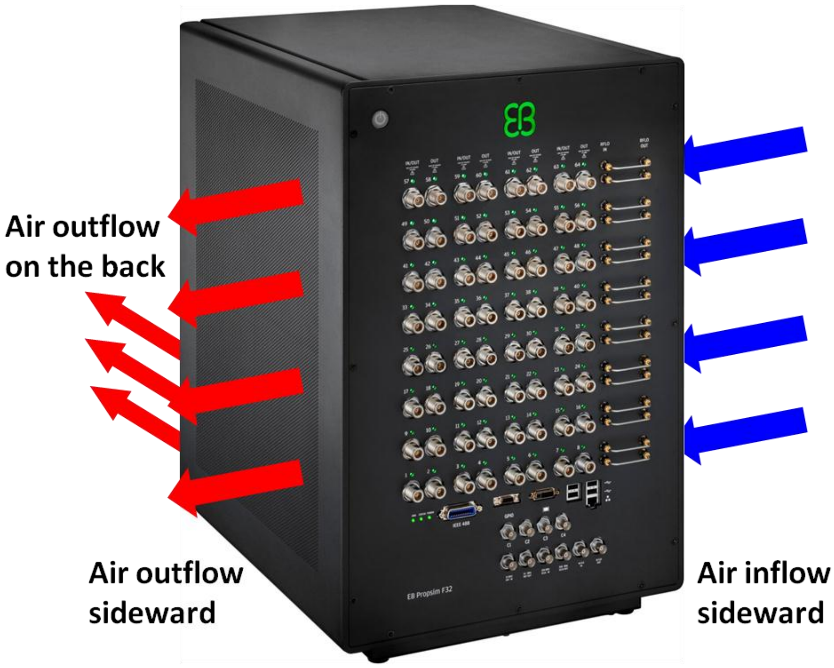


Figure 25. Air inflow and outflow on sides and back of the emulator

3.9.2 Transportation requirements

3.9.2.1 Transportation with package

ETS 300-019-1-2, class 2.2 careful transportation

- Air temperature
  - -25 ... +70°C (unventilated enclosures)
  - -25 ... +30°C, when direct transfer of the product between the two given temperatures and condensation is presumed.
- Relative humidity
  - up to 95% (not combined with rapid temperature changes at +40°C)
- Relative humidity
  - up to 95% (combined with rapid temperature changes at high relative humidity)

3.9.3 Electrical safety regulations

Tested according to IEC 61010-1 regulations.

3.9.4 Electromagnetic compatibility (EMC)

IEC 61326-1:

- Electrical equipment for measurement, control and laboratory use - EMC requirements – Part 1 General Requirements

IEC 61326-2-1:

- Electrical equipment for measurement, control and laboratory use - EMC requirements – Part 2-1.

EMC USA & Canada:

- FCC part 15.107 and 15.109 and IC standard RSS-210. ICES-003

3.9.5 Compatible standards and regulations

- WEEE according to EU regulations in case of product disposal, 2002/96/EC
- ETL Control number 3064827
- CB Certification number SE-56678, TRF 816104-1
- CE tested according to EMC: IEC 61326-1 & IEC 61326-2-1 Safety: IEC 61010-1

4 Graphical User Interface

The Graphical User Interface (GUI) of the EB Propsim opens when the emulator has been started up and the user has logged in, see Figure 26.

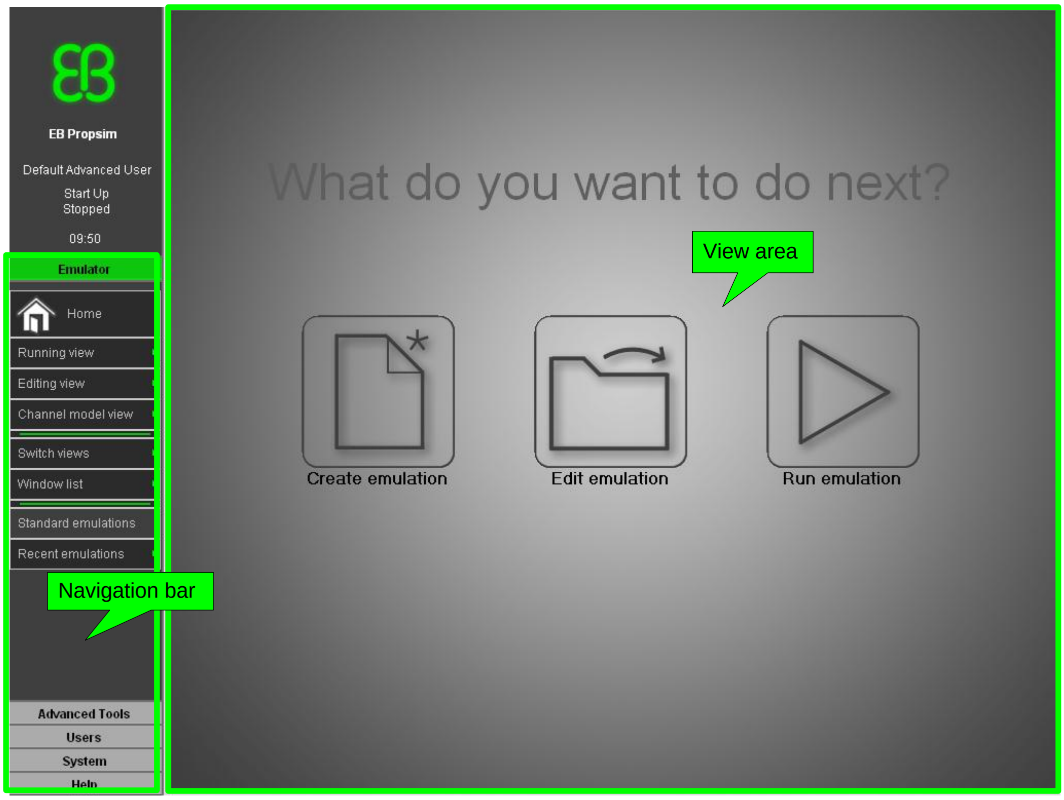


Figure 26. Emulator Home view

The GUI is divided into navigation bar and view area.

- The navigation bar is organized as a column in the left side of the GUI and provides easy access to all views and applications of the EB Propsim.
- The main views and applications of EB Propsim open in the view area. The content of the view area (including the menus and the tools available in the toolbar) thus depend on the selected view or application.

Note that the navigation bar is not always shown in GUI screen shot examples.

4.1 Navigation bar

The navigation bar provides access to the following views and applications:

CATEGORY	ITEM	DESCRIPTION
Emulator	 Home	Scenario wizard and access to pre-installed standard emulations and user emulations, see chapter 4.3
	Running view	Emulation running tool, see chapter 4.4.
	Editing view	Emulation generation tool, see chapter 4.5
	Channel model view	Channel modeling tool, see chapter 4.6
	Switch view	Enables selection of Ate Monitor, Ate Lan and Ate Gpib views
	Window List	Lists all open applications
	Standard Emulations	Access to pre-installed standard emulations
	Recent Emulations	Quick access to recently used emulations and channel models
Advanced Tools		Access to Correlation Editor. Batch Builder and RTC plus optional tools (Aerospace Model Editor, GCM Tool, OTA Tool, 802.11 Tool and WES)
Users	Change User	Opens dialog for changing user profiles
	Create User	Opens dialog for creating new user profiles
	Delete User	Opens dialog for deleting user profiles
System	System Configuration	Includes information about emulator hardware and option configuration
	Control Panel	Configuration of several emulator details, such as locale, time zone and network
	Text Editor	Opens Notepad application suitable for small note keeping
	File Manager	Can be used for browsing, deleting or moving emulations
	Command Prompt	Text-based file- and control utility
	Unplug or Eject Device	Utility to safely unplug removable devices, such as USB mass memories
	Restart/Shutdown	Controlled restart and shutdown of emulator
Help	About	Includes firmware release information
	Quick Guide	Opens user documentation in pdf format
	Release Note	Opens a release note document in txt format
	User Documentation	Opens User Reference document in pdf format
	Application Notes	Access to application notes in pdf format

Table 57. Navigation bar

4.2 Menus and Toolbar

In addition to menu commands, the main controls of each view are available as tools located in the toolbar, as shown in Figure 27. All toolbar icons marked with white color are available, whereas the grey colored icons are disabled. The tools available in Running view, Editing view and Channel model view are described in the corresponding chapters.

Figure 27. Emulator run controls in toolbar.

4.3 Home view

Home view is opened when emulator has been started and user has logged in, or by clicking Emulator -> Home from the navigation bar on the left side of the GUI.

Home view offers an access to pre-installed standard emulations, user defined emulations and Scenario Wizard to create emulations for different test scenarios and technologies. User can select these functionalities from the Home view by using three navigation buttons (see Figure 28).

- New emulation with Scenario Wizard can be created by selecting **Create emulation** button. Scenario Wizard is described in chapter 4.3.1.
- Existing emulation can be edited by selecting **Edit emulation** button. Editing is described in chapter 4.3.2.
- Emulation can be loaded to Running view by selecting **Run emulation** button. Loading emulations from Home view is described in chapter 4.3.3.

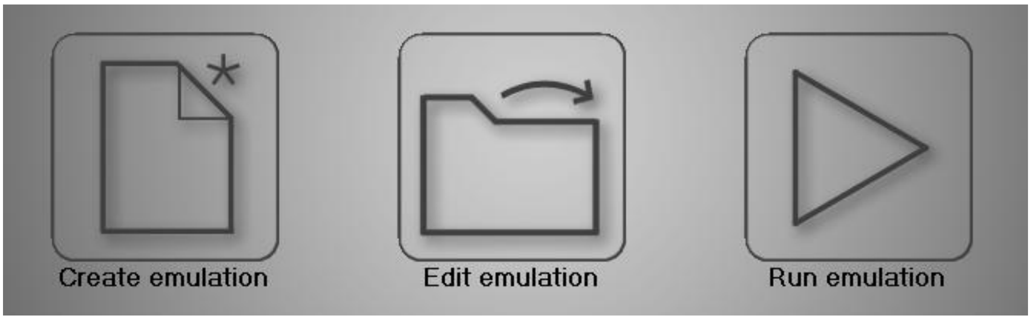


Figure 28. Home view navigation buttons

4.3.1 Create emulation with Scenario Wizard

Selecting Create emulation button from the Home view launches the Scenario Wizard (see Figure 29).

With Scenario Wizard user is able to create new emulations for different test scenarios and technologies. Emulation is created in steps from basic settings to finalizing the test setup. There is own page for each step in creation process.

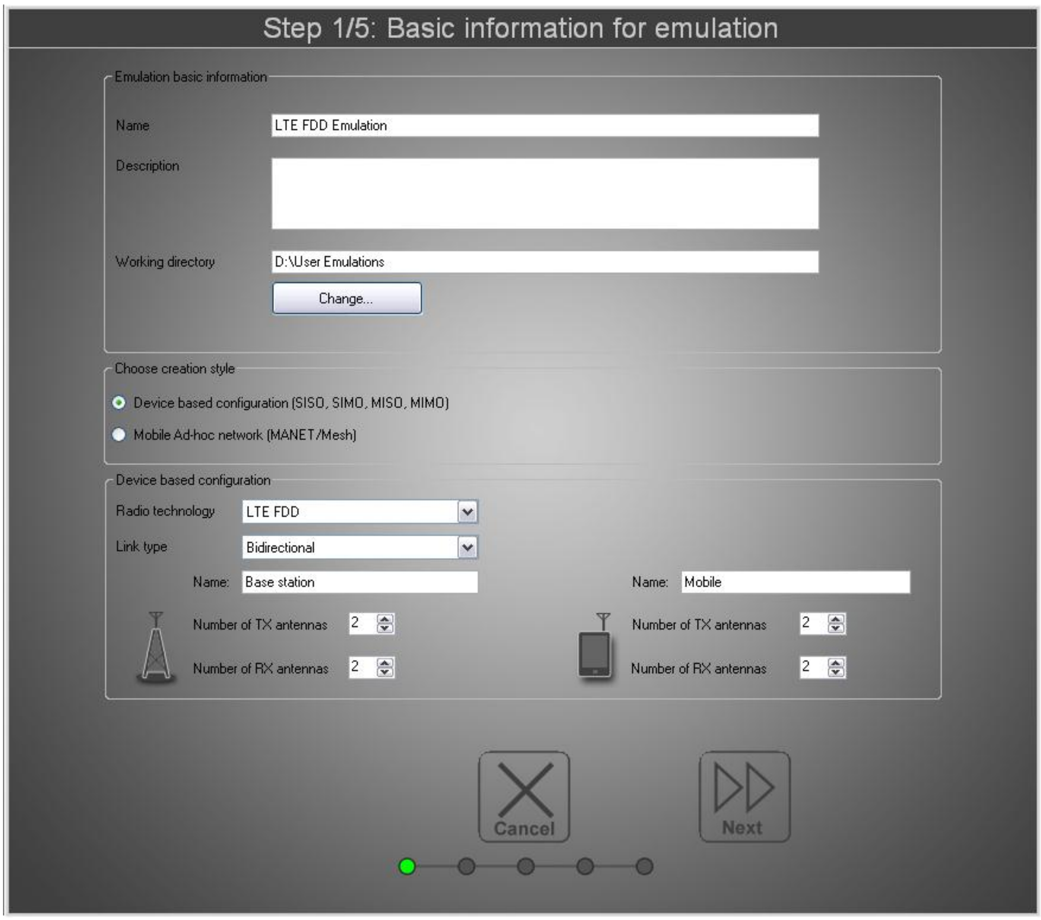


Figure 29. Scenario Wizard - First step of creating new emulation.

4.3.1.1 Scenario Wizard navigation

Scenario wizard contains following navigation buttons (presented in Figure 30) for navigation between wizard steps:

- **Back** for browsing back to previous step.
- **Cancel** for cancelling emulation creation and discarding all changes.
- **Next** for accepting changes and proceeding to next step.

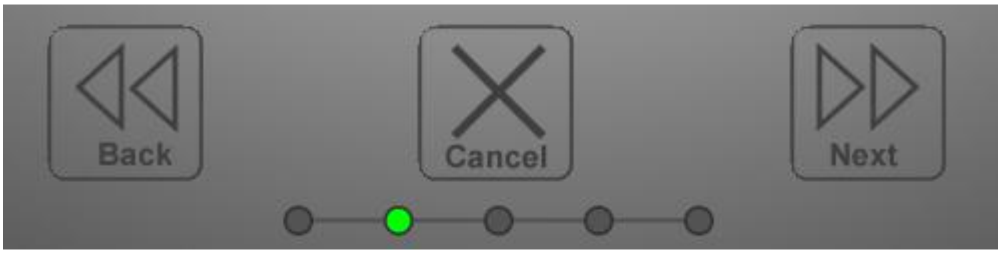


Figure 30. Scenario Wizard navigation

Navigation buttons are located at the bottom of wizard page. Current step is indicated with green color under the navigation buttons.

If an error has occurred in page and proceeding to next step is not possible, Next button is shown in red color. In that case, also invalid field(s) in current step with corresponding error information is marked in red color.

4.3.1.2 Step1: Basic information for emulation

Following basic settings are defined in the first page of the wizard.

Emulation basic information (presented in Figure 31)**:**

- **Name**: Emulation name.
- **Description**: Emulation description
- **Working directory**: Emulation working directory. Scenario wizard creates emulations to user emulations folder. It is possible to browse the default working folder or make new working folders in it by clicking **Change…** button or typing folder name to empty working directory field.

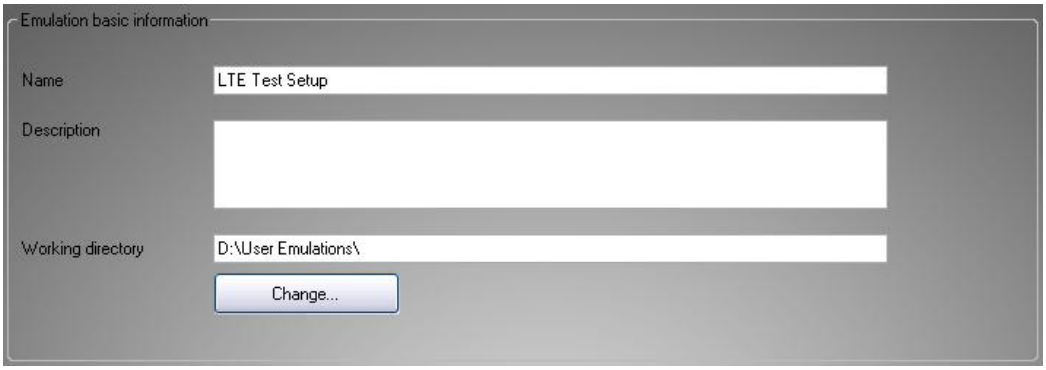


Figure 31. Emulation basic information

Bandwidth:

In EB Propsim F8, emulation bandwidth is selected (see Figure 32).

Figure 32. Bandwidth selection in EB Propsim F8

Available bandwidth options depend on the EB Propsim F8 channel bandwidth configuration and installed MIMO Extension option license. Table 62 in chapter 4.5.2.1 shows the maximum available fading channel count and bandwidth with different MIMO extension options in EB Propsim F8.

Creation style:

Creation style is selected from two options (see Figure 33): **Device based configuration** to enable SISO, SIMO, MISO and MIMO emulation creation or **Mobile Ad-hoc network** to enable MANET/Mesh emulation creation.

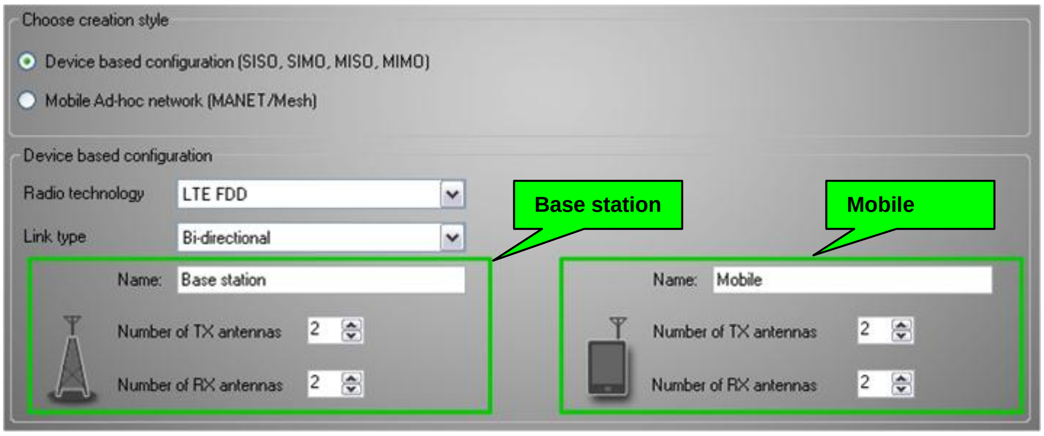


Figure 33. Creation style – Device based configuration

Device based configuration

By default, device based configuration option is selected for new emulation. It enables following options for creating SISO, SIMO, MISO or MIMO emulation:

Radio technology:

- Pre-defines common environment variables for emulation (maximum transmit power, center frequency, crest factor, etc.).

Technology for test scenario is selected from drop-down list.

Note:

Changing technology always resets all emulation environment variables to default values (see chapter 4.3.1.4).

Link type:

- Bi-directional: to create bi-directional channel emulation
- Downlink: to create downlink channel emulation
- Uplink: to create uplink channel emulation

Link type is selected from drop-down list. Selection enables/disables corresponding antenna amount selections for devices (Base station and Mobile).

Note:

Two RF locals are required to run the FDD bi-directional emulation.

In EB Propsim F8, uplink/downlink duplexing must be handled outside the EB Propsim, for example with external duplexers or EB IFU (Interface Unit).

Device (Base station and Mobile):

- Name: Device name (can be edited, default names presented in Figure 33)
- Number of TX antennas: device transmitter antenna amount
- Number of RX antennas: device receiver antenna amount

Maximum number of TX and RX antennas depends on hardware configuration.

Mobile Ad-hoc network (MANET/Mesh)

Figure 34. Mobile Ad-hoc Network options

Selecting Mobile Ad-hoc network creation style enables following options for creating MANET/Mesh emulation (see Figure 33):

Radio technology:
- Defines common environment variables for MANET/Mesh emulation

Number of radios:
- Number of radios in MANET/Mesh network. Maximum number of radios depends on hardware configuration.

4.3.1.3 Step 2: Channel models and shadowing (optional)

Channel models setup page presents links with representative device symbols and antenna amount based on first page selections (see chapter 4.3.1.2.).

Figure 35 presents bi-directional MIMO channel model setup with two base station and mobile TX and RX antennas. Transmitter device with antenna amount is illustrated on the left side of the link and receiver on the right.

Figure 35. Channel model setup page for bi-directional 2x2 MIMO channel emulation

Each link includes following information:

Channel model

Channel models for link can be set in two ways:

- By browsing pre-defined channel model for all channels in the link (**Browse…** button).
  - Channel model setup is described in chapter 4.3.1.3.1.
- By browsing pre-defined channel model for each channel between TX and RX antennas separately (**Advanced…** button). In advanced channel model setup it is also possible to rename TX and Rx antennas.
  - Advanced channel model setup is described in chapter 4.3.1.3.2.

Shadowing profile (optional)

Shadowing for the link is enabled by checking "Use shadowing" option under the channel model selection (see Figure 36).

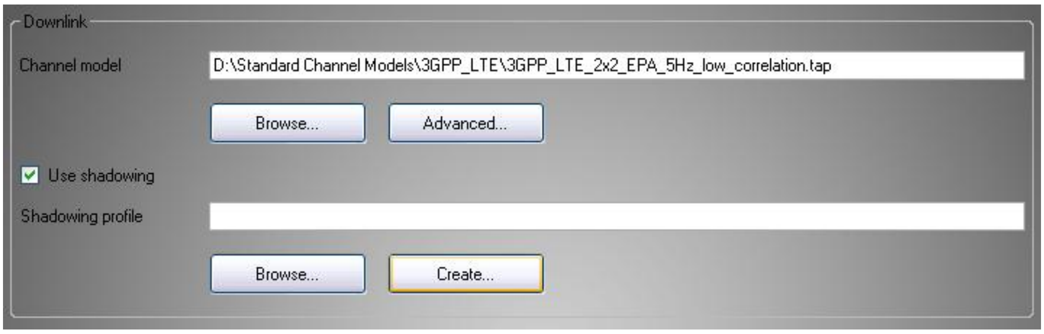


Figure 36. Shadowing enabled

Shadowing profile for link can be added in two ways:

- By browsing pre-defined shadowing profile for the link (**Browse…** button).
  - Shadowing profile setup is described in chapter 4.3.1.3.3.
- By creating new shadowing profile for the link (**Create…** button).
  - Shadowing profile creation is described in chapter 5.1.

4.3.1.3.1 Channel model setup

Clicking **Browse…** button under the channel model field opens a Channel model setup dialog (presented in Figure 37).

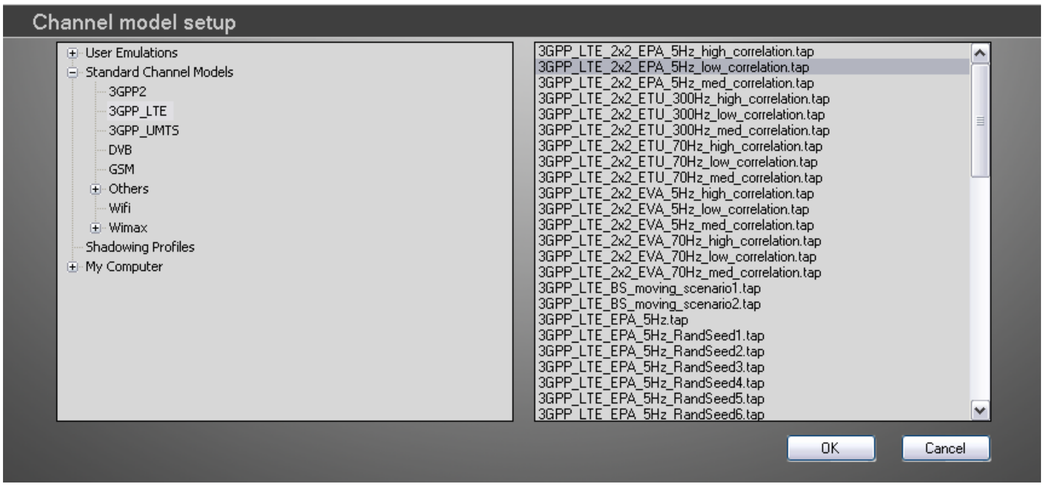


Figure 37. Channel model setup dialog

In channel model setup dialog, folders are listed on the left and channel model files on the right. When a folder is selected, channel model files that fit to antenna amounts of selected link are shown in the file list.

File is selected by double clicking file in file list or clicking OK button. Channel model is set for all channels in the selected link.

Note:
Scenario Wizard supports following channel model files:

- .tap
- .ir
- .rtc
- .asc
- .mat

4.3.1.3.2 Advanced channel model setup

Clicking Advanced… button in channel model setup page opens the Advanced channel model setup dialog (see Figure 38).

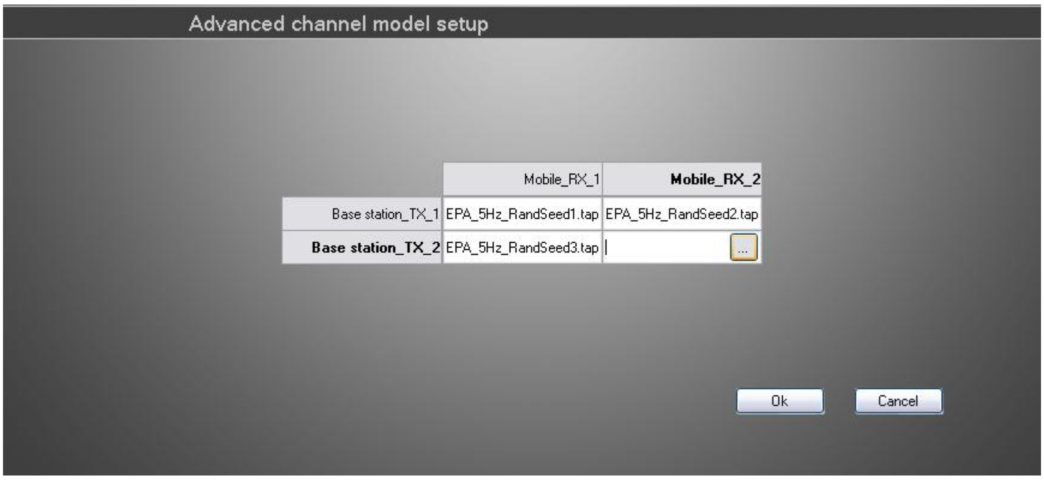


Figure 38. Advanced channel model setup

In advanced channel model setup it is possible to set pre-defined channel model for each channel between Tx and Rx antennas separately and rename Tx and Rx antennas.

If correlated MIMO channel model has been selected, only Tx and Rx antenna names can be changed.

Dialog contains following functions:

Antenna names

- Antennas default names are based on Tx and Rx device names.

Channel model fields

- Own field for each channel
- Selected channel field is indicated with browse button and TX and RX antenna names of channel are marked in bold.

Browse

- Opens channel model setup dialog.

Renaming antennas:

Renaming antennas can be done by clicking wanted antenna field and typing a new name for antenna.

Setting channel models:

Channel model is set by selecting wanted channel field between TX and RX antennas and clicking browse button. It opens channel model setup dialog like presented in Figure 37. Selected channel model file name appears to channel model field.

Channel model path information can be checked from the tooltip (see Figure 40).

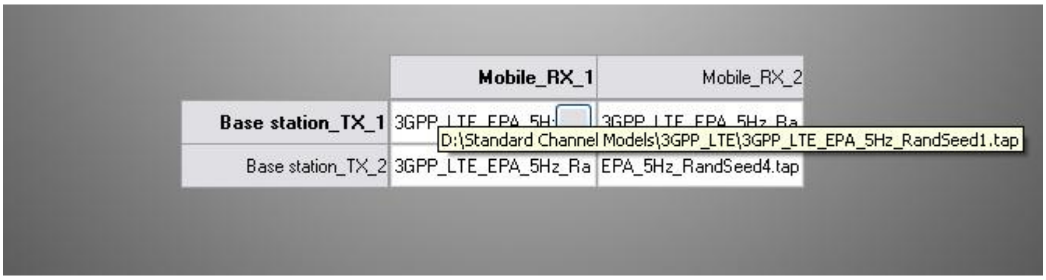


Figure 39. Selected channel model path information in tooltip

Changes are accepted by clicking **OK** button.

4.3.1.3.3 Browse shadowing profile (optional)

Clicking **Browse…** under Shadowing profile field opens a Shadowing profile setup dialog is opened (presented in Figure 40).

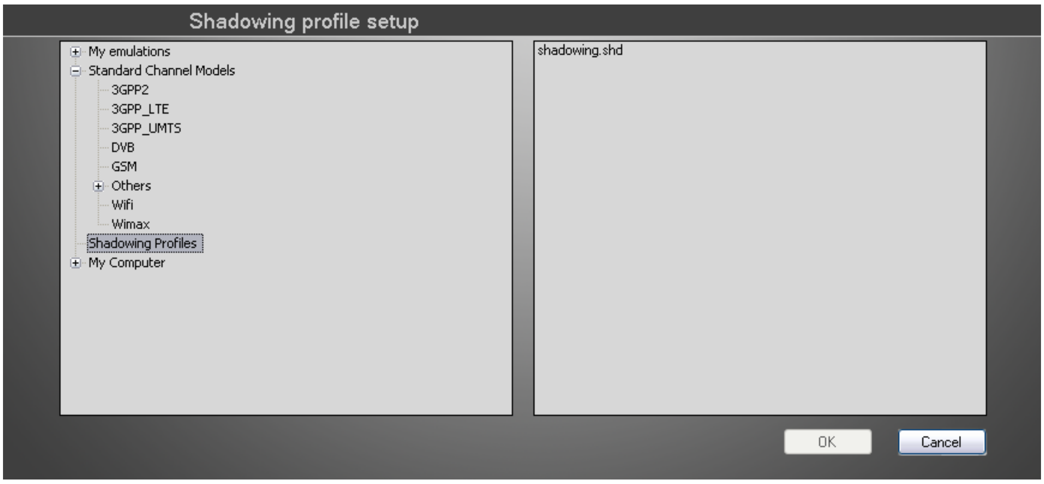


Figure 40. Shadowing profile selection

In shadowing profile setup dialog, folders are listed on the left and shadowing profiles on the right. When a folder is selected, shadowing profiles in folder are listed to the right.

File is selected by double clicking file in file list or clicking OK button. Selected shadowing profile is added to Shadowing profile field (see Figure 41). Shadowing profile can be individually defined for each link.

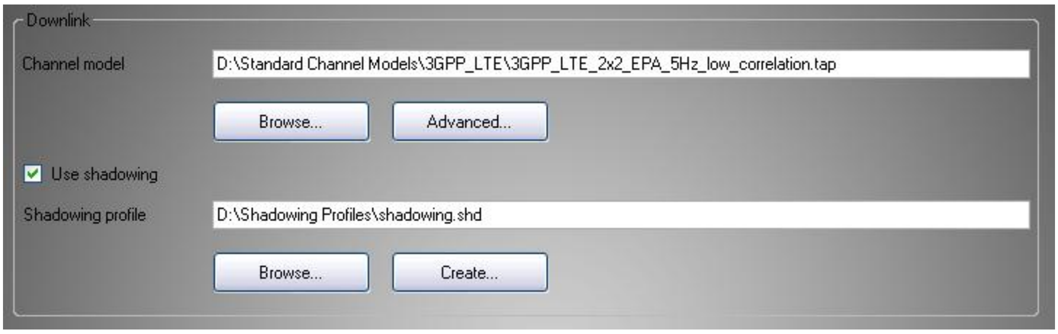
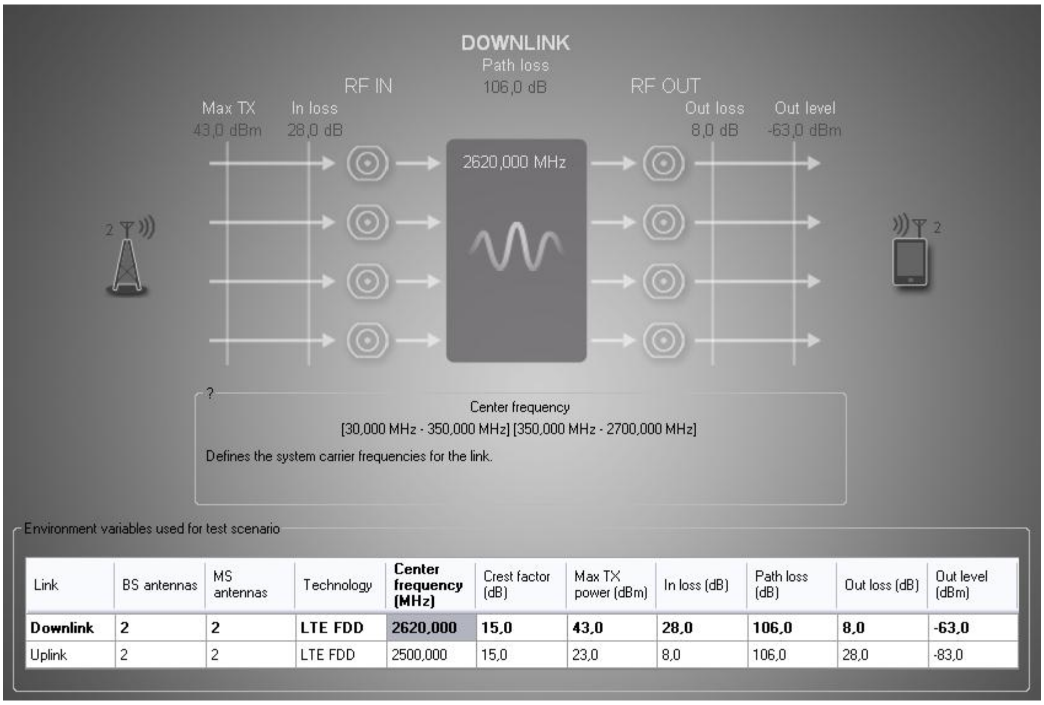


Figure 41. Selected shadowing profile

4.3.1.4 Step 3: Environment variables

Environment variables page presents links with their predefined environment variables. Predefined values are based on the technology selected in first page (see chapter 4.3.1.2). Variables of the selected link are also illustrated in example picture as presented in Figure 42.

Explanation and possible limit values for selected variable are shown above the variable table.



Link	BS antennas	MS antennas	Technology	Center frequency (MHz)	Crest factor (dB)	Max TX power (dBm)	In loss (dB)	Path loss (dB)	Out loss (dB)	Out level (dBm)
Downlink	**2**	**2**	**LTE FDD**	**2620,000**	**15,0**	**43,0**	**28,0**	**106,0**	**8,0**	**-63,0**
Uplink	2	2	LTE FDD	2500,000	15,0	23,0	8,0	106,0	28,0	-83,0

Figure 42. Emulation environment variables for bi-directional FDD 2x2 MIMO channel emulation.

Environment variables used for emulation:

Link
- Defines the name or identification of the link.

BS Antennas (read only)
- Displays the number of base station antennas defined on first wizard page.

MS Antennas (read only)
- Displays the number of mobile station antennas defined on first wizard page.

Technology (read only)
- Defines the name of the selected technology.

Center Frequency
- Defines the system carrier frequencies for the link.

Crest Factor
- Defines the crest factor (peak/average transmit power) of used modulation.

Max TX Power
- Defines the maximum transmit power of the base station or mobile without cables or external losses.

In loss
- Defines the loss of cables and external components connected between device (BS or MS) and EB Propsim.

Path loss
- Defines the total loss between devices.

Out loss
- Defines the loss of cables and external components connected between EB Propsim and device (BS or MS).

Out level

- Defines the signal level seen by the device (BS or MS), after fading, cable losses and external components.
- This level is achieved when transmitter sends with its maximum TX power.

For MANET/Mesh emulations, environment variables are configured automatically. If needed, settings can be modified later in Running view, see chapter 4.4.5.

4.3.1.5 Step 4: Active connectors selection

Scenario Wizard automatically allocates physical RF connectors that are used when running the emulation. Allocation is based on selections in previous steps. In this page it is possible to verify these default connector settings, or modify connector locations to match test setup cable connections.

Figure 43 illustrates the layout of the active connector selection page:

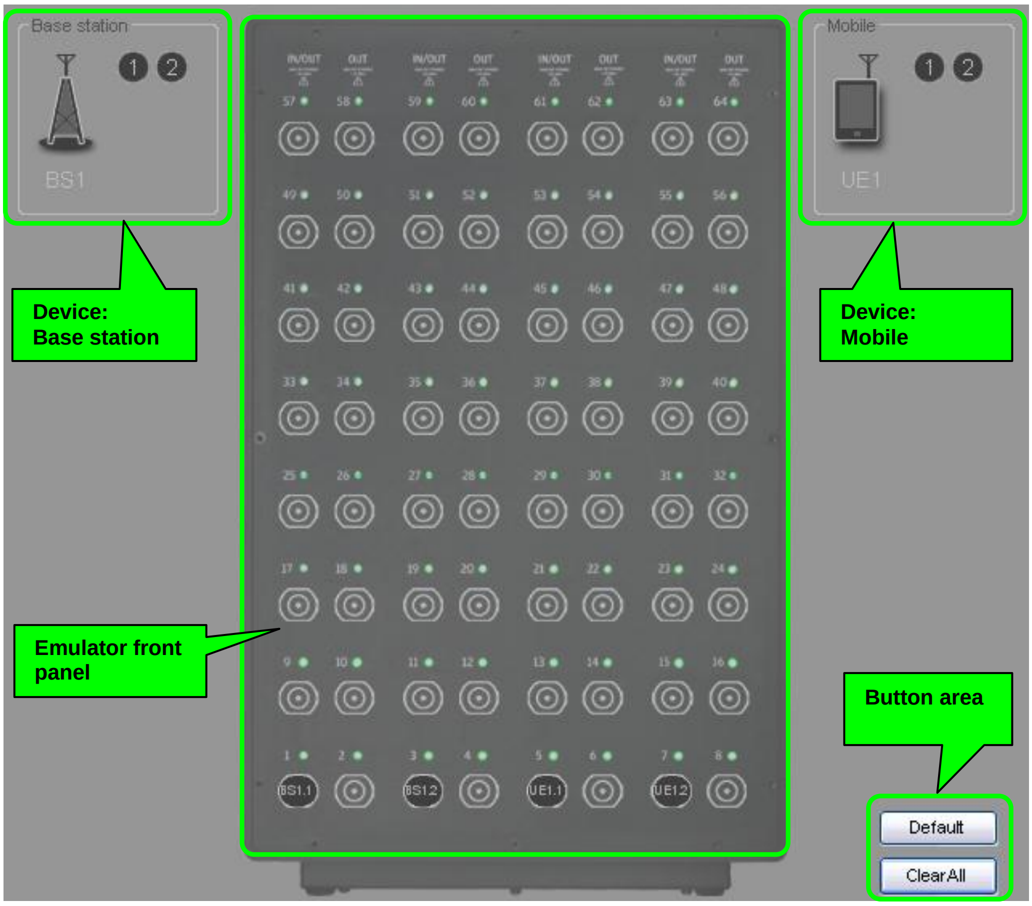


Figure 43. Active connector selection page layout - F32

Emulator front panel

- Presents available RF connectors and allocated RF connectors in current hardware configuration
  - Available RF connectors are shown in grey color: 
  - Allocated RF connectors are shown with identification information: 

Identification information is based on device identification (BS1 and UE1 in Figure 43) and connected device antenna number.

Devices

- Illustrated with representative symbols and antenna amounts on the both sides of the emulator front panel
- If Mobile ad-hoc network (MANET/Mesh) option selected on the first page of wizard (see chapter 4.3.1.2), MANET radios are shown on the left of the front panel

Button area

- Button(s) for restoring/reseting all allocated connectors.
  - **Default** button restores the default connector setup
  - **Clear All** clears all active connectors (button is used in EB Propsim F32 and FS8, see chapter 4.3.1.5.1)

4.3.1.5.1 Active connectors selection – EB Propsim F32 and FS8

In this page it is possible to verify default connector settings, or modify connector locations to match test setup cable connections. Connectors can be relocated one by one to suitable RF connectors.

Figure 44 presents the EB Propsim F32 default active connectors for bi-directional 2x2 MIMO channel emulation, as located in the emulator front panel. In bi-directional channel emulations, EB Propsim F32 and FS8 use RF IN/OUT connectors. For example in Figure 44, Base station TX 1 and RX 1 antennas (BS1.1) are allocated to RF1 (IN/OUT).

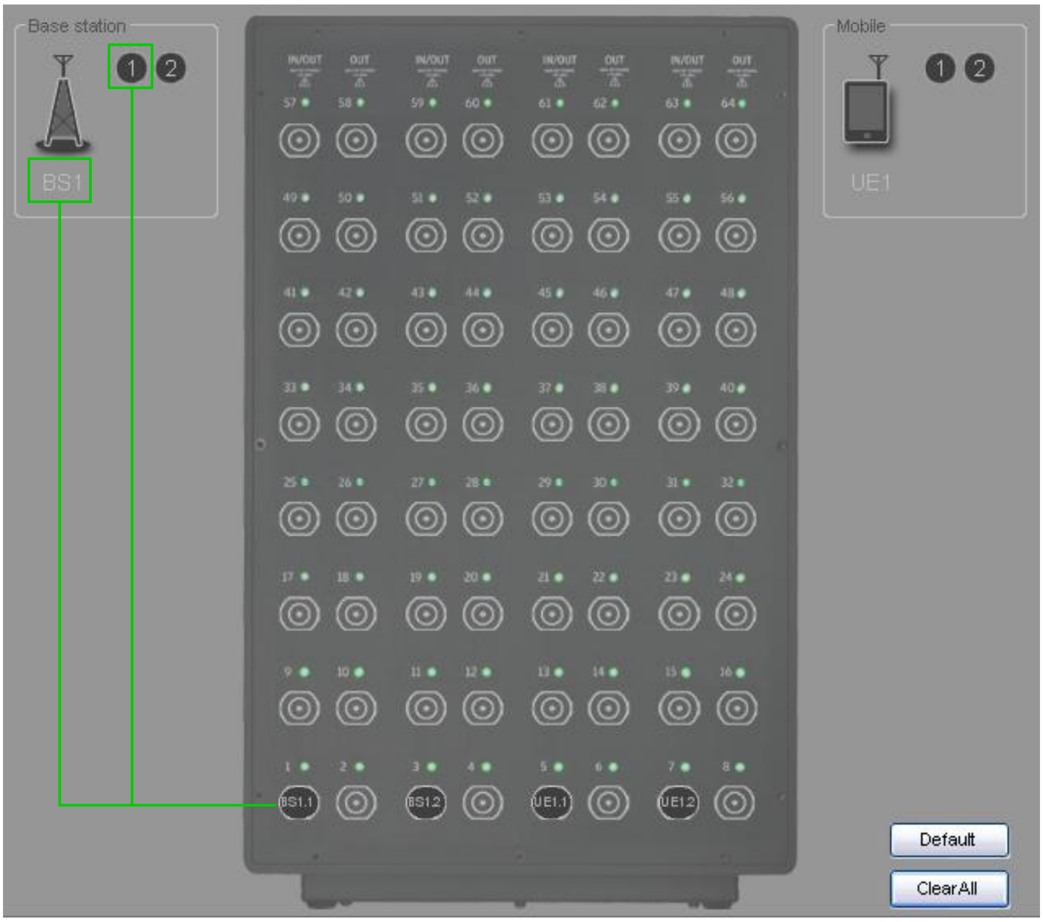
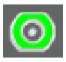


Figure 44. Active connectors selection - F32

Relocating connectors:

In EB Propsim F32 and FS8 it is possible to change the default connector setup. It can be done in two ways:

- All active connectors are cleared by clicking **Clear All** button and then all antennas are dragged one by one to wanted RF connector or
- Antenna is dragged from the front panel to the wanted RF connector and dropped there. When dragging antenna to new connector, suitable connector is shown as green in the front panel: 

Default button restores the default connector setup.

4.3.1.5.2 Active connectors selection – EB Propsim F8

In this page it is possible to verify default connector settings, or modify link connector group locations to match test setup cable connections.

Figure 45 presents presents the EB Propsim F8 default active connectors for bi-directional 2x2 MIMO channel emulation, as located in the emulator front panel.

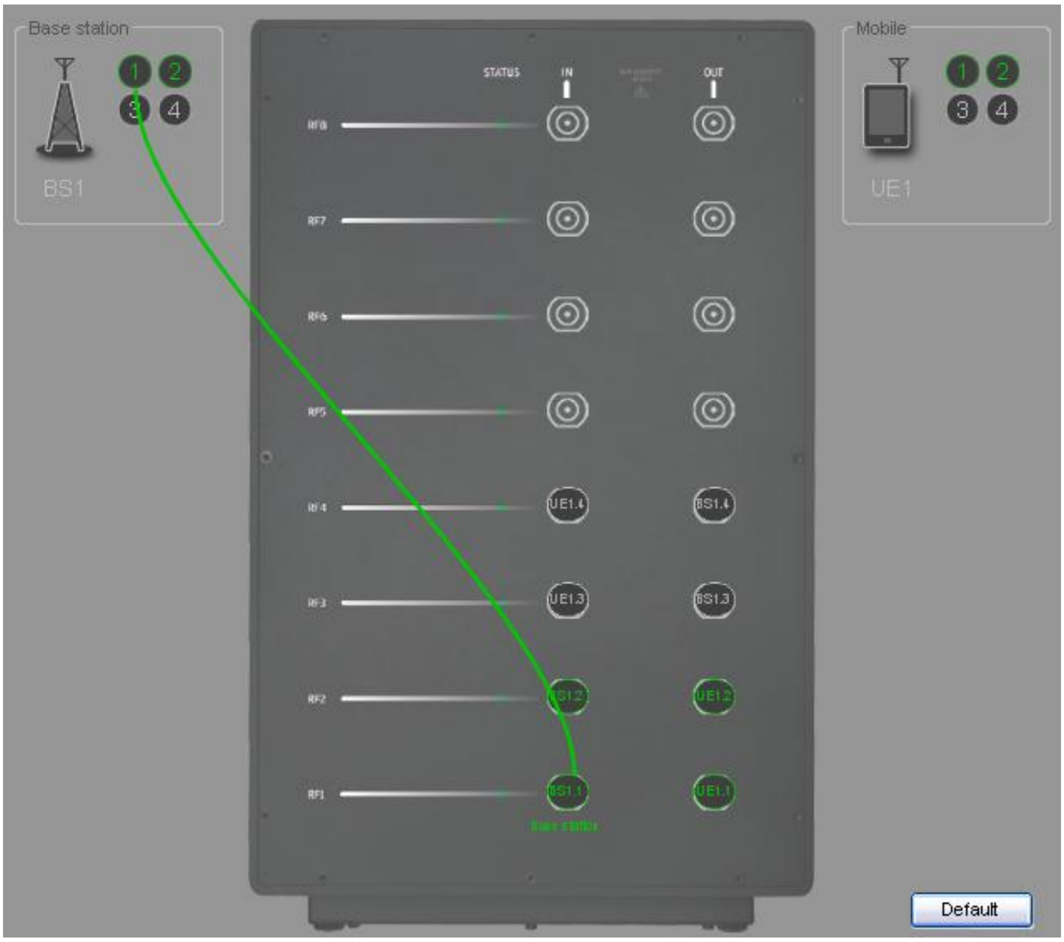


Figure 45. Active connectors selection – F8

When RF connector or device antenna is clicked, cable connection between selected RF connector and matching TX/RX antenna is shown and all antennas and connectors that belong to same link are highlighted in green color (as illustrated in Figure 45).

Relocating link connectors:

In EB Propsim F8, connectors can be relocated by moving all connectors of the link, starting from the uppermost connector group.

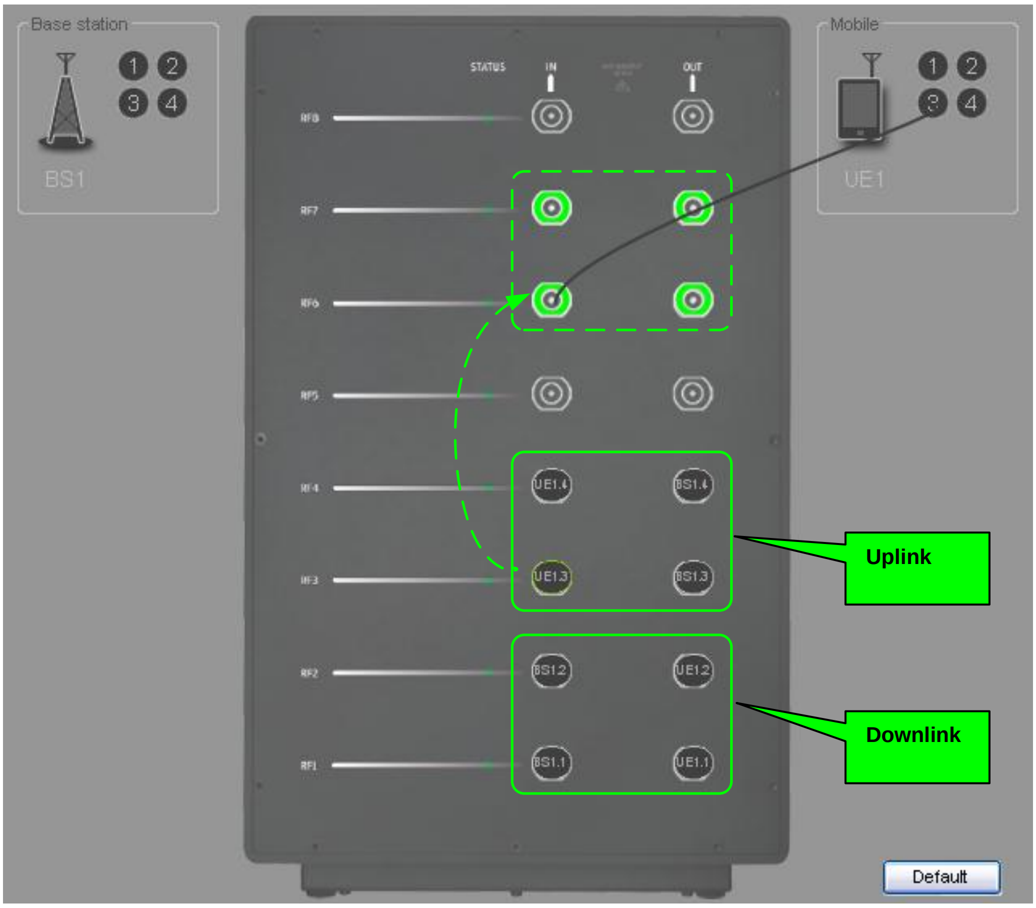


Figure 46. Relocating link connectors in EB Propsim F8

Figure 46 illustrates the relocating of uplink connectors to upper RF connectors in emulator front panel. When one of the link's connectors has been selected, group can be dragged and dropped to new location in the front panel. While dragging connector group to new location, suitable locations are shown as green connector groups in the front panel.

After moving uplink connectors, it is possible to move downlink connectors.

Default button restores the default connector setup.

4.3.1.6 Step 5: Summary view

Last page of the wizard presents a brief summary of selected emulation configuration (see Figure 47).

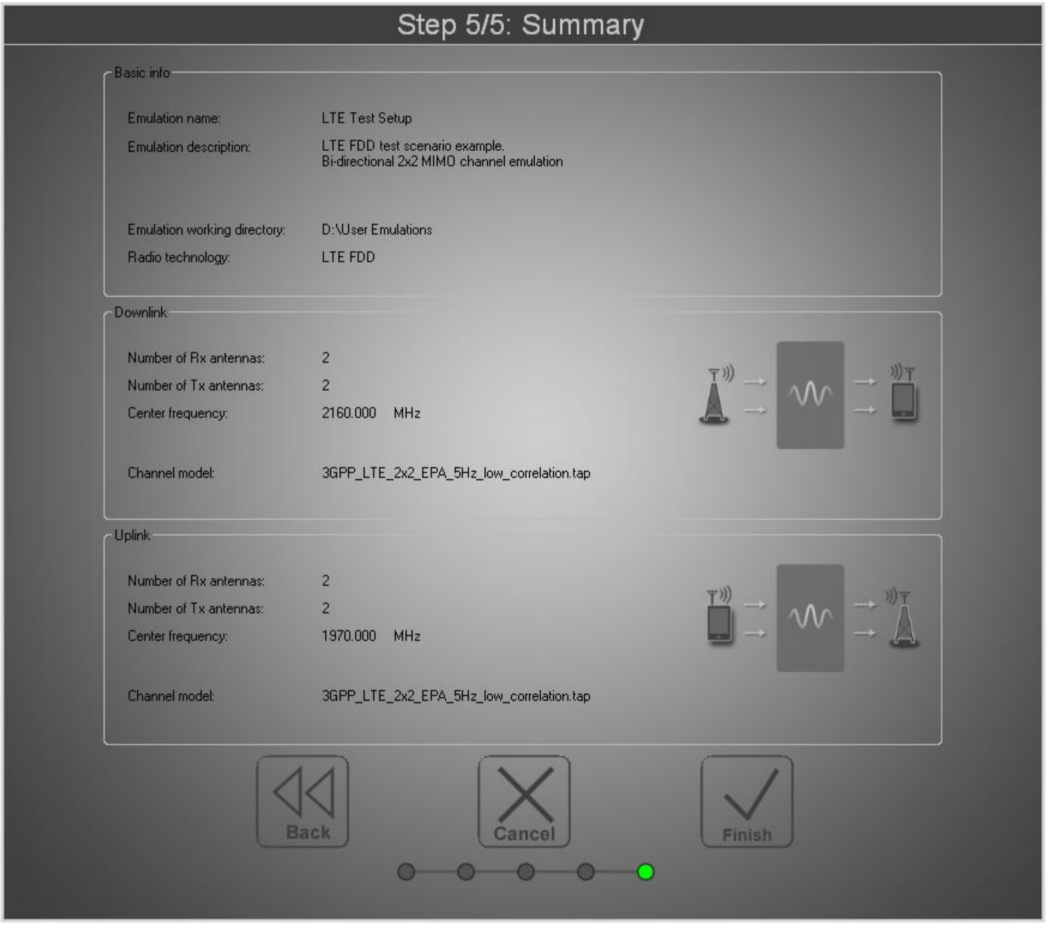


Figure 47. Emulation summary

In summary page, **Finish** button replaces Next button. When **Finish** button is clicked, following options are opened:

- **Finish and Build:** Save emulation and build it.
- **Finish and Run:** Save emulation, build it and open to the Running view.
- **Finish:** Save emulation.
- **Cancel:** Cancel and return to summary page.

Emulation is saved to individual folder in emulation working directory, containing all related files. Folder name consists of emulation name and .wiz extension in format <emulation name>.wiz (e.g. MyEmulation.wiz).

Emulation building is described in chapter 4.3.1.6.1

4.3.1.6.1 Emulation building

Emulation build is started when **Finish and Build** or **Finish and Run** option has been selected from summary page. Build progress window (presented in Figure 48) appears to the bottom of the window.

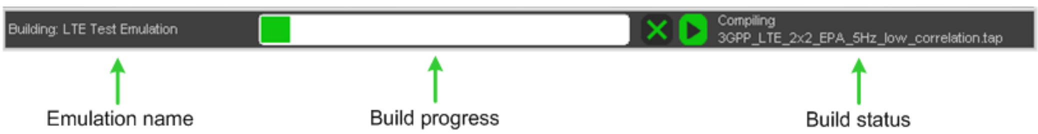


Figure 48. Build progress window

Build progress window has following information and functionality:

Emulation name
Name of the emulation

Build progress
Indicates the status of the current build in progress bar

Build status
Current status description of the build
More specific build messages can be viewed by clicking build status area

 **Cancel**
Cancel the ongoing build

 **Run**
Open emulation to the Running view after build has finished

4.3.2 Edit emulation

Selecting Edit emulation button (shown in green color in Figure 49) from the Home view opens a dialog where user can browse and select and emulation file to be opened for editing.

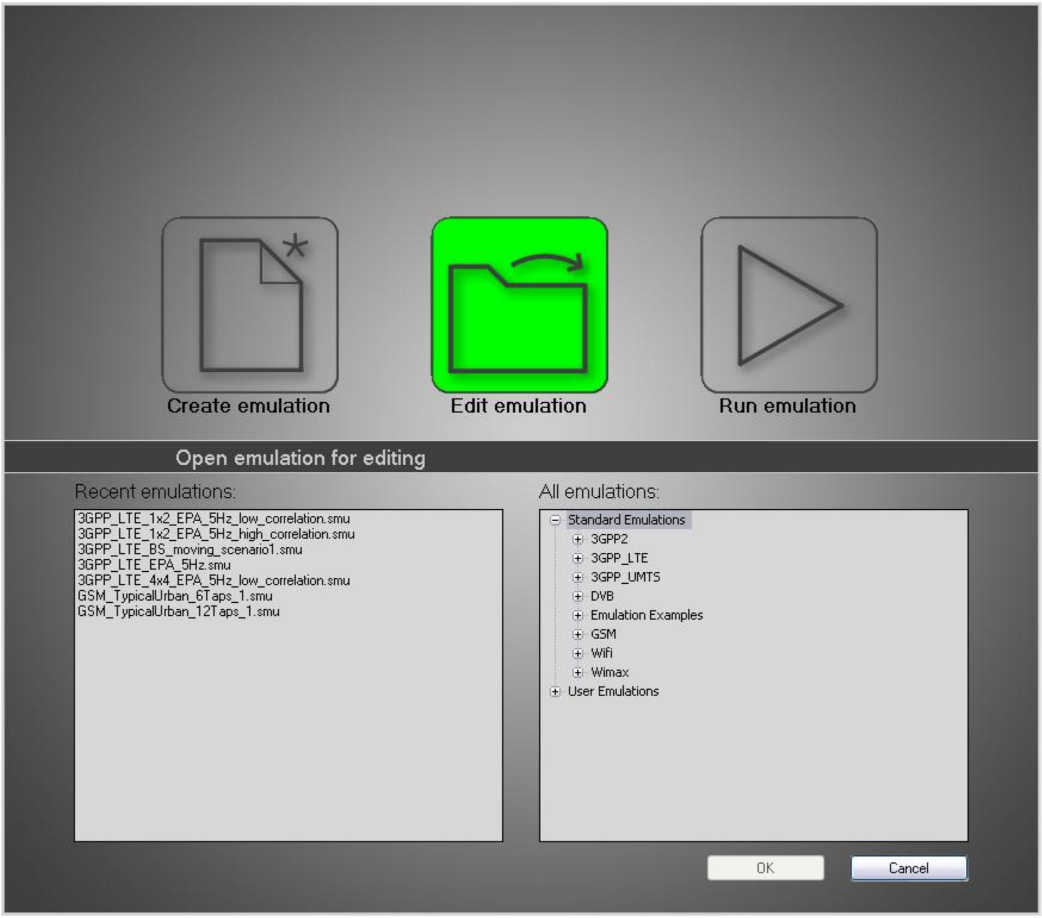


Figure 49. Open emulation for editing.

Dialog includes following lists for browsing the emulation file:

- **Recent emulations**: List of recently used emulation files of current user.
- **All emulations**: Pre-installed standard emulations (Standard Emulations) and user defined emulations (User Emulations) in hierarchical list.

File selection is confirmed by clicking OK button. Dialog is closed and selection is cancelled by clicking Cancel button.

Emulation is opened to Scenario Wizard or to Editing view, based on emulation creator. Emulation creator information is shown in tooltip in browse dialog.

If emulation has been created with Scenario Wizard, it is possible to edit emulation description, technology, channel models and connector configuration by following steps that were described in chapter 4.3.1. Changing emulation name creates a new copy of selected emulation.

For instructions to edit the emulation in Editing view, please see chapter 4.5.

4.3.3 Run emulation

Selecting Run emulation button (shown in green color in Figure 50) from the Home view opens a dialog where user can browse and select an emulation file to be opened to Running view.

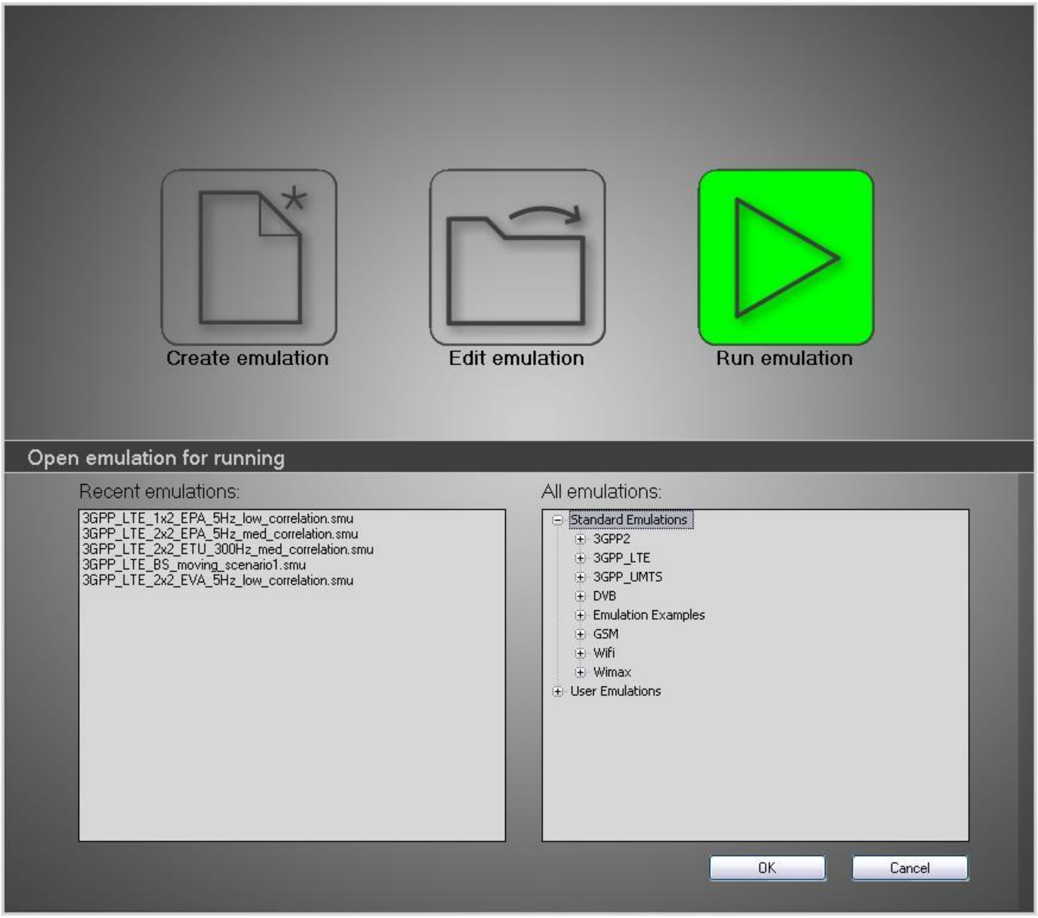


Figure 50. Open emulation for running.

Dialog includes following lists for browsing the emulation file:

- **Recent emulations**: List of recently used emulation files of current user.
- **All emulations**: Pre-installed standard emulations (Standard Emulations) and user defined emulations (User Emulations) in hierarchical list.

File selection is confirmed by clicking OK button. Dialog is closed and selection is cancelled by clicking Cancel button.

For instructions to run the selected emulation in Running view, please see chapter 4.4.

4.4 Running view

Running view is opened by clicking Emulator category button of the navigation bar in the left side of the GUI and then selecting Running view. The Running view is used to load and run standard emulations or emulations that have been created with channel modeling tools.

In Running view user can view settings of the emulation and the hardware connectors, change parameters of the emulation, control how the emulation is run and see the change of the Channel Input Responses (CIRs) during the emulation. User can also save the changes to the emulation for future use.

Pre-generated emulation file (.SMU) is first loaded to the Running view. Modified emulation parameters can be saved to the emulation file if desired. The actual emulation data for each channel is stored in the emulation hardware control files (.SIM) not visible to the user.

User can select between three different views within the Running view. The selection can be done from Window-menu or using the toolbar buttons.

- The emulation settings can be modified in Emulation settings view (Figure 51). The view is described in chapter 4.4.5.
- The change of the CIRs during the emulation can be seen in the **CIR Graphics** view. The CIR Graphics controls are described in chapter 4.4.7.
- Active Connectors view illustrates graphically which connectors the emulation uses for current emulation.

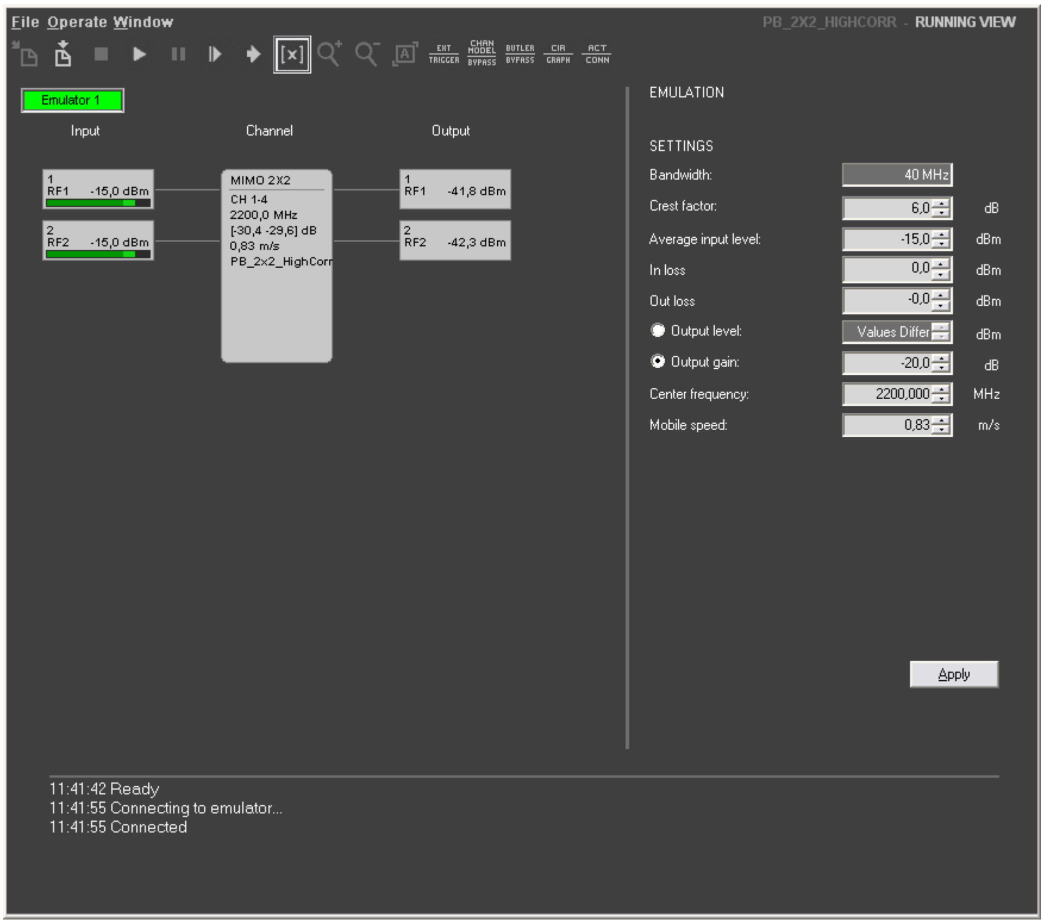


Figure 51. Running view with emulation settings

4.4.1 Running view toolbar

The tools available in Running view toolbar are described below.

SYMBOL	TOOL	DESCRIPTION

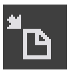
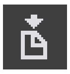
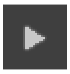
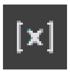
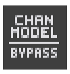
SYMBOL	TOOL	DESCRIPTION
	Open	Opens a dialog where user can browse and select an emulation file to be opened.
	Save	Saves edited emulation with the same name or prompts the user to define a name if the emulation has not been saved previously.
	Run	Starts the emulation and fading process. The emulation runs continuously even though individual models are not designed as continuous.
	Stop	Stops the emulation. The emulation is set to initial state. Clicking Run after stopping the emulation starts the emulation from the beginning.
	Pause	Pauses the emulation. The emulation is stopped to the impulse response, which was active when Pause was pressed. Pressing Pause again will continue running the emulation from the impulse response where it was paused.
	Step	Steps to the next impulse response of the emulation. Step will run the emulation until there is an impulse response change in any channel. Can be used only when the emulation is stopped or paused. Step will wrap back to the first impulse response in the end.
	GoTo	Runs emulation to a given position. For details, see chapter 4.4.2
	Show or hide status window	Shows or hides the status area at the bottom of the view.
	Zoom in	Can be used to scale the CIR graphics in CIR Graphics View.
	Zoom out	Can be used to scale the CIR graphics in CIR Graphics View.
	Autoscale	Selects such a scaling in CIR Graphics View that all taps in all channels are visible.
	Ext trigger	Enables or disables external trigger. • When disabled, the emulation is run normally. • When enabled, the emulator waits for external trigger signal to run, pause or stop emulation. External trigger can be disabled at any time and emulation runs normally thereafter. For detailed description of the operation, see chapter 4.4.5.2.1.
	Channel model bypass	Generates a constant radio channel model in which delay is according to delay of the shortest path in channel model and gain is average gain of channel model. Enables quick verification of test setup performance and signal levels. For details, see chapter 4.4.3.1.
	Butler bypass	Generates a similar constant radio channel model as “Channel model bypass” with Butler phase shifts applied to each channel. Enables quick verification signal levels and MIMO throughput performance. For details, see chapter 4.4.3.2.
	CIR Graph	Switches between CIR Graphics and Emulation Settings.
	Active Connectors	Shows or hides Active Connectors dialog.

Table 58. Running view toolbar

4.4.2 GoTo

Emulation is run to a given position by clicking the GoTo button. Figure 52 shows the Go To dialog.

Figure 52. Advancing emulation by specifying time from the beginning of emulation

Two options exist for defining the position where the emulation will be advanced to:

- Define the time in seconds from the beginning of the emulation. After clicking the **OK** button the emulation is advanced to the defined position.

- Define the impulse response number where the emulation will be run.
  - The limits on the IR number depend on the current channel model. If there are several channels in the emulation user may select the channel number for the defined impulse response. After clicking the **OK** button the selected channel is advanced to the defined impulse response. The other channels will be advanced simultaneously to the corresponding position.
  - If the given impulse response number is greater than the last IR number of the channel model, the channel is run to the last impulse response and stopped there.

NOTE: Depending on the model and update rate the GoTo operation can take considerable time.

4.4.3 Channel bypass functionality

Different channel bypass modes replace all the fading channels in the emulation with static channels allowing quick verification of test setup, signal levels and performance. Details of static models depend on the selected bypass mode.

4.4.3.1 Channel Model Bypass

Channel Model Bypass replaces the all fading channels in the emulation with static 1-path model with attenuation at the average level of the corresponding fading channel model. Delay of the static channels is defined by the shortest path delay of the current position in the original channel model, as shown in Figure 53.

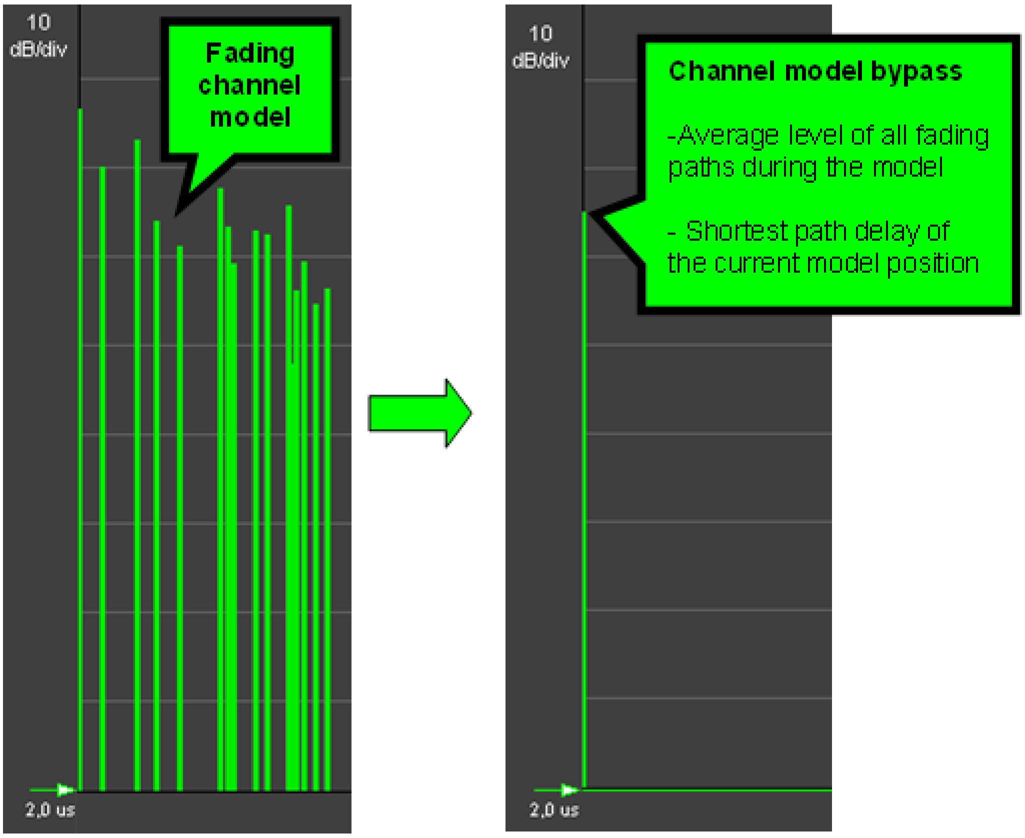


Figure 53. Channel model bypass behaviour principle

Note: Channel Model Bypass is not indicated in Cir Graphics View as shown as a principle in Figure 53.

4.4.3.2 Butler Bypass

Butler Bypass replaces the fading channel models with static channels similar way as Channel Model Bypass but it also modifies the phase components of the individual channels according to Butler matrix. This emulates the beaming effect, where receiver will see all of the MIMO input streams arriving from different angles. This will create enough diversity for the receiver to separate different input streams and MIMO link can be established.

Butler phases are calculated with following formula (Radiowave Propagation and Antennas for Personal Communications, Second edition, Kazimierz Siwiak, 1998):

$$phase(input, output) = angle\left(\exp\left[j\left[\left[input - \frac{N+1}{2}\right]\left[output - \frac{N+1}{2}\right]\frac{2\pi}{N}\right]\right]\right) -$$

$$angle\left(\exp\left[j\left[\left[input - \frac{N+1}{2}\right]\left[1 - \frac{N+1}{2}\right]\frac{2\pi}{N}\right]\right]\right)$$

where,

phase	is phase shift of the channel between input and output in radians
input	is input index in MIMO channel group (1…N)
output	is output index in MIMO channel group (1…N)
N	is maximum MIMO dimension of inputs and outputs

A beaming with Butler matrix is based on assumption that antenna elements are on horizontal plane with a distance λ/2 between elements.

For 4x4 MIMO the equation gives following phase matrix in degrees:

$$\begin{bmatrix} 0 & -135 & 90 & -45 \\ 0 & -45 & -90 & -135 \\ 0 & 45 & 90 & 135 \\ 0 & 135 & -90 & 45 \end{bmatrix}$$

Matrix rows represent the MIMO inputs and matrix columns MIMO outputs. Figure 54 shows the MIMO topology with Butler angles.

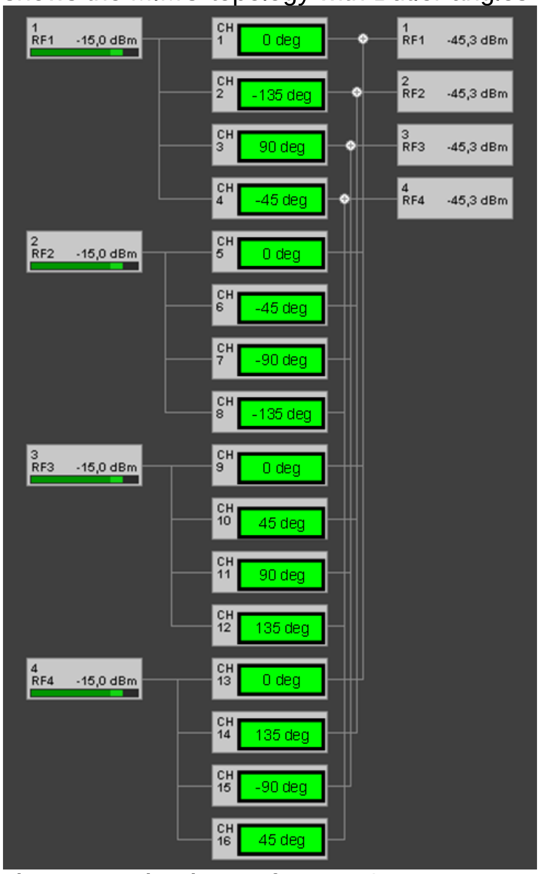


Figure 54. Butler phases of 4x4 MIMO

4.4.4 Running view menus

The Running view menus contain the following items:

MAINMENU	SUBMENU	SHORTCUT	DESCRIPTION
File	Open	Ctrl-O	Opens emulation
	Close		Closes emulation
	Save	Ctrl-S	Saves emulation
	Save As		Saves emulation with new name
	Settings		Running view settings
	Recent file list		Four recently used emulation files

MAINMENU	SUBMENU	SHORTCUT	DESCRIPTION
Operate	Run		Starts emulation
	Pause		Pauses emulation
	Stop		Stops emulation
	Step		Steps emulation to next impulse response
	Goto		Runs emulations to given position
	Ext. Trigger		Enables / disables external trigger
Window	Show Emulation settings		Switches Emulation settings view to the work area
	Show CIR Graphics		Switches CIR Graphics to the work area
	Show Active Connectors		Opens active connectors dialog
	Show Status Window		Enables / disables status area to be shown

Table 59. Running view menus

4.4.4.1 Settings

Selecting File → Settings menu item opens a dialog for defining Running view settings. Note that the settings are kept in shut down of the device.

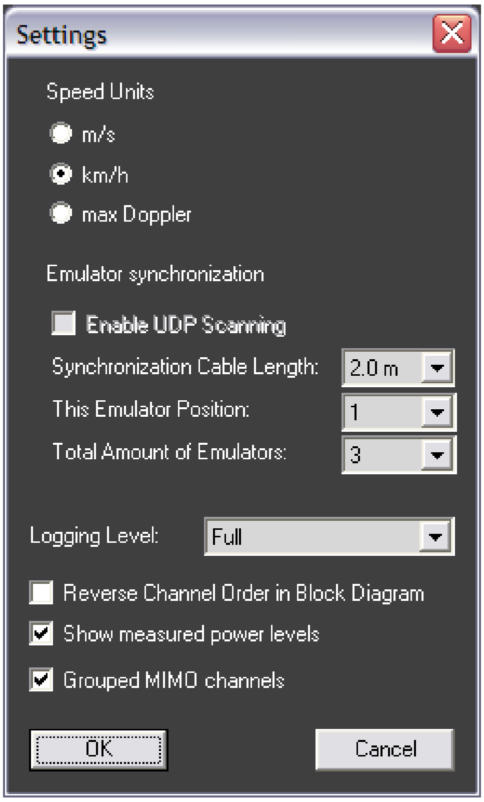


Figure 55. Running view settings dialog

Speed Units

- Meters per second, kilometers per hour or maximum Doppler can be selected as the unit to be used for speed definitions.

Emulator synchronization

Synchronization Cable Length

- Synchronization cable length specifies the length of cables (2.0 m, 4.0 m or 6.0 m) used for synchronization when multiple emulators are used. Note that the cables need to be of the equal length.

This Emulator Position

- Defines the position of the emulator in multi-emulator configuration.

Total Amount of Emulators

- Defines the total number of emulators in multi-emulator configuration.

Other settings

Logging Level

- The logging level setting is for troubleshooting purposes only and does not directly affect the emulator functionality. The default setting (minimal) should not be changed unless explicitly instructed by EB. Logging level setting can be changed also from system configuration, see chapter 4.11.

Reverse Channel Order in Block Diagram

- If Reverse Channel Order is selected, the emulation block diagram is shown so that Channel 1 is shown as lowest channel, matching the emulator's physical channel order.

Show measured power levels

- If selected, the emulation block diagram will show measured power levels for inputs.

Grouped MIMO channels

- If Grouped MIMO channels is selected, the channel objects of emulation block diagrams are replaced with channel group boxes if loaded emulation type is either MIMO or MANET/Mesh. The setting is on by default. In Figure 60 grouped MIMO channels selection is disabled and in Figure 57 it is enabled.

4.4.4.2 Show Active Connectors

This command opens Active Connectors window. It shows graphically which connectors the emulation uses for current emulation, see Figure 56.

To see all connectors, scroll the screen with scrollbar on the right. Note that modifying the emulation might change the signal inputs and/or outputs used. It is thus recommendable to verify the connections always after creating or modifying emulation. Connectors can be relocated in Scenario wizard and Editing view.

The following items are shown in active connector view:

- Active RF connectors (green connectors), with small number indicating which input or output signal the connector is for. Also input and output names given by the user are shown.
- RFLO jumper cables if Internal Local Oscillator is used, or needed RFLO signal frequency for RFLO In connector.

- External RFLO signals and required RFLO frequency, if needed (in case no Internal Local Oscillator option installed). (EB Propsim F8 only)
- Reference clock in/out, SCLK, and active sync in/out connectors.

Figure 56. Active connector display with 2x2 MIMO model

4.4.5 Emulation Settings view

The Emulation settings of Running view contains emulation block diagram and parameter settings area, see Figure 57.

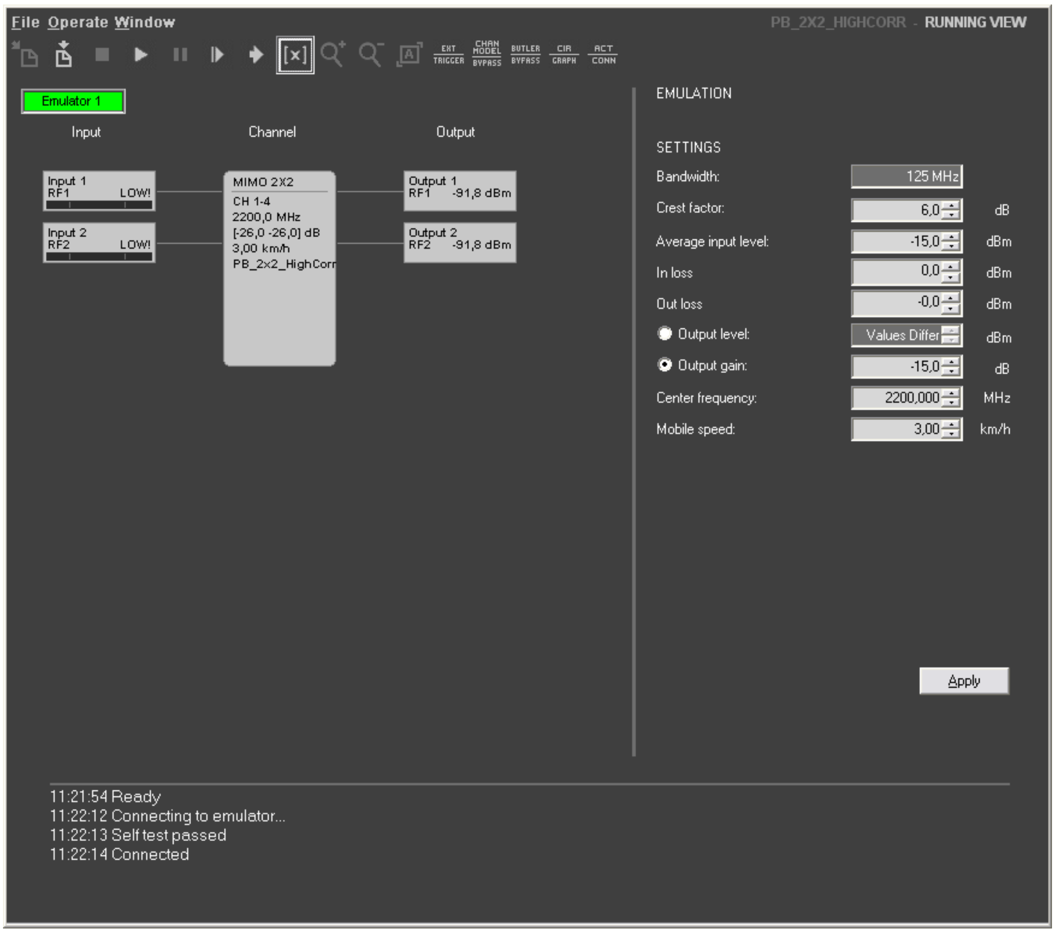


Figure 57. Running view - Emulation settings

The block diagram illustrates the current emulation setup. Note that for MIMO configurations, the look of the block diagram varies depending on the Grouped MIMO channels selection in File → Settings (see Chapter 4.4.4.1). In Figure 57 the Grouped MIMO channels selection is enabled (default) i.e. the channels are illustrated as single group.

The block diagram can have the following elements:

- Input(s)
  - User defined name (or logical input number)
  - RF connector
  - Measured input power level and visual indication of the power level compared to defined level
- Channel(s) (with SISO configurations and MIMO configurations when Grouped MIMO channels setting is disabled)
  - Channel / channel group number
  - Center frequency
- Channel group(s) (for MIMO configurations when Grouped MIMO channels setting is enabled)
  - Channel numbers
  - Center frequency
  - Channel model gain (min, max)
  - Mobile speed
  - Name of the channel model
- Output(s)
  - User defined name (or logical output number)
  - RF connector
  - Output power level

- Interference source
  - Interference settings based on selected interference source type

The Settings pages have a hierarchical structure. Different settings pages can be accessed by clicking repeatedly any of the input, channel, channel group or output blocks.

- **Emulation settings.** This page is for top level settings. Parameter changes in this view affect all lower level views.
- **Emulator settings**
- **Group settings.** Channel group consists of channels that have common inputs or outputs.
- **Channel settings**
- **Output settings**
- **Input settings**

Most of the edit fields in the channel group and emulation settings pages show values that may apply to several channels, inputs or outputs. If the value is not the same in all channels, inputs or outputs, the text "Values Differ" is shown in the edit field, see Figure 58. The user may force all values to be the same by entering a numeric value to the edit field and clicking Apply.

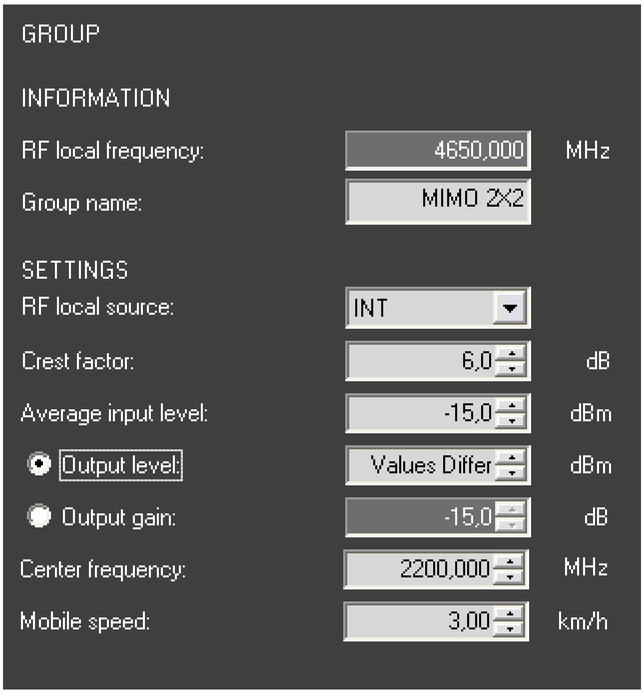


Figure 58. Differing output levels

Most of the numeric edit fields have a tool tip that shows the range of accepted values, see Figure 59. If a too large or small value is entered, values are automatically set to nearest acceptable limit.

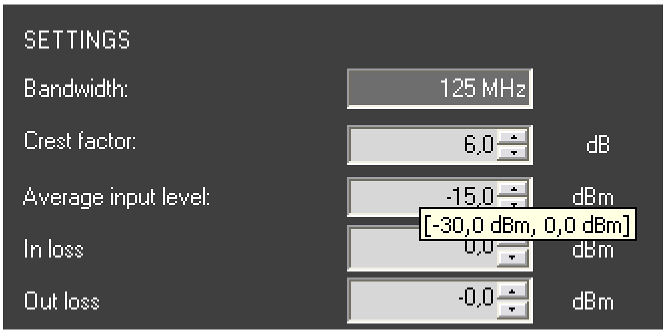


Figure 59. Tooltip showing limits of the edit field

4.4.5.1 Emulation settings

The emulation settings can be accessed by repeatedly clicking in the block diagram objects or once in the background of block diagram (Figure 60).

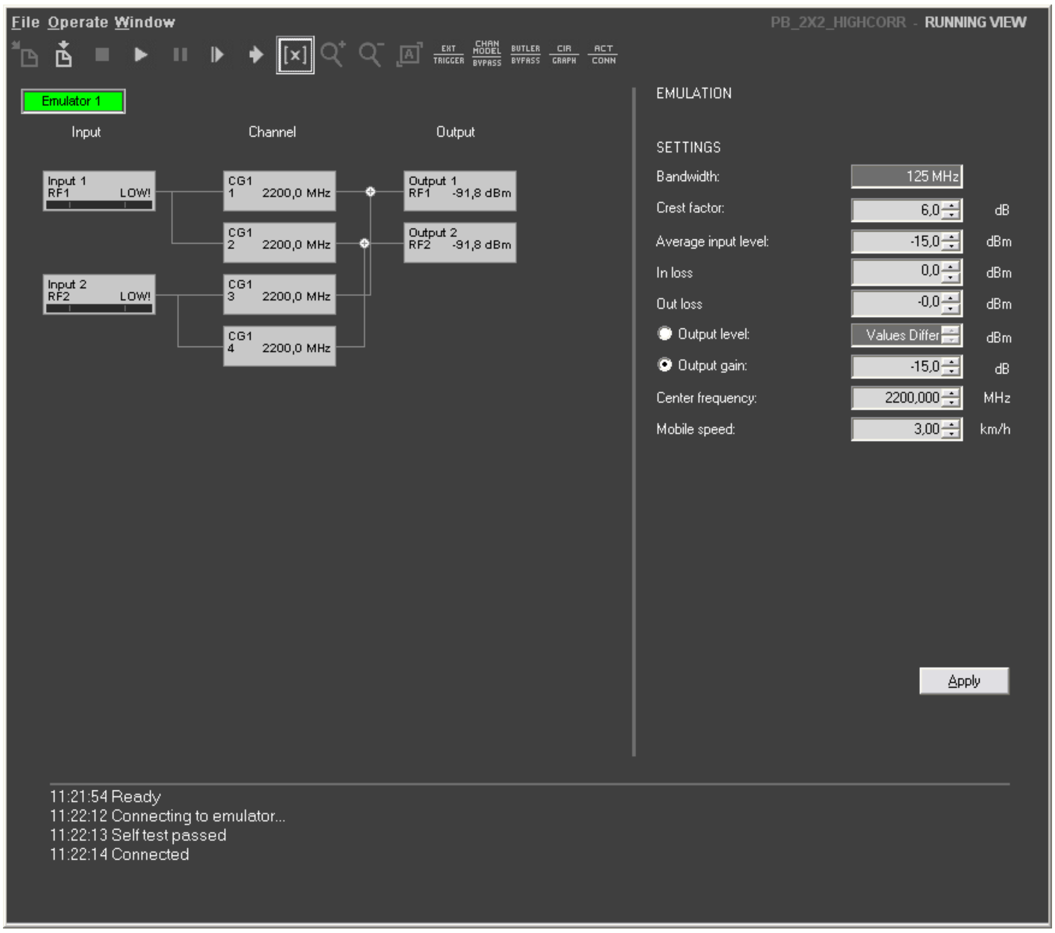


Figure 60. Emulation settings view

Instead of a numeric value an edit field may show the text “Values Differ” which indicates that all corresponding values in the emulation are not the same. By entering a value in this field all the corresponding values in the whole emulation are changed to be the same.

Settings

Bandwidth
- Current emulation bandwidth

Crest factor
- Crest factor of channel inputs in the emulation.

Average input level
- Average input signal level of channel inputs in the emulation.

In loss
- Test setup loss between the transmitting device and EB Propsim input.

Out loss

- Test setup loss between EB Propsim output and the receiving device.

Output level

- Output level of all channel outputs in the emulation.

Output gain

- Output gain of all channel outputs in the emulation.

Center frequency

- Carrier wave center frequency of all channels in the emulation.

Mobile speed

- Mobile speed of all channels in the emulation.

4.4.5.2 Emulator settings

The Emulator settings (see Figure 61) can be accessed by clicking repeatedly in the block diagram.

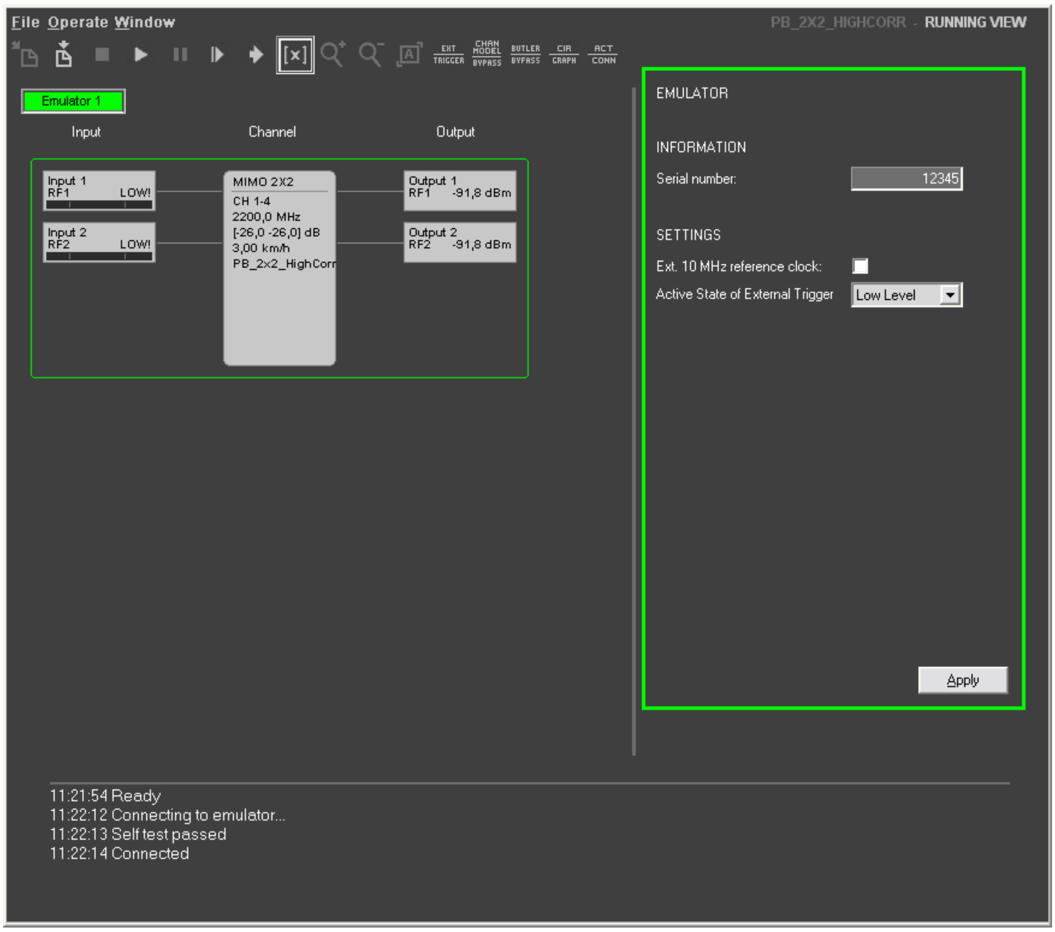


Figure 61. Emulator settings

Information

Serial number

- Emulator serial number.

Settings

Ext. 10 MHz reference clock

- Sets the emulator to use external 10MHz reference clock instead of the internal signal. The external signal is also passed to Ref Out connector of the emulator. Note that this setting is system wide and not bound to current emulation.
- When disabling external 10 MHz reference clock, please note that warming up of internal reference can take up to 10 minutes.

Active State of External Trigger

- Configuration for external triggering when triggering is enabled.
- Four options exist: Low Level, High Level, Rising Edge and Falling Edge. Trigger configuration selection is confirmed by clicking Apply.
- Note that the setting can be changed only when the emulation is stopped.

4.4.5.2.1 External Trigger

External trigger can be used for starting and stopping the emulation by using external connection. Trigger signal is connected to Sync In connector of the EB Propsim.

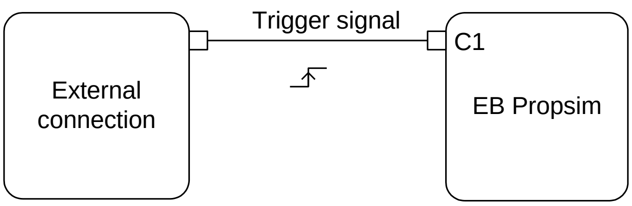


Figure 62. Example trigger connection

The current running state of the emulation can be acquired from the Sync Out connector of the emulator. When emulation is running, Sync Out is low, and when stopped, Sync Out is high.

Emulation start, stop and pause can be trigged in order to define the time of the actions precisely. Either level or edge active triggering can be used. Commands are however still needed.

Note: If triggering is disabled while the emulator is waiting for the external trigger, the emulation continues similarly as if trigger had occurred.

Level Active triggering

Starting emulation

- When Run is selected from Operate menu or from toolbar of **Running view**, emulator starts to wait for trigger signal. Text “Waiting for external trigger” is shown in status view.
- When changes to active level is detected on the signal connected to Sync In, the emulation starts to run and text “Emulating” is shown in the status view.

Running emulation

- Emulation stops running, when trigger signal goes to non-active state. Text “Waiting for external trigger” is shown in the status view. Emulation continues running and “Emulating” text is shown again when trigger signal goes to active state.
- Note: If triggering is disabled while the emulator is waiting for external trigger the emulation continues and “Emulating” text is shown.

Pausing emulation

- When **Pause** is selected from Operate menu or from toolbar of **Running view**, emulation pauses immediately. Clicking pause again re-starts emulation.

Stopping emulation

- When **Stop** is selected from Operate menu or from toolbar of **Running view**, emulation stops immediately.

Figure 63 shows an example of low level triggering.

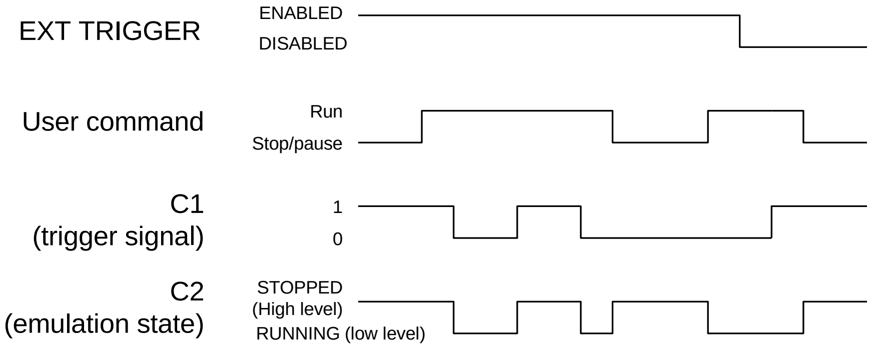


Figure 63. Low level triggering example

Edge Active triggering

Starting emulation

- When Run is selected from Operate menu or from toolbar of **Running view**, emulator starts to wait for trigger signal. Text "Waiting for external trigger" is shown in status view.
- When active edge (rising or falling, depending on configuration) is detected on the signal connected to Sync In, the emulation starts to run and text "Emulating" is shown in the status view.

Running emulation

- Emulation runs normally.

Pausing emulation

- When Pause is selected from Operate menu or from toolbar of **Running view**, emulation pauses when signal connected to Sync In interface has active edge. Clicking pause again starts the emulation on next active edge in input sync signal.

Stopping emulation

- When Stop is selected from Operate menu or from toolbar of **Running view**, emulation stops when signal connected to Sync In interface has active edge.

Figure 64 shows an example of falling edge triggering.

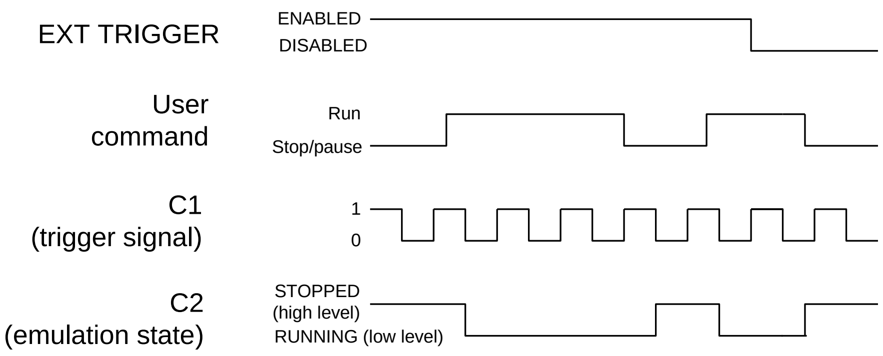


Figure 64. Falling edge triggering example

4.4.5.3 Channel group settings

The Group settings can be selected by simply by clicking the channel group box, see Figure 65.

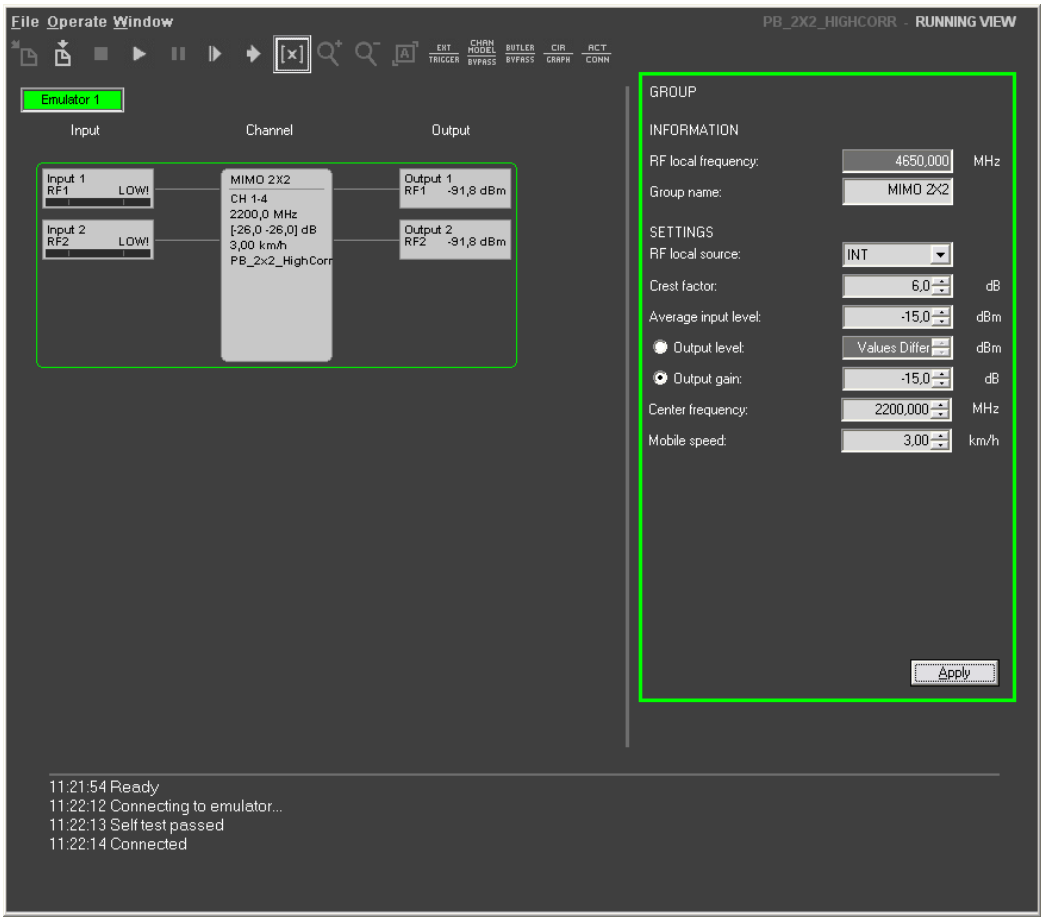


Figure 65. Channel group settings with Grouped MIMO channels

Information

RF local frequency

- RF local frequency [MHz] defines the signal frequency needed at the RF local frequency connector (RFLO IN) of the RF interfaces belonging to the channel group.

Group name

- Custom name for current channel group: the name is also shown in the top of the group block in the diagram.

Settings

RF Local Source

- Used RF local source.

INT	Internal Local Oscillator. This selection is available only if Internal Local Oscillator is installed in the emulator.
DIRECT (EB Propsim F8 only)	External Local Oscillator. The correct connector for the external Local Oscillator connection can be checked in the Active Connector View, see chapter 4.4.4.2.

Crest factor

- Crest factor of channel inputs in the channel group.

Average input level

- Average input signal level of channel inputs in the channel group.

Output level

- Output level of all channel outputs in the channel group.

Output gain

- Output gain of all channel outputs in the channel group.

Center frequency

- Carrier wave center frequency of the channel group.

Mobile speed

- Mobile speed of all channels in the channel group.

Note:
Figure 65 shows Channel group settings when Grouped MIMO channels selection is enabled in File → Settings (see Chapter 4.4.4.1). When setting is disabled, individual channels are shown in the block diagram. The channel group settings can be accessed by repeatedly clicking any of the input, channel or output blocks belonging to the group and browsing through the various settings pages, see Figure 66.

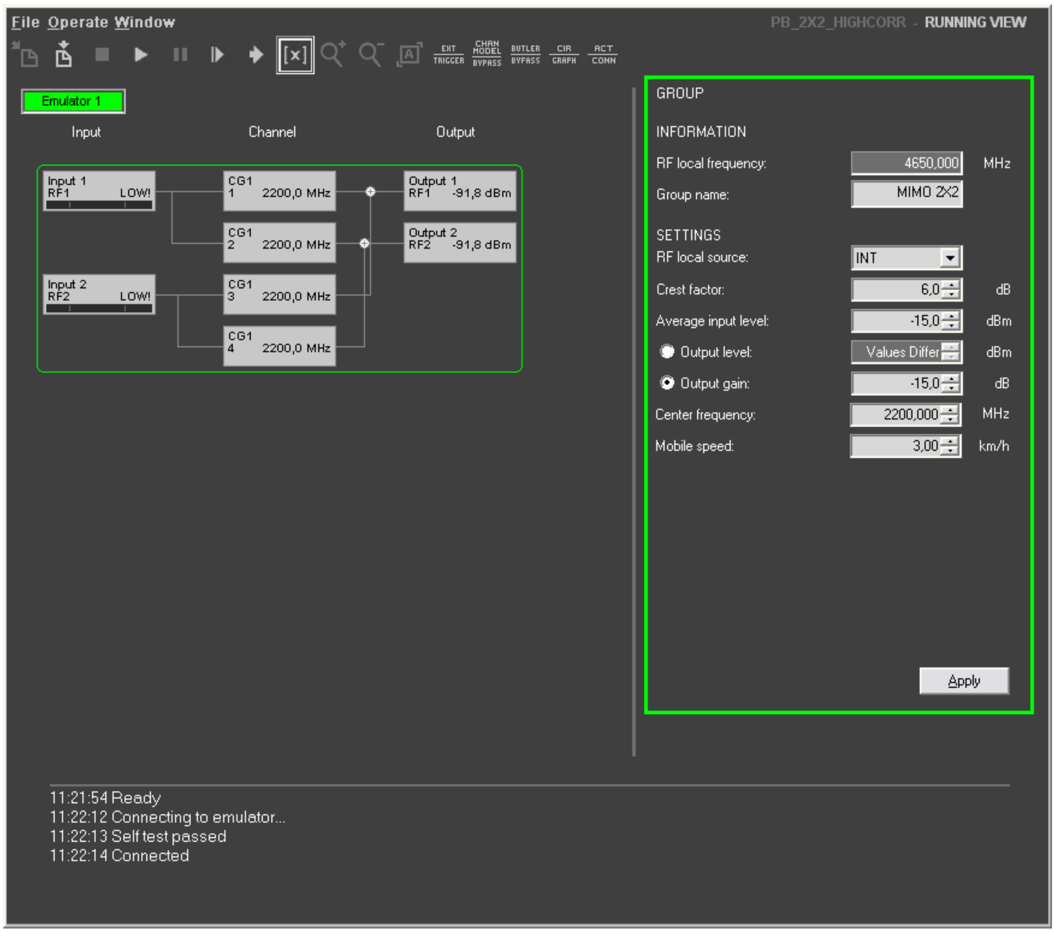


Figure 66. Channel group settings with Grouped MIMO channels disabled

4.4.5.4 Channel settings

The Channel settings can be selected by clicking the channel block in the block diagram, see Figure 67.

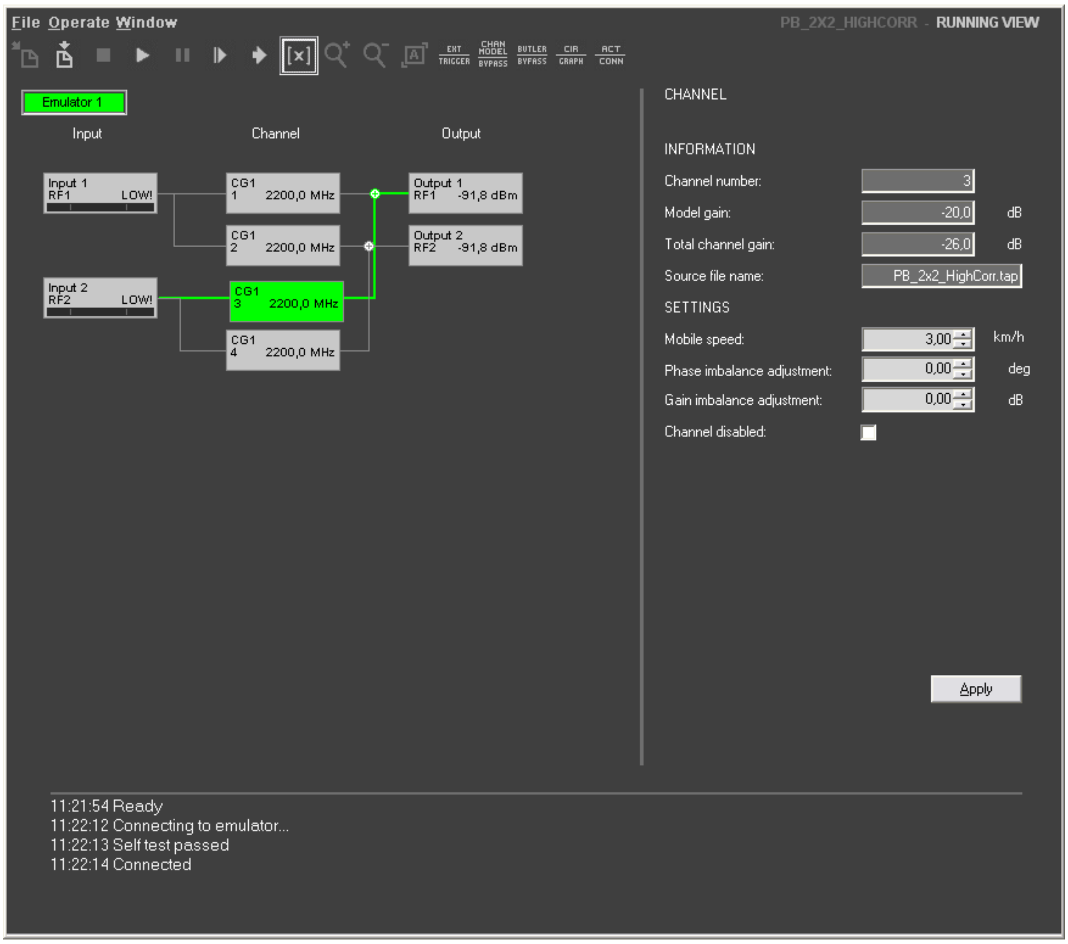


Figure 67. Channel settings with channel 3 selected

Information

Channel number

- Selected channel number.

Model gain

- Model gain is the average gain of the channel model over the full bandwidth (standard 40MHz, 80MHz F8 optional, 125MHz F8 optional). Note: There is always some variation due to the interaction of the input signal and the frequency selectivity of a channel model.

Total channel gain

- Total channel gain from input to output.

Source file name

- Name of the channel model source file.

Settings

Mobile speed

- Mobile speed affects the CIR update rate according to the following equation:

$$f_{upd} = \frac{2 \cdot SD \cdot v \cdot f_c}{c},$$

where
- *SD* sample density, samples per half-wave (in the channel model file),
- *c* speed of light,
- *v* mobile speed,
- f_{upd} CIR update rate,
- f_c center frequency from channel group.

- Mobile speed is read-only if the CIR update rate has been locked in the IR file. If sample density was zero in the original channel model file, the mobile speed cannot be calculated and the edit field will be empty.

Gain imbalance adjustment
- Imbalance adjustment value. The field can be used to adjust the gain of separate channels.
- Range for the adjustment is 0 … -100 dB with 0.1 dB steps
- **Note:** Adjustment are taken place when the emulation is running

Phase imbalance adjustment
- Imbalance adjustment value. The field can be used to adjust the phase of separate channels.
- Range for the adjustment is -180 … 180 degrees with 0.1 degree steps
- **Note:** Adjustment are taken place when the emulation is running

Channel disabled
- Disables channel when selected

Note: the Channel settings are not available for MIMO configurations when Grouped MIMO channel setting is enabled in File → Settings (see Chapter 4.4.4.1). In this mode Channel specific information can be accessed by right-clicking the channel group block and selecting “Channel info…”, see Figure 68.

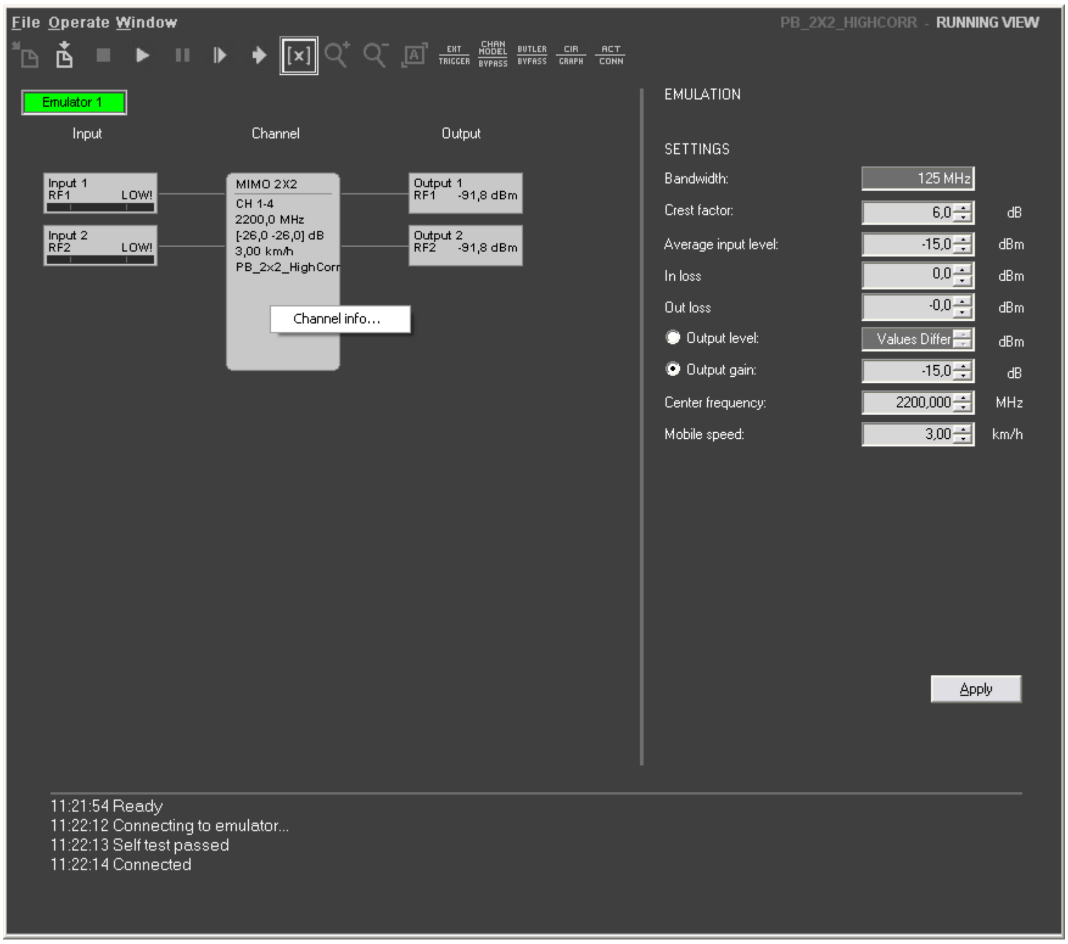


Figure 68. Accessing channel info

Channel Info dialog is shown in Figure 69. The information of each channel is arranged in rows.

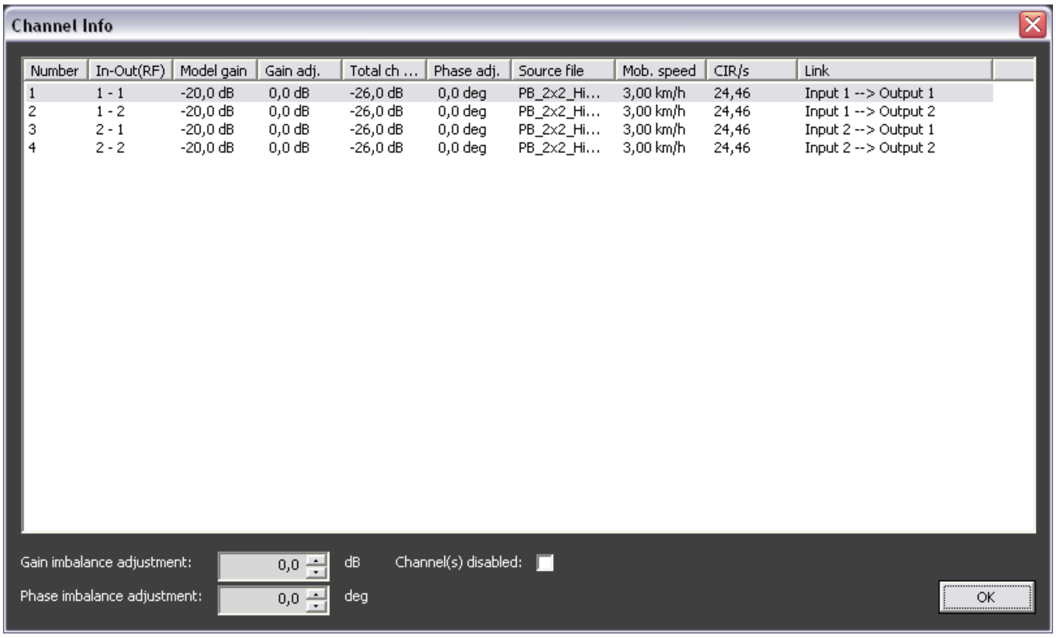


Figure 69. Channel Info dialog

Gain imbalance adjustment

- Channel gain can be adjusted by selecting row(s) in the table with mouse and setting the desired imbalance value in Gain imbalance adjustment.
- Range for the adjustment is 0 … -100 dB with 0.1 dB steps
- **Note:** Adjustment are taken place when the emulation is running

Phase imbalance adjustment

- Channel phase can be adjusted by selecting row(s) in the table with mouse and setting the desired imbalance value in Phase imbalance adjustment.
- Range for the adjustment is -180 … 180 degrees with 0.1 degree steps
- **Note:** Adjustment are taken place when the emulation is running

Channel(s) disabled

- Disables selected channel(s).

4.4.5.5 Input settings

The Input settings can be selected by repeatedly clicking the input block and cycling through the various settings pages.

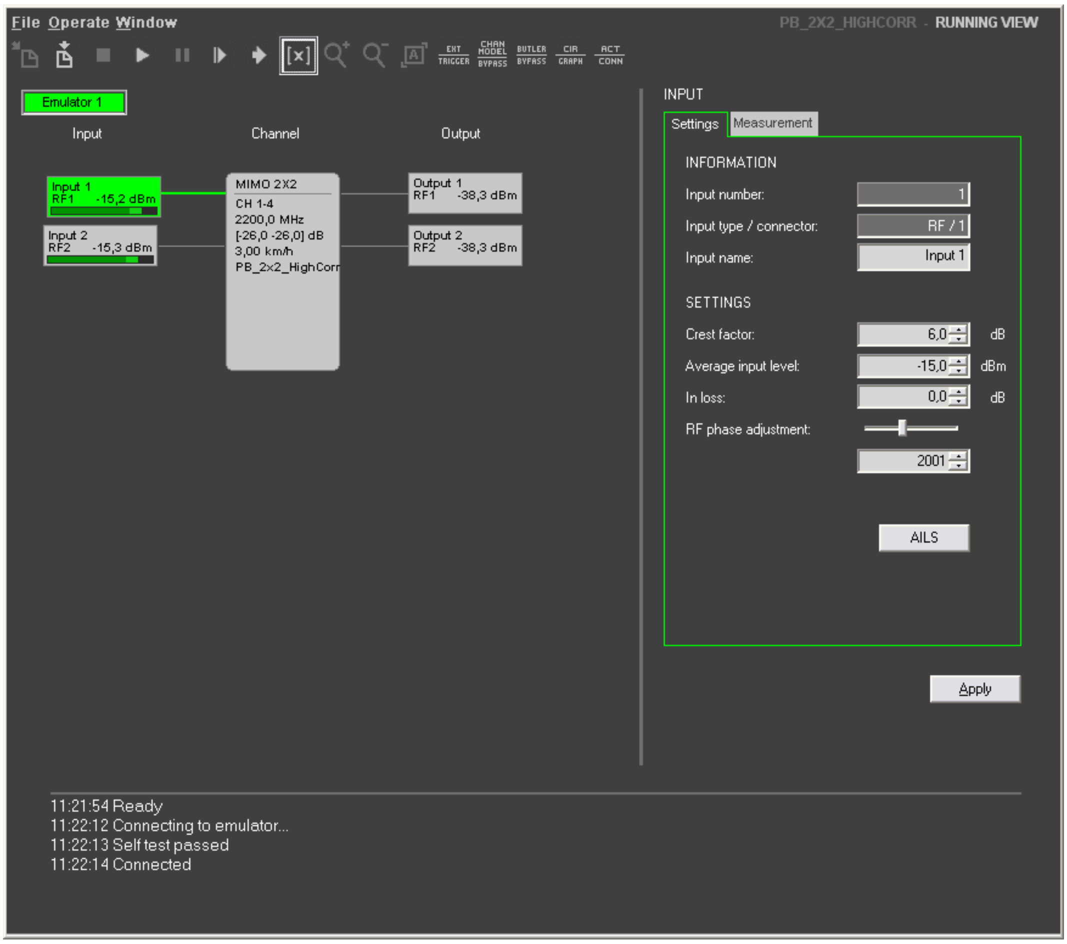


Figure 70. Input settings

4.4.5.5.1 Input parameters - Settings tab

Information

Input number

- Selected input number: the same number is shown in the top left corner of the input block in the diagram (if name is not defined). This number is used when accessing the input settings via remote interface (ATE).

Input type / connector

- RF Radio frequency.
- Input HW connector number. HW connector number is visible also in input elements of the block diagram.

Input name

- Custom name for current input: the name is also shown in the top left corner of the input block in the diagram

Settings

Crest factor

- Crest factor indicates the difference between the average power level and the peak power in the input signal.

Average input level

- Average input signal level, in [dBm]. This is used to calculate the input attenuation for the input signal so that the dynamics of the AD/DA converters is used optimally.
- The input's AD converter reaches max peak values with a sinusoidal input signal having the same average level as inputted to the settings when the crest factor is 0 dB. If the crest factor is set to 6 dB, an extra 6 dB is reserved until the AD converter input cuts the signal.

In loss

- Test setup loss between the transmitting device and EB Propsim input (external cables, attenuators and other equipment) in [dB]. When this value is measured and set correctly, average input level defines the level in the beginning of the input cable – ie. TX power of the transmitting device. Value in this field affects the limits of average input level. Example: if attenuation in the cable between transmitting device and EB Propsim input is 3.6 dB, value 3.6 dB is used as In loss. Value can be also negative if external amplifiers are used.

RF phase adjustment

- The user can adjust the input phase more than 360 degrees with this slider or by entering a value to edit field below. Adjustments are made immediately, without need to click Apply.
- Clicking with the mouse to the left or right of the slider handle makes the minimum possible adjustment. The adjustment range is 1200 … 3200. Each click represents 0.2 degrees.
- Adjustments are hardware interface specific and changing the emulation will not affect them.

AILS

- When input power level and the crest factor of the transmitted signal are constant but unknown AILS feature can be used to measure these values and to automatically set. Following use cases are supported by AILS:

  Normal signal power

  1. User connects signal source to EB Propsim input(s).
  2. User loads emulation in EB Propsim.

3. User selects one channel input from channel diagram.
4. User opens AILS settings dialog by clicking 'AILS' button in input settings page.
5. User specifies measurement time. Possible values are 1, 3, 10 and 30 seconds.
6. To set input level and crest factor for current channel user click 'Autoset Input' button.
7. To set input level and crest factor for all inputs in current emulation user click 'Autoset All' button.
8. During measurement status is displayed.
9. When measurement is finished measured input level and crest factor values are updated.
10. User can cancel measurement by clicking 'Cancel' button after measurement is started.

Too high signal power

1. Same as steps 1-7 in previous case.
2. Input level measured by AILS is too high and message to user is shown at status box.

Too low power

1. Same as steps 1-7 in the first case.
2. Input level is too low for AILS and message to user is shown at status box.

- AILS setting dialog is shown in Figure 71.

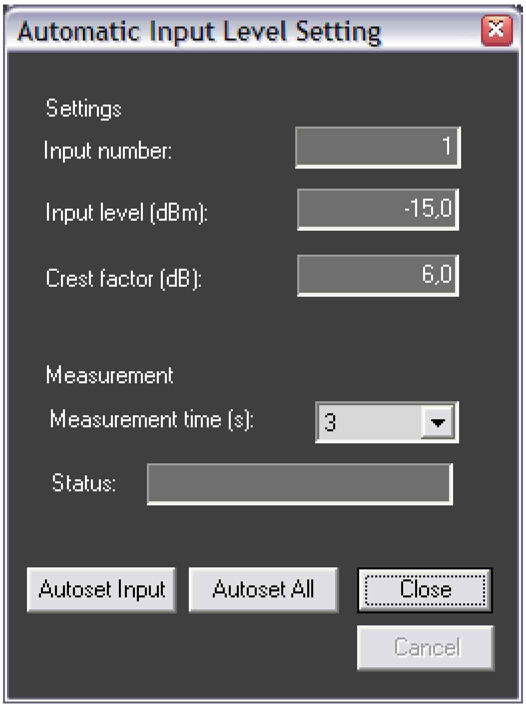


Figure 71. AILS settings dialog

4.4.5.5.2 Input settings - Measurement tab

Measurement tab of input settings allows user to change measurement parameters of each input, see Figure 72.

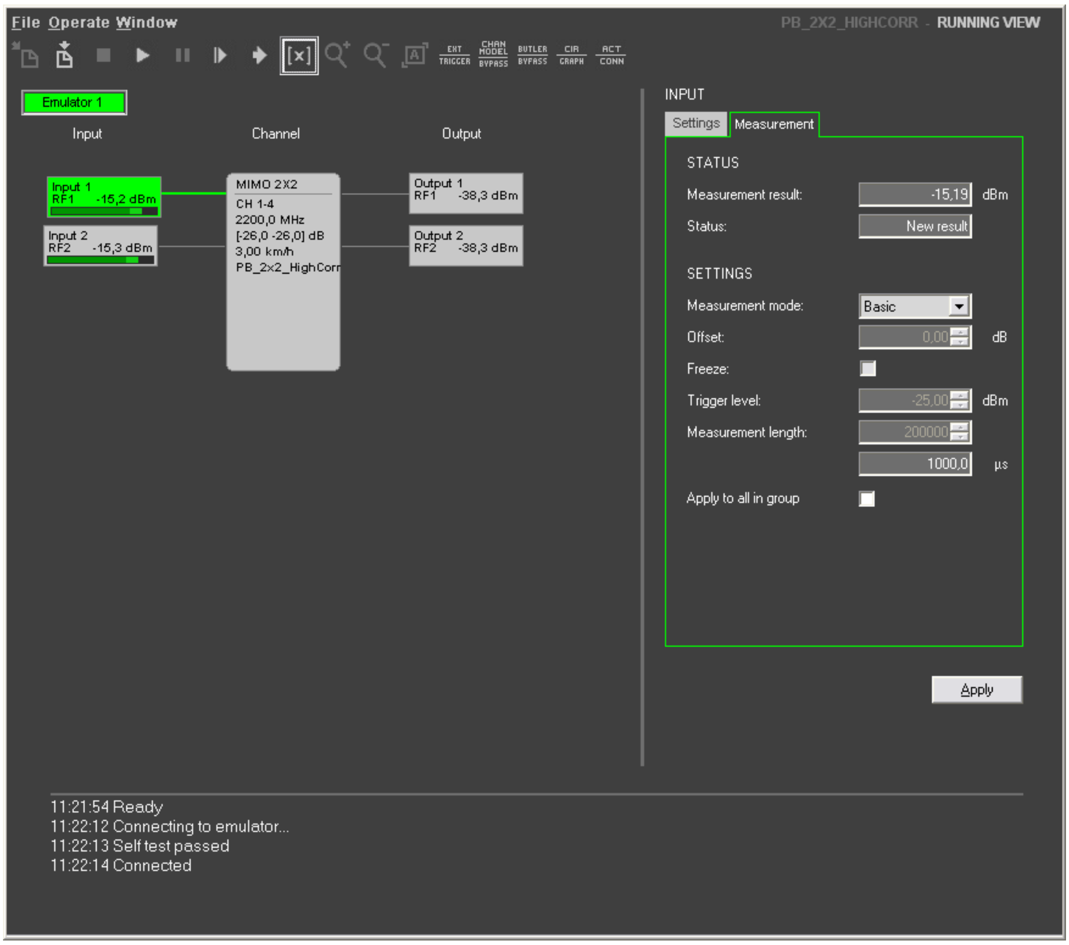


Figure 72. Input measurement settings

Status

Measurement result

- Displays the measurement result

Status

- Displays the status of the measurement:

Idle	The system is in idle state.
Measuring	The system is measuring a new result.
Result not ready	The system is measuring. If system seems to be in this state too long, verify burst measurement parameters.
New result	New up-to-date result is ready and shown in result box.
Frozen	The measurement result is frozen to value shown in measurement result box.

Settings

Measurement mode

- Measurement mode selection:

Disabled	Measurement of selected input is disabled completely.
Basic	Input is constantly measured, and offset setting is not applied to measurement.
Continuous	Continuous signal measurement. Offset setting is applied to measurement, and result can be frozen.

Burst	TDD measurement mode. Measurement is done during duty period of signal. Offset and freeze are available.

Offset

- User settable offset value for measurement fine-tuning. The value set here is directly added to measurement result, and also affects C/I setting. The range is -3 … +3 dB with 0.1 dB steps.

Freeze

- Freezes the SNR (or C/I) measurement, the same measurement result will be used until the user unfreezes the measurement.

Trigger level

- Specifies the trigger level of the Burst (TDD) measurement. The power greater then trigger level will be measured. The range depends on input settings. Resolution is 0.1 dBm.

Measurement length

- Specifies the number of averages that are used to determine the Burst (TDD) input level. The range is 32 … 134217728 (i.e. 0.2us to 671 ms).

Apply to all in group

- Apply the current setting to all inputs in current channel group.

4.4.5.6 Output settings

The Output settings can be selected by repeatedly clicking the output block and cycling through the various settings pages.

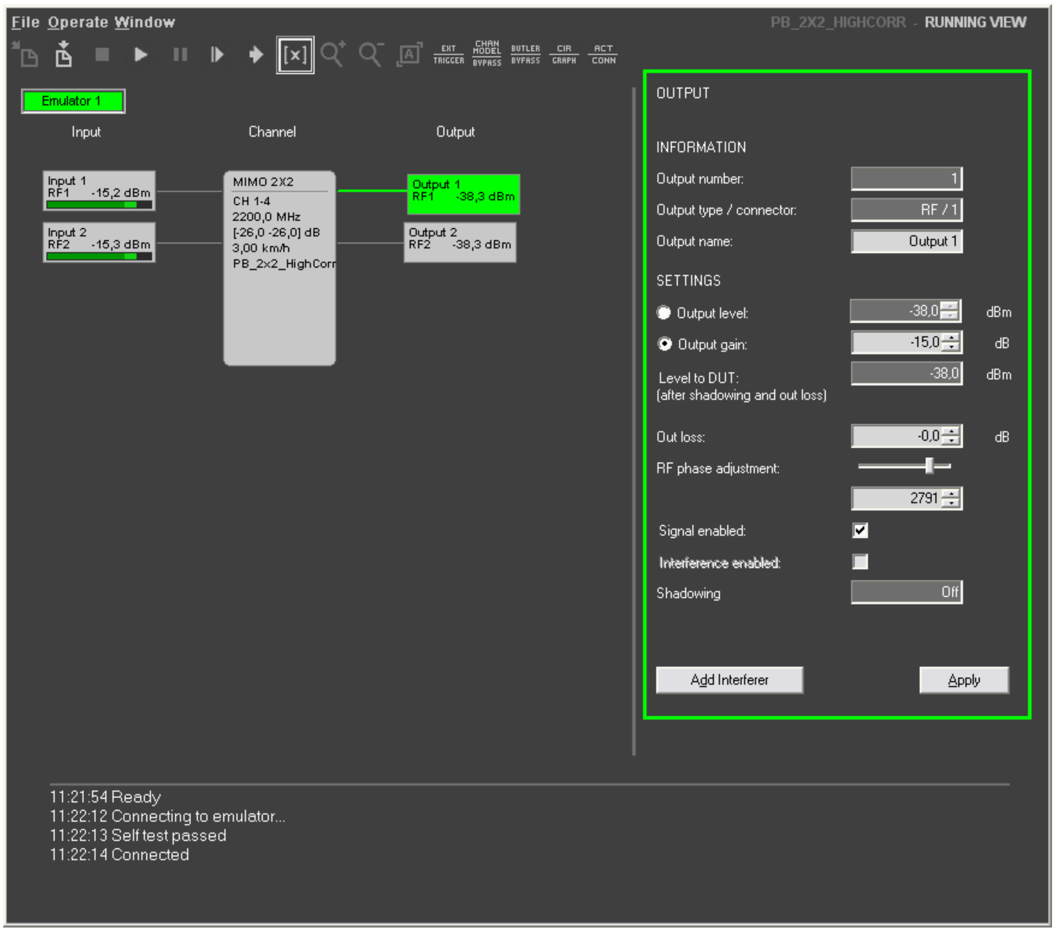


Figure 73. Channel output settings

Information

Output number

- Selected output number: the same number is shown in the top left corner of the output block in the diagram (if name is not defined). This number is used when accessing the output settings via remote interface (ATE).

Output type / connector

- RF Radio Frequency.
- Output HW connector number. HW connector number is visible also in output elements of the block diagram.

Output name

- Custom name for current output: the name is also shown in the top left corner of the output block in the diagram

Settings

Output gain

- Output gain value in [dB]. All signals connected to the output are attenuated by the same amount.

Output level

- Output level value at RF output [dBm].

Out loss

- Test setup loss between the EB Propsim output and the receiving device (external cables, attenuators and other equipment) in [dB]. When this value is measured and set correctly, output level, interference level and “Level to DUT” define the level in the end of the output cable – i.e. the power level fed to the receiving device. Value in this field affects the limits of output level and interference level. Example: if attenuation in the cable between EB Propsim and the receiving device is 3.6 dB, value 3.6 dB is used as Out loss. Value can be also negative if external amplifiers are used.

Level to DUT

- Current output level after shadowing and out loss, calculated from the total channel gain and average input level on current shadowing position. Note: There is always some variation due to the random nature of the statistical channel models.

RF phase adjustment

- The user can adjust the output phase more than 360 degrees with this slider. Adjustments are made immediately without clicking Apply.
- Clicking with the mouse to the left or right of the slider handle makes the minimum possible adjustment. The adjustment range is 1200 … 3200. Each click represents 0.2 degrees.
- Adjustments are hardware interface specific and changing the emulation will not affect them.

Signal Enabled

- Enables or disables output signal, leaving interference sources connected to output enabled.

Interference Enabled

- Enables or disables all interference sources connected to output. The field is visible only if interference option is available. If there is no interference attached to output, this setting has no effect.

Shadowing

- Shows whether shadowing profile is used in the current output (On / Off).

Add Interferer

- Adds an interference source to channel output. The button is visible only if interference option is available.

4.4.6 Running emulations with internal duplex ports (F32 and FS8)

This chapter shows how to run and adjust the duplex-emulation in Running view. Creating the duplex-emulation with Editing view is explained in Chapter 4.5.3.1 and with Scenario Wizard in Chapter 4.3.1.2.

When the emulation with internal duplexing is opened to Running view, uplink and downlink are drawn as overlapping layers. Figure 74 shows full duplex 2x2 MIMO emulation between base station and mobile. Clicking any element from downlink or uplink will bring the corresponding link to top. To view uplink and downlink separately, uncheck the “Grouped MIMO channels” checkbox from “File -> Settings”.

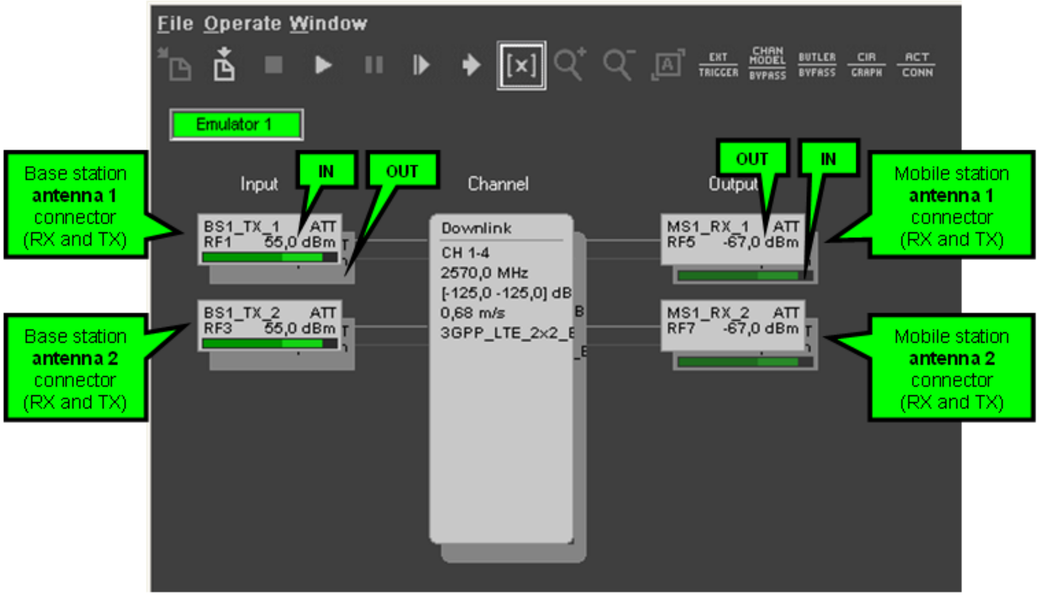


Figure 74. Running emulation with duplex ports

Figure 75 shows duplex emulation with uplink shown on top. Cabling principle is also shown in the figure – full duplex 2x2 MIMO connections between EB Propsim, base station (or communication tester) and mobile phone requires four cables.
Figure 76 shows the same cabling with physical IN/OUT connectors of EB Propsim.

Tip: To ease up the cabling, select any antenna port (EB Propsim input or output) from EB Propsim GUI - corresponding connector will be indicated with a blue led on the EB Propsim front panel.

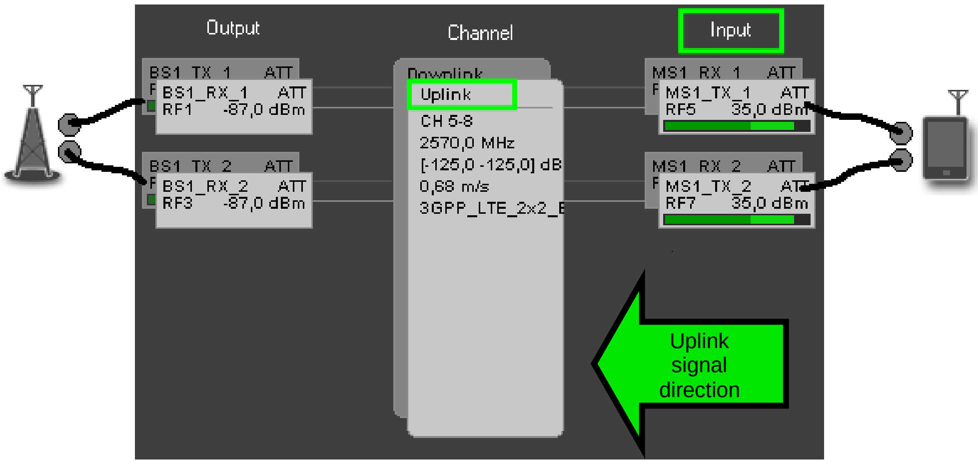


Figure 75. Duplex emulation with uplink on top. Note the signal flow from right to left (EB Propsim uplink inputs are on the right)

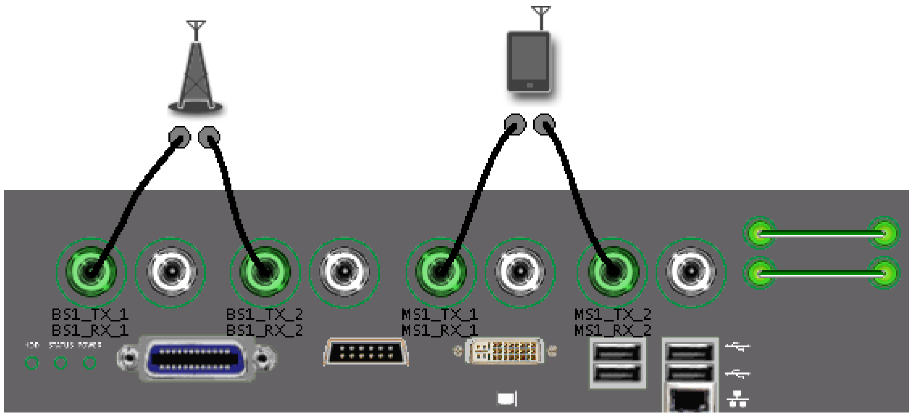


Figure 76. Connecting base station and mobile phone to duplex connectors, 2x2 full duplex MIMO

4.4.6.1 Input attenuator setting

EB Propsim duplex ports (F32 and FS8) contain an "Input Attenuator" setting as shown in Figure 77. Input attenuator affects the power level limits of both **input** and **output** located in the **same duplex connector** (IN/OUT connector). Input attenuator is enabled by default and allows higher input levels to be fed into EB Propsim input. At the same time the attenuator is limiting the maximum output level in the same connector. If higher output level is desired and lower input level is not an issue, internal input attenuator can be turned off. This is typical when the test setup contains communication tester devices instead of real base stations – output power of communication testers is normally quite low, but they require reasonably high incoming signal level.

As an example: when input attenuator is enabled, maximum input level is +15 dBm (with crest factor 15dB), and when disabled, maximum input level is -10 dBm (with crest factor 15dB).

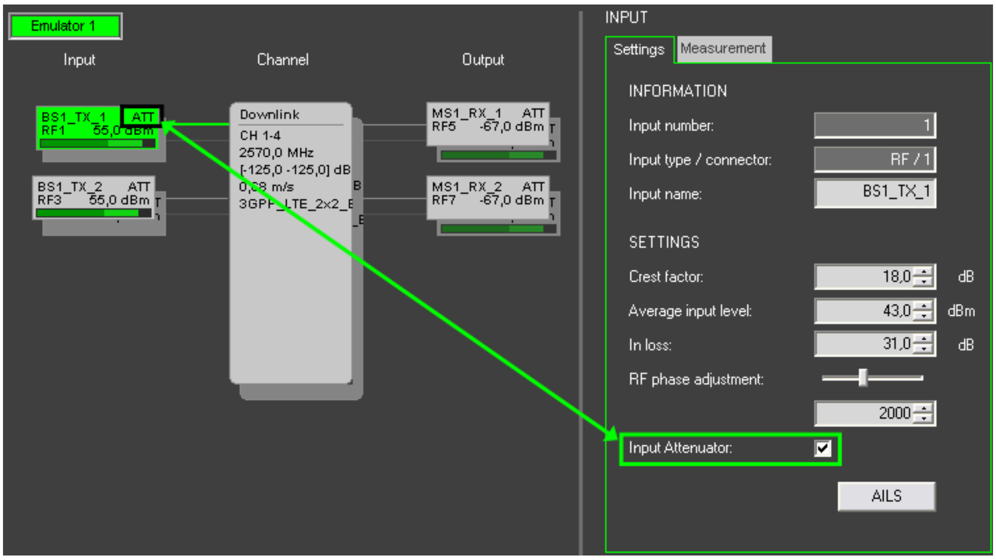


Figure 77. Input attenuator setting

When using EB Propsim duplex ports, also the frequencies of the uplink and downlink affect the maximum achievable output levels. In FDD scenarios where the frequency separation between uplink and downlink is 50 MHz or more, maximum output level can be up to 10 dB higher compared to the scenario where frequency separation is less than 50 MHz. The purpose of this limitation is to prevent infinite signal looping when frequency bands of uplink and downlink overlap inside EB Propsim.

4.4.7 CIR Graphics view

The change of the Channel Input Responses (CIRs) during the emulation can be seen in the CIR Graphics view. The view can be selected from Window menu of the Running view or from the toolbar.

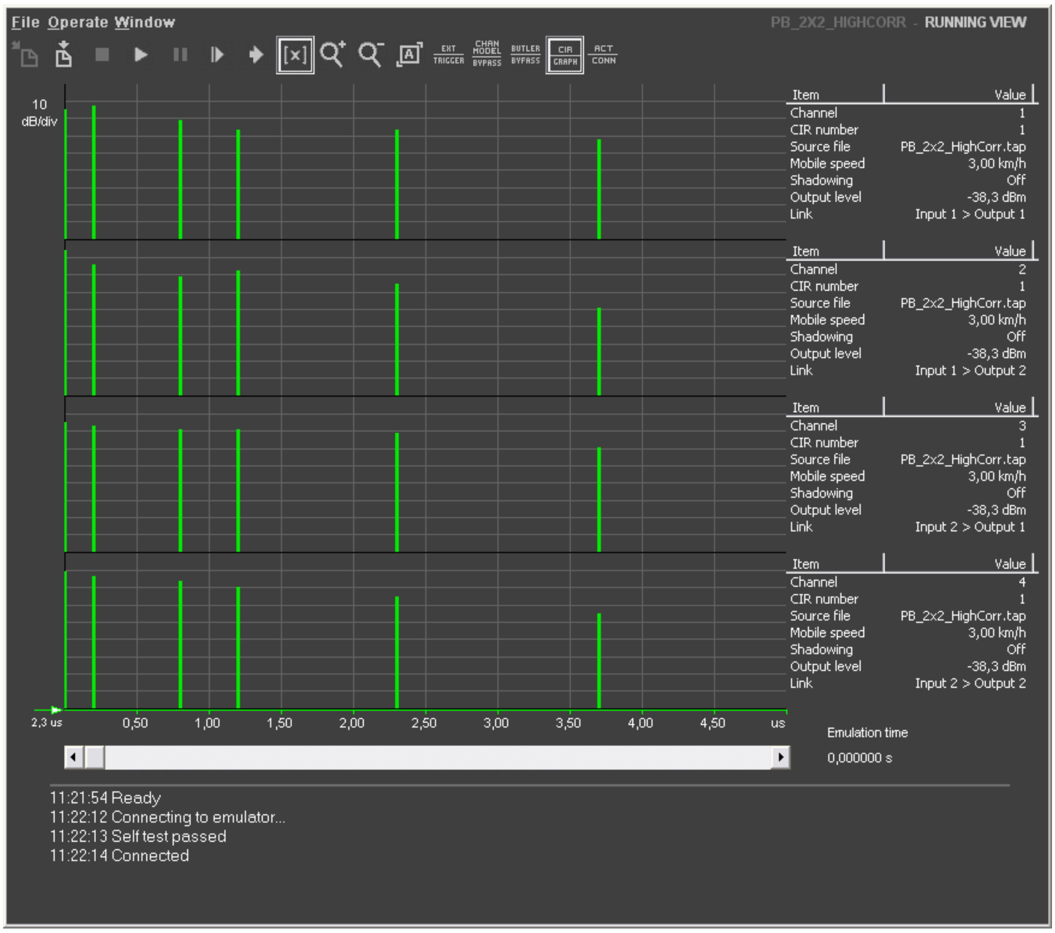


Figure 78. Channel impulse response display view

Signal paths defined in the channel model are shown in the CIR graphics window as discrete taps by default. The taps are shown in the delay positions where they have been defined in the channel model. Amplitudes of the taps have been scaled so that relative powers between channels are preserved.

4.4.7.1 Controlling CIR Display

The CIR graphics can be scaled using the “Zoom in” and “Zoom out” or “Autoscale” buttons in the **toolbar**. “Zoom in” is also possible when keyboard CTRL button is pressed down by sweeping CIR Graphics Window with mouse left button pressed down and realeasing in desired zooming end point or clicking mouse left button for start point and again for end point of the zooming area. The horizontal scroll bar at the bottom of CIR Graphics Window adjusts the visible delay area.

Emulation may have more channels than will fit comfortably to the visible CIR Graphics Window display area. The vertical scroll bar in the left side of window can be used for browsing through channels. Right-clicking on any channels in the channel impulse response display will give a menu shown in Figure 79. Using this menu the user can select to view only some of the channels. The menu item “Channels…” will show a list of all channels and their display status; see Figure 80.

Figure 79. Channel menu of CIR display

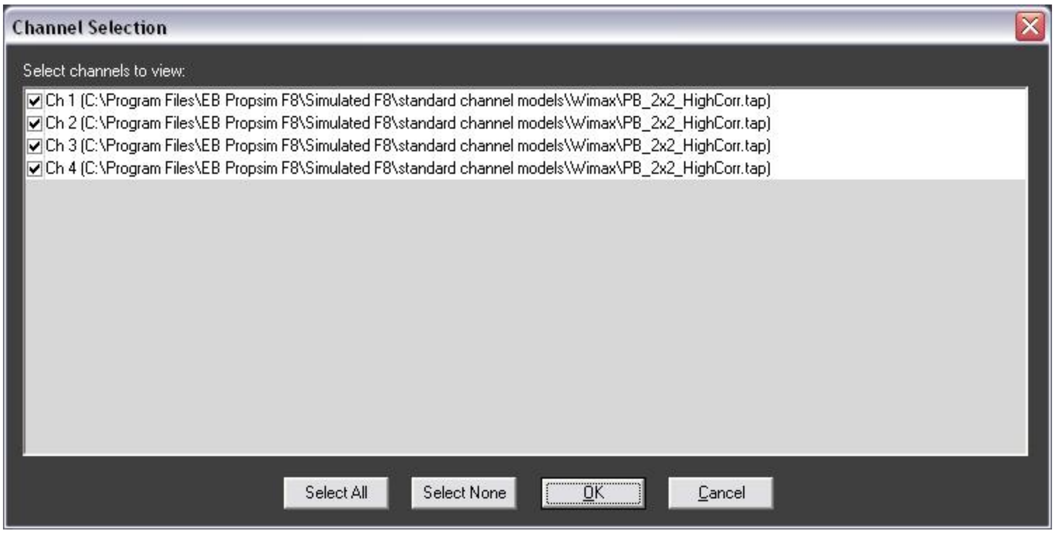


Figure 80. CIR display channel selection dialog

Figure 81 shows an example of CIR display in Single Channel View mode.

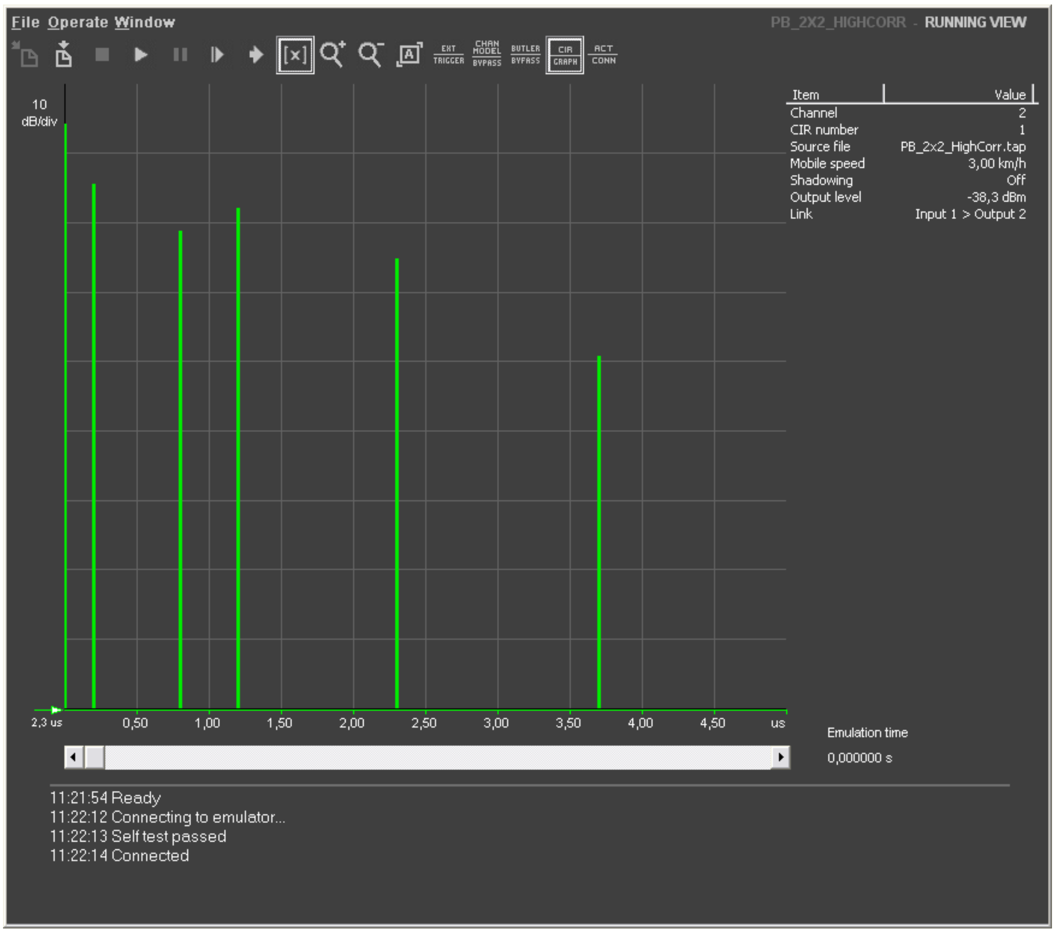


Figure 81. CIR display in single channel view

The channel impulse response graphics can also be changed to show a continuous line graph instead default discrete bar graph, see Figure 82. This graph format may be useful e.g. for measured channel models. The Graph Format dialog can be opened by right-clicking a channel and selecting “Properties…”, see Figure 79.

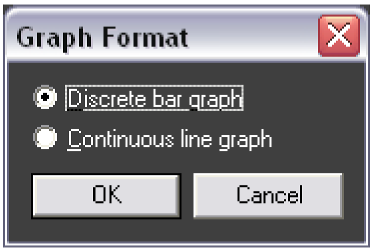


Figure 82. CIR display graph format selection dialog

4.4.7.1.1 Channel Information

Following items are shown in channel information list to the right of the CIR graph:
Channel number
CIR number
Source file name
Mobile speed
Shadowing status
Output level
Link information

The current CIR number keeps changing when the emulation is running.

Total running time of the emulation is shown in the bottom right corner of the CIR graphics window. The emulation time indicates the time the emulation has been run in seconds. The emulation time is cleared when the emulation is stopped.

4.4.7.1.2 Insertion delay

Insertion delay is shown at the bottom of the CIR graph.

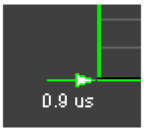


Figure 83. Insertion delay

Insertion delay describes the internal hardware delay with the used channel model. The total real delay of a tap is the sum of the insertion delay and the delay value defined in the channel model. If the smallest delay in the channel model is greater than the insertion delay, the insertion delay is zero.

4.5 Editing view

The Editing view is used for creating and editing emulation block diagrams. When a diagram is ready, the Emulation view generates hardware control files (recognized from .SIM file extension) that can be loaded into the emulator.

The Editing view (Figure 84) contains a menu bar, toolbar, emulation block diagram, settings view and status view.

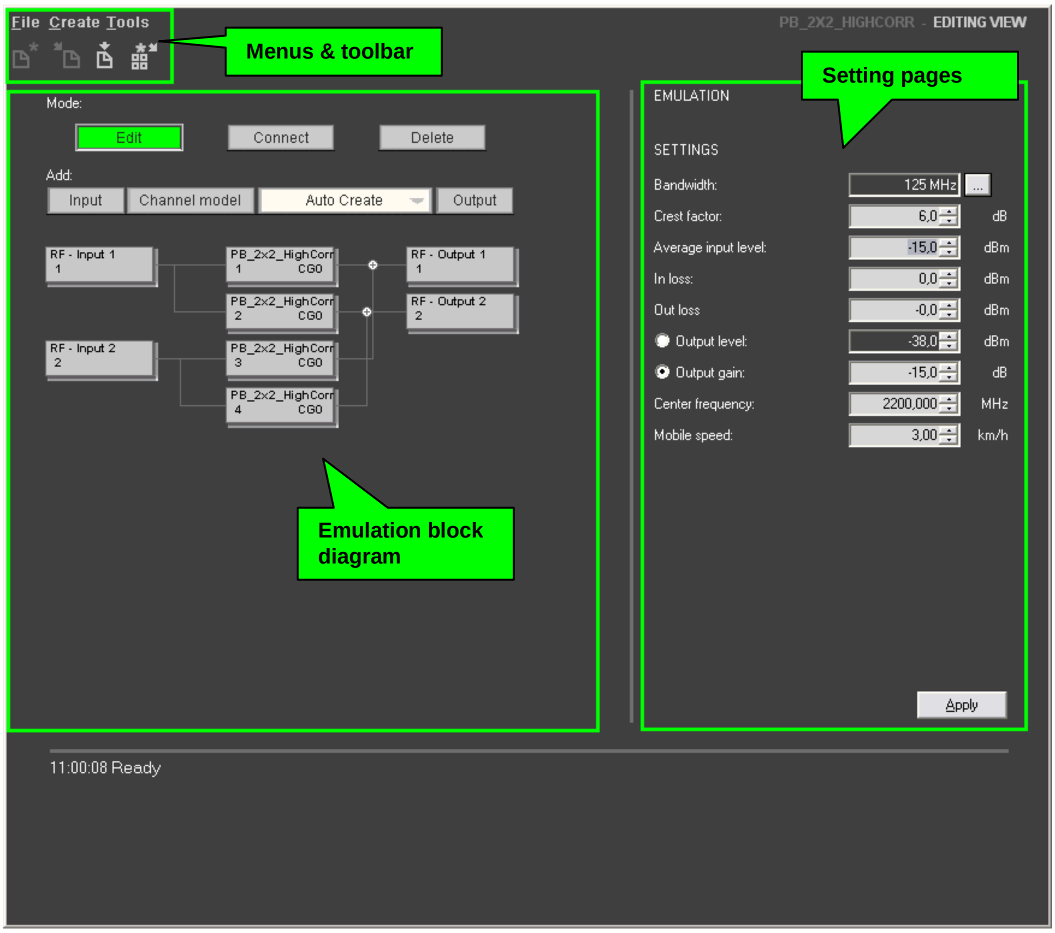


Figure 84. Editing view main window

Emulation block diagram has buttons for editing, connecting and deleting channels, interference sources, inputs and outputs.

Settings pages show the settings for the selected piece of the block diagram.
The Settings pages have a hierarchical structure. Different settings pages can be accessed by clicking repeatedly any of the input, channel or output blocks.

- **Emulation settings.** This page is for top level settings. Parameter changes in this view affect all lower level views.
- **Channel group settings.** The group consists of all channels that have common inputs or outputs.
- **Channel, Output or Input settings**
- **Interference settings**

Status view area at the bottom of the Editing view shows the emulation generation outputs and error messages.

4.5.1 Editing view toolbar

The tools available in Editing view toolbar are described below.

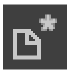
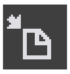
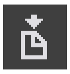
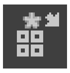
SYMBOL	TOOL	DESCRIPTION
	Create new emulation	Creates new emulation.
	Open emulation	Opens a dialog where user can browse and select an emulation file to be opened.
	Save emulation	Saves edited emulation with the same name or prompts the user to define a name if the emulation has not been saved previously.
	Generate model	Builds the emulation.

Table 60. Editing view toolbar

4.5.2 Editing view menus

The Editing view menus contain the following items:

MAINMENU	SUBMENU	SHORTCUT	DESCRIPTION
File	New…	Ctrl-N	Creates new emulation
	Open…	Ctrl-O	Opens emulation
	Close	Ctrl-F4	Closes emulation
	Save	Ctrl-S	Saves emulation
	Save As…		Saves emulation with new name
	Settings…		Editing view settings
	Recent files list		Four recently used emulation files
Create	Generate Model	Ctrl-G	Starts model generation (building emulation)
Tools	Conversions - ASC to IR		Converts ASC file to IR file, see chapter 4.5.2.2.
	Conversions - IR to ASC		Converts IR file to ASC file, see chapter 4.5.2.2.
	Conversions - Matlab to IR		Converts MAT file to IR file, see chapter 4.5.2.2.
	Run current emulation		Opens the emulation in Running view

Table 61. Editing view menus

4.5.2.1 Create New Emulation

Selecting File → New opens Create New Emulation dialog (Figure 85).

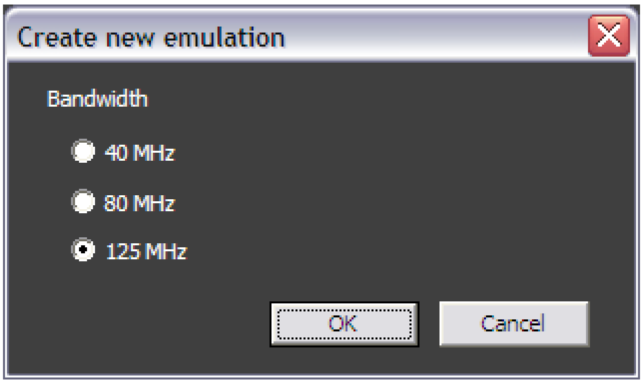


Figure 85. Create New Emulation dialog with EB Propsim F8

Available channel bandwidth options depend on the EB Propsim F8 channel bandwidth configuration and installed MIMO Extension option license. Table 62 below shows the maximum available fading channel count and bandwidth with different MIMO Extension options.

	MAX. BANDWIDTH	**125 MHZ**	**125 MHZ**	**125 MHZ**	**80 MHZ**
	MIMO extension mode	none	1 to 2 MIMO	1 to 4 MIMO	1 to 8 MIMO
Fading channels	8 RF channels	8	16	32	64
	6 RF channels	6	12	24	48
	4 RF channels	4	8	16	N/A
	2 RF channels	2	4	N/A	N/A

Table 62. Capacity of different MIMO Extension options (F8)

Clicking OK will create a new emulation file. The block diagram view with the Emulation Settings will be opened.

4.5.2.2 Conversions

The conversion tools in Tools menu enable file conversion from one format to another. Conversion from ASC to IR file is shown in Figure 86. Browse the source file and define the file to be created and click OK.

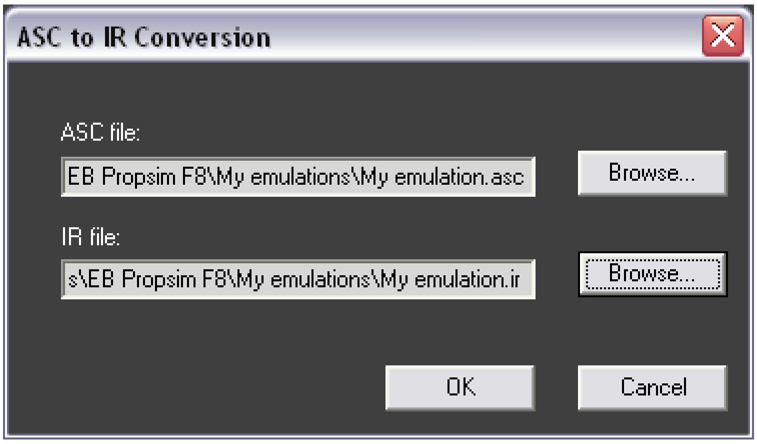


Figure 86. ASC to IR conversion dialog

ASC file is a simple text file format for importing and exporting channel impulse response data to/from IR-files. The ASC file format is explained in *File Formats* section of the User Reference documentation.

IR to ASC and MAT to IR conversions work similarly, see Figure 87 and Figure 88.

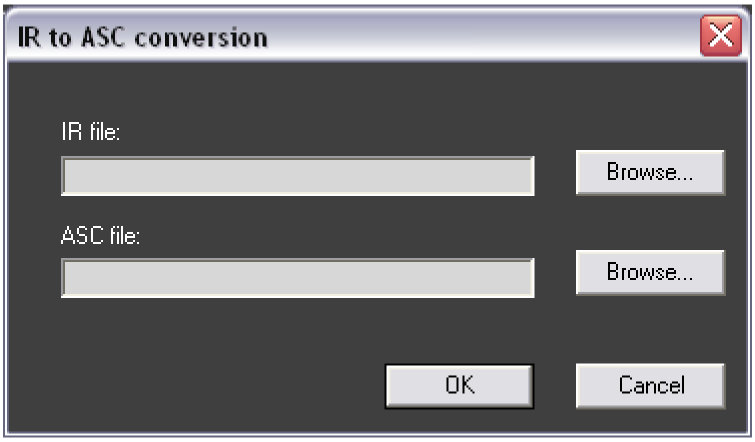


Figure 87. IR to ASC conversion dialog

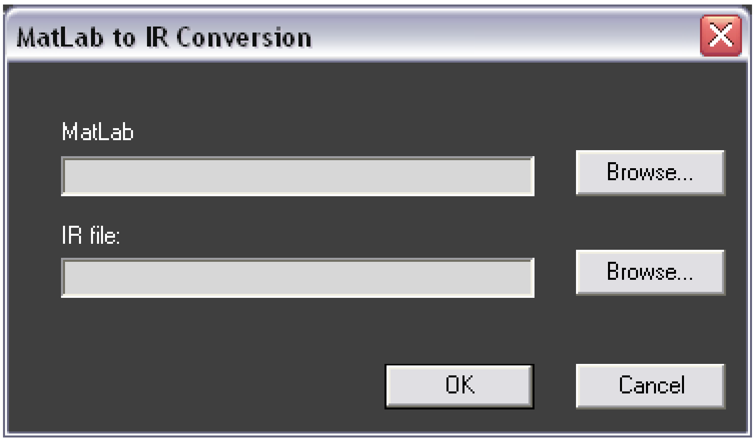


Figure 88. MAT to IR conversion dialog.

MAT (MatLab) is file format used by MathWorks, Inc.'s MatLab®. Specific MAT files can be converted to IR files to be used as channel models. The MAT file format is also explained in *File Formats* section of the User Reference documentation.

4.5.3 Emulation block diagram

The emulation block diagram illustrates the emulation setup.

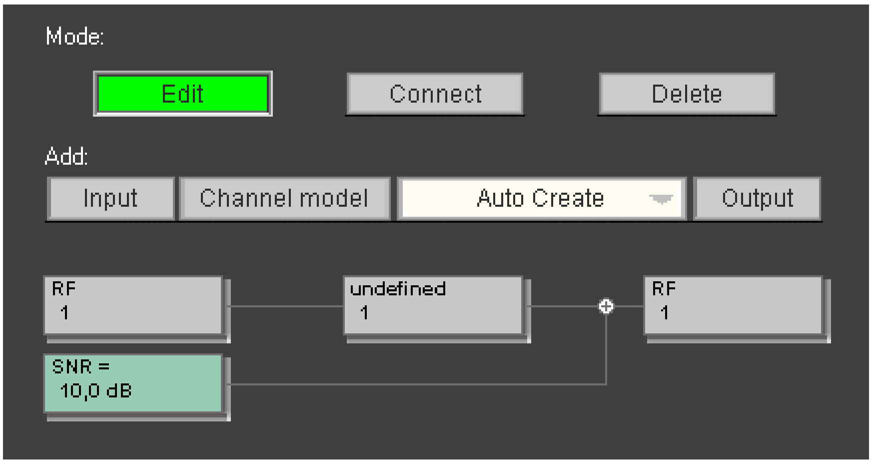


Figure 89. Emulation block diagram edit control

For inserting inputs, channel models and outputs

- The user clicks the Edit button in the Mode row.
- The user adds the needed number of inputs, channels and outputs by clicking the Input, Channel model and Output buttons.

For connecting inputs or outputs to the channel models

- The user clicks the Connect button in the Mode row.
- The user clicks one of the buttons (Input, Output or Channel Model) and then clicks another side button. Inputs and outputs can be connected to channel models.

For deleting input, channel models or outputs

- The user clicks the Delete button in the Mode row.
- The user clicks the button (Input, Output or Channel Model) that needs to be deleted. It is also possible to delete only the connection between blocks by clicking the line.

4.5.3.1 Auto Create

Auto Create (see Figure 90) allows user to automatically create special channel groups that in some cases cannot be created otherwise. Auto create is the way to create emulations using duplex ports with EB Propsim F32 and FS8.

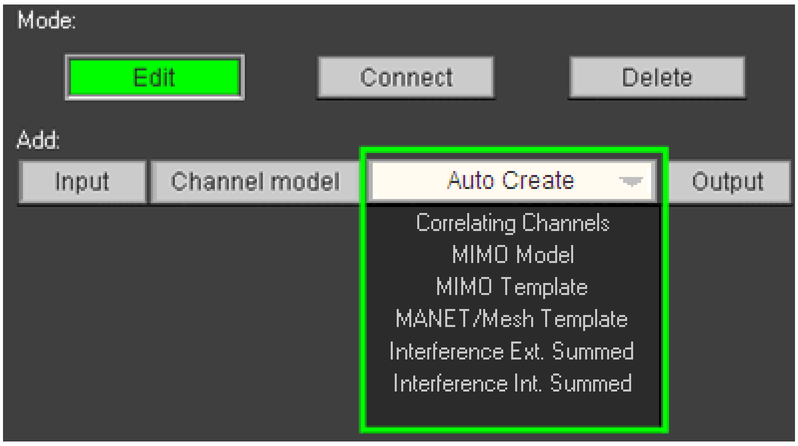


Figure 90. Auto Create selection

Correlating Channels

- Normal correlation (non-MIMO) or DoA models are done with correlating channels selection. After selecting this item, user has to select .TAP or .ICS file on which the correlation is based. Emulation Editor will then automatically create required number of channels, as well as their inputs and outputs, and load them with correlating model.

MIMO Model

- MIMO model creation works similarly to normal correlation creation, but it will automatically create all needed connections for a MIMO correlation model based on input .TAP or .ICS file.
- Select “Full Duplex” as in Figure 91 to use EB Propsim F32/FS8 internal duplex ports.

MIMO Template

- MIMO template creates inputs, channels and outputs for user-defined MIMO topology, allowing user to import own channel models for emulation.
- Select “Full Duplex” as in Figure 91 to use EB Propsim F32/FS8 internal duplex ports.
- To create bi-directional SISO link, select “Full Duplex” and define the number of inputs and outputs to 1 for both directions.

MANET/Mesh Template

- MANET/Mesh template creates inputs, channels and outputs for user-defined MANET/Mesh topology, allowing user to import own channel models for emulation.
- Select “Use duplex ports” as in Figure 91 to use EB Propsim F32/FS8 internal duplex ports.

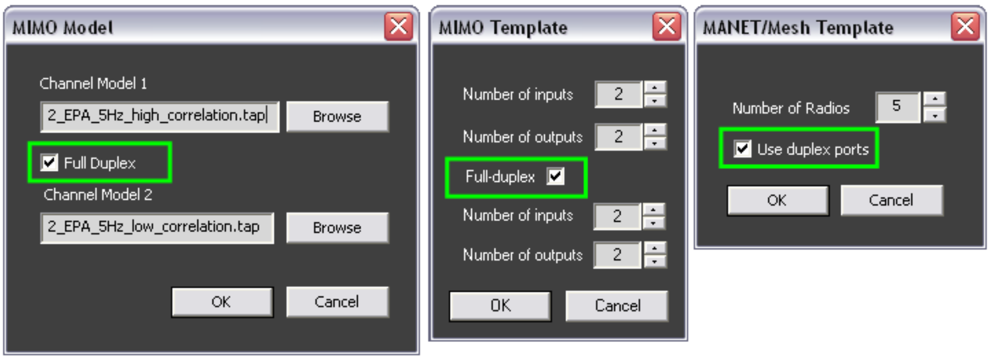


Figure 91. Selecting the usage of internal duplex ports when creating MIMO, Manet/Mesh channel topology. Duplex ports are available with EB Propsim F32 and EB Propsim FS8.

Interference Ext. Summed

- Externally summed non-correlating interference.
- It is possible to set the CW interference to external summing mode. This uses two channels, and user has to use an external RF signal coupler to sum signals from the outputs. In this mode the emulator automatically adjusts the output level of the signal and interference.

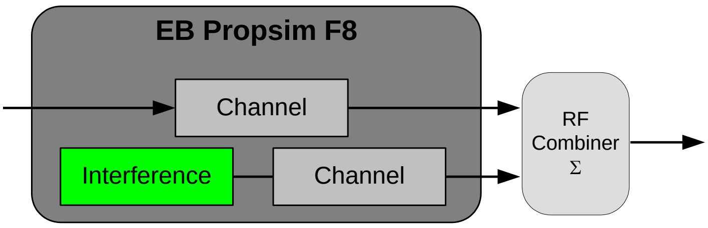


Figure 92. Externally summed fading interferer

Interference Int. Summed

- Internally summed non-correlating interference.
- By using two channels it is possible to create test cases where CW interference is passed through similar, but non-correlating fading channels before the signal and the interference are summed. In this configuration fading is enabled and C/I ratio is accurate over the emulation length. The channels used can have similar default profile but be non-correlating or have specific, known correlation, or channels can be completely different.

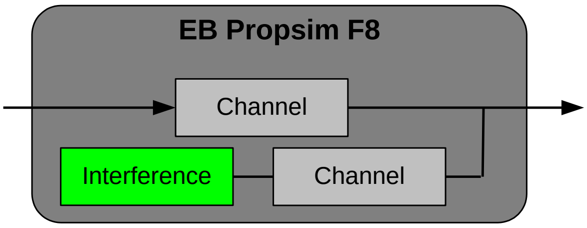


Figure 93. Internally summed non-correlated fading interference

4.5.4 Settings

The different settings and parameters for the emulation can be selected to the view by repeatedly clicking any of the input, channel or output blocks and cycling through the various setting pages in the block diagram.

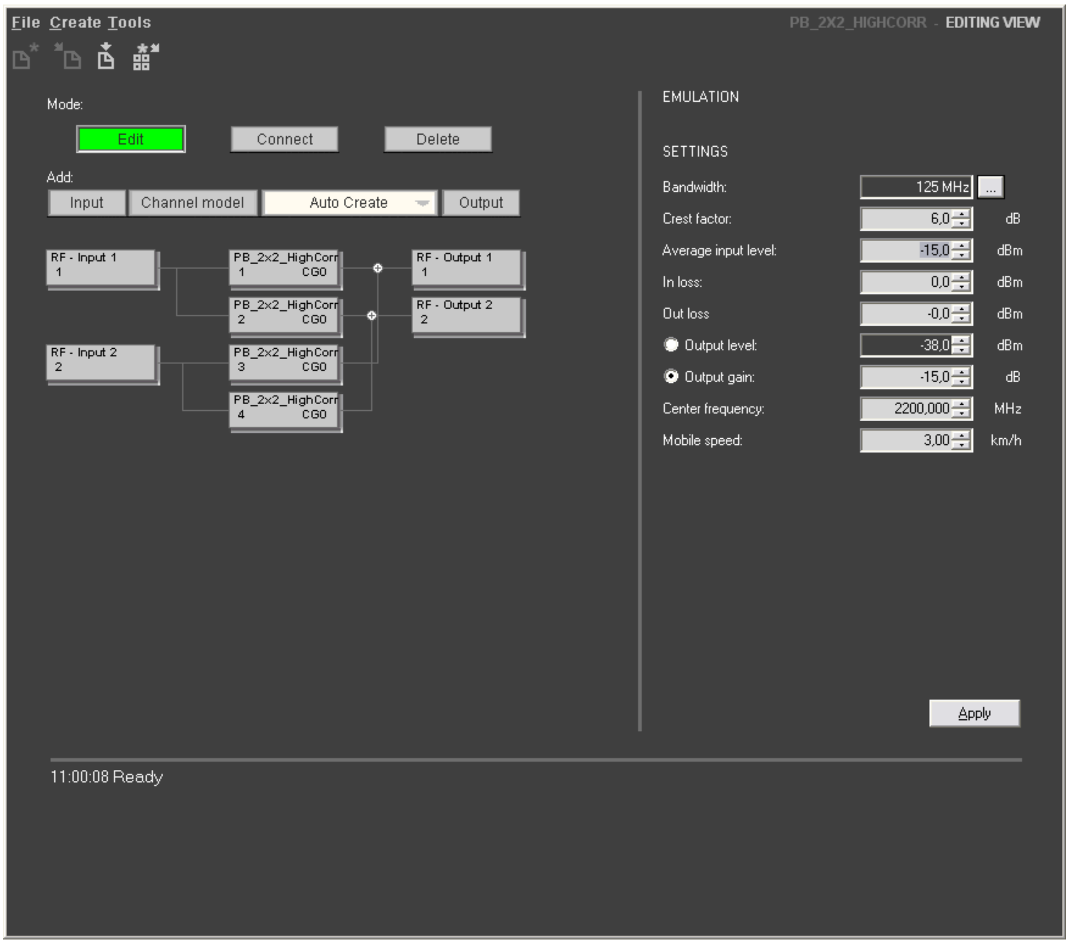


Figure 94. Editing view with emulation settings

The Settings pages have a hierarchical structure.

- **Emulation settings.** This page is for top level settings. Parameter changes in this view affect all lower level views.
- **Emulator settings**
- **Group settings.** Group consists of all channels that have common inputs or outputs.
- **Channel settings**
- **Output settings**
- **Input settings**

4.5.4.1 Emulation settings

The Emulation settings are shown in Figure 94.

Settings

Bandwidth

- Bandwidth for the current emulation. Content of the bandwidth selection window varies depending on available device configuration and MIMO extension option. Figure 95 and Figure 96 show options with two different setups. Bandwidth, channel amount and delay resolution always affect each others: reducing digital bandwidth increases available channels with less precise delay resolution.

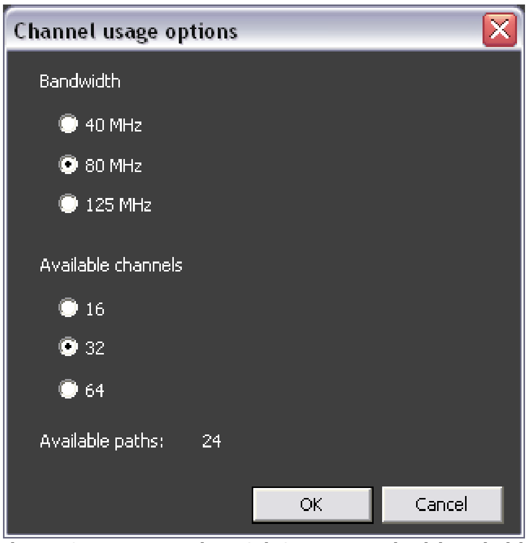


Figure 95. EB Propsim F8 (125 MHz option) bandwidth and channel amount selection

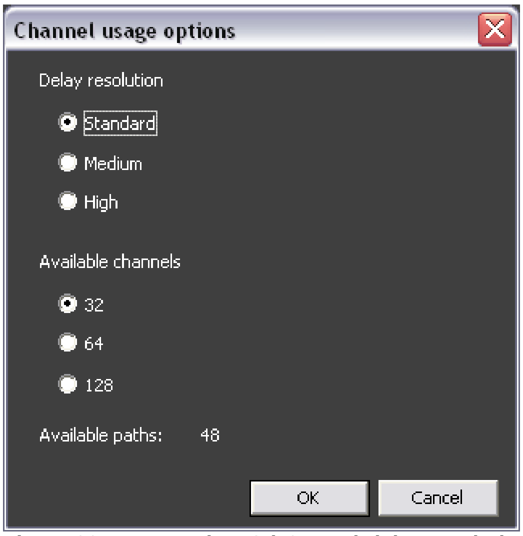


Figure 96. EB Propsim F8 (40 MHz) delay resolution and channel amount selection

Crest factor

- Defines the crest factor of the input signal. Crest factor defines the ratio between the peak and average of the input signal. Default value for crest factor is 6 dB.

Average input level

- Average input signal level of channel inputs in the emulation.

In loss

- Test setup loss between the transmitting device and EB Propsim input.

Out loss

- Test setup loss between EB Propsim output and the receiving device.

Output level

- Output level at DUT of all channel outputs in the emulation.

Output gain

- Output gain of all channel outputs in the emulation.

Center frequency

- Carrier wave center frequency of all channels in the emulation.

Mobile speed

- Mobile speed of all channels in the emulation.

4.5.4.2 Channel group settings

The Group settings can be accessed by repeatedly clicking any of the input, channel or output blocks belonging to the group and cycling through the various setting pages.

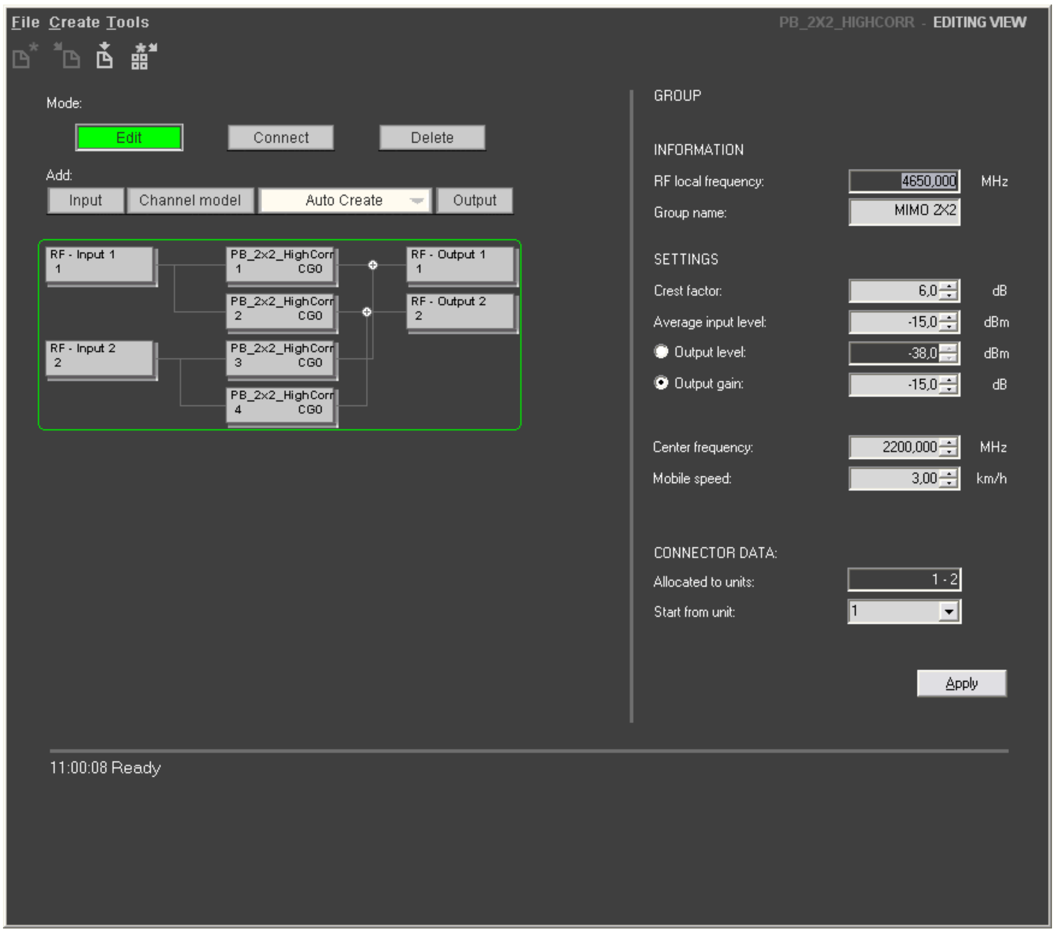


Figure 97. Editing view with channel group settings

Information

RF local frequency

- RF local frequency [MHz] defines the signal frequency needed at the RF local frequency connector (IN RFLO) of the RF units belonging to the channel group. If emulator is equipped with Internal Local Oscillator option, RFLO can be generated internally and external generators are not needed.

Group name

- Custom name for current channel group.

Settings

Crest factor

- Defines the crest factor of the input signal. Crest factor defines the ratio between the peak and average of the input signal. Default value for crest factor is 6 dB.

Average input level

- Average input signal level of channel inputs in channel group.

Output level

- Output level at all outputs in the channel group.

Output gain

- Output gain of all outputs in the channel group.

Center frequency

- Carrier wave center frequency of channel group.
- Center frequency is read-only if it has been locked in any IR file used in channels belonging to the channel group. The statistical models containing Suzuki or lognormal taps and statistical multi-channel models using the direction of arrival correlation model have locked CIR update rates. To change the CIR update for these models you have to edit them in Channel model view.

Mobile speed
- Mobile speed of all channels in the channel group.

Connector data (F8)

Allocated to units
- Display the units where current channel group (link) is allocated to.

Start from unit
- Number of the first unit where current channel group (link) is allocated to.

4.5.4.3 Channel settings

The Channel settings can be accessed by repeatedly clicking the channel block and cycling through the various setting pages.

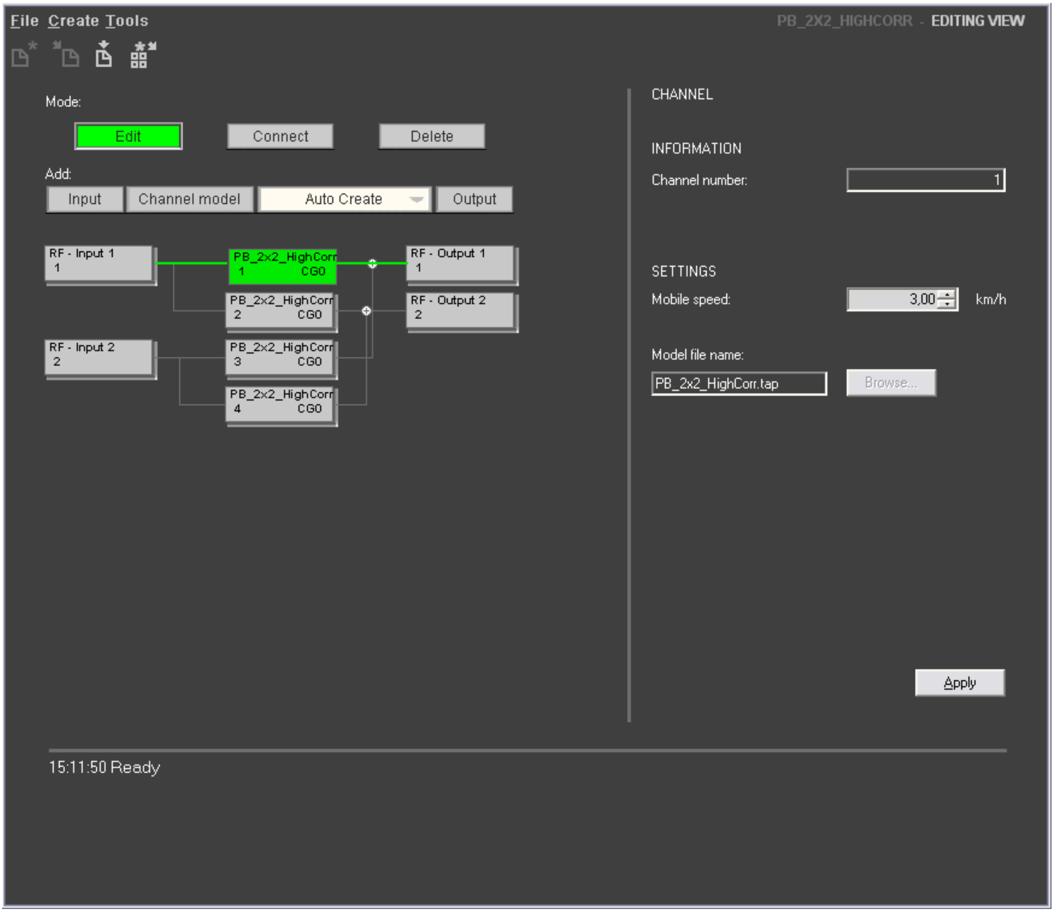


Figure 98. Editing view with channel 1 settings

Information

Channel number

- Selected channel number: the same number is shown in the bottom left corner of the channel block in the diagram.

Settings

Mobile speed

- Mobile speed affects the CIR update rate according to the following equation:

$$f_{upd} = \frac{2 \cdot SD \cdot v \cdot f_c}{c},$$

where

SD	sample density, samples per half-wave (in the channel model file),
c	speed of light,
v	mobile speed,
f_{upd}	CIR update rate,
f_c	center frequency from channel group.

- Mobile speed is read-only if the CIR update rate has been locked in the IR file. If sample density was zero in the original channel model file, the mobile speed can not be calculated and the edit field will be empty.

Model file name

- Channel model file name: .TAP, .IR, .ICS, .ASO, .ASC, .RTC or .MAT. The user can import channel models by converting them to the generic .IR or .ASC format. The file format syntax specifications can be found in *File Formats* section of the User reference documentation.

4.5.4.4 Input settings

The Input settings can be accessed by repeatedly clicking the input block and cycling through the various setting pages.

Figure 99. Editing view with input 1 settings

Information

Input number
- Selected channel input number: the same number is shown in the bottom left corner of the input block in the diagram.

Input name
- Custom name for current input.

Settings

Input type
- RF Radio Frequency.

Crest factor
- Defines the crest factor of the input signal. Crest factor defines the ratio between the peak and average of the input signal. Default value for crest factor is 6 dB.

Average input level
- Average input signal level. This is used to calculate the input attenuation for the input signal so that the dynamics of the AD/DA converters is used optimally.
- The input’s AD converter reaches max peak values with a sinusoidal input signal having the same average level as inputted to the settings when the crest factor is 0 dB. If the crest factor is set to 6 dB, then an extra 6 dB is reserved until the AD converter input cuts the signal.

In loss

- Test setup loss between the transmitting device and EB Propsim input (external cables, attenuators and other equipment) in [dB]. When this value is measured and set correctly, average input level defines the level in the beginning of the input cable – ie. TX power of the transmitting device. Value in this field affects the limits of average input level. Example: if attenuation in the cable between transmitting device and EB Propsim input is 3.6 dB, value 3.6 dB is used as In loss. Value can be also negative if external amplifiers are used.

Measurement

These settings can enable burst power measurement, where effective power (duty period) of time-divided signal is measured in order to determine the input power level of EB Propsim. This kind of measurement is needed for example with WLAN testing, since the signal duty cycle can not be accurately pre-determined.

Measurement mode

- Measurement mode selection:

Disabled	Input measurement is disabled.
Basic	Measurement enabled. No freeze or offset compensation is available.
Continuous	Measurement where Freeze and Offset are available.
Burst	TDD measurement mode. Measurement is done during duty period of signal. Offset and freeze are available.

Offset

- User settable offset value for measurement fine-tuning. The value set here is directly added to measurement result, and also affects C/I setting. The range is -3 … +3 dB with 0.1 dB steps.

Trigger level

- Specifies the trigger level of the Burst (TDD) measurement. The power greater then trigger level will be measured. The range depends on input settings. Resolution is 0.1 dBm.

Measurement length

- Specifies the number of averages that are used to determine the Burst (TDD) input level. Range is 32 … 134217728 (i.e. 0.2us to 671 ms).

Connector data

Unit number

- Unit number for selected input.

Connector (F32 and FS8)

- Connector location for selected input, see Figure 100.

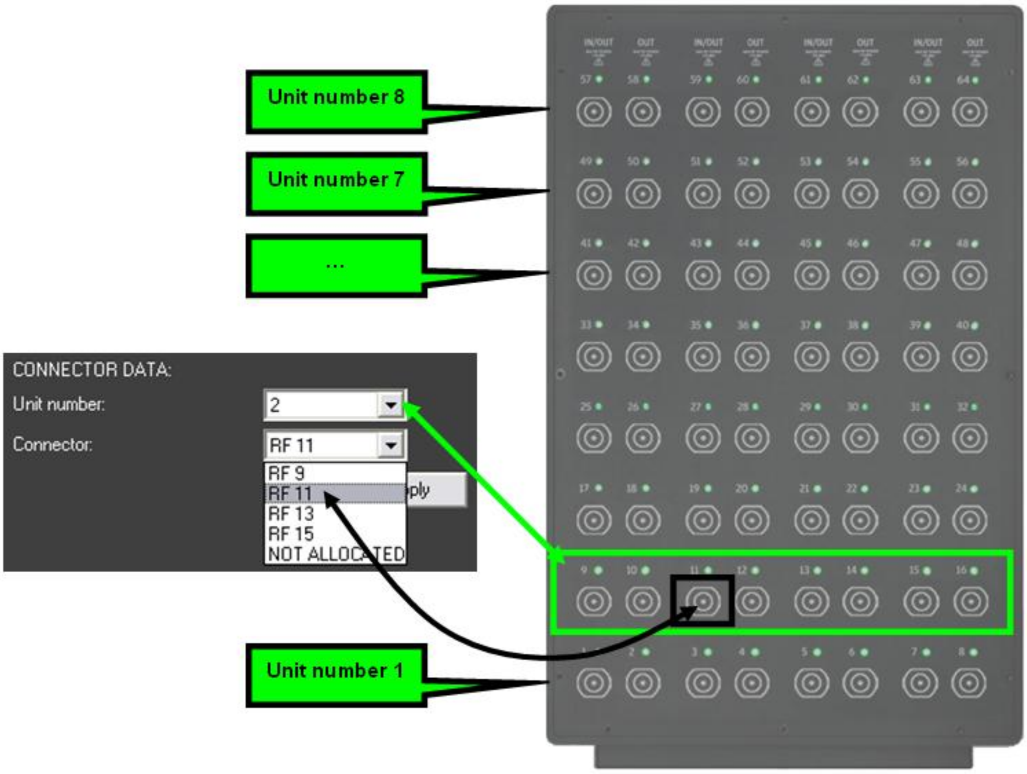


Figure 100. Input connector selection and mapping to physical device F32.

4.5.4.5 Output settings

The Output settings can be accessed by repeatedly clicking the output block and cycling through the various setting pages of the Editing view.

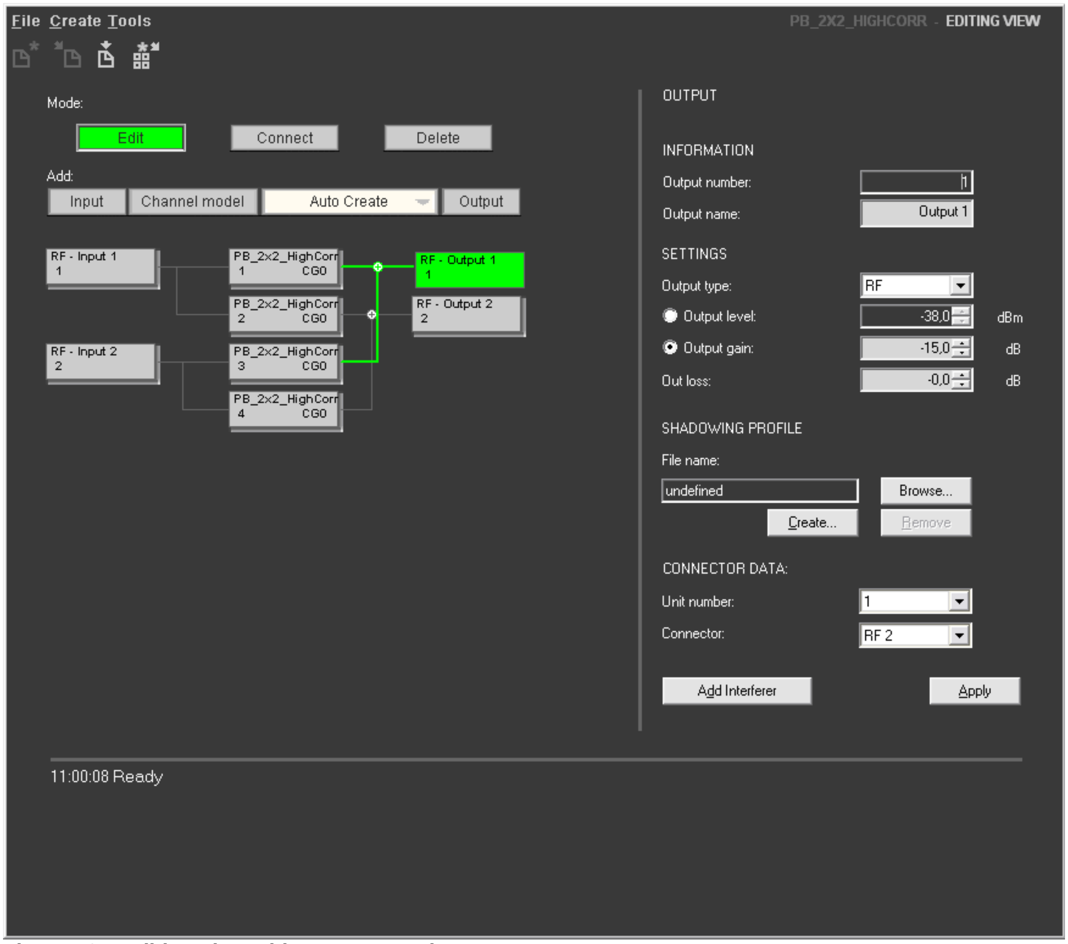


Figure 101. Editing view with output 1 settings

Information

Output number

- Selected channel output number: the same number is shown in the bottom left corner of the output block in the diagram.

Output name

- Custom name for current output.

Settings

Output type

- RF Radio Frequency.

Output level

- Output level at RF output, value in [dBm].

Output gain

- Output attenuation value in [dB].

Out loss

- Test setup loss between the EB Propsim output and the receiving device (external cables, attenuators and other equipment) in [dB]. When this value is measured and set correctly, output level and interference level define the level in the end of the output cable – i.e. the power level fed to the receiving device. Value in this field affects the limits of output level and interference level. Example: if attenuation in the cable between EB Propsim and the receiving device is 3.6 dB, value 3.6 dB is used as Out loss. Value can be also negative if external amplifiers are used.

Shadowing Profile
- Existing shadowing profile can be selected to be applied by using Browse.
- New shadowing profile can be generated by clicking Create.
- The optional Shadowing feature is described more detailed in chapter 5.

Connector data

Unit number
- Unit number for selected output.

Connector (F32 and FS8)
- Connector location for selected output, see Figure 100.

Add Interferer
- Interference can be defined to be used for the emulation by clicking Add Interferer.
- The optional interference source is described more detailed in chapter 6.

4.6 Channel model view

The Channel model view is targeted for creating a simplified statistical channel model for stationary scenarios and scenarios where the mobile station is moving away or towards the base station at a constant speed.

The main function of the Channel model view is to create and edit channel model files.

The view can be launched by selecting Emulator → Channel model view from the navigation bar.

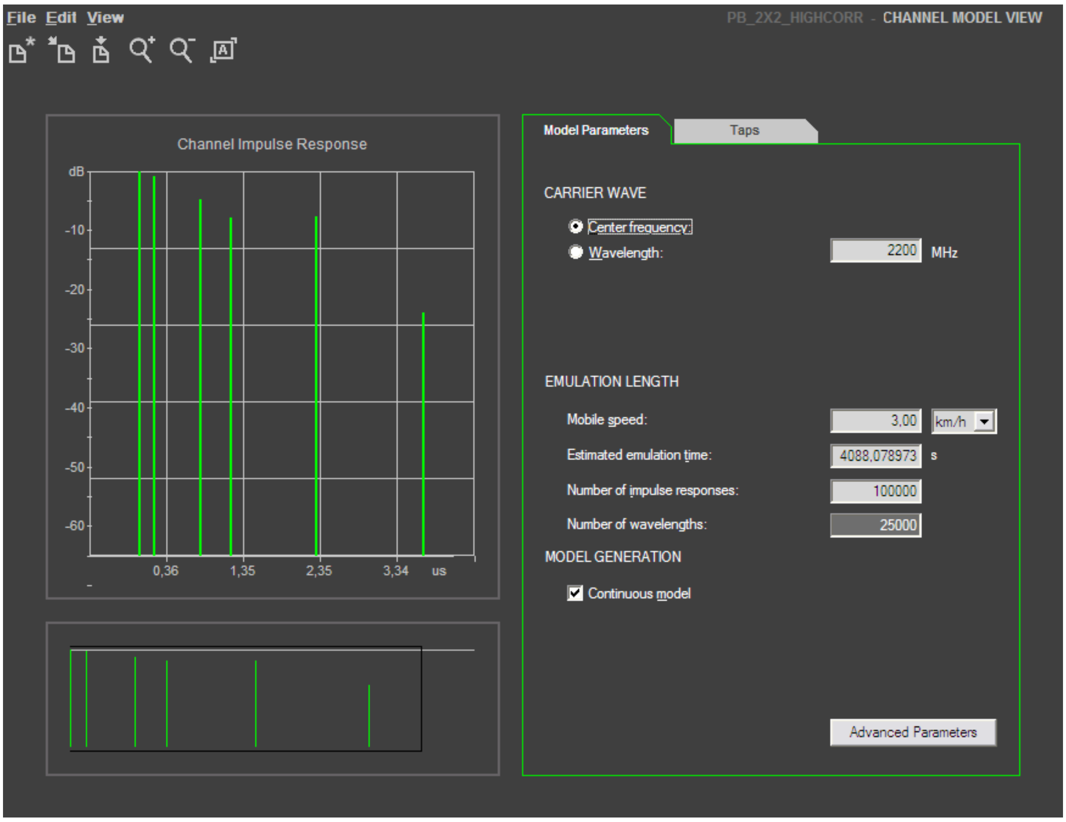
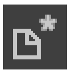
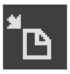
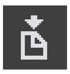


Figure 102. Channel model view

The Channel Model view contains a menu bar, a toolbar, a channel impulse response graph window and two tabs for channel model settings (Model parameters and Taps).

4.6.1 Channel model view toolbar

The tools available in Channel model view toolbar are described below.

SYMBOL	TOOL	DESCRIPTION
	Create new channel model	Creates new channel model file
	Open	Opens a dialog where user can browse and select a channel emulation file to be opened.
	Save	Saves edited channel model with the same name or prompts the user to define a name if the channel model has not been saved previously.

SYMBOL	TOOL	DESCRIPTION
	Zoom in	Can be used to scale the CIR graphics in CIR Graphics View.
	Zoom out	Can be used to scale the CIR graphics in CIR Graphics View.
	Autoscale	Selects such a scaling in CIR Graphics View that all taps in all channels are visible.

Table 63. Channel model view toolbar

4.6.2 Channel model view menus

The Menu bar contains the following menus:

MENU	SUBMENU	SHORTCUT	DESCRIPTION
File	New	Ctrl-N	Creates new channel model, see chapter 0
	Open	Ctrl-O	Opens channel model
	Save	Ctrl-S	Saves channel model
	Save As…		Saves channel model with new name
	Change model type	Ctrl-G	Enables model type change
	Recent file list		Recently used channel model files
Edit	Cut	Ctrl-X	
	Copy	Ctrl-C	
	Paste	Ctrl-V	
	Delete	Del	
	Select All	Ctrl-A	
	Invert Selection		
	Add Tap		Adds a new tap
	Remove Tap		Removes selected tap
	Properties…		Opens Tap Properties dialog
View	Zoom In		
	Zoom Out		
	Autoscale		

Table 64. Channel model view menus

4.6.2.1 New Model Generation

Selecting File –> New opens New Model Generation Wizard dialog, see Figure 103. The wizard first asks for the type of the model. The model can be a single-channel model, geometry based model, MISO- or SIMO-type correlative model or a MIMO-type correlative model.

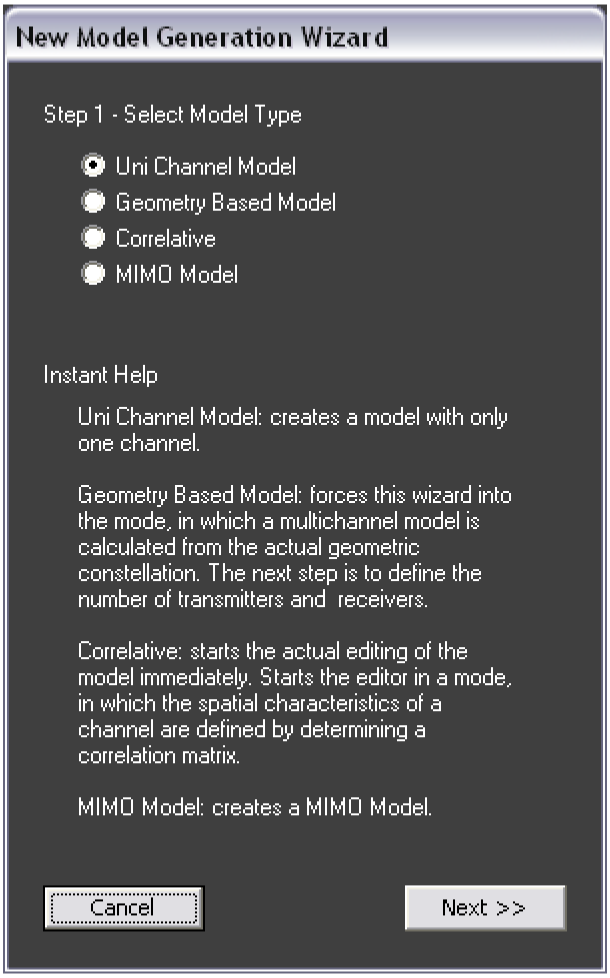


Figure 103. New Model Generation Wizard

Next step is to give the number of channels. If the model is correlative, single number is required. If a MIMO emulation is to be created, both number of transmitter and receiver antenna elements must be specified.

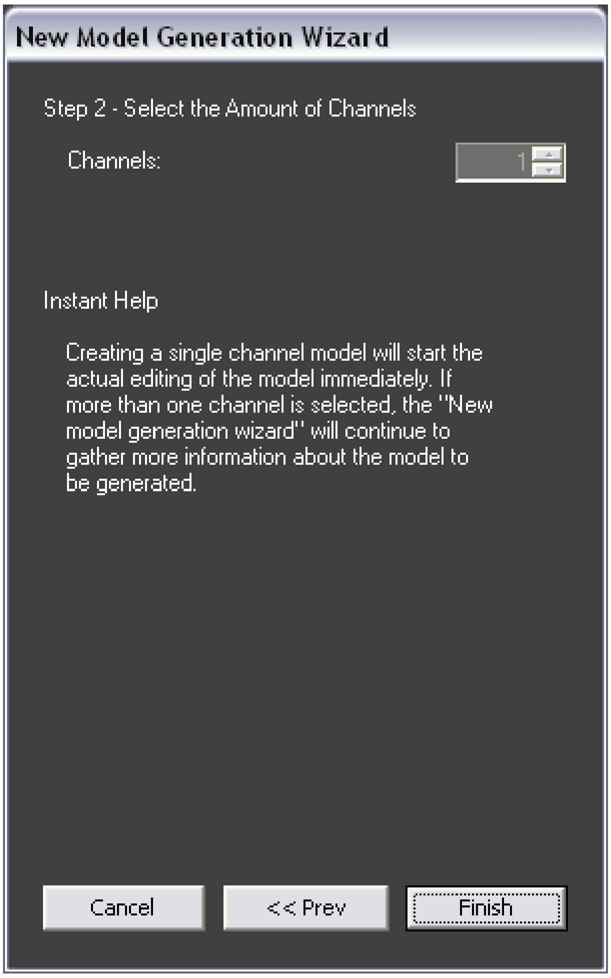


Figure 104. Number of channels

In case of a MIMO or Geometric model the wizard asks for the number of Transmitters and Receivers.

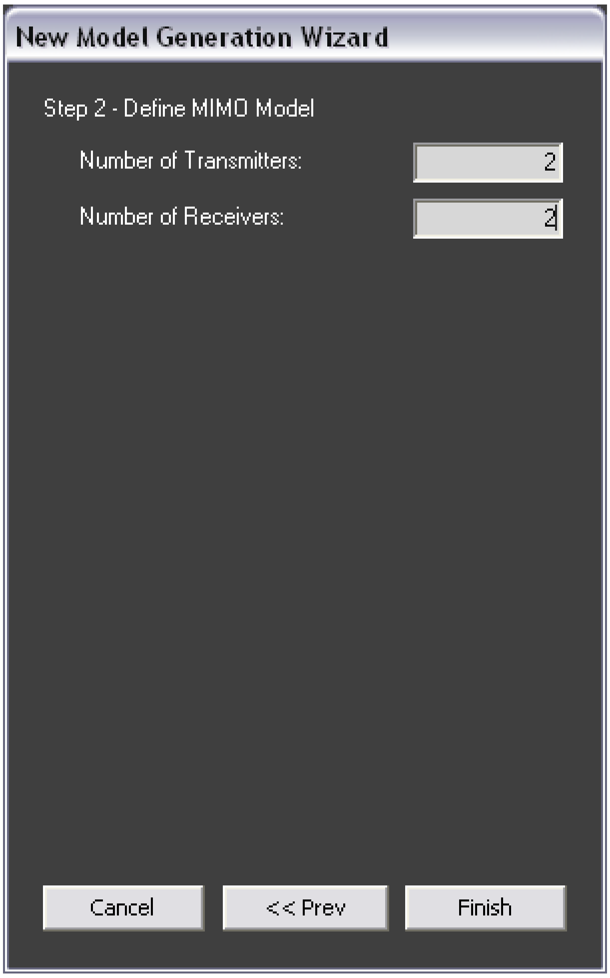


Figure 105. MIMO model definition

If Geometric Model case is selected, the antenna array coordinates must be given or loaded from an already existing .ARR file, see Figure 106. This is done for both transmitter and receiver arrays, if they are specified to contain more than one antenna element.

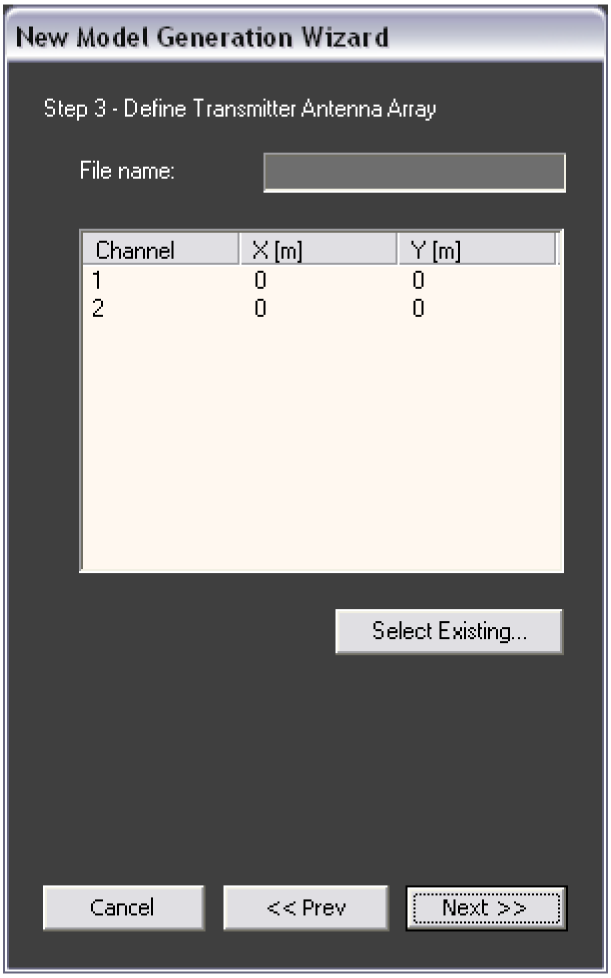


Figure 106. Antenna array definition

4.6.3 Channel Impulse Response Graph

The Channel Impulse Response graph of Channel model view consists of two windows. The lower window is fixed to show the whole impulse response with all existing taps and the upper window the zoomed section of the whole IR.

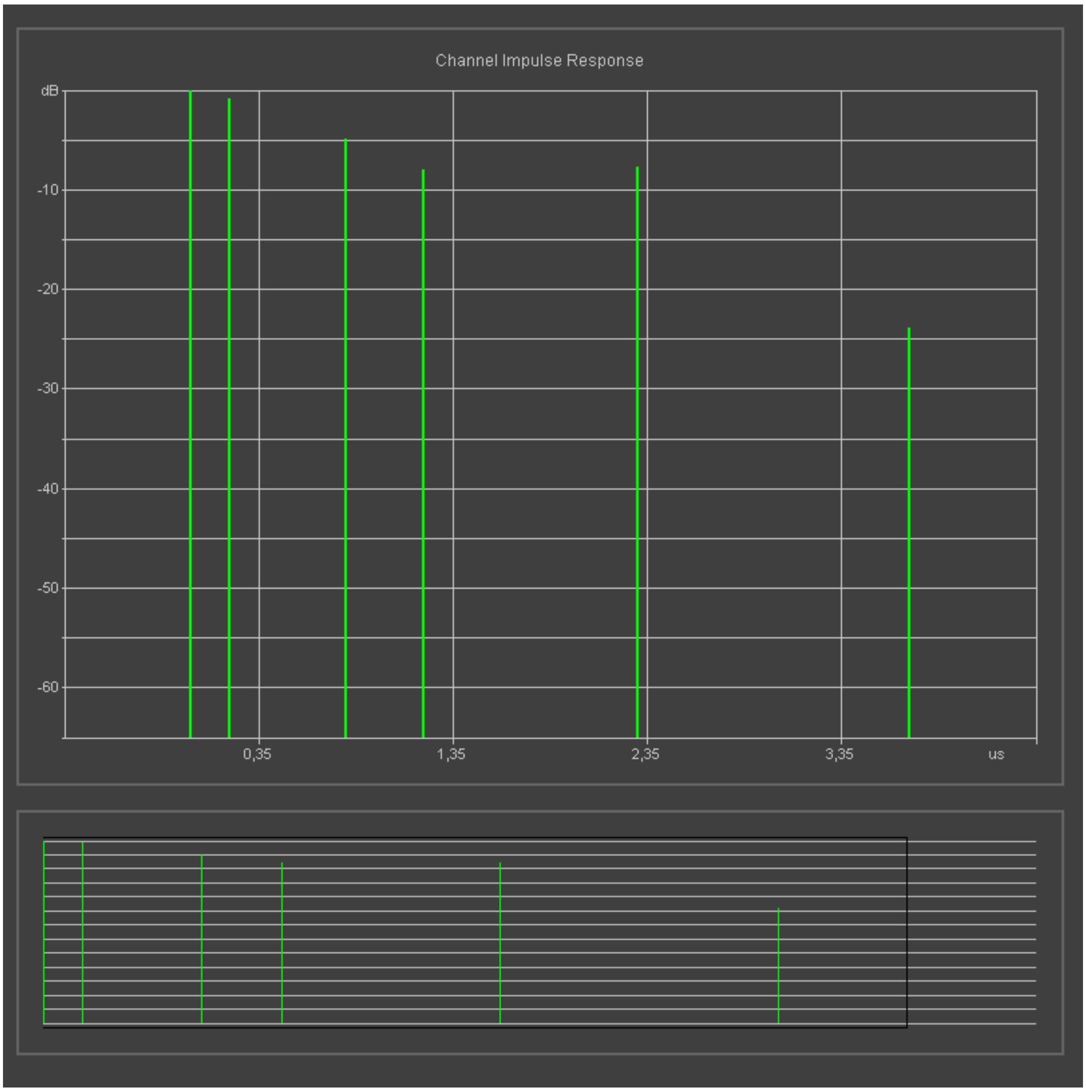


Figure 107. The channel impulse response graph

4.6.4 Channel model settings – Model parameters

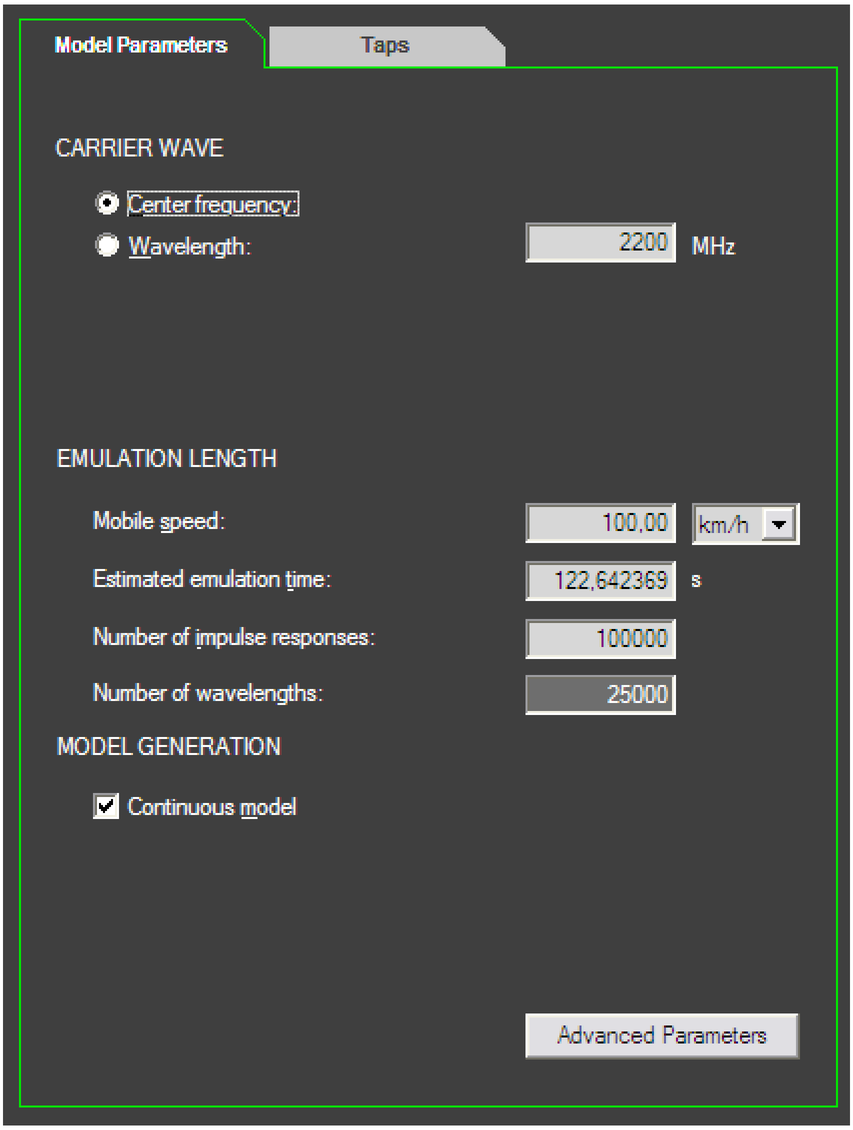


Figure 108. Model parameters

Carrier Wave

The carrier wave parameter can be given either as

- Center frequency [MHz] or
- Wavelength [cm]

(Sampling)
(Sample density / Sample distance)

- Fields are visible when opening models generated earlier with sample density other than 2. Default sample density is 2. Using greater sample density doesn't improve fading quality but increases the model size and building time (and decreases available model length). HW interpolator increases the sample density when emulation is running.

Emulation Length

Mobile speed

- Specifies the emulated mobile speed.

Estimated emulation time

- Estimated emulation time (length), based on number of impulse responses and mobile speed.

Number of impulse responses

- Number of impulse responses in emulation.

Number of wavelengths

- Number of wavelengths in emulation, based on the number of impulse responses and sample density.

Model Generation

Continuous model

- The amount of CIRs is adjusted so that the channel model is continuous from the last CIR to first.

Advanced Parameters

Channels

- Shows the number of channels.

Distribution seed

- Set the channel model seed. When the same channel model is re-generated with the same seed, the result is exactly the same as the original. If the seed is changed, new channel(s) are not correlating with the original ones.

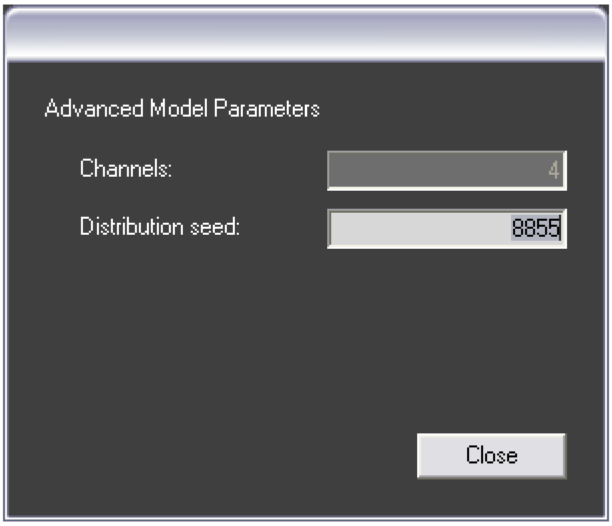


Figure 109. Advanced Model Parameters dialog with one single-channel channel model

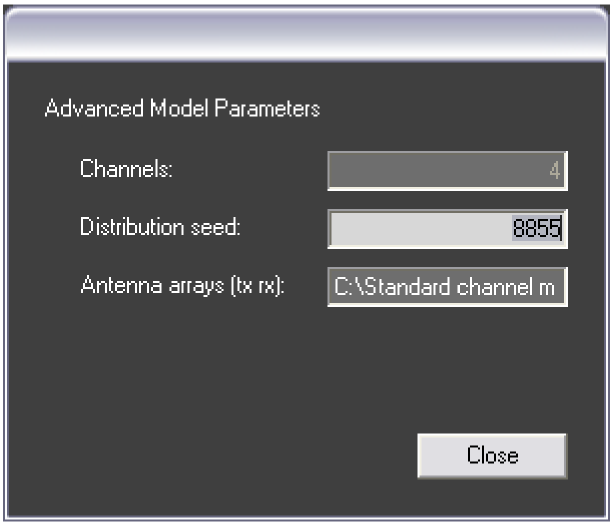


Figure 110. Advanced Model Parameters dialog in Geometric Model mode

4.6.5 Channel model settings – Taps

The Taps tab has a list of channel model taps with three columns:

- **Delay** shows the delay of the tap. Text "Rand" indicates hopping delay.
- **Mean Amplitude** shows the mean amplitude of the tap.
- **Fading** drop-down list allows the user to change distributions. If the distribution function requires parameters, defaults are used.

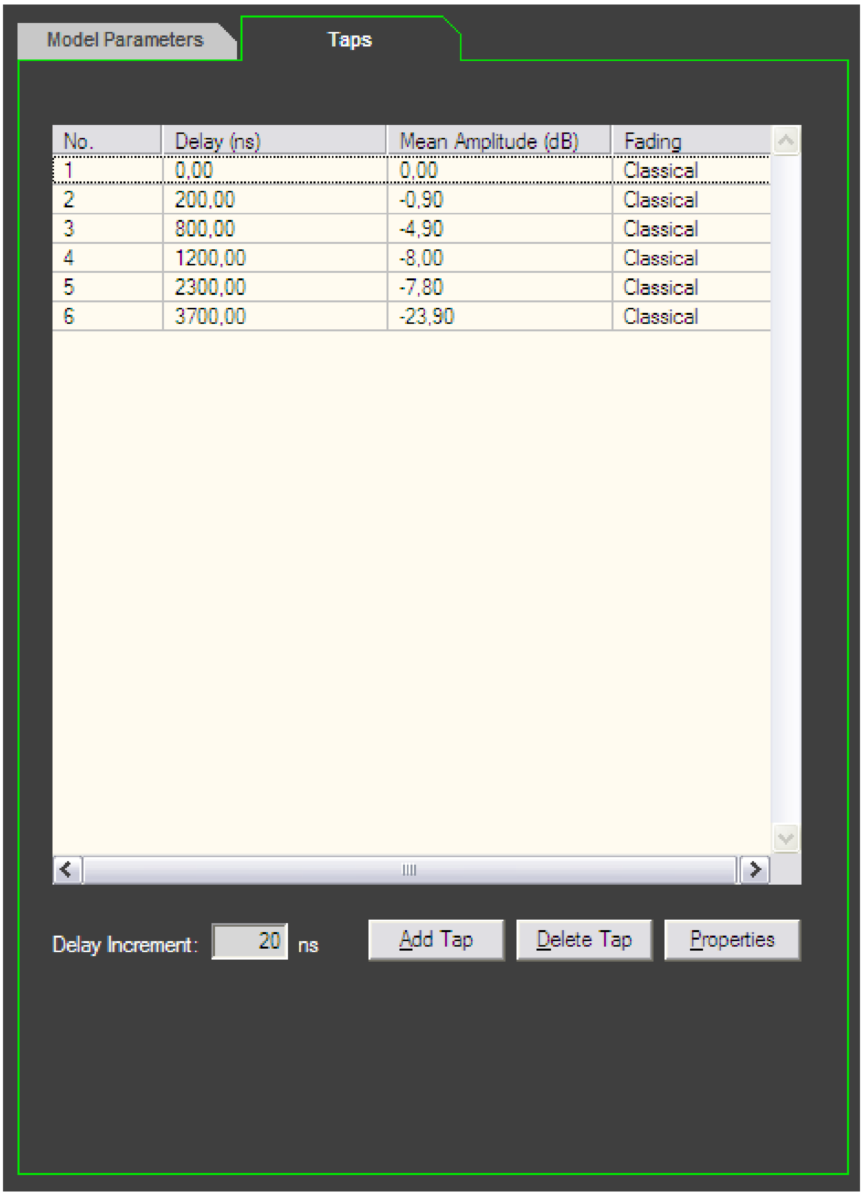


Figure 111. Channel model tap list

Delay increment

- Delay increment for next tap

Add Tap control adds one tap, i.e. one row, to the tap list and sets the default parameters for it:

- Delay: next multiple of Delay increment
- Mean amplitude level: 0 dB
- Distribution: Classical

Delete Tap removes currently selected tap from the list.

4.6.5.1 Tap Properties

Clicking Properties in channel model tap list opens Tap Properties dialog.

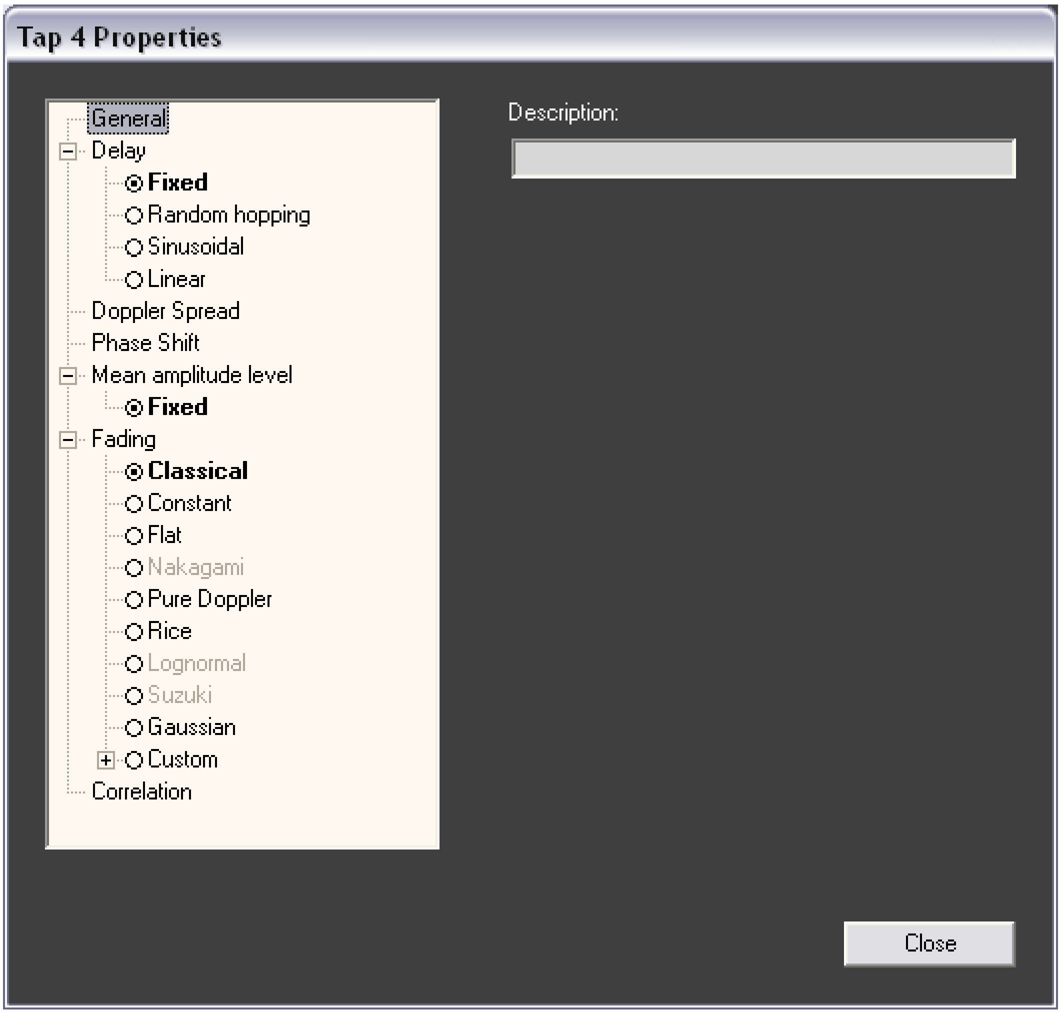


Figure 112. Detailed tap properties dialog

The Properties dialog title shows the number of the selected tap. When this dialog is open user can select taps on the grid and edit properties.

User can set different properties in the list. The values are as follows:

- General
  - Description field (only informative)
- Delay
  - Delay function indicates the function according to how the delay behaves. Alternatives are: Fixed, Random Hopping, Linear and Sinusoidal.
- Doppler Spread
- Phase Shift
- Mean amplitude
- Fading
- Correlation (appears in MIMO and correlating multi-channel cases)
- Geometric Model (appears in geometric multi-channel case only)

If a property is selected and Enter is pressed, the controls related to the active property appear on the right side of the dialog. Controls in the properties are described in the following subchapters.

4.6.5.1.1 Delay

Fixed

With fixed delay, the delay value of the current tap is fixed to the same value during the whole emulation period.

- Parameters:
  - Delay [ns]

Random Hopping

In random hopping the user defines the delay positions, which are repeated throughout the emulation. Each hopping tap actually generates a pair of hopping taps. Only one tap is moved at a time and a new time position is selected so that it is neither of the previous ones.

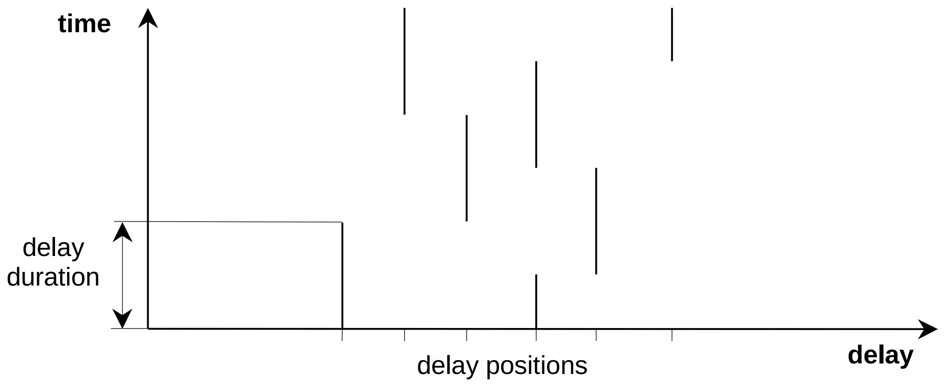


Figure 113. Random hopping delay parameters

- Parameters:
  - Delay duration as milliseconds. The period for keeping taps on one delay position. Note that when lifetime expires, only one tap is moved at a time. This way both taps have an "individual" period of 2x lifetime
  - Table of discrete delay values for tap positions

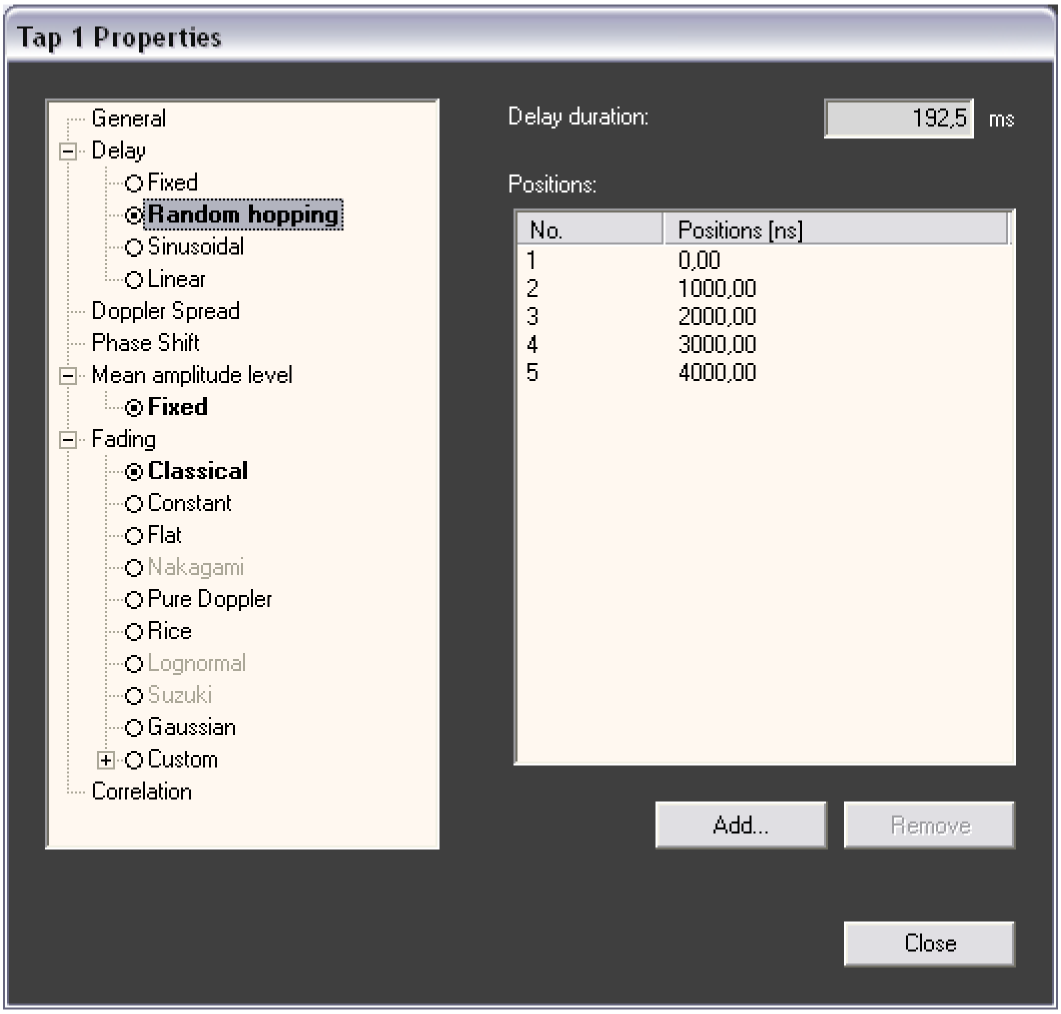


Figure 114. Random hopping properties

Sinusoidal

With sinusoidal delay, the delay of the current tap slides between the defined minimum and maximum delays sinusoidally. The start point of sinusoidal sliding tap is in the middle of the sliding area. Speed of sliding can be adjusted with the period time, which is the time when the tap has been slid a full period from beginning to end.

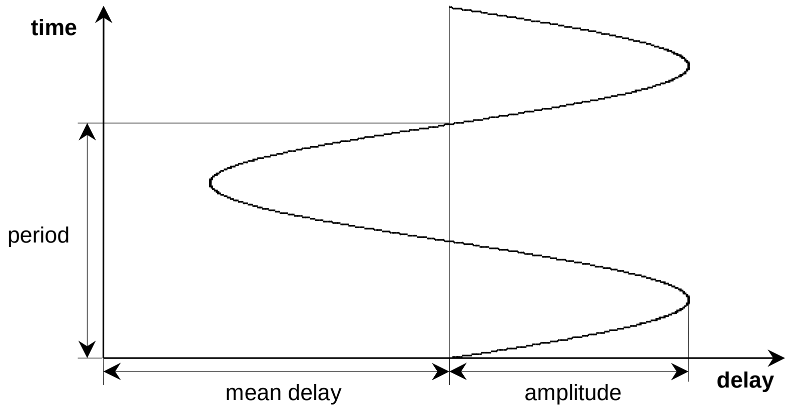


Figure 115. Sinusoidal delay parameters

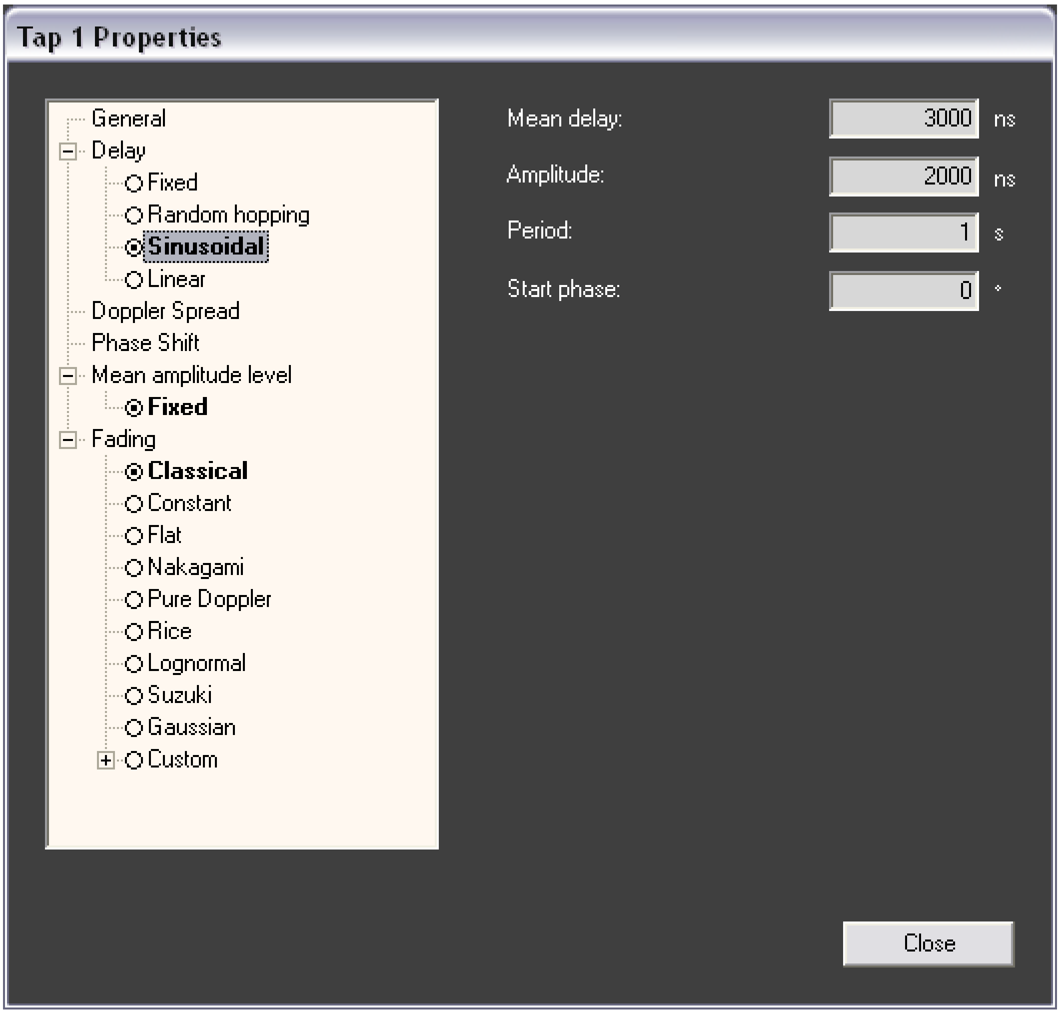


Figure 116. Sinusoidal delay form

Linear

With linear delay, the delay of the current tap slides between the defined minimum and maximum delays linearly. The start point of a linearly sliding tap is the defined minimum delay. Speed of sliding can be adjusted with the period time, which is the time when the tap has been slid a full period from beginning to end.

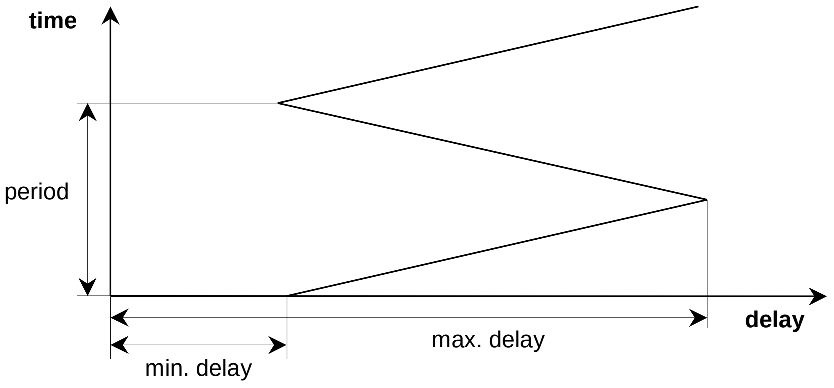


Figure 117. Linear delay parameters

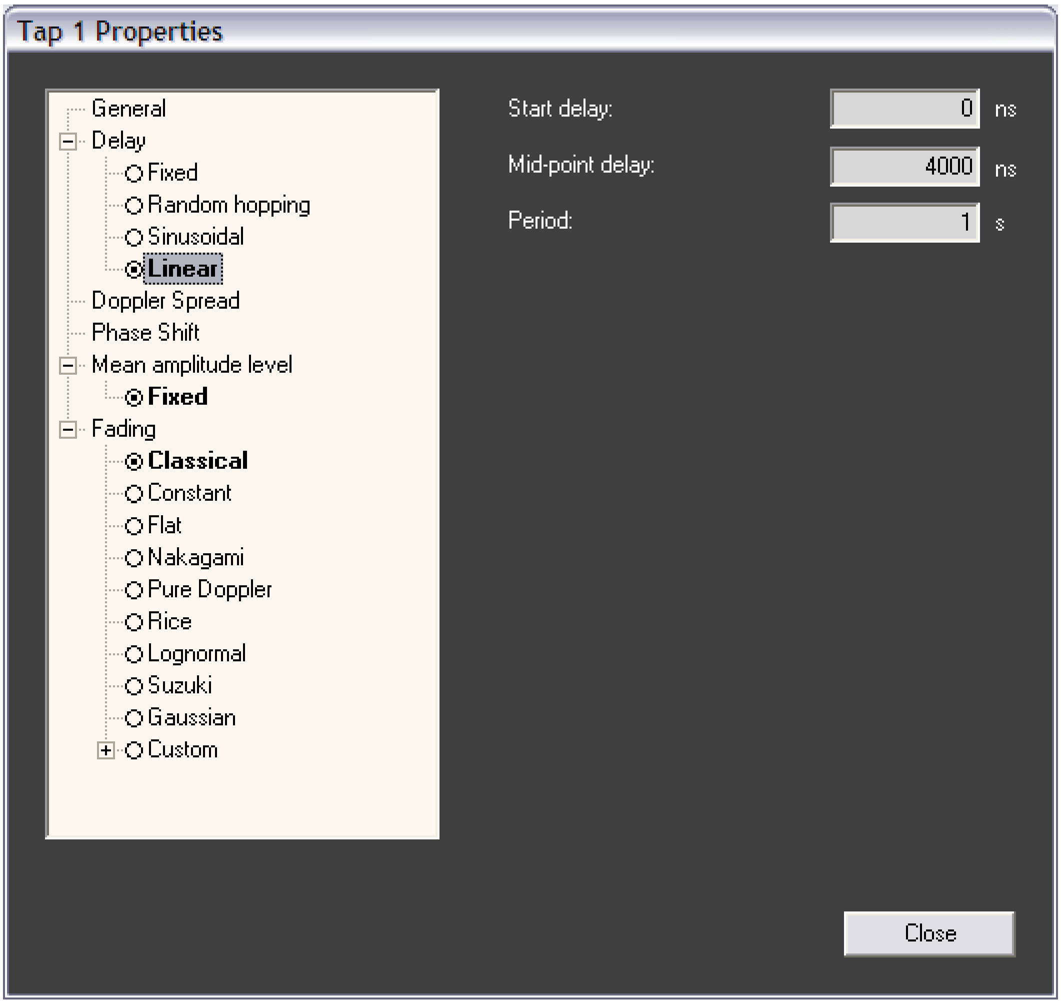
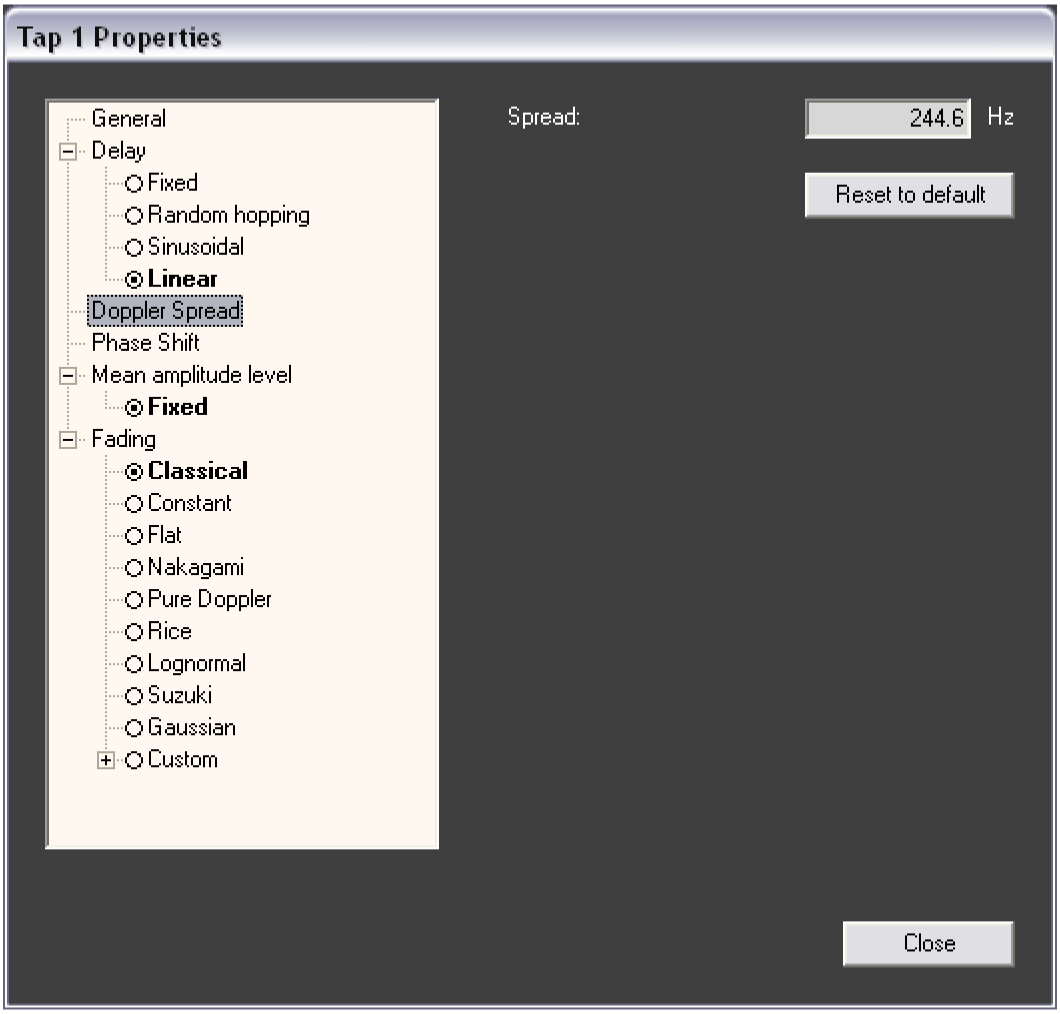


Figure 118. Linear delay form

4.6.5.1.2 Doppler Spread



4.6.5.1.3 Phase Shift

Constant phase shift can be applied to any tap, in addition to other fading parameters. The phase shift is given in degrees.

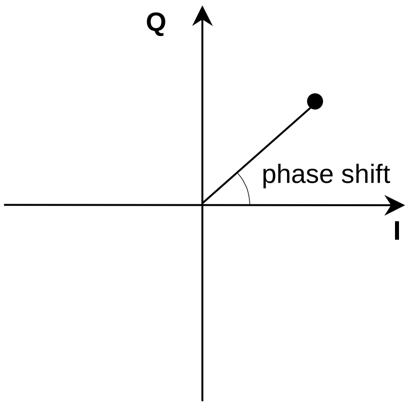


Figure 119. Phase shift

4.6.5.1.4 Mean amplitude level

The amplitude of a tap in a specific channel impulse response is determined by two factors:

- Fast-fading characteristics determine the short term variations in signal level.
- Overall mean amplitude determines the level over several fast-fading cycles.

The long term behavior of individual taps can be specified with mean amplitude level. Currently only a fixed mean amplitude can be specified. The specified amplitude is relative to all other taps in all channels that belong to the emulation. For example, in a case of two channel emulation, the real tap amplitudes depend on tap amplitudes of both channels: the relative differences between channels are preserved.

4.6.5.1.5 Fading

The user has to select one of the radio channel fading models shown under category. More information about fading types, see *Wireless Propagation Environment* section of the User Reference documentation.
Fading type cannot be selected in geometric model case.

Classical

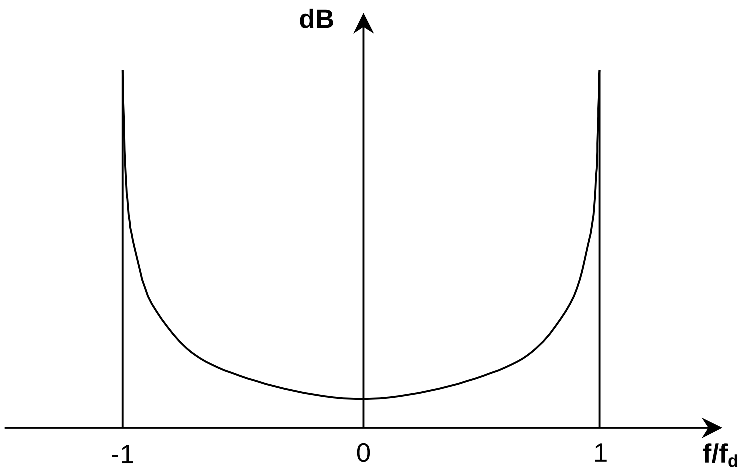


Figure 120. Doppler spectrum of classical fading model (Jakes)

No parameters.

Constant

No fading is applied: tap has constant amplitude.
No parameters.

Flat

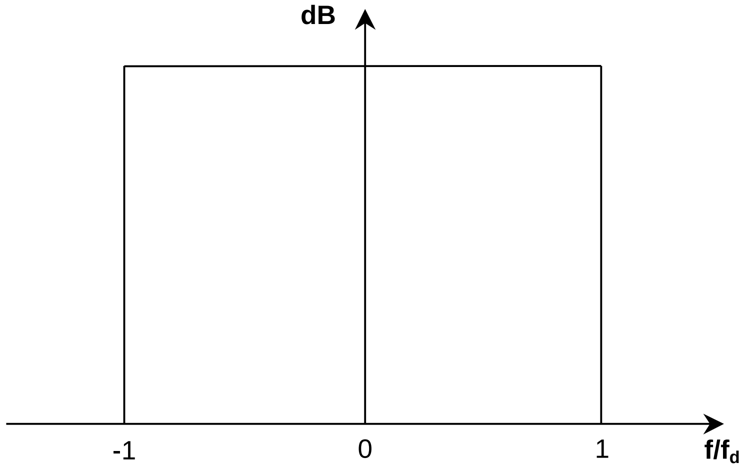


Figure 121. Doppler spectrum of flat fading model

No parameters.

Nakagami

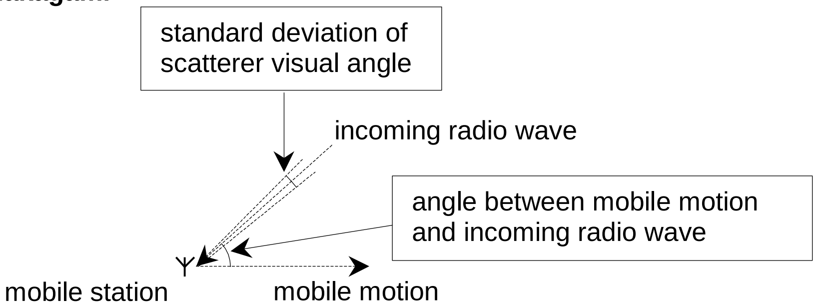


Figure 122. Nakagami parameters

Parameters:

- Angle between mobile motion and incoming radio wave [deg]
- M-parameter, describes the severity of fading, the bigger the M-parameter is the channel is less fading. More information can found in *Wireless Propagation Environment* section of the User Reference.
- Standard deviation of scatterer visual angle [deg]

Pure Doppler

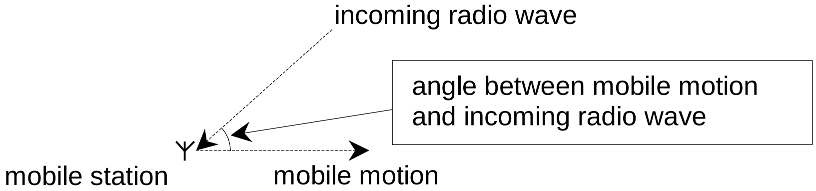


Figure 123. Pure Doppler parameters

Parameters:

- Angle between mobile motion and incoming radio wave [deg]

Rice

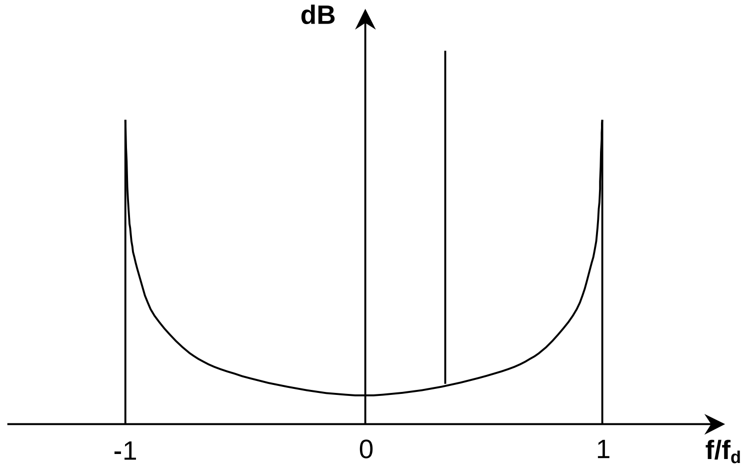


Figure 124. Doppler spectrum of rice fading model

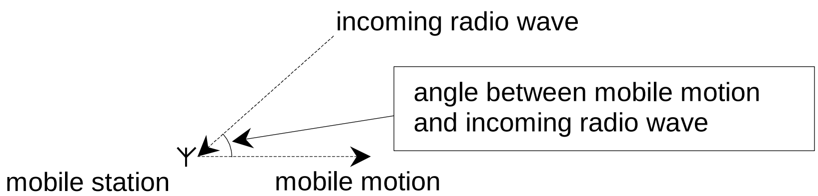


Figure 125. LOS component parameters with rice distribution

Parameters:

- Angle between mobile motion and direct ray [deg]
- Power ratio direct / scattered [dB]

Lognormal

Parameters:

- Standard deviation [dB], describes the amount of fading. Bigger values produce deeper fading.
- Correlation length [m], describes the time correlation of the channel model. After this length correlation is low, i.e. the channel has changed a lot.

Suzuki

Parameters:

- Standard deviation [dB]
- Correlation length [m]

Gaussian

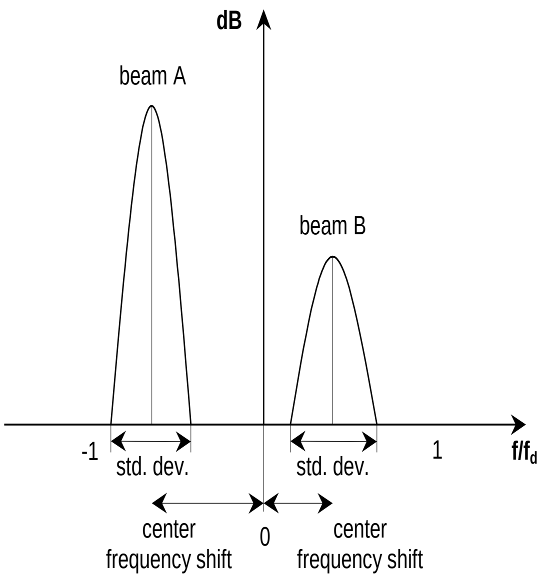


Figure 126. Doppler spectrum of gaussian fading model

Parameters:

- Beam A center frequency shift
- Beam A standard deviation
- Beam B center frequency shift
- Beam B standard deviation
- The power ratio of the beams A and B [dB]

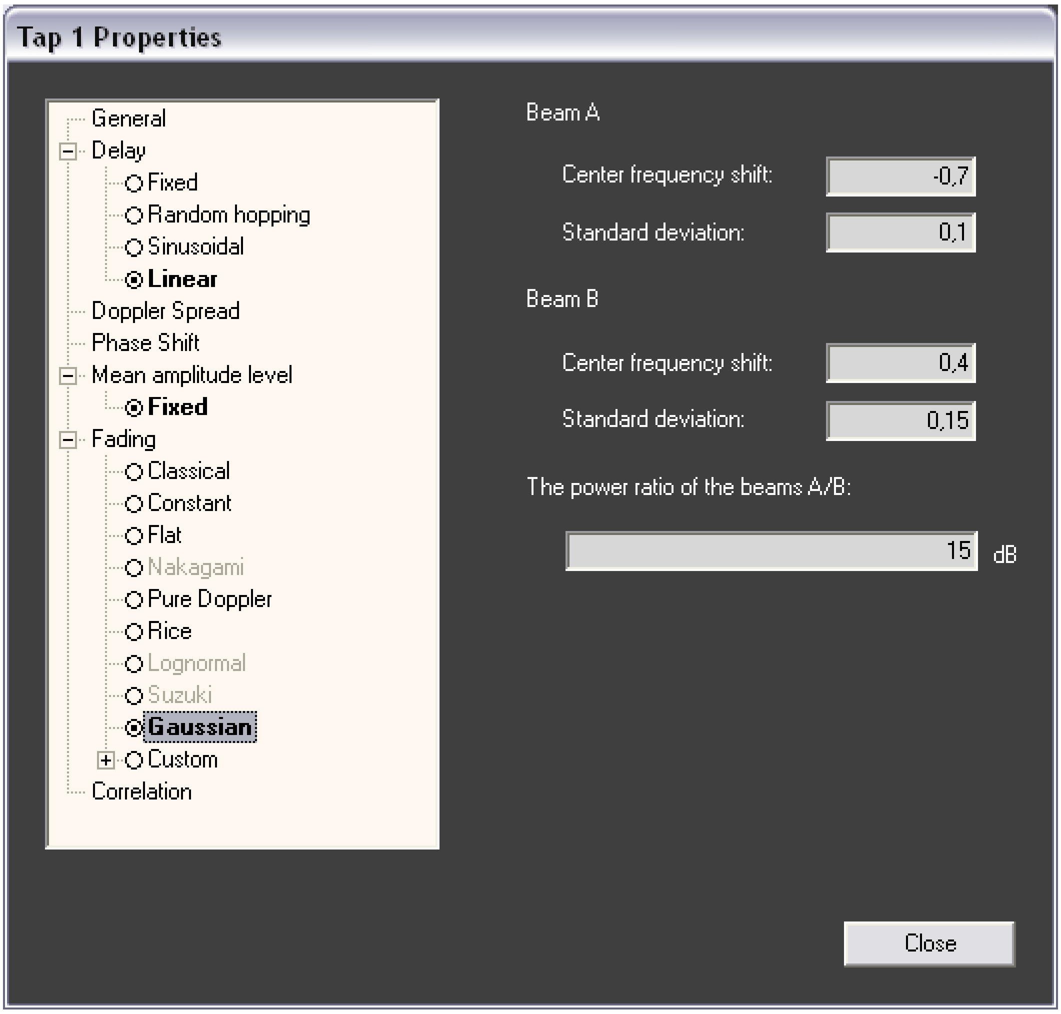


Figure 127. Gaussian fading form

Custom

Amplitude distributions:

- Constant
- Rayleigh
- Rice

Doppler spectrums:

- Pure Doppler
- Jakes
- Gaussian
- Flat
- Butterworth
- Rounded

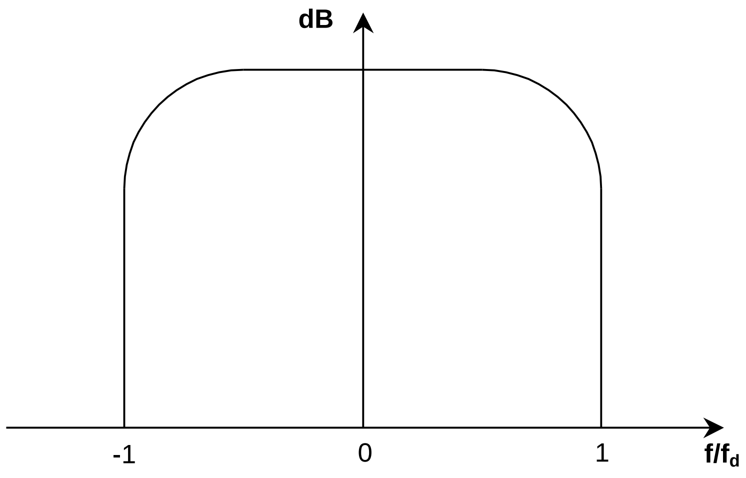


Figure 128. Rounded Doppler spectrum

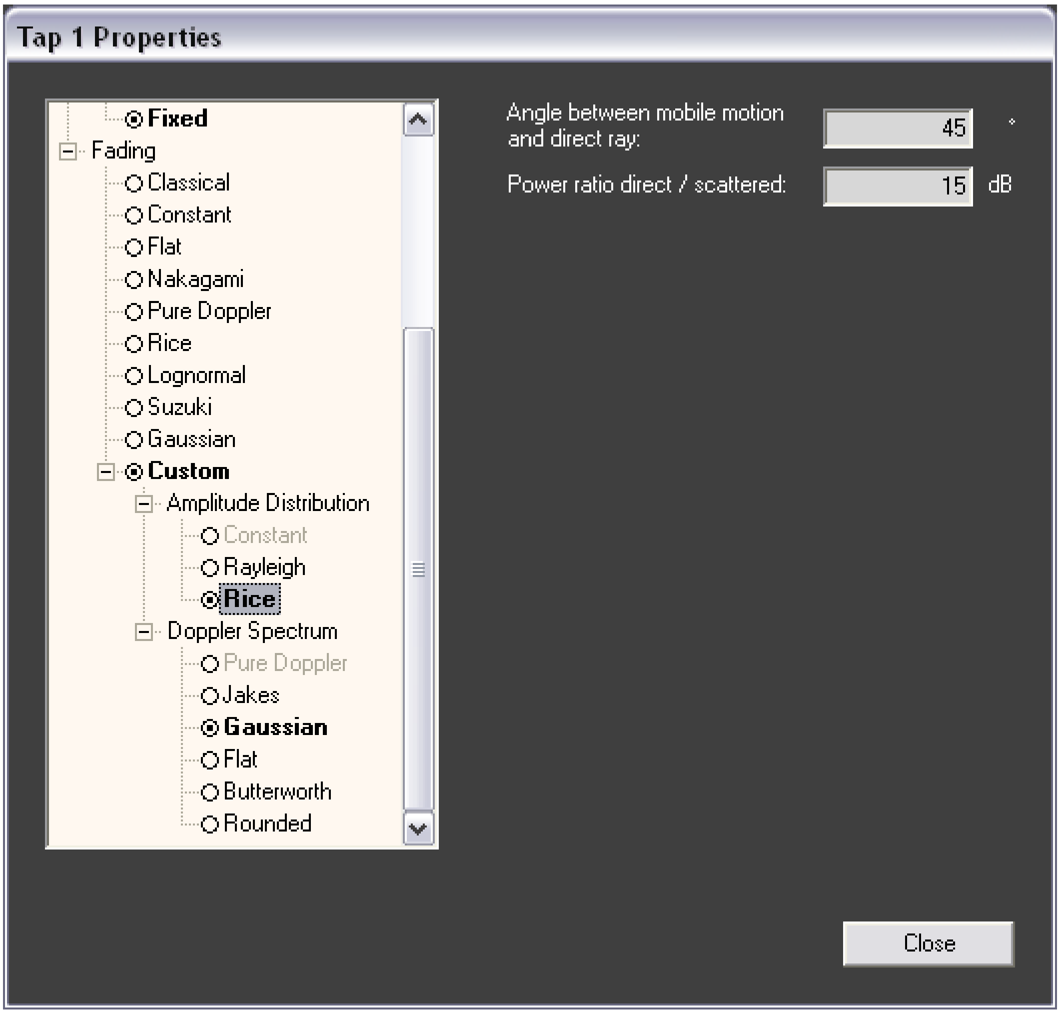


Figure 129. Custom fading (Rice with gaussian doppler)

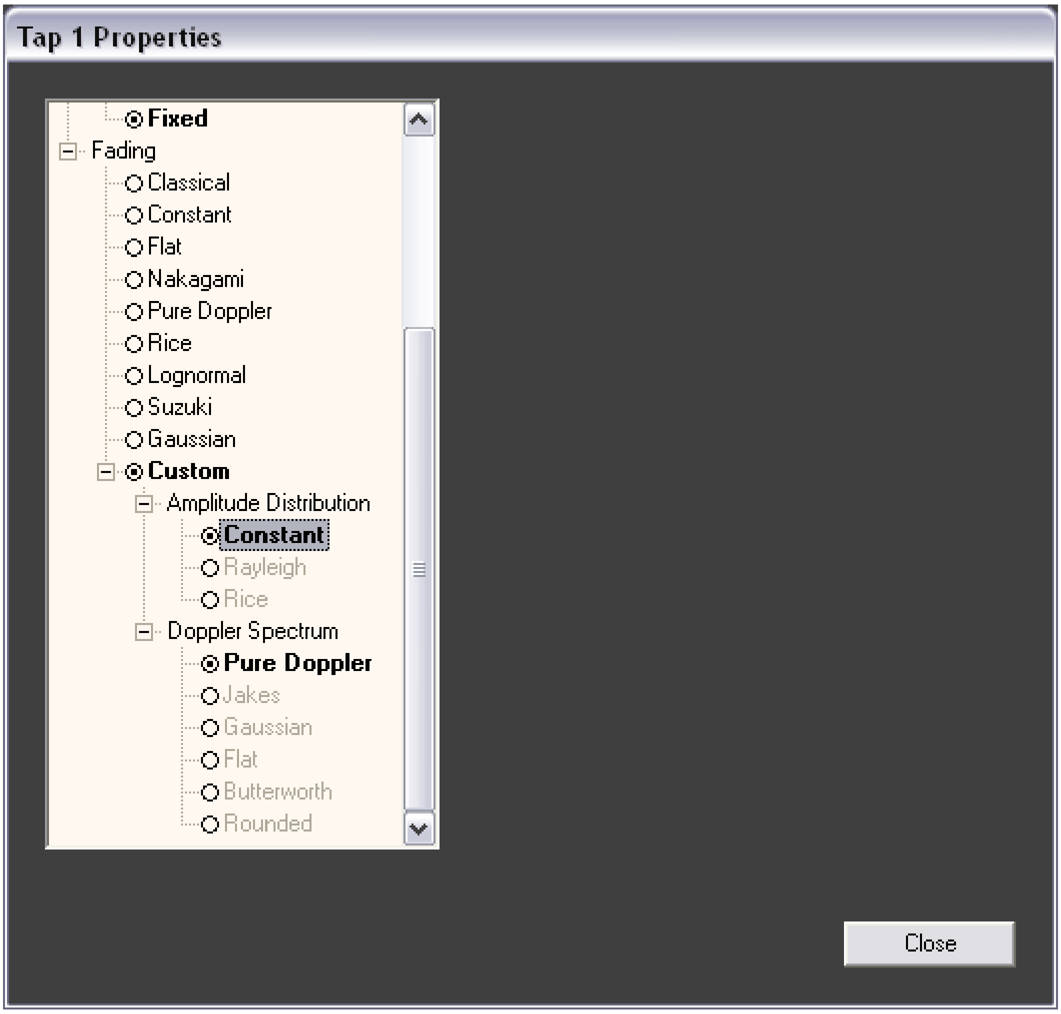


Figure 130. Custom fading (Constant)

Geometrical Model

These parameters are available if the current model is a geometrical multi-channel model.

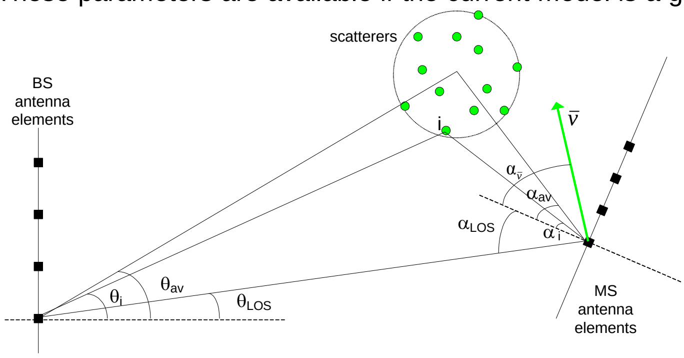


Figure 131. Direction of arrival parameters

Geometric model parameters

- Mean DoD, mean Direction of Departure from transmitter array
- DoD spread (if distribution is non-uniform), how much variance there is in DoD ang
- DoD distribution, how spread is distributed.
- Mean DoA, mean Direction of Arrival to receiver array

- DoA spread (if distribution is non-uniform), how much variance there is in DoA angle
- DoA distribution, how spread is distributed
- Mobile heading, defining to which receiver array is going
- Number of scatterers
- LOS (Line-of-Sight) component parameters (if enabled)
- Transmitter LOS direction, to which direction from transmitter LOS is
- Receiver LOS direction, from which direction receiver LOS comes
- Ricean K factor of LOS component

DoD defines the mean direction of departure of signal from the transmitting antenna array towards the scatterers. DoD distribution and DoD spread define the distribution type of variance from mean signal, and how large variance is. Available types are uniform (there is no variance), gaussian and laplacian distributions.

DoA parameters define same parameters as DoD parameters, but for receiving (mobile) antenna array. Mean DoA is relative to the mobile heading.

If Line of Sight component is enabled, model changes from classical to rician distribution. The transmitter's and receiver's directions define the doppler shift of LOS component, and rician K factor it's relation to power of non-LOS components.

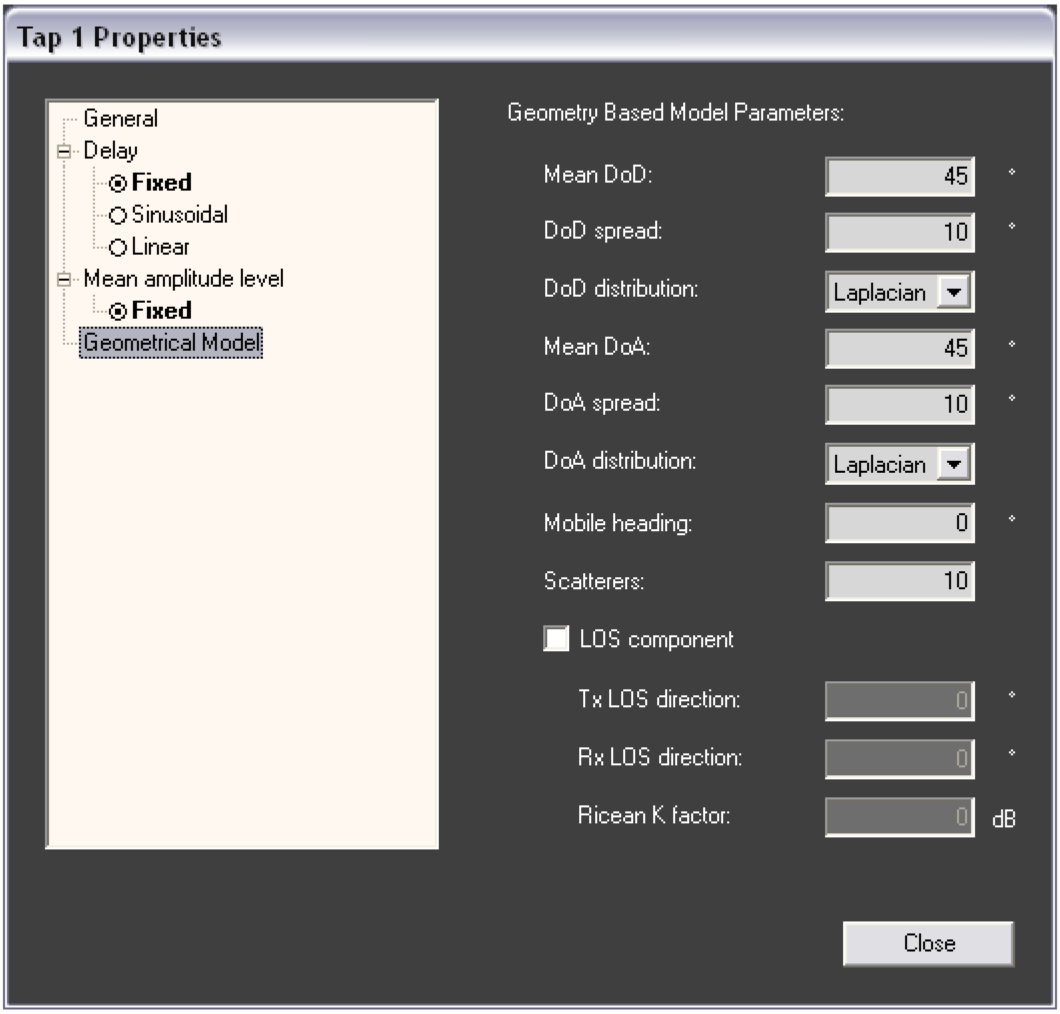


Figure 132. Direction of arrival parameters

4.6.5.1.6 Correlation

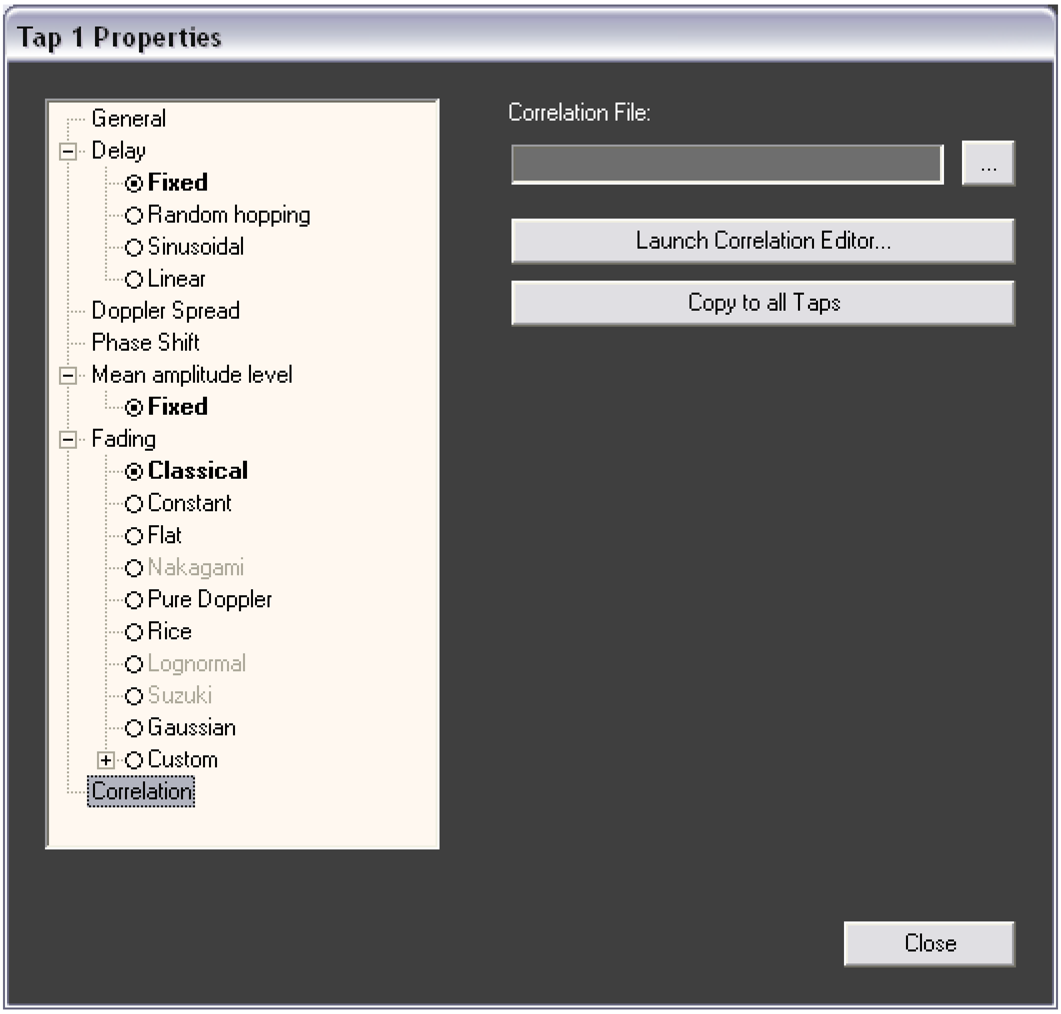


Figure 133. Correlation parameters

These parameters are available if the current model is a correlation matrix multi-channel model (channel amount > 1) or MIMO model.

Correlation parameters:

- File name. Name of the file where correlation matrix is stored
- Correlation matrix can either be selected from existing correlation matrices or created from scratch. If existing matrix is loaded, the type of the model is checked and if correlation file type does not match, editor refuses to load it.

You can create a new correlation matrix by using Launch Correlation Editor pushbutton. By using Copy To All Taps button you can copy single tap's correlation matrix to all taps, if same matrix is used by all channels.

4.7 Correlation Editor

The Correlation Editor (Figure 134) is used to create and edit amplitude or complex correlation matrices. The correlation matrix defines correlation coefficients between channels.

The Correlation Editor can be started from the Advanced Tools menu at the navigation bar or by clicking Launch Correlation Editor… in the Tap Properties dialog of the Channel model view, see Figure 133.

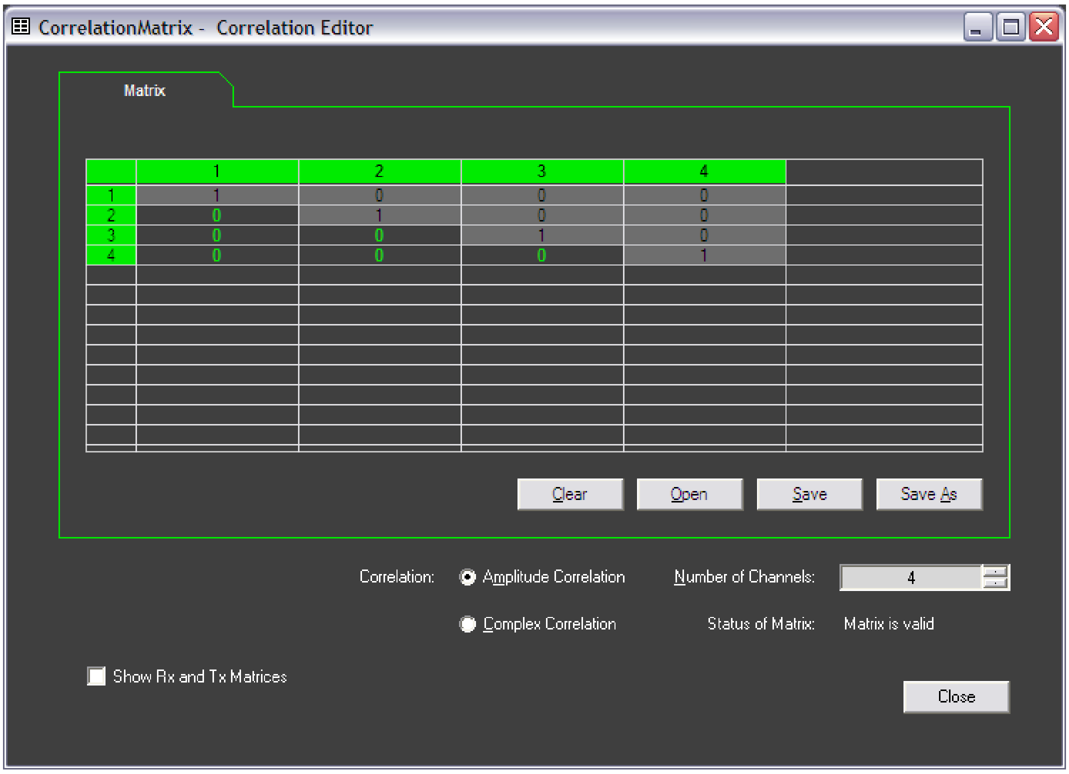


Figure 134. Correlation Editor

Matrix

- The correlation coefficients are viewed and edited in this diagonally symmetric matrix.
- Correlation matrix is Hermitian (N×N), where N is the number of channels. Values on the diagonal [c_{11} c_{22} c_{33} c_{44}] are '1' and in general case other entries are complex. Therefore, it is enough to set the correlation values to the lower triangular of the matrix. The upper triangular values in Correlation Editor are read-only.

$$C = \begin{bmatrix} 1 & c_{21}^* & c_{31}^* & & c_{N1}^* \\ c_{21} & 1 & c_{32}^* & \cdots & c_{N2}^* \\ c_{31} & c_{32} & 1 & & c_{N3}^* \\ & \vdots & & \ddots & \\ c_{N1} & c_{N2} & c_{N3} & \cdots & 1 \end{bmatrix}$$

- Any coefficient values defined on the matrix will be retained even if the number of channels is increased afterwards.

Correlation

- Either Amplitude Correlation of Complex Correlation can be selected.

- Number format:
  - Amplitude Correlation 0.1234
  - Complex Correlation 0.1234+0.3220i
- The decimal separator can be a '.' or ',' depending on the country settings.
- Note that the editor does not accept complex numbers in Amplitude Correlation mode.

Number of Channels
- The size of a matrix can be set from 1 to 128 channels.
- Note that when Correlation Editor is launched from Channel model view, channel count is locked to number of channels in channel model, and can not be changed.

Status of Matrix
- Indicates whether the matrix entered is valid:

Not Valid	Matrix is not nonnegative definite, which means that correlation between channels is not realistic. Change some of the values, so that the matrix becomes valid.
Valid	Matrix is nonnegative definite, which means that correlation between channels is OK, and the matrix can be used by the TapToIrCompiler.

Show Rx and Tx matrices
- Show or hide Rx and Tx matrices.

Calculate Kronecker
- Kronecker calculation of the result matrix can be enabled or disabled.

4.7.1 Editing the correlation coefficients

To enter or replace a coefficient value, click on the desired item in the editable area (indicated with graphite gray background by default). An edit field and text cursor will appear.

An alternative method is to select the desired line with the up and down arrow keys and then press Enter. Any subsequent press of Enter will select the next editable location. The mouse and key controls are listed in Table 65.

MOUSE	OPERATION
Left click	Select a cell
Double click	Edit a cell

KEY	SELECTION MODE	EDITING MODE
Up	advance cell up	advance cell up
Down	advance cell down	advance cell down
Left	advance cell left	advance char/cell left
Right	advance cell right	advance char/cell right
Enter	edit next cell down	edit next cell down, wrap
Tab	---	edit next cell right, wrap
Shift+Tab	---	edit next cell left, wrap
Escape	no operation	cancel edit
F2	toggle edit mode	Toggle select mode
Ctrl+Tab	no operation	next dialog item

Table 65. Correlation editor mouse and key controls

Notes:

- These commands only apply when the number of channels is 3 or more.
- While in Edit mode, the Left and Right Arrow keys will advance one character at a time until the beginning or end of the text is reached. When there is no more text in the desired direction, the next cell will be edited.

4.7.2 Correlation Editor in MIMO mode

When a MIMO model has been defined in Channel model view the Correlation Editor will open in MIMO mode, see Figure 135. This differs from the normal model so that matrices can be entered for both transmitter and receiver arrays and the editor will calculate correct result as Kronecker product from these matrices.

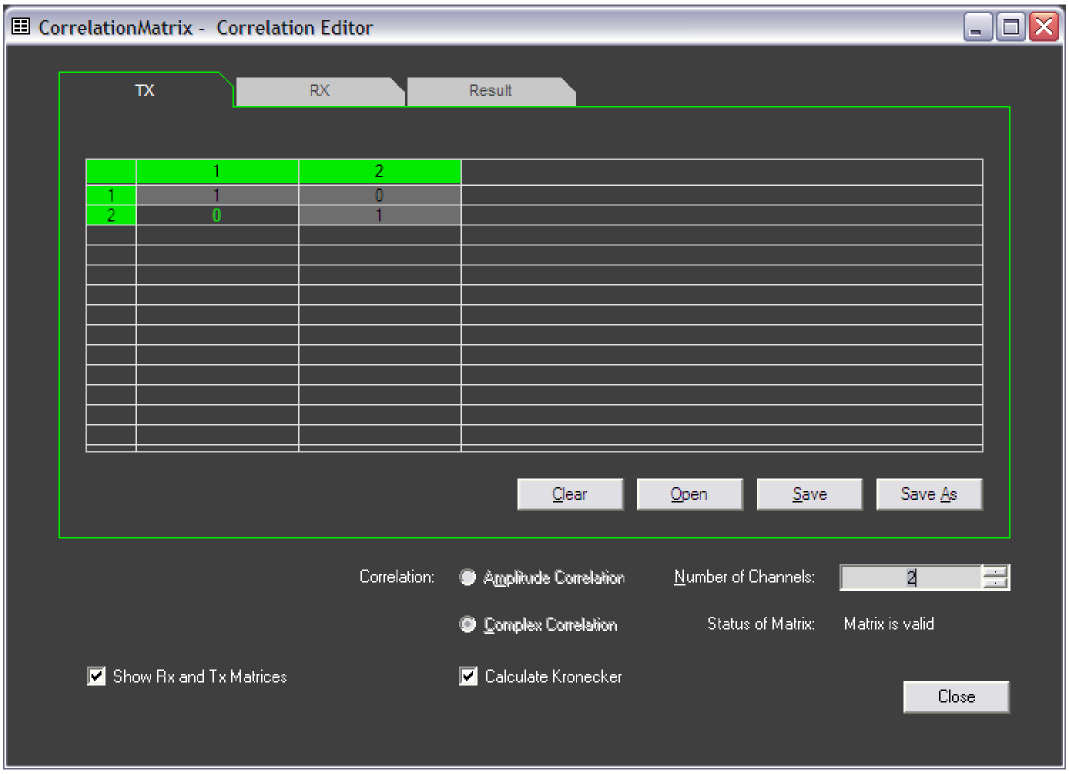


Figure 135. MIMO correlation editing

User can also enter the MIMO mode by opening the Correlation Editor in stand alone mode and by loading a MIMO result matrix saved earlier with the editor. This will load all the three matrices simultaneously and editing can be continued.

The editor has three tabs; Tx matrix, Rx Matrix and Result Matrix. The contents of the Result Matrix are Kronecker Product of the Tx and Rx matrices and it will be recalculated when user switches to the Result Matrix page.

All the matrices can be edited when page view is visible, but the Result Matrix will revert back to calculated values when user switches back to the Result Matrix page.

As a result of Kronecker product calculation, result matrix is special MIMO matrix (when saved to file as .COR file) that contains all the Tx, Rx and Result matrices that can be loaded to the Channel model view. When MIMO matrix is saved, all matrices are saved to the result file. It is not possible to save only Tx or Rx matrices.

Because MIMO matrix file contains three matrices - Rx matrix, Tx matrix and the result matrix - the editor will ask user which of the matrices he wants to load to current page view, see Figure 136. Note that it is not possible to load a normal matrix as MIMO Tx or Rx matrices; loading the matrix will lose MIMO matrix.

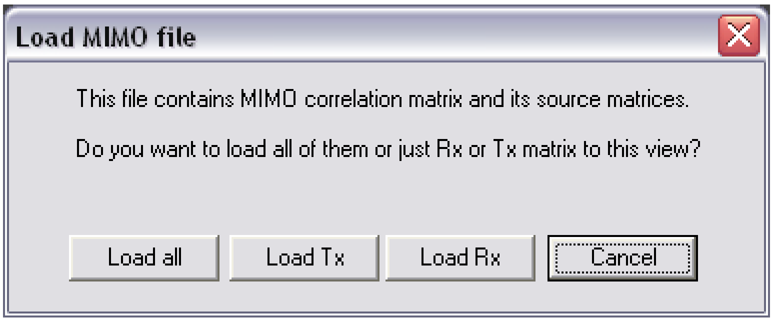


Figure 136. Loading MIMO file

By selecting the Load all option, the editor will load all three matrices and transfer to MIMO mode.

If user changes TX or RX matrix and clicks Result matrix tab user must choose whether to update result matrix or not.

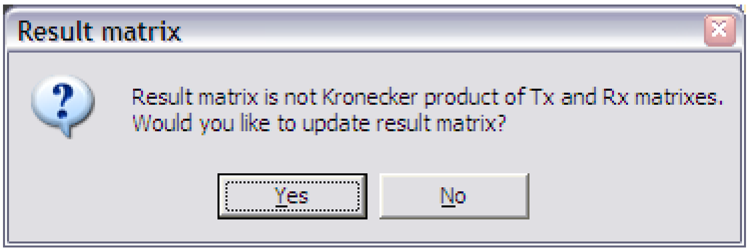


Figure 137. Result matrix update

4.8 Emulation Batch Builder

Emulation Batch Builder (Figure 138) is tool for building multiple emulations at the same time. The Emulation Batch Builder can be started from the Advanced Tools menu at the navigation bar.

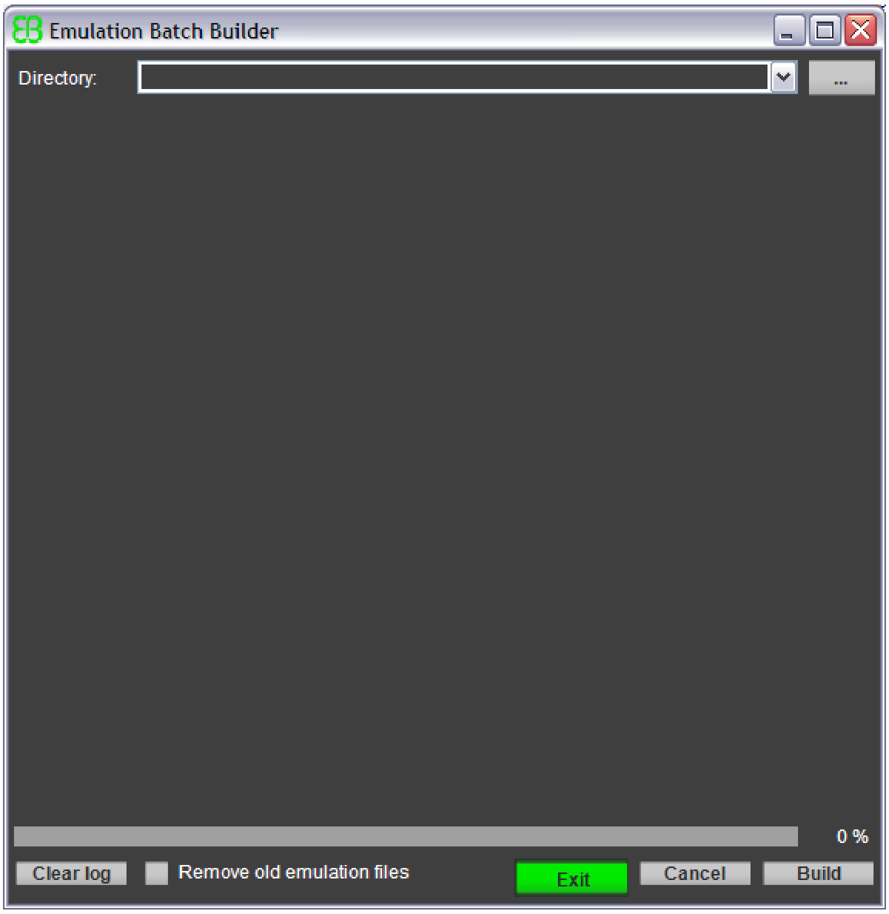


Figure 138. Emulation Batch Builder.

4.8.1 Usage

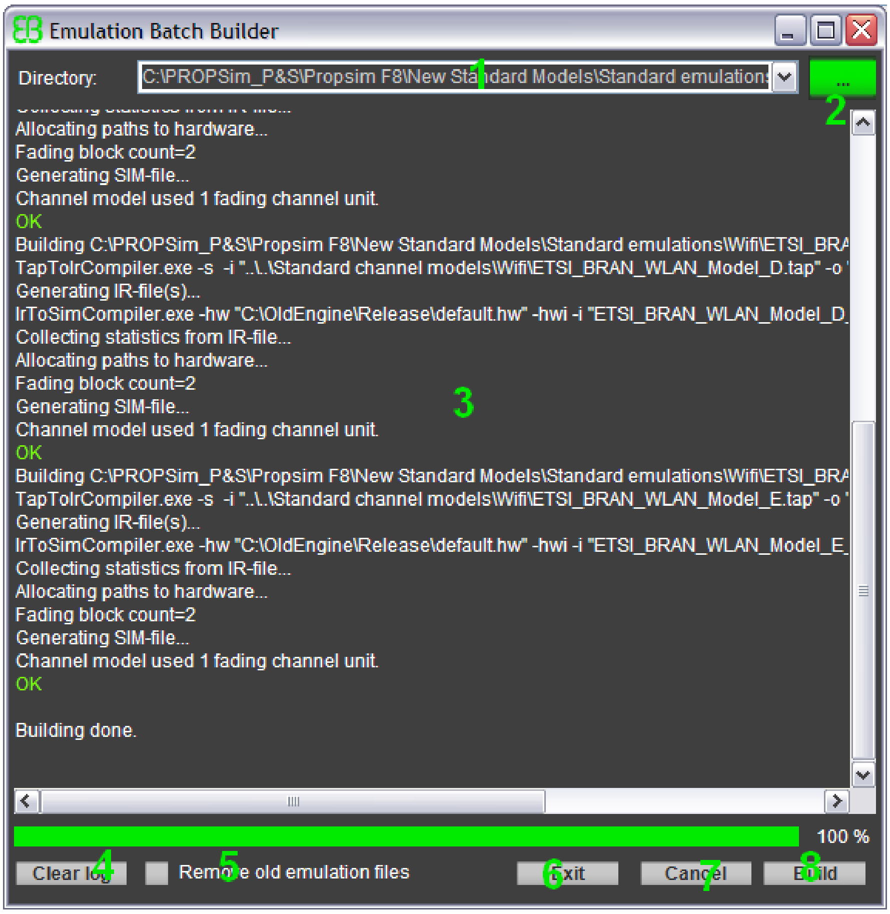


Figure 139. Emulation Batch Builder GUI components.

Emulation Batch Builder GUI components in Figure 139:

1. Currently selected directory. Use drop down menu for directory history.
2. Button for browsing emulation directory.
3. Build log.
4. Button for clearing the build log.
5. Check box for option to delete previous emulation files before building.
6. Button for exiting the application.
7. Button for canceling current build process.
8. Button for starting build process for selected directory.

4.9 Multi Emulator Scaler

EB Propsim scales the output gain of the emulator optimally for each emulation when the emulation is build. The scaling is dependent of the emulator hardware characteristics and the results can thus vary between emulators even for the same emulation. The Multi Emulator Scaler (Figure 140) can be used to scale the output gains of different emulators to match for multi-emulator use.

The emulation files (.SMU) of each emulator in the multi-emulator setup are provided to the Multi Emulator Scaler. The Scaler creates new .SMU files with modified output gain value for each emulation.

Optionally also calibration file (.XML) can be provided for each emulator. The calibration file includes calibration values (dB) for each input and output of the emulator. It can be used e.g. to compensate cable losses for different channels.

The generated .SMU files are stored in the same folder the original .SMU files were stored in. A backup folder is created for the original files “<path to .SMU files>\Backup”.

Multi Emulator Scaler can be started from the Advanced Tool menu at the navigation bar.

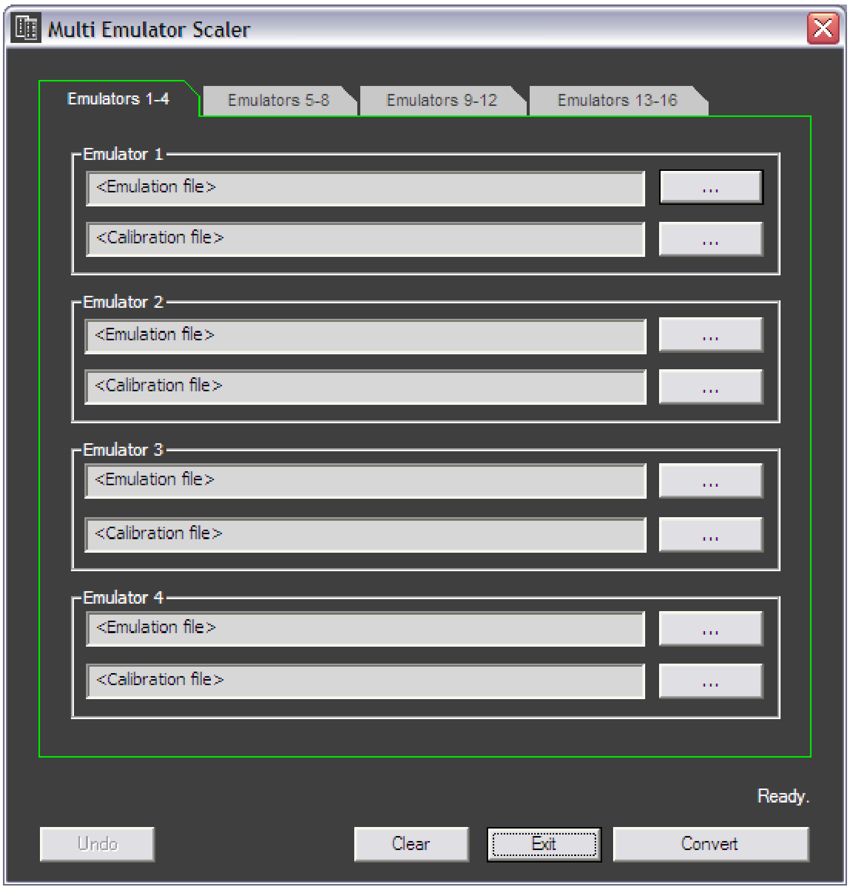


Figure 140. Multi Emulator Scaler

4.9.1 Using scaler tool

Scaler tool can be used in GUI or CLI mode. To start program in CLI mode user must provide emulation files and optional calibration files as a command line parameters. Without command line parameters program starts in GUI mode.

In both operation modes scaler tool writes new .SMU file for each emulation. Original .SMU files are stored inside folder “<path to .SMU files>\Backup”.

4.9.1.1 Command line usage

In Command Line Interface (CLI) mode the user must provide emulation files as command line parameters. Optionally also the calibration files can be provided. If no parameters are provided, the program starts in GUI mode (see Chapter 4.9.1.2).

Scaler tool usage in CLI mode:

```
C:\>Scaler.exe <SMU for emulator 1> <XML for emulator 1> … <SMU for
emulator n> <XML for emulator n>
```

Calibration files (.XML) are optional.

4.9.1.2 GUI usage

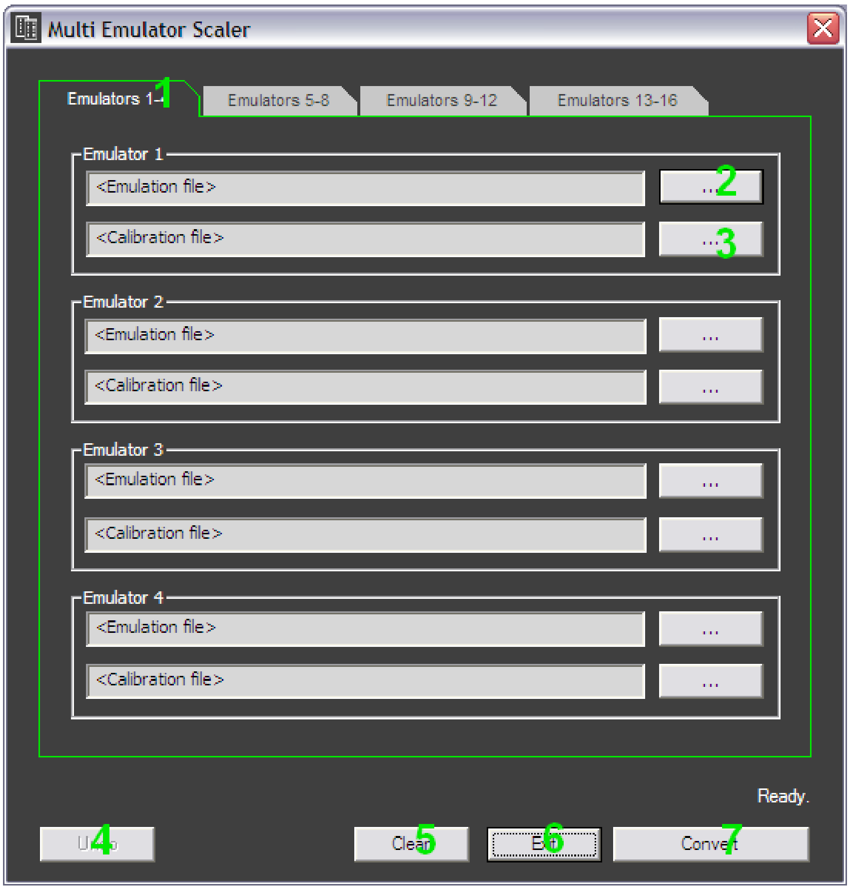


Figure 141. Multi Emulator Scaler GUI components

Scaler tool GUI components in Figure 141:

1. Tab pages for emulations. Each tab page contains input fields for 4 emulators. Currently maximum amount of emulators is 16.
2. Button for browsing .SMU file for selected emulator.
3. Button for browsing .XML calibration file for selected emulator. More information in *File Formats* section of User Reference documentation.
4. Undo converting.
5. Button to clean text boxes.

6. Button to exit application.
7. Button to convert .SMU file.

4.10 User profiles

The menu items in Users menu enable changing the current user, creating new user profiles and deleting existing user profiles.

There are two kinds of user profiles with different usage rights in EB Propsim:

- User with **Basic user** rights can load and run emulations (**Running view**) and use system utilities. All the changeable parameters in **Running view** are editable.

- User with **Advanced user** rights can additionally edit channel models and emulations and create new users profiles.

4.10.1 Create New User

Selecting Create New User opens a dialog for creating a new user profile, see Figure 142.

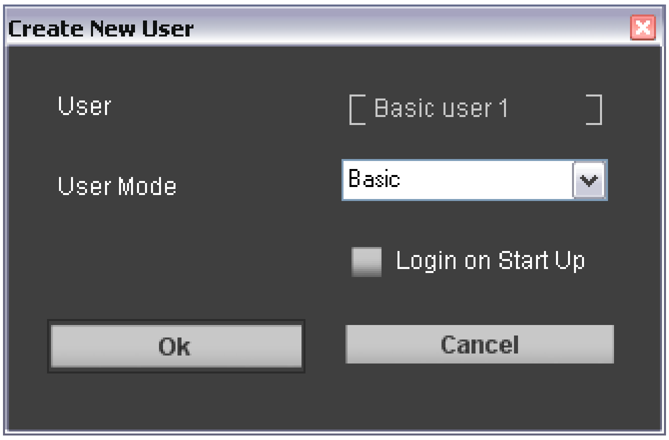


Figure 142. Create New User dialog

User
- Name of the new user profile

User Mode
- Selection of user type for the new user profile
  - Advanced
  - Basic

Login on Start Up
- If selected, the created user profile will be set as the default user at start-up

4.10.2 Change User

Selecting Change User opens a Login dialog, see Figure 143.

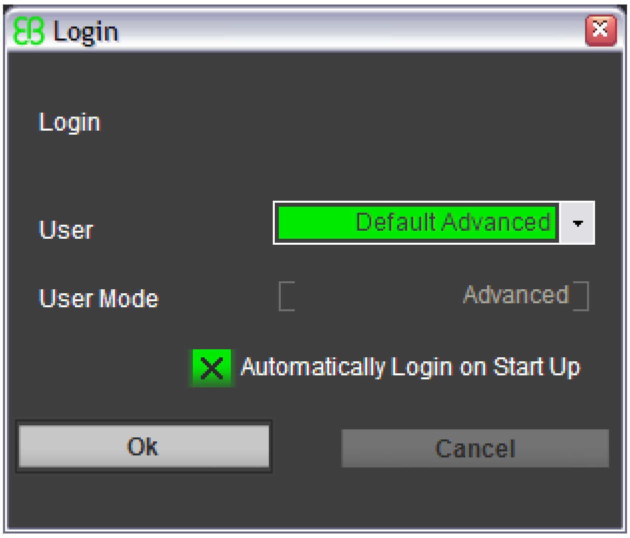


Figure 143. Login dialog

User

- Username can be selected from the drop-down list

User Mode

- Shows the user type (Basic / Advanced) of the selected user

Automatically Login on Start Up

- If selected, the username will be set as the default user at start-up

4.10.3 Delete User

Selecting Delete User opens a dialog in Figure 144.

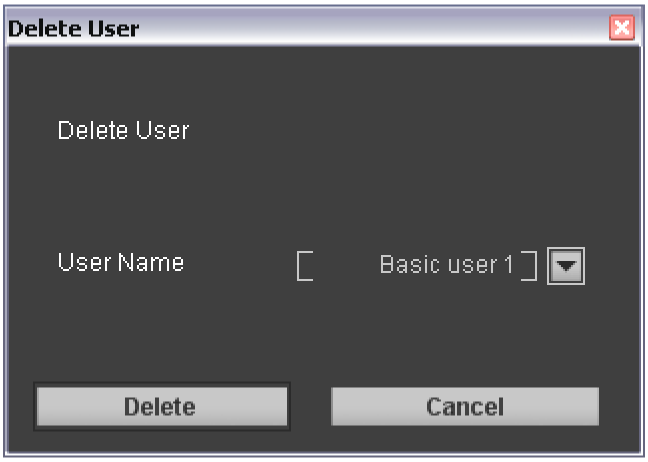


Figure 144. Delete User dialog

User

- User name can be selected from the drop-down list

4.11 System Configuration

Selecting System → System Configuration opens the configuration dialog of the EB Propsim containing e.g. device serial number, calibration status and expiration date plus configuration and options, see Figure 145.

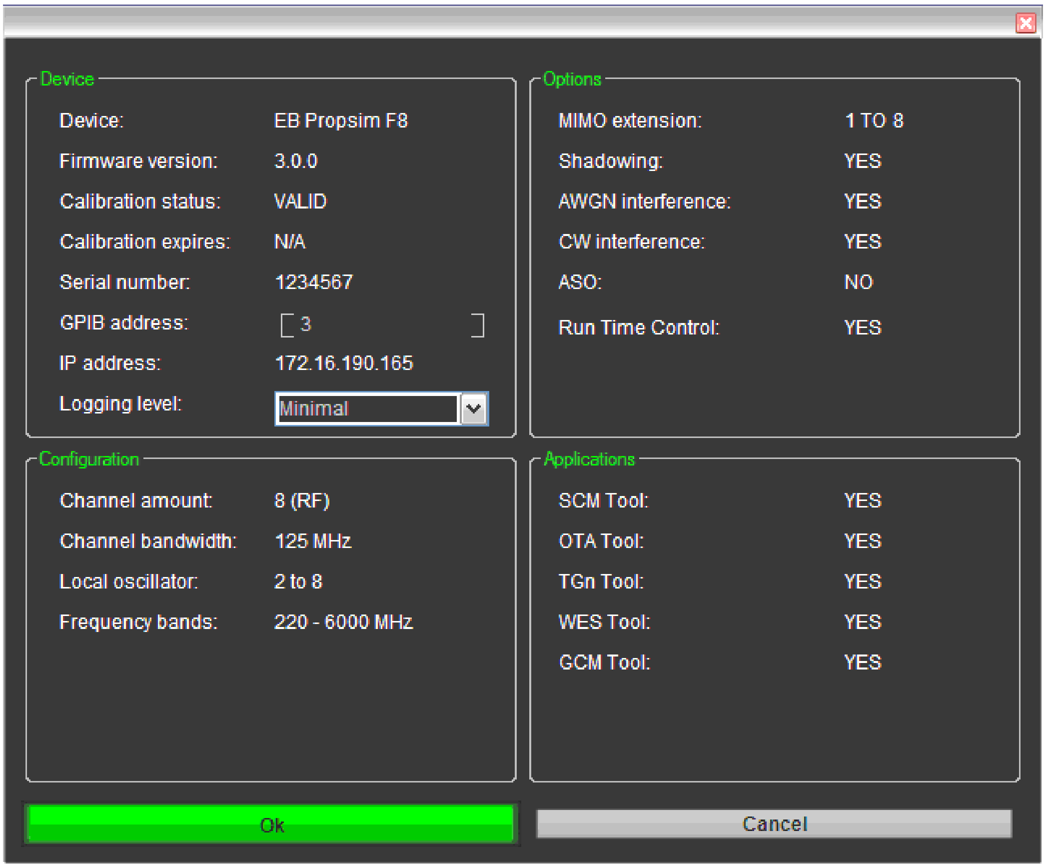


Figure 145. System Configuration

4.12 Restart/Shutdown

The EB Propsim can be shut down either by pressing the Standby-button in the emulator front panel or by selecting System → Restart/Shutdown in the navigation bar, see Figure 146. User can also select to restart the emulator.

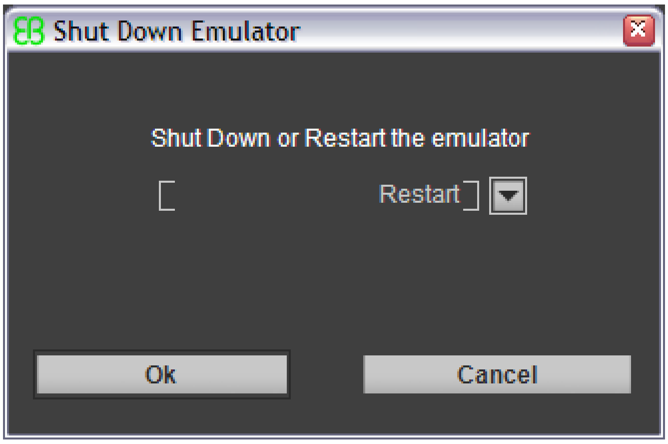


Figure 146. Shut Down Emulator dialog

5 Shadowing (optional feature)

Shadowing is a phenomenon in which a large obstruction such as a hill or a large building obscures the main signal path between a transmitter and a receiver, causing attenuation in the received signal power. The Shadowing option of the EB Propsim enables modeling of the shadowing phenomenon with a variety of functions.

Shadowing profiles can be defined from Scenario Wizard and Editing view.

In Scenario Wizard, it is possible to load an existing shadowing profile or create a new profile. Shadowing profile can be defined for each link from the channel model setup page (see Figure 147 and chapter 4.3.1.3). Selected shadowing profile is applied for all Rx antennas in link.

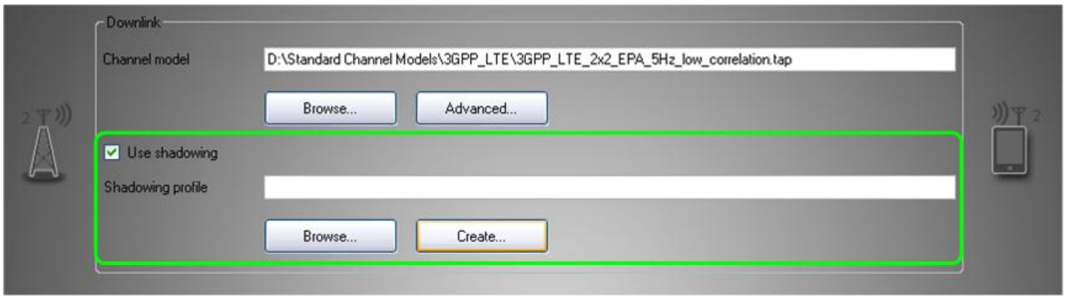


Figure 147. Shadowing profile field in the Scenario Wizard

In Editing view, the shadowing profile to be used in the emulation can be defined in the Output settings page of the Editing view, see Figure 148.

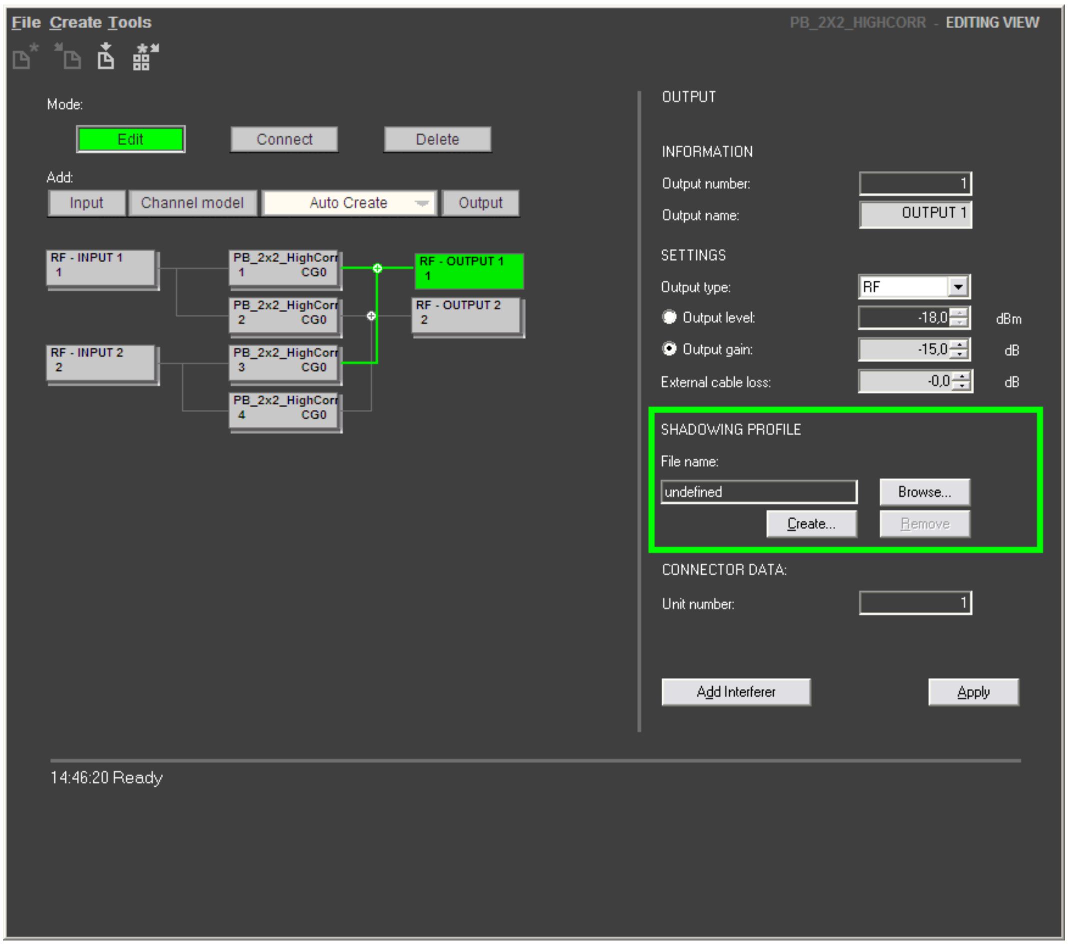


Figure 148. Shadowing profile field in the Editing view

In Editing view, the user can load an existing shadowing profile file, create a new profile or edit a loaded shadowing profile.

Shadowing profile is added in an output by clicking Apply after selecting or creating a profile. The shadowing profile becomes a part of the emulation when the emulation is built. Shadowing profile can be removed from the channel output by clicking Remove in the Shadowing field.

5.1 Shadowing profiles

Clicking Create or Edit in the Shadowing profile field opens the Shadowing Editor, see Figure 149 (Editing view) and Figure 150 (Scenario Wizard). The editor opens with the default Sinusoidal profile type when a new profile is created.

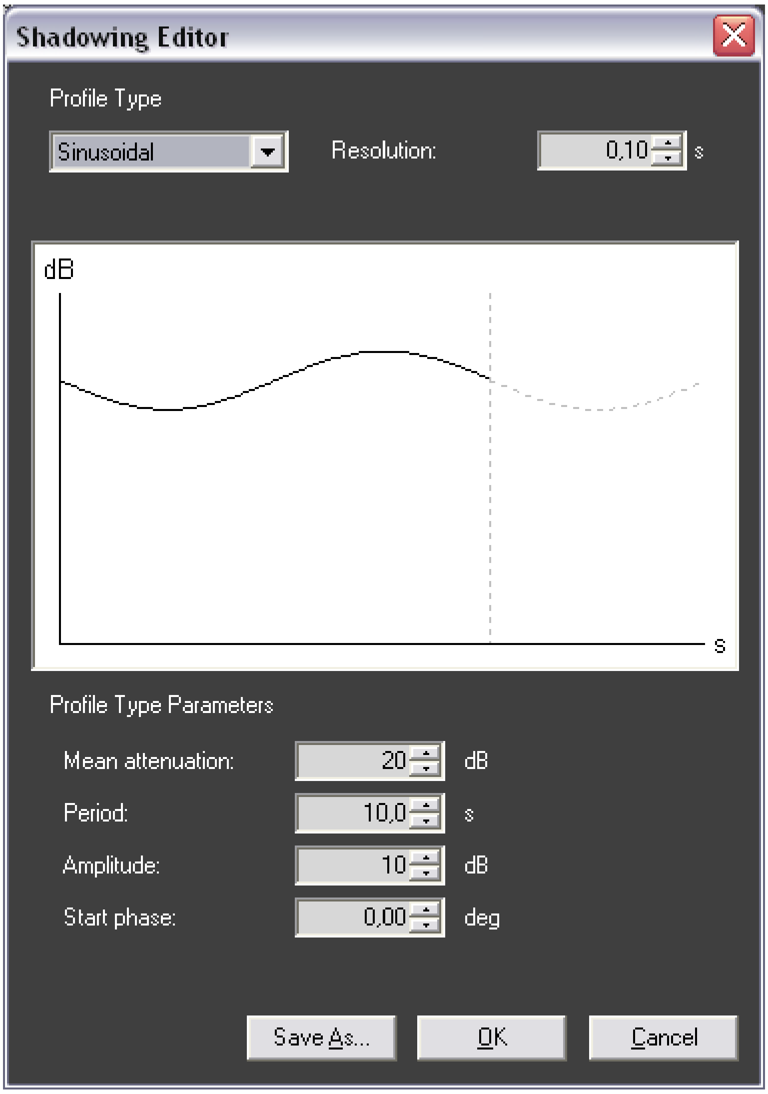


Figure 149. Shadowing editor in Editing view

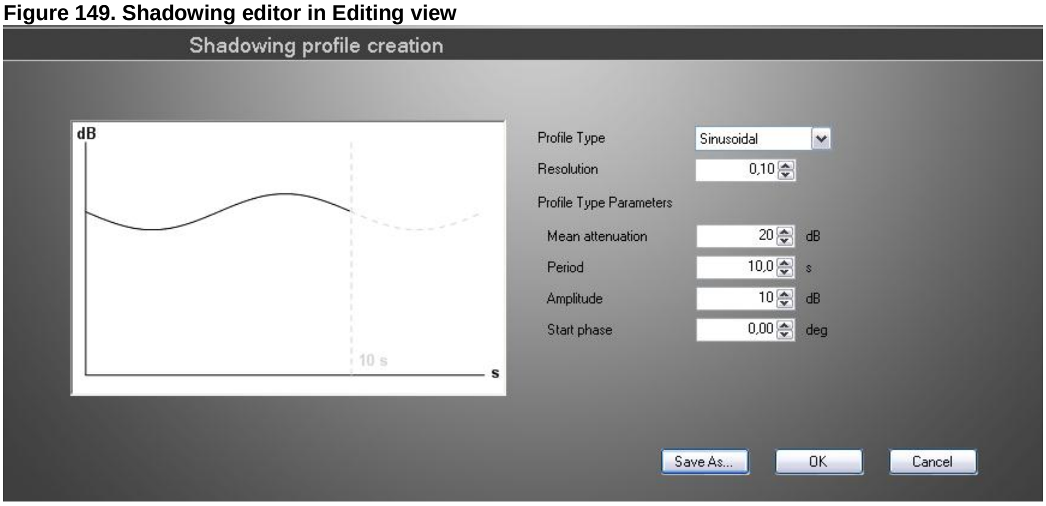


Figure 150. Shadowing profile creation in Scenario Wizard

Note of attenuation

- Under no circumstances can attenuation values be smaller than 0 dB or larger than 100 dB; if values that would cause such situation are entered, profile is changed automatically to match these limitations.

Profile type

- The following profiles types are available Lognormal, Sawtooth, Sinusoidal, Triangle and User defined. See the following subchapters for more information about each profile type.

Resolution (s)

- Specifies how often shadowing is updated.

5.1.1 Lognormal shadowing profile

Lognormal shadowing profile is a distribution of a random variable whose decibel value follows the normal distribution.

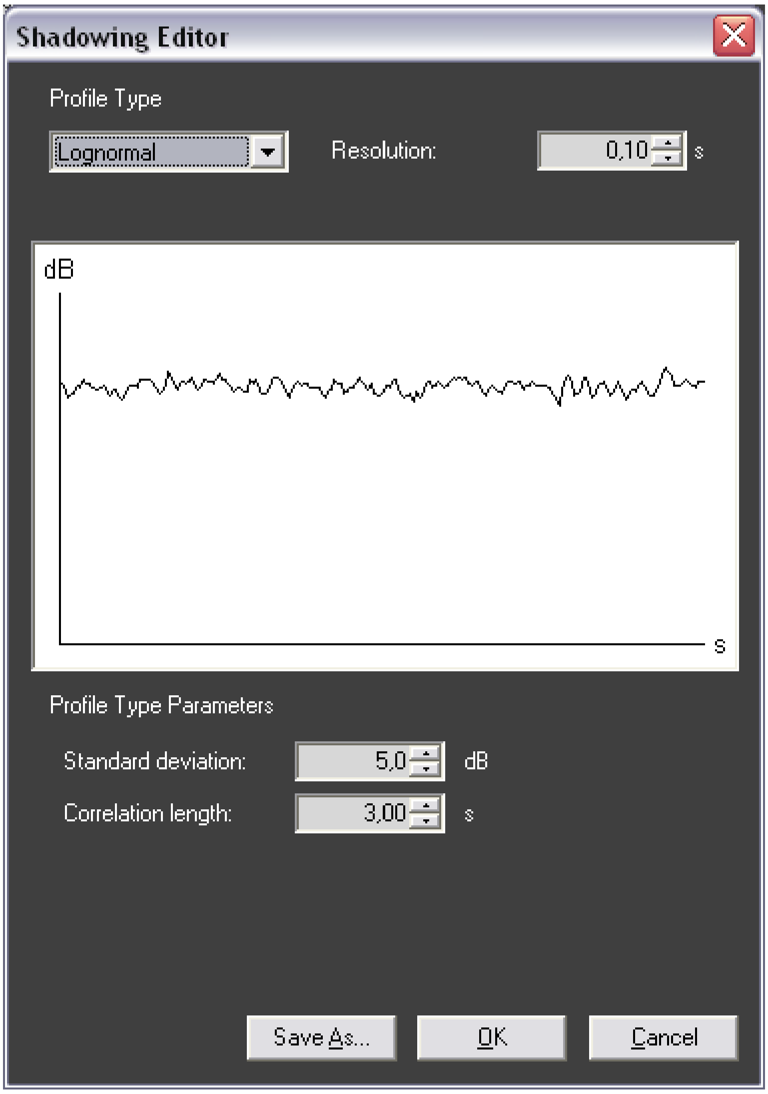


Figure 151. Lognormal shadowing profile

Profile Type Parameters

Standard deviation (dB)
- Standard deviation describes the dispersion of attenuation values. With larger values more deviation from mean value will occur. Default value is 5.0 dB.

Correlation length (s)
- Correlation length describes the time correlation of the Lognormal profile. Slow fading between points separated by correlation length is uncorrelated. Default value is 3.00 s.
- Mean attenuation of Lognormal profile depends on the standard deviation. The mean attenuation is automatically adjusted so that the profile is not cut.

5.1.2 Sawtooth shadowing profile

With a sawtooth profile the attenuation increases or decreases linearly between defined start and end attenuations during a defined period time.

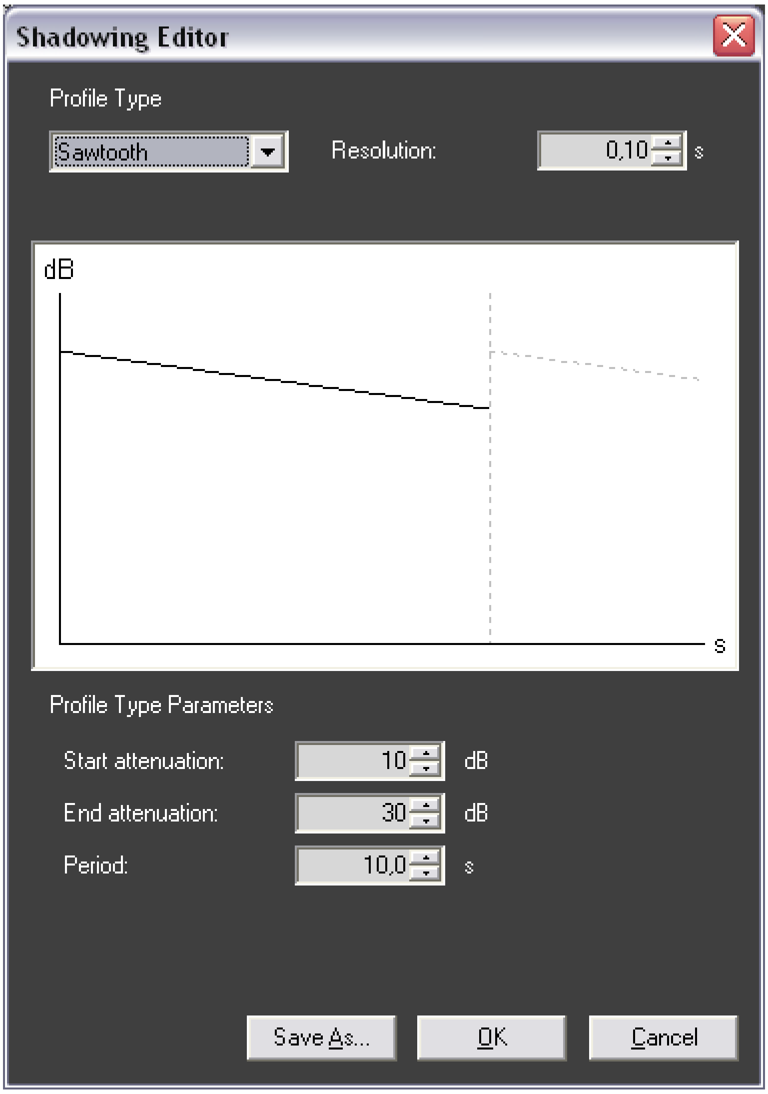


Figure 152. Sawtooth shadowing profile

Profile type parameters

Start attenuation (dB)
- The start attenuation of sawtooth profile. Default value is 10 dB.

End attenuation (dB)
- The end attenuation of sawtooth profile. Default value is 30 dB.

Period (s)
- The period time of sawtooth profile. After the period time the profile is started again from the start attenuation. Default value is 10.0 s.

5.1.3 Sinusoidal shadowing profile

With a sinusoidal profile the attenuation varies according to the defined sinusoidal curve.

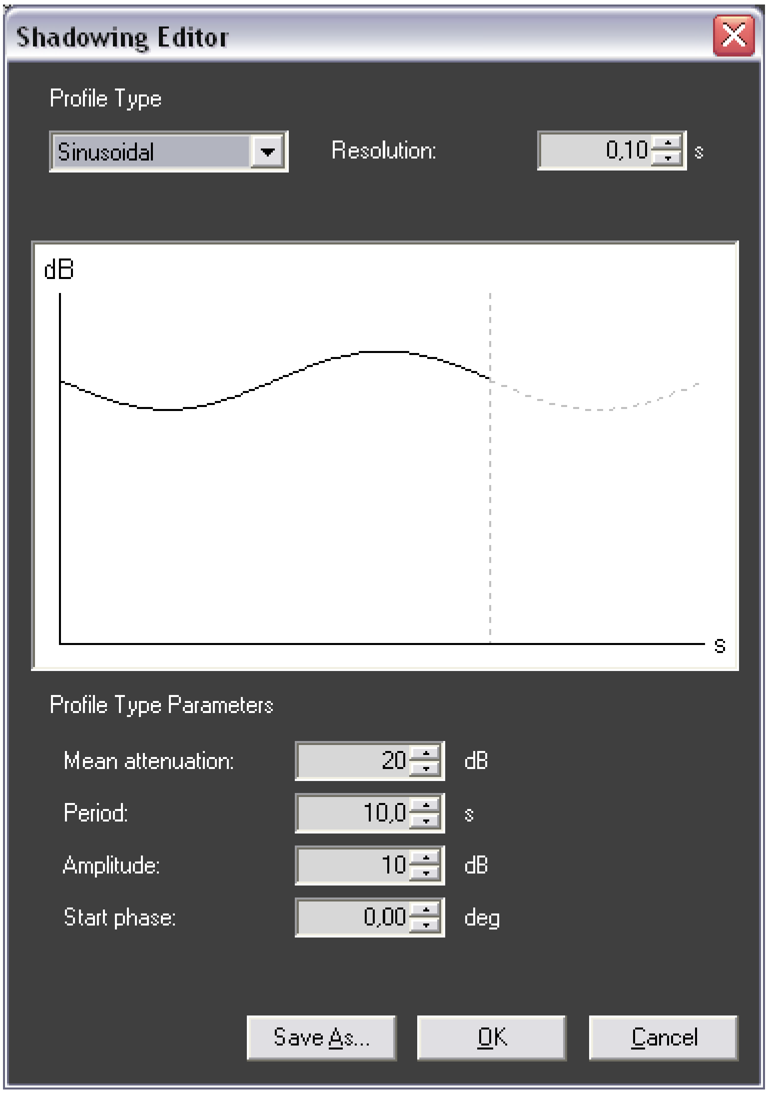


Figure 153. Sinusoidal shadowing profile

Profile Type Parameters

Mean attenuation (dB)
- Average attenuation of sinusoidal profile. Default value is 20 dB.

Period (s)

- The period time of sinusoidal profile. After the period time the profile is started again from the start point. Default value is 10.0 s.

Amplitude (dB)

- The amplitude of sinusoidal profile. Default value is 10 dB.

Start phase (deg)

- The phase angle in which the sinusoidal profile is started to run. Default value is 0.00 deg.

5.1.4 Triangle shadowing profile

With a triangle shadowing profile the attenuation varies linearly from start attenuation to mid-point attenuation and back to start attenuation during the period time.

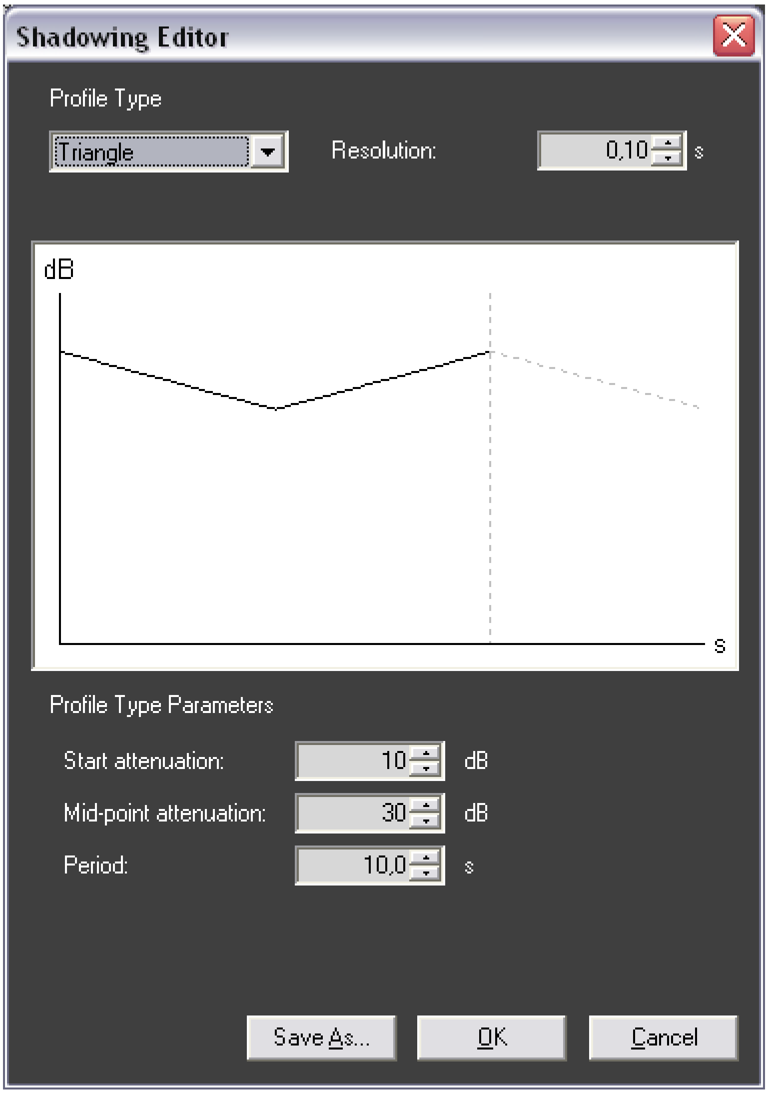


Figure 154. Triangle shadowing profile

Profile Type Parameters

Start attenuation (dB)

- The start attenuation of triangle profile. Default value is 10 dB.

Mid-point attenuation (dB)

- The attenuation where attenuation starts to vary back towards the start attenuation. Default value is 30 dB.

Period (s)

- The period time of triangle profile. After the period time the profile is started again from the start point. Default value is 10.0 s.

5.1.5 User defined shadowing profile

In a user defined profile the user can define the attenuation values and the time in seconds between value changes. Adding time positions and attenuation values in the normal fashion produces ramps from point to point. To add a fast step, add two points having the same time value, the latter having the attenuation after the step.

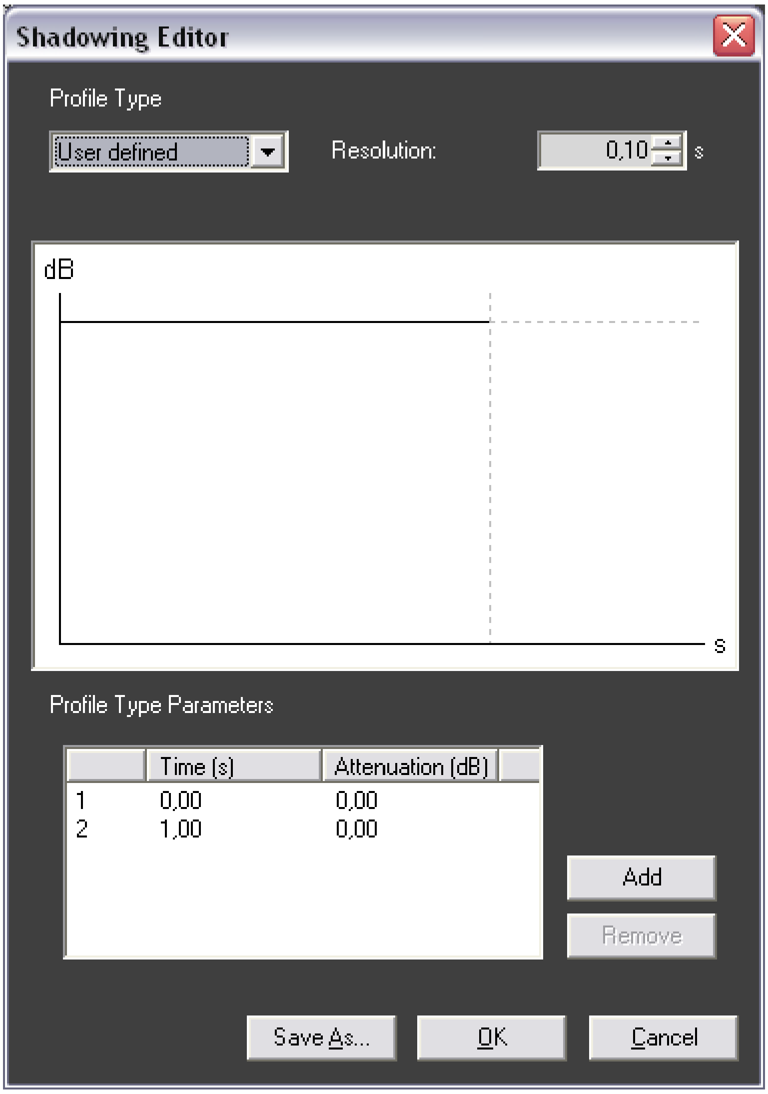


Figure 155. User defined shadowing profile

Profile Type Parameters

Time (s)

- The time from the start point of shadowing profile

Attenuation (dB)

- The attenuation value in the defined time

6 Internal Interference Generator (optional feature)

Internal interference generator is available for EB Propsim as an option. AWGN and CW can be selected as interference sources.

The interference generator can be controlled through the EB Propsim GUI or GPIB/Ethernet with ATE commands the same way the emulator itself. The interference can be set either to constant power mode, C/I mode or disabled.

The control through GUI is handled either in Editing view or Running view.

An interference source can be added from the output parameters. Output parameters can be accessed by selecting the output block in the work area. After selecting the Add Interferer the type of the interferer can be chosen. The added interfering signal is summed automatically to the wanted signal using internal summing. The interfering signal is set to no-fading mode.

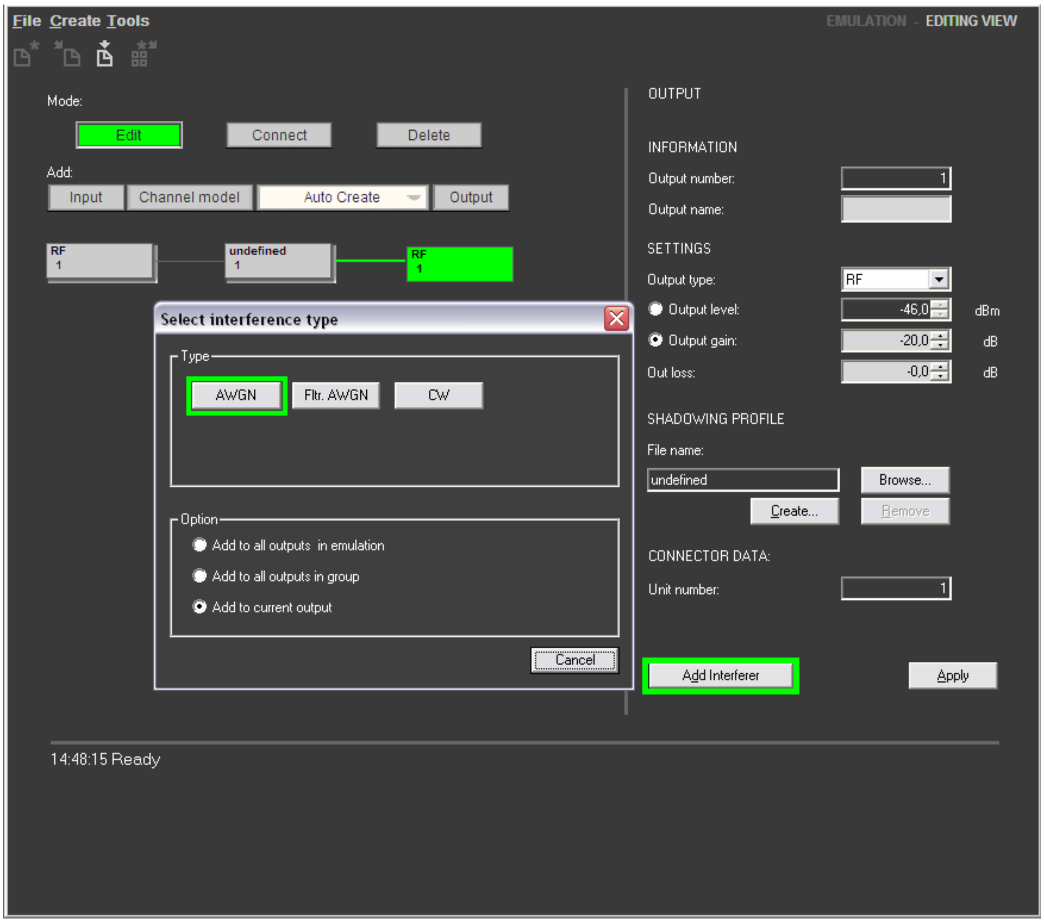


Figure 156. Adding Interference to emulation

The summing of signals can be done also externally by using a RF power combiner. External summing of signal and interference allows higher signal/interference dynamics but requires the use of two EB Propsim outputs and the external RF combiner, whereas internally summed signals are routed through a single EB Propsim output.

To add an externally summed interference, select Auto Create → Interference Ext. Summed.

NOTE: Only those interferers that are defined in the Editing view are available for use in Running view.

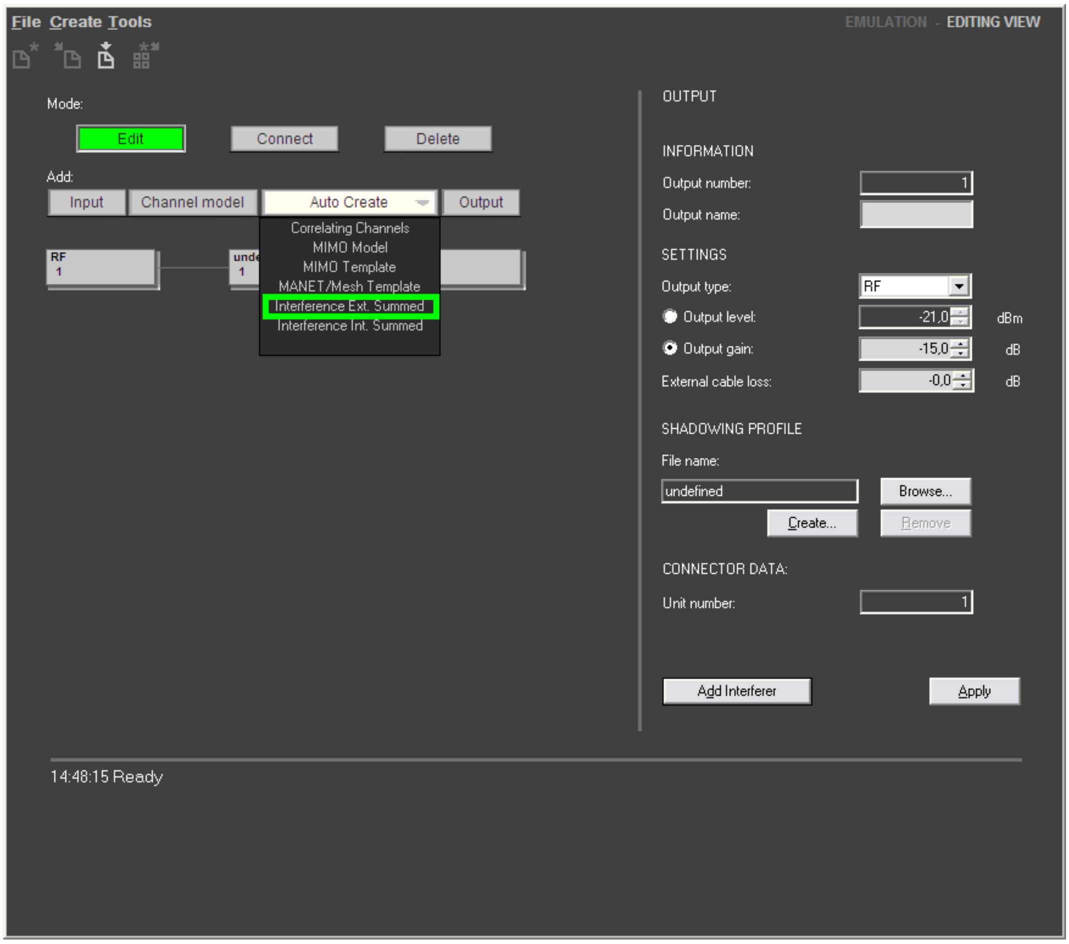


Figure 157. Adding independently faded interference to emulation

6.1 Interference settings

Interference settings can be accessed by selecting the interference block in the block diagram. The settings can be modified either in Running or Editing view.

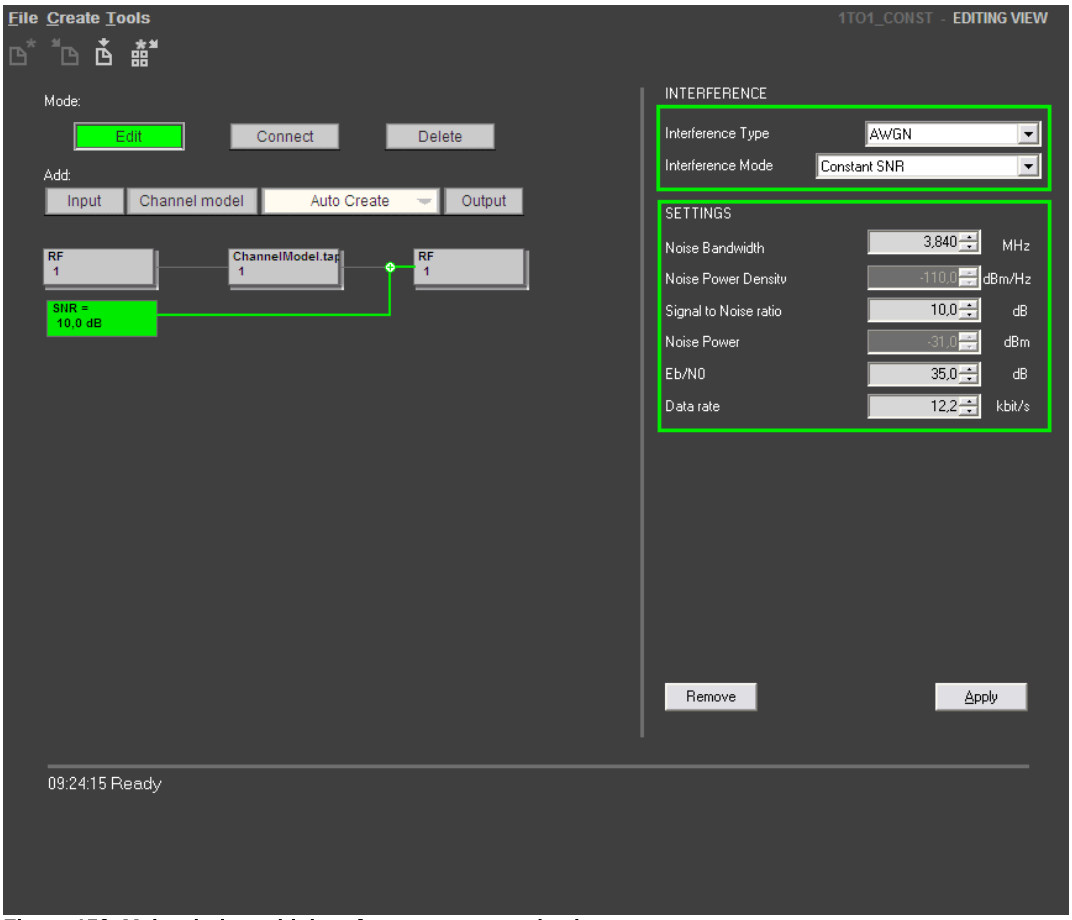


Figure 158. Main window with interference source selection

The interferences can be controlled also through GPIB/LAN. When using interference in remote mode all the same parameters and settings are available as in GUI control mode. More information on ATE commands for controlling the interference can be found in the *Remote Control* section of the User Reference documentation.

6.1.1 General interference settings

In addition of general parameters interference generators may have generator specific parameters, which are described in interferer specific chapters below.

Interference Type

- User can choose the interference signal to be AWGN, Filtered-AWGN or CW. Note that the options available depend on the product configuration.

Interference Mode

- Allows the selection of interference modes.
- Three operational modes are available for AWGN interferences:
  - **Constant SNR** where the ratio between signal and noise power is set by the user.
  - **Constant noise power density** where noise power level is set by the user.

   – **Const. SNR, fixed noise power,** where the noise is set by the user and kept at constant level. Also the SNR ratio is set by the user and the signal level adjusted accordingly.

- Two adjustment modes are available for CW interference:
   – **Constant interference power** where interference power level is set by the user.
   – **Constant C/I ratio** where the ratio between signal and interference power is set by the user.

6.1.2 AWGN settings

The AWGN settings are shown in Figure 159.

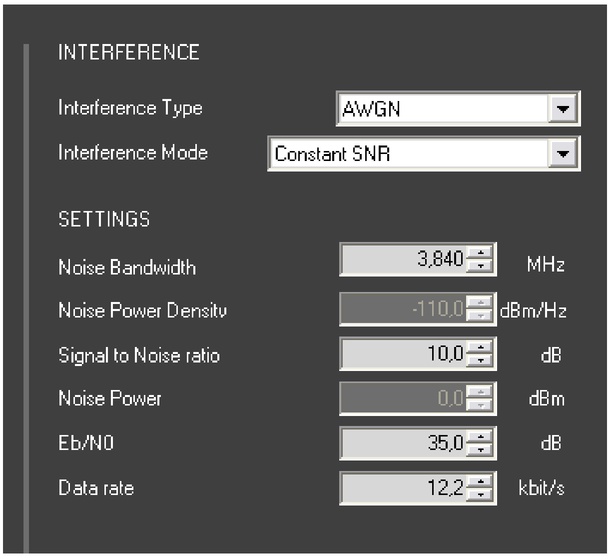


Figure 159. AWGN settings

Settings

Noise Bandwidth [MHz]
- User defines the wanted bandwidth which is used to calculate the required spectral noise power density.

Noise Power Density [dBm/Hz]
- The noise generated by emulator has 40 MHz bandwidth (80 or 125MHz optional). User defines the absolute noise power spectral density in dBm/Hz, which is used to calculate the required noise power spectral density over the defined bandwidth for noise power and SNR calculations. Correct noise level is calculated by assuming that exactly this amount of noise is received in the receiver. The relationship between noise power spectral density (N_0) and N, the power of the noise at specified band is calculated as
  $N = N_0 + 10\log_{10}(NBW)$.

Signal to Noise ratio [dB]
- User defines the signal to noise ratio in dB. The EB Propsim measures the signal level continuously and adjusts the noise power density based on the measured signal levels and user specified Noise Bandwidth. The measurement period is currently fixed and always approximately 500 ms.

Noise Power [dBm]
- User defines the wanted noise power over the defined Noise Bandwidth.

Eb/N0 [dB]
- User defines the signal to noise ratio per transmitted bit in dB. The EB Propsim measures the signal level continuously and adjusts the noise power density based on the measured signal levels and user specified data rate. The relationship between noise power spectral density (N_0) and the wanted signal is calculated from equation:

$$\frac{E_b}{N_0} = \frac{C}{N} - 10\log_{10}\left(\frac{Data\_rate}{NBW}\right).$$

Data rate [kbit/s]
- User defines the data rate in kbps which is used to calculate the required spectral noise power density.

6.1.3 Filtered AWGN settings

The settings for Filtered AWGN interference are shown in Figure 160 below.

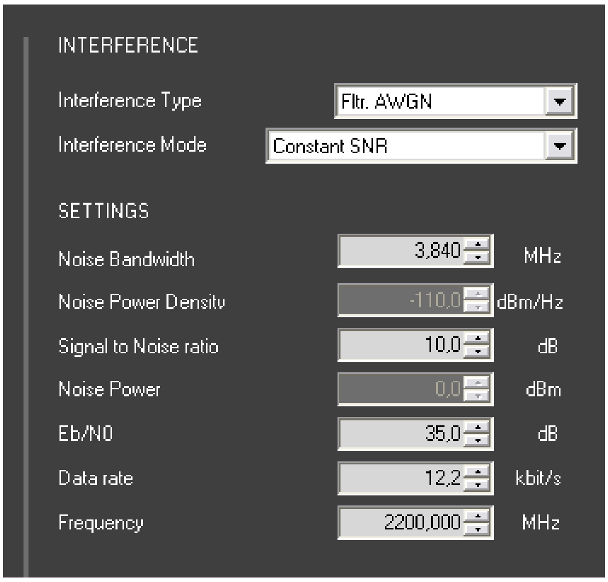


Figure 160. Filtered AWGN settings

Settings

Noise Bandwidth [MHz]:
- User defines the applied signal bandwidth (1 – 30 MHz).

Frequency [MHz]:
- The available range depending on EB Propsim bandwidth, ±5 MHz (40 MHz BW), ±25 MHz (80 MHz BW), ±47.5 MHz (125 MHz BW) around the specified channel center frequency.

The other settings are identical to AWGN settings described in previous chapter.

6.1.4 CW settings

The CW settings are shown in Figure 161 below.

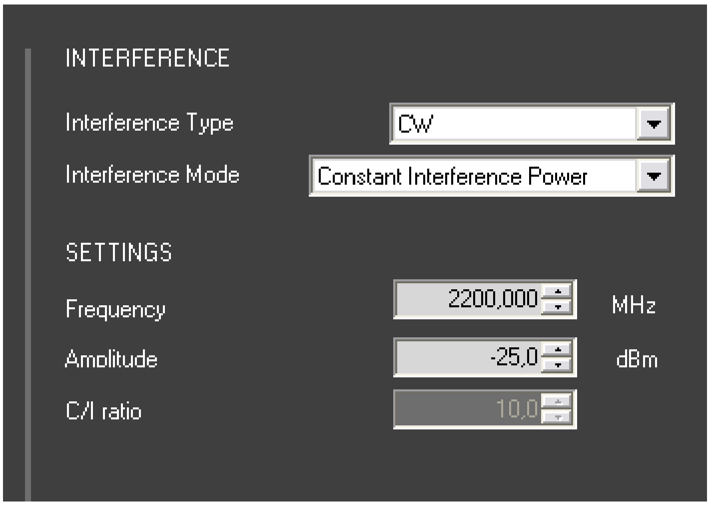


Figure 161. CW settings

Settings

Frequency [MHz]:
- The available range is ±20 MHz (±40 MHz or ±62.5 MHz optional) around the specified channel center frequency.

Amplitude [dBm]
- The absolute power of interference signal on output connector. This setting is not available when using Constant C/I ratio mode.

C/I Ratio [dB]:
- Desired carrier to interference ratio which should be kept. This selection is not available in when using Constant Interference Power mode.

Figure 162. Common interference settings

Common interference settings in Running view

Interference Enable
- Activate/deactivate interference

Measure all inputs
- When enabled, interference is following signal levels of all inputs in group
- When disabled, interference is following signal level only from single input

Apply to all in group
- All settings are applied to all interferences within the channel group (link)

7 Aerospace and Satellite Option – ASO (optional feature for EB Propsim F8 REL 1.4)

ASO enables EB Propsim F8 to emulate high velocity radio channels from ground-to-air and air-to-air links typical in space and satellite applications, aircraft communication, missile control and radars.

The option contains Aerospace Model Editor application and specific channel emulation engine with capability to run models with longer path delays than in normal emulation mode and extremely high velocities. The editor application is used to generate, edit and visualize ASO models.

Key Features

- Capability to emulate radio channels with long delay and high velocity. These emulations have very high accuracy and realistic Doppler effects in both frequency and in code domain.
- Open text based file format for importing customer specific radio channel and location parameters
- Functionality to validate the model format and data values
- Visualization of user defined dynamic channel models
- Geometric graphs showing channel parameter curves for three dimensional movement
- Playback functionality to see model evolution in time
- Wizard for generating an editing models

Aerospace Model Editor is opened in the navigation bar at the left side of GUI by clicking Advanced Tools category button and then selecting Aerospace Model Editor. The Aerospace Model Editor is used to load and view models in ASO file format.

The Aerospace Model Editor allows user to:

- Validate user created models
- Observe parameters like distance, speed, range rate and gain at different time and object positions.
- Visualize geometry of coordinate based model movement in three dimensions
- Observe Doppler, delay and range rate curves
- Make partial selections from the most interested part of the model
- Preview time evolution of the models

In addition the Aerospace Model Editor includes model generation wizard by which the user can create simple models to different types of ASO file formats. For more information of the ASO file formats, refer to chapter File Formats of User Reference documentation.

7.1 Aerospace Model Editor

The application window is divided into graphical views and regions with different functions as shown in Figure 163 below.

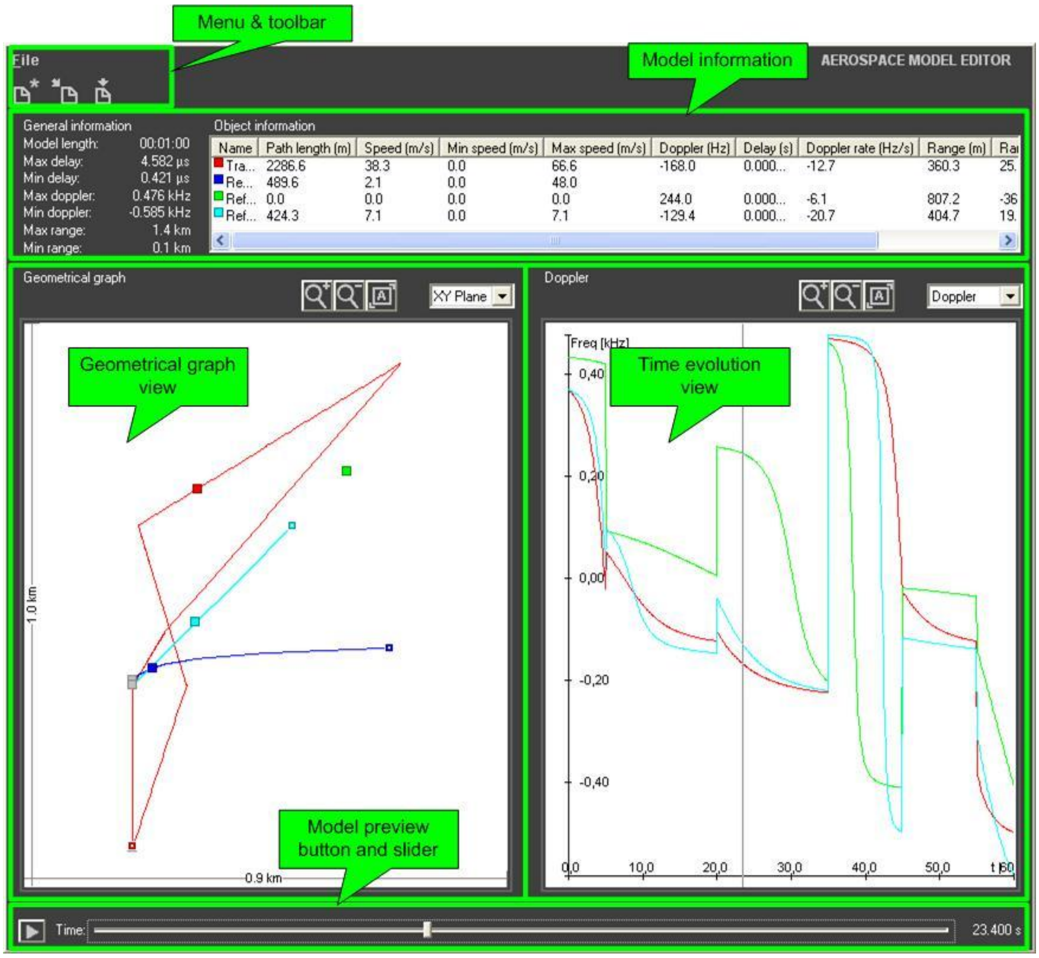


Figure 163: Aerospace Model Editor

7.1.1 File menu

The Aerospace Model Editor file menu contains the following items:

MAIN MENU	SUBMENU	SHORTCUT	DESCRIPTION
File	Create New ASO model…	Ctrl-N	Opens New Model Generation Wizard
	Open ASO model…	Ctrl-O	Opens existing ASO model
	Edit ASO model…	Ctrl-E	Opens model editing window
	Partial model…	Ctrl-P	Opens Partial model dialog
	Save	Ctrl-S	Saves model
	Save As…		Saves model with new name
	Exit		Closes Aerospace Model Editor

Table 66. File menu of Aerospace Model Editor

7.1.2 Model Information

General information field shows some main properties of the model.

GENERAL INFORMATION	DESCRIPTION
Model length	Model length in seconds
Max delay	Maximum propagation delay of the model
Min delay	Minimum propagation delay of the model. Note: if the minimum delay of the model is smaller than EB Propsim F8 insertion delay, fixed delay offset is added to the model at emulation generation phase.
Max Doppler	Maximum Doppler frequency over the model
Min Doppler	Minimum Doppler frequency over the model
Max range	Maximum range i.e. maximum radio path distance, over the model
Min range	Minimum range (radio path distance) over the model

Table 67. General information of model

Object information field shows the current status of objects in the model. Object names and information columns shown are dependent on what type of model is selected.

Most of the parameters in this field are time dependent. Current time can be varied by using the model preview slider (see 7.1.4). By clicking the object name with left mouse button the associated curves in graphs will be highlighted.

OBJECT INFORMATION	DESCRIPTION
Name	Object name: Transmitter, Receiver, Reflector or Link
Path length [m]	Static parameter showing length of the object movement over the model. Information available only for coordinate based (N-type) ASO models
Speed [m/s]	Speed of the object movement. Information available only for coordinate based (N-type) ASO models
Min speed [m/s]	Static parameter showing the minimum speed of the object movement over the model. Information available only for coordinate based (N-type) ASO models
Max speed [m/s]	Static parameter showing the maximum speed of the object movement over the model. Information available only for coordinate based (N-type) ASO models
Doppler [Hz]	Doppler frequency of the link (*
Delay [s]	Delay of the link (*
Doppler rate [Hz/s]	Change rate of the link Doppler (*
Range [m]	Range of the link (e.g. radial distance) (*
Range rate [m/s]	Link range changing rate due to movement of objects (*
Gain [dB]	Actual gain of the propagation link (*
Period [s]	Static parameter showing the time of the periodic cycle usetd in function based models. Information available only for Function based ASO models.

*) Note: Propagation links are calculated from other objects to receiver e.g. from transmitter to receiver, directly or via reflections. Therefore certain link parameters (Doppler, Delay, Doppler rate, Range, Gain) are not shown on the receiver object line.

Table 68. Object information of model

7.1.3 Graphical Views

Model time evolution can be observed in the graph on the right-hand side. Doppler / Delay / Range rate curves are shown as a function of time. If there are multiple objects in the model, their graphs are shown in different colors.

In Geometrical graph, on the left side of the Aerospace editor view, object paths can be observed in selected plane (XY, YZ or XZ). The object movement and location geometries are shown only for Coordinate based ASO models.

Both graphical views can be zoomed in and out or reset via corresponding buttons above graphs. Views can be zoomed also by using the scroll wheel of the mouse. Pressing the Ctrl key simultaneously zooms faster when using the scroll wheel. Both views can be panned by holding down the left mouse button and moving mouse around.

7.1.4 Model preview

User is able to use preview functionality to see how the objects will move over the time and how different values of various parameters will change during the actual emulation run. The preview can be started by pressing the start/pause animation button on the left side of the time slider. User can also move the time slider manually by using mouse or use keyboard arrow buttons when the slider is selected.

Values in the Object information box reflect the object states at current time position. Current time is shown at the right end of the slider. If partial model is selected (see 7.1.7) the preview slider will play only the selected active region.

7.1.5 New Model Generation Wizard

Create new ASO model command opens following dialog box (
Figure 164) where the user is requested to select what kind of ASO-model will be created. The selected type will determine the actual ASO file type. Details about the ASO file formats can be found in File Formats section of User Reference.

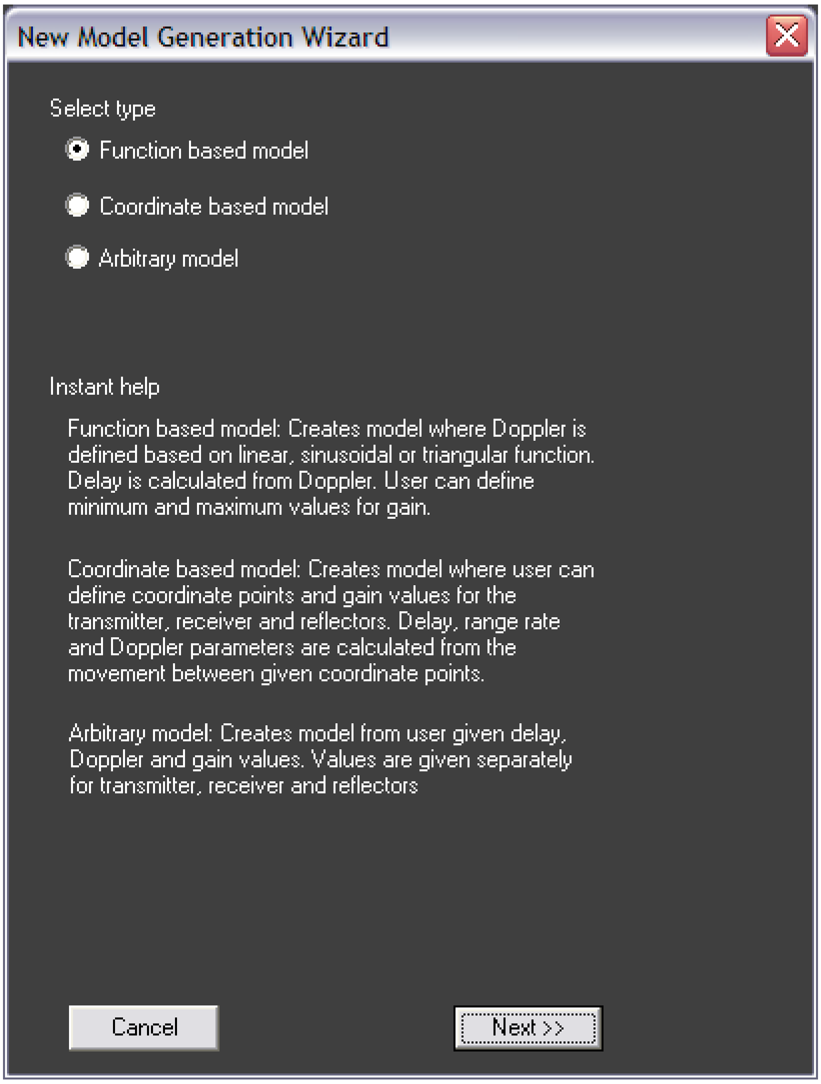


Figure 164: New Model Generation Wizard model type selection dialog

Emulation center frequency allows user to define different frequencies for the model and for the emulation. Doppler is always calculated by using emulation center frequency value. RF center frequency defines the actual frequency used in connections to and from the EB Propsim F8 emulator.

This feature is useful if the user application frequency band is outside EB Propsim F8 frequency range. In this case the RF connection to EB Propsim F8 can be done for example at the customer application IF frequency or by selecting testing frequency so that it matches with the emulator frequency range.

7.1.5.1 Function based model

Function based model creates model where Doppler is defined based on linear, sinusoidal or triangular function. Delay is calculated from Doppler. User can define minimum and maximum values for gain. When function based model is selected following dialog will open (Figure 165).

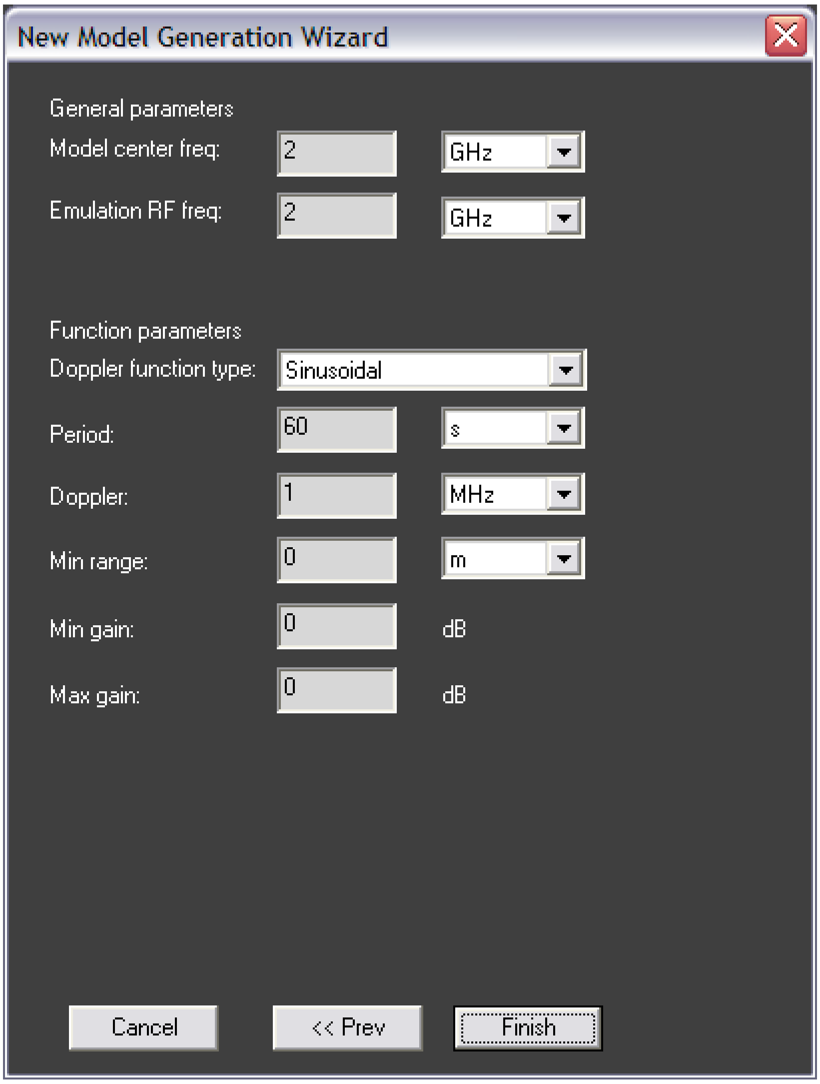


Figure 165: Function based model parameter dialog

7.1.5.2 Coordinate based model

Coordinate based model creates model where user can define coordinate points and gain values for the transmitter, receiver and reflectors. Delay, range rate and Doppler parameters are calculated from the movement between given coordinate points. Coordinate based model parameter dialog is shown in
Figure 166.

Values in table fields can be written by using scientific expressions.

Examples:
- 100km can be written 100e3 [m] and
- 15 microseconds 15e-6 [s].

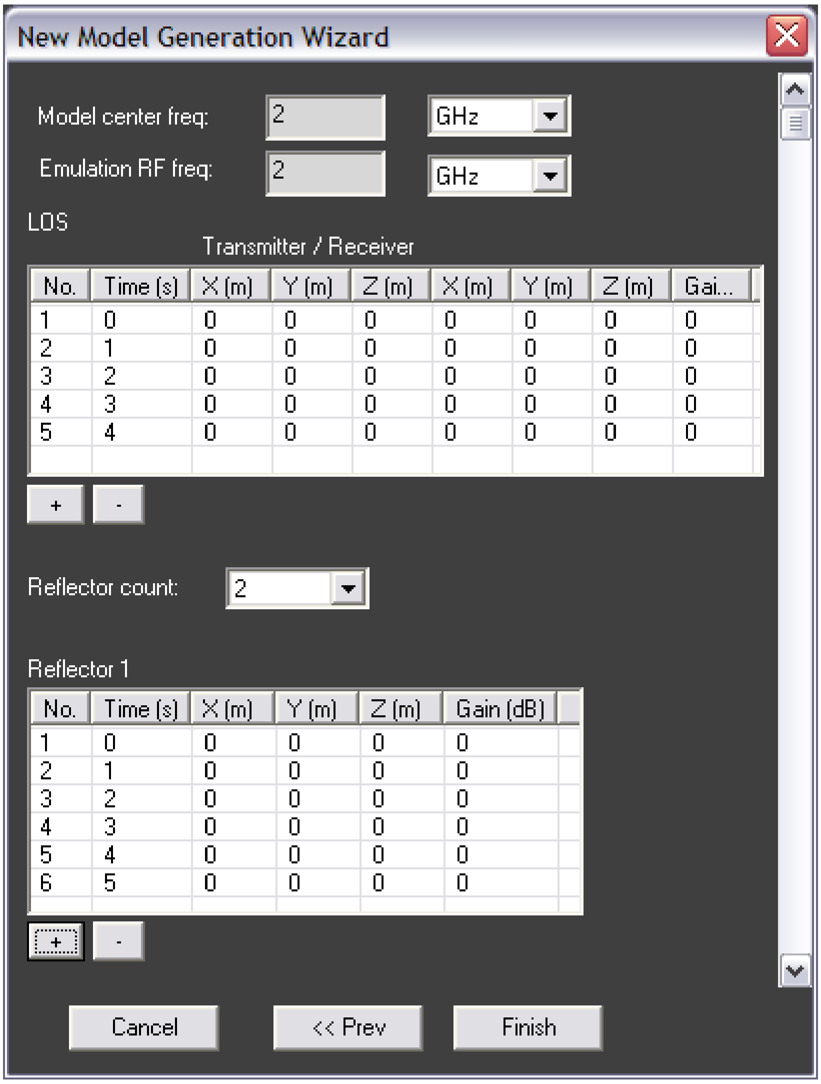


Figure 166: Coordinate based model parameter dialog

7.1.5.3 Arbitrary model

Arbitrary model creates model from user given delay, Doppler and gain values. Values are given separately for transmitter, receiver and reflectors. Arbitrary model parameter dialog is shown in Figure 167.

Values in table fields can be written by using scientific expressions.

Example: 15 microseconds can be written 15e-6 [s].

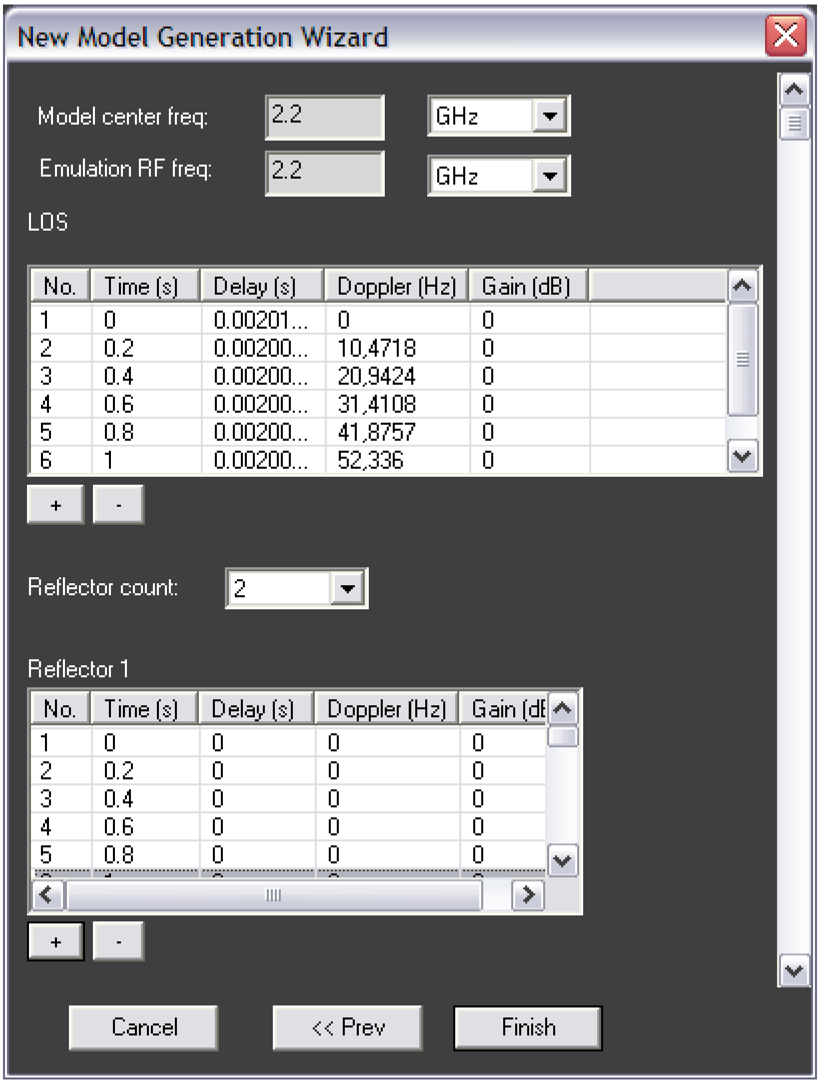


Figure 167: Arbitrary model parameter dialog

7.1.6 Editing existing Aerospace models

The ASO files generated by the Aerospace Model Editor can be further modified by simply editing the file content with any text editing tool. It is recommended that the models are always opened in Aerospace model editor before loading the model into emulation in Editing view. The Aerospace model editor will validate that the model parameters are inside specified range and that the model file syntax is correctly written. It is also useful to see if the model looks as it was intended to be before compiling the emulation file.

7.1.7 Partial models

This functionality enables user to select a smaller part of the model. Partial modeling functionality is useful, when only a specific portion of the longer model needs to be used in the final emulation. Partial modeling functionality can also be used for creating several shorter models from different time instances from the longer model range.

By selecting Partial model from the File menu a new dialog box will appear, see Figure 168.

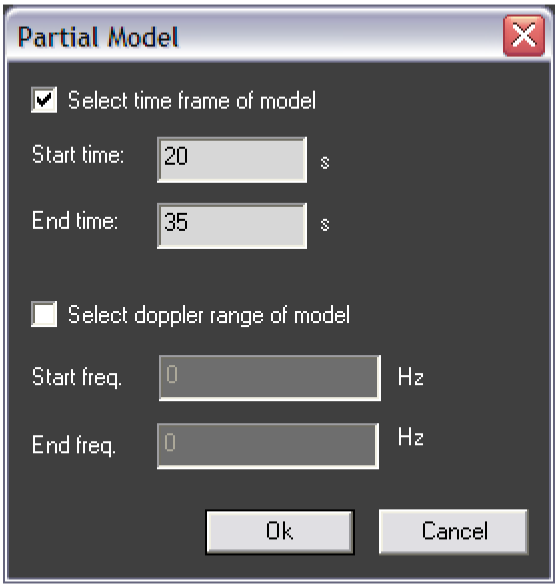


Figure 168: Selecting partial model range

User can define the model range by giving start and end times or by giving Doppler frequency limits of the range. Alternatively user may use the left mouse button to drag range markers on the Doppler or delay view, see Figure 169. Range markers appear if user has checked the Select time frame of model or the Select Doppler range of model option in dialog.

Please note that the Doppler range selection is an alternative way of defining start and stop times for the model. The selection is always done in time axis. Doppler frequency selection is available only for function based ASO models.

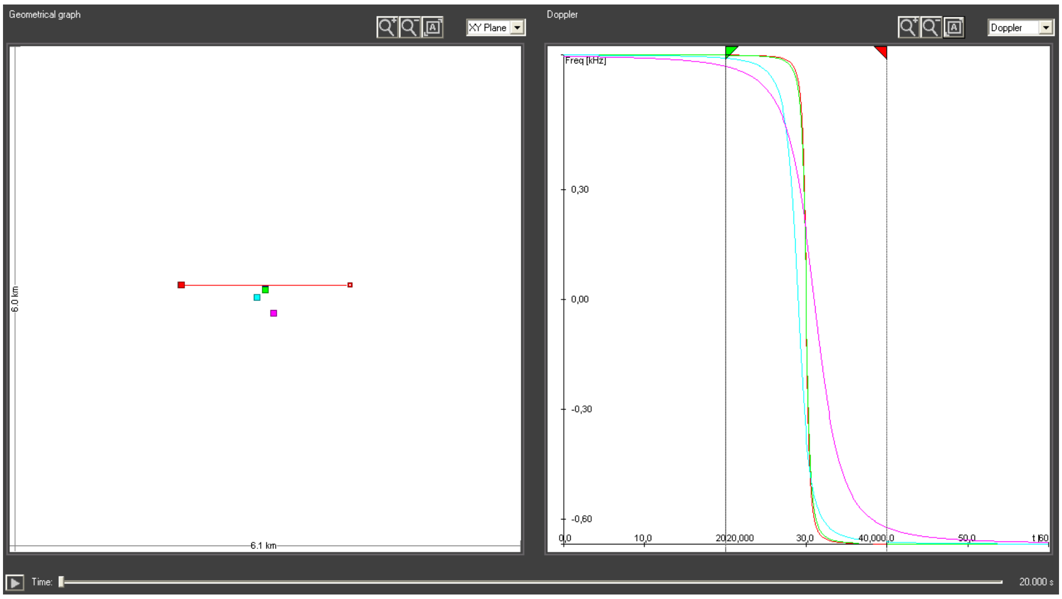


Figure 169: Partial modeling

7.2 Creating and Running Emulation

This chapter describes how the ASO models are used for creating Aerospace emulation and how to run it in real time. Generating and running Aerospace emulation with EB Propsim F8 requires generally following steps:

- Creating ASO channel model
- Creating and compiling emulation in Editing view
- Loading and controlling generated emulation in Running view

Please note that the focus in this section is in ASO specific operation. Refer to chapters 4.4 and 4.5 for more information about Running view and Editing view.

7.2.1 Creating Aerospace emulation in Editing view

Editing view (see Figure 170) is used for creating emulations to EB Propsim F8. Workflow in creating Aerospace emulation is similar to normal emulations. Start by creating new standard emulation and add Input, Channel model and Output objects by clicking buttons in Add region. Please note: ASO models can not be mixed with normal channel models in the same emulation.

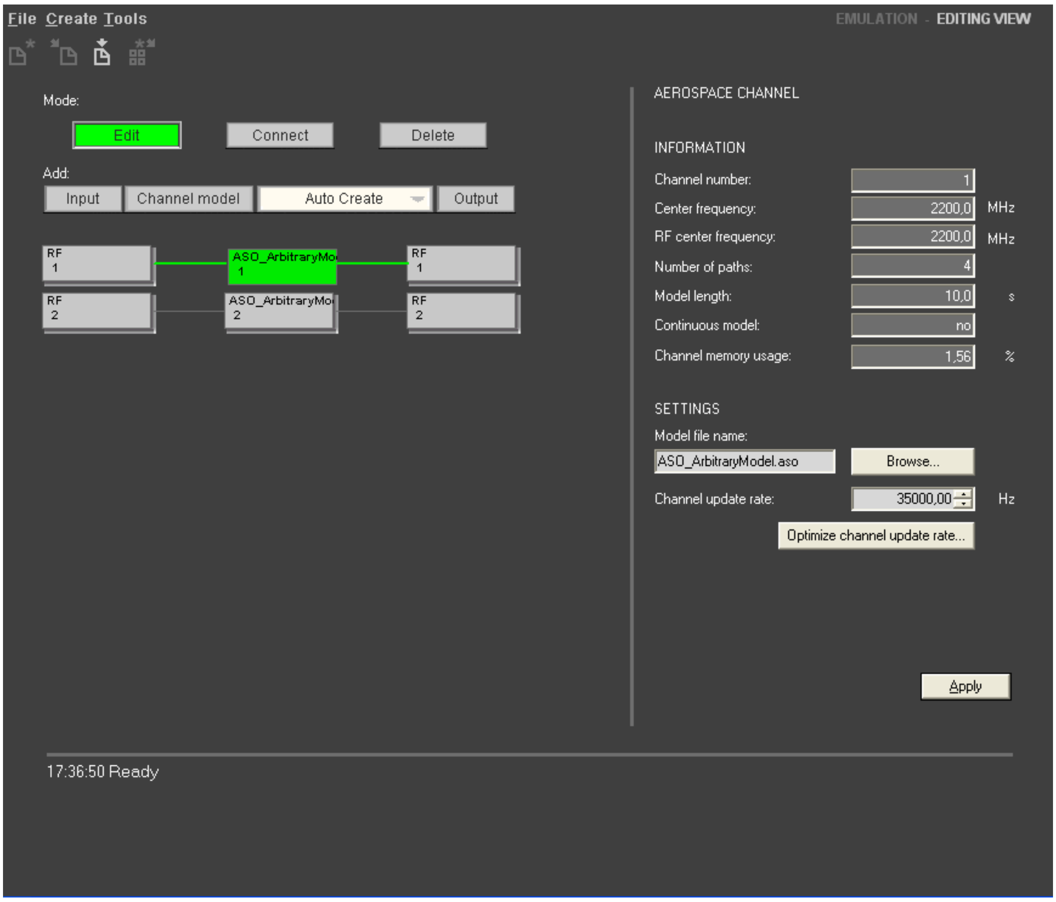


Figure 170: Aerospace model loaded into Channel model box in Editing view

When model box is selected, the right side of the Editing view shows model specific Information and Settings for that particular channel. Modifications to models are done in Aerospace model editor or directly to ASO file with any text editor.

7.2.1.1 Update rate optimization

The channel update rate affects to the accuracy and length of the emulation and size of the emulation file. Optimize channel update rate button opens a dialog (Figure 171) that helps in choosing the correct channel update rate to minimize Doppler error or range error to certain limits, or to find the best combined accuracy with available hardware resources.

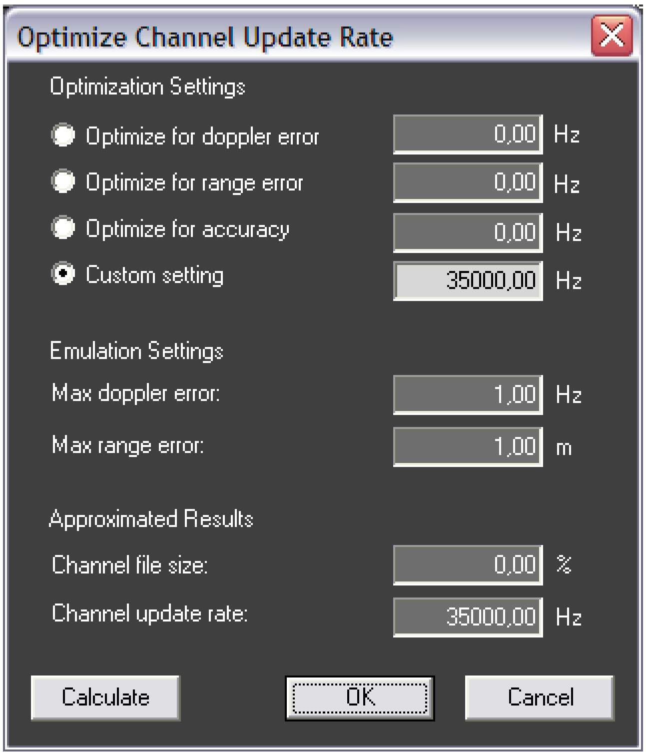


Figure 171: Channel update optimization dialog

Depending on the ASO model type user has 1..3 optional methods to optimize the channel update rate. In addition user can adjust the update rate parameter manually.

Optimize for Doppler error method uses user defined Max Doppler error value and calculates the minimum update rate that fulfils the criterion.

Optimize for Range error method uses user defined Max Range error value and calculates the minimum update rate that fulfils the criterion.

Optimize for Accuracy method calculates the highest model update rate that allows the emulation to fit into the emulator memory.

Custom Setting allows user to define model update rate manually.

Resulting update rate and estimated size of the model are shown after pressing Calculate button. Calculating Doppler and range errors for long models might take some time to complete.

7.2.1.2 Generating emulation file

When the models are loaded to emulation, necessary adjustments are done and the model is saved, the emulation can be generated. The generation starts by pressing Generate model button in the toolbar.

When the emulation file is generated user can launch the Running view from the Tools menu in Editing view.

7.2.2 Running Aerospace emulation

Running view (Figure 172) is used for controlling and running emulations in EB Propsim F8.

A generated aerospace emulation is loaded by selecting Open from File menu and then choosing the emulation file (*.smu). When the emulation is loaded channel parameters are initialized. Pressing Run button in the toolbar will eventually start the emulation. Stop button will stop the emulation and return it to beginning of the emulation. Pause and Go To functionalities are disabled with ASO models.

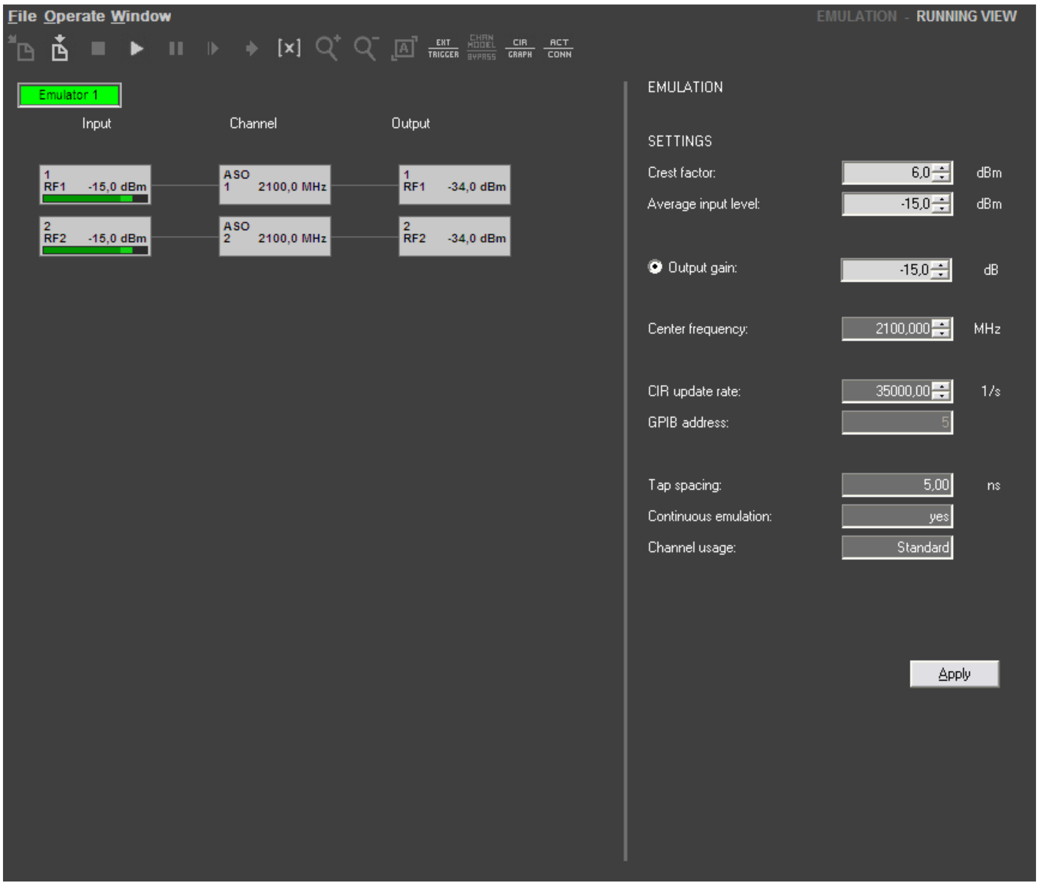


Figure 172: Aerospace emulation loaded in Running view

Emulation settings view shows channel connection diagram and emulation related information. By selecting Show CIR Graphics in the Window menu, a following view will appear (Figure 173). In this view user can observe impulse responses and aerospace channel state parameters channel-by-channel.

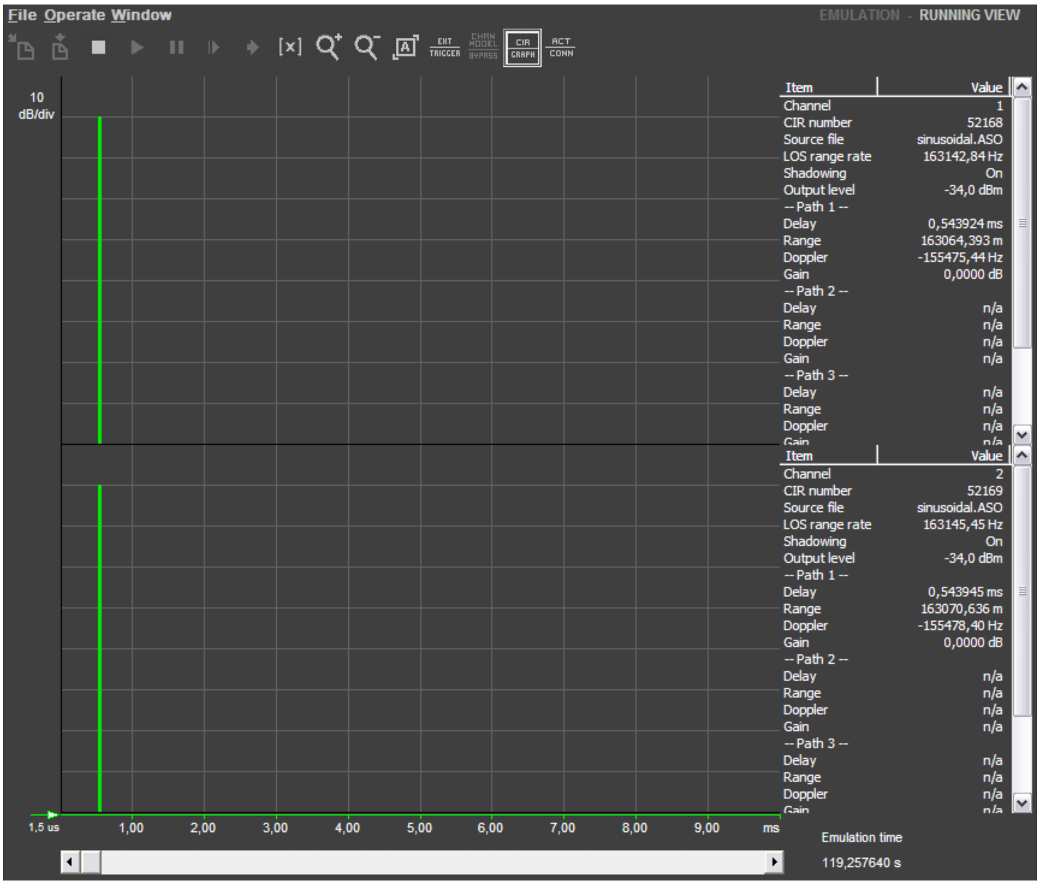


Figure 173: CIR graphics window

8 Remote Control

ATE (Automatic Test Equipment) services of EB Propsim provide the user a possibility to control the emulator remotely, either via GPIB or LAN interface. Both of the interfaces obey the same ATE commands explained in this document.

When EB Propsim emulator boots up, both the ATE GPIB Server and the ATE LAN Server are started. As ATE commands are received to either interface, they are displayed in the corresponding monitor console, which also shows possible responses and error messages.

ATE provides to a large extent the functionality of the Running View of the graphical user interface and can be seen as equivalent to it.

8.1 GPIB

With GPIB interface, the EB Propsim is used like any GPIB device in listener mode, while your application will be the controller-in-charge. There are some preconfigured settings for the emulator ATE GPIB interface, of which only primary address is reconfigurable by the user.

ITEM	SETTING
Control mode	LISTENER/TALKER
IbcHSCableLength	2
Primary address (default)	1
Secondary address	-
Timeout	100 ms
EOS mode	0

Table 69. Configuration settings for ATE GPIB interface

8.2 LAN

The LAN interface can be accessed with the following LAN parameters:

ITEM	USER SETTING
LAN IP address	IP address of the emulator.
Port	3334
EOS	‘\n’ (line feed)

Table 70. Access settings for ATE LAN connection

8.2.1 LAN Example

Below you can find a short code example that opens connection to the emulator by using ATE LAN interface, requests identification, and resets it. The example is written in MatLab (registered trademark of The MathWorks, Inc.). Please refer to the documentation of test environment you are using for details on how to open and use LAN connections.

```
% Open communication to EB Propsim

% Replace 192.168.0.1 with IP address of the Propsim
fid = tcpip('192.168.0.1', 3334);
fopen (fid);

% Read instrument's ID

fprintf (fid, "*IDN?\n');
reply = fscanf(fid);
disp([ 'Connected: ' reply(1:length(reply)-1) ] );

% Reset device

fprintf (fid, '*RST\n');

% Close connection

fclose (fid);
```

8.2.2 LAN Example using Microsoft Visual Studio

The following example demonstrates how EB Propsim can be controlled by using Microsoft Visual Studio .NET 2003, using C++ language and .NET framework.

```
// This is simple example program that communicates with Propsim
// emulator using ATE LAN interface.
// Written in C++, using Microsoft Visual Studio 2003 .NET
// and .NET framework

#include "stdafx.h"
#using <mscorlib.dll>
using namespace System;
using namespace System::Text;
using namespace System::IO;
using namespace System::Net;
using namespace System::Net::Sockets;
using namespace System::Collections;

 // Connect to a remote socket by using network API.
Socket * ConnectSocket(String* server, int port)
{
  Socket* s = 0;
  IPHostEntry* hostEntry = 0;

   // Get host related information.
  hostEntry = Dns::Resolve(server);

  IEnumerator* myEnum = hostEntry->AddressList->GetEnumerator();
  while (myEnum->MoveNext())
    {
      IPAddress* address = __try_cast<IPAddress*>(myEnum->Current);
      IPEndPoint* endPoint = new IPEndPoint(address, port);
      Socket* tmpS = new Socket(endPoint->AddressFamily,
                            SocketType::Stream, ProtocolType::Tcp);
```

```
      tmpS->Connect(endPoint);
      if (tmpS->Connected)
        { s = tmpS;
          break;
        }
      else
        { continue;
        }
    }
  return s;
}

 // Send an ATE command to Propsim. Line termination is added to end of
string,
// and does not need to be included in parameter string.
void sendCommand(Socket *s, String *cmd)
{
  String* request = String::Concat(cmd,S"\n");
  Byte bytesSent[] = Encoding::ASCII->GetBytes(request);
s->Send(bytesSent, bytesSent->Length,
        static_cast<SocketFlags>(0));
}

 // Read last response from socket. Note that if there is no
 // response to be read, this will block until program is
 // terminated. Be sure to only call this after a request has
 // been sent to ATE server.
String* readResponse(Socket *s)
{

  Byte bytesReceived[] = new Byte[2];
  String* response;

   // Receive message one character at a time; ineffient, but does
   // not require additional buffering.
  do {
int bytes = s->Receive(bytesReceived, 1,
                       static_cast<SocketFlags>(0));
 response = String::Concat(response,
                   Encoding::ASCII->GetString(bytesReceived,
                   0, bytes) );
    } while (response->IndexOf('\n') == -1);

  return response;
}

 // Main program
int _tmain()
{
   // Open connection to Propsim. Change "localhost" to Propsim's
   // IP address to connect remotely.
  Socket* s = ConnectSocket("localhost", 3334);

   // Send IDN query.
  sendCommand(s, "*IDN?");

   // Read response to previous IDN query.
  String *response = readResponse(s);

   // Write response to console
  Console::WriteLine(response);
  return 0;
}
```

8.3 ATE monitor interface

When the ATE connection is established, either with LAN or GPIB, and an emulation has been loaded the Ate monitor (see Figure 174) will pop up automatically on the screen. The Ate monitor will be updated after every sent command and will show the current status of the emulation. The interface can be used in local or remote mode.

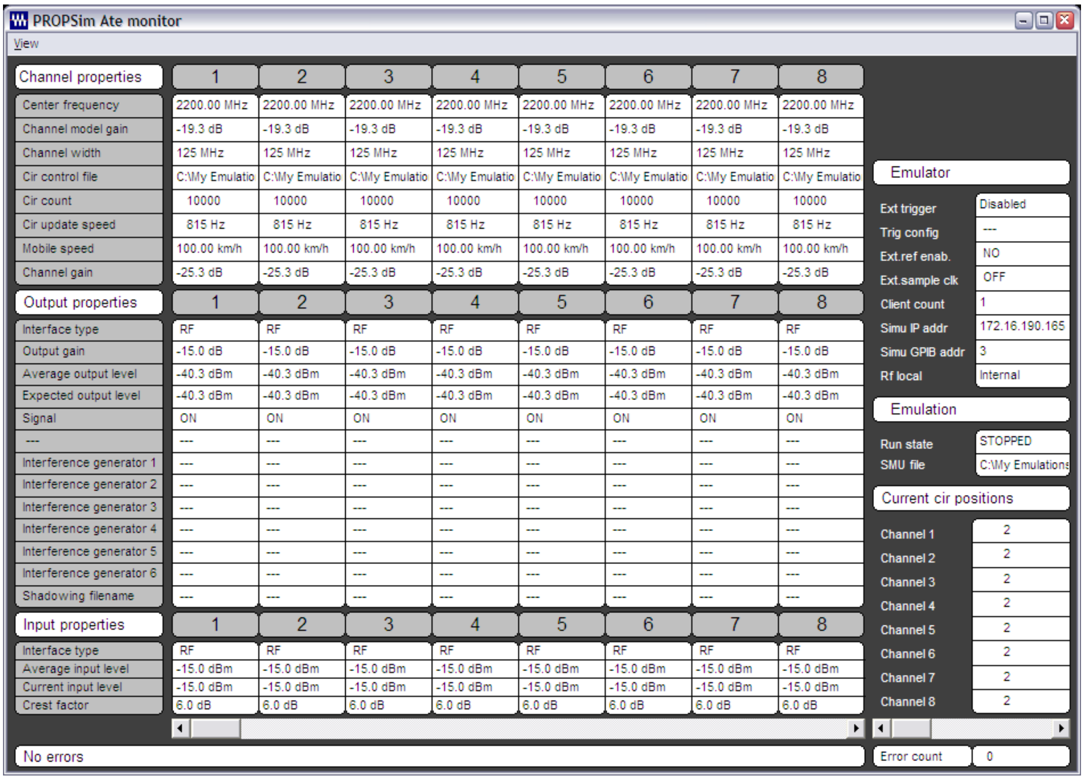


Figure 174. ATE monitor in local mode

8.3.1 Local and remote modes

The monitor will be in local mode in the emulator. It will be started automatically when the emulator receives ATE command to open emulation; either from LAN or GPIB interface.

The Ate monitor can also be started in remote mode. In remote mode, additional **Connect** menu exists (see Figure 175).

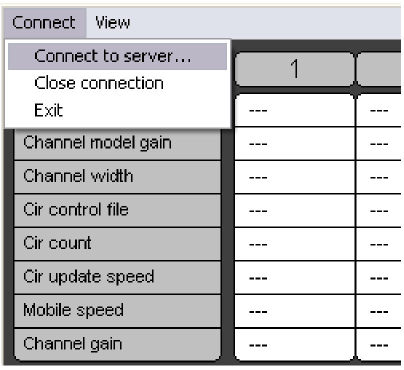


Figure 175. Connect menu

By using the connect menu the user can connect to the emulator just by entering its IP-address (or the name of the emulator, if known).

Figure 176 shows the dialog that will open when **Connect to server...** menu entry is selected.

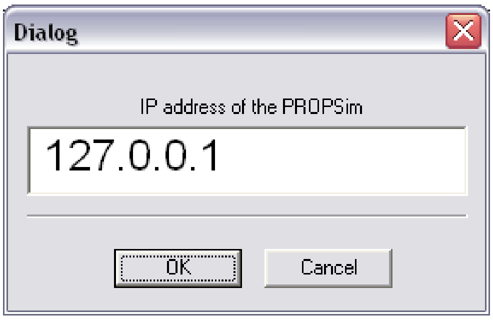


Figure 176. Connecting to EB Propsim

The IP-address of the emulator must be entered with dot notation or as a name of the emulator. Both can be obtained with IPCONFIG /ALL command in the command window of the emulator.

Command window can be opened by selecting **System -> Command Prompt** in the main menubar.

8.3.2 Error window

All possible errors will be shown in the log window. Also the date and time of the errors are shown. The log window (see Figure 177) will only show the errors received when the monitor has been open and not those happened earlier. The last received error is shown at the bottom line of the monitor window.

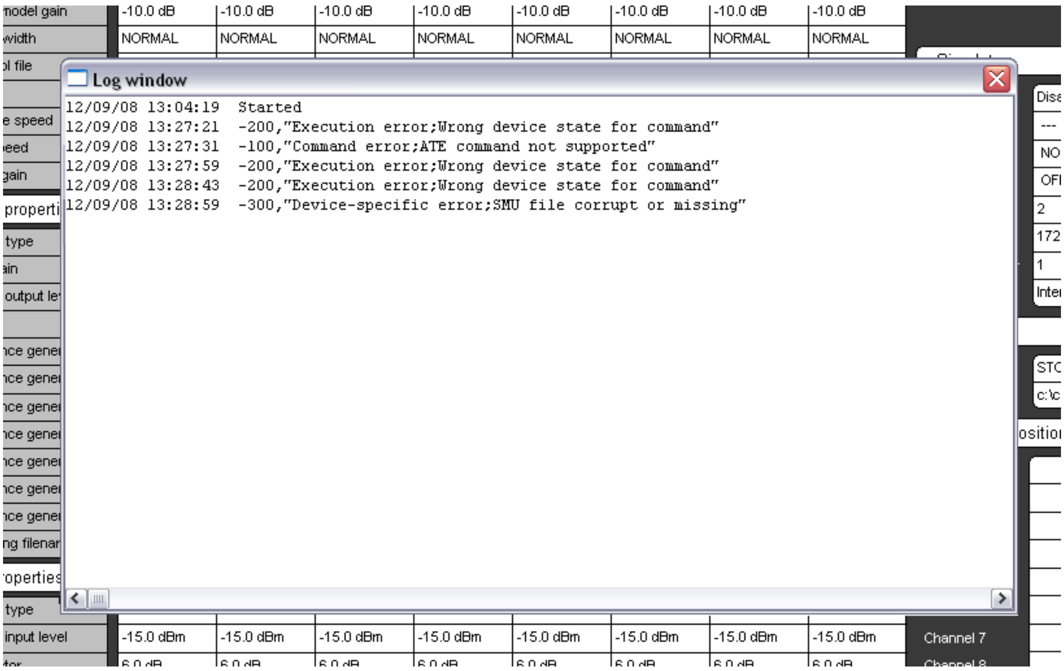


Figure 177. Log window

The window can be cleared with **Clear log window** menu entry, see Figure 178. Entries on the log window can be copied to the clipboard by selecting them with a mouse and selecting **Copy to clipboard** entry from menu that opens with right mouse button.

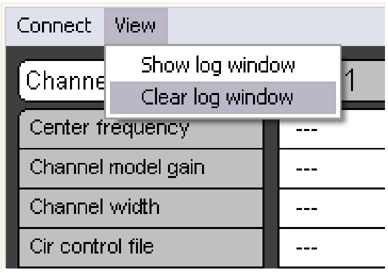


Figure 178. Clearing the log window

8.3.3 ATE monitor display

The monitor will show properties of 32 inputs, 32 outputs and 128 channels. Only the properties for channels / inputs / outputs that are used in the emulation are shown.

Each channel, input and output has its own column. By clicking the shown value with left mouse button it will be shown in its entirety. The slider on the bottom of the screen slides the main window sideways to show all the channels.

8.3.3.1 Channel properties

Channel parameters shown in Figure 179 can be monitored.

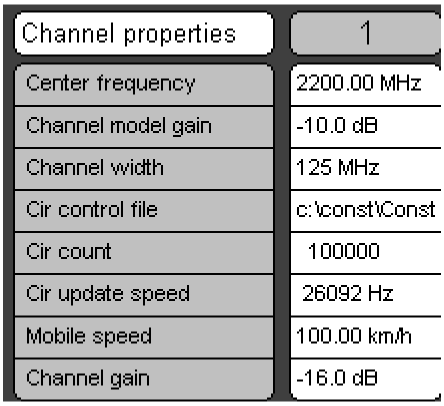


Figure 179. Channel properties

- Center frequency:
  - center frequency of the model in MHz
- Channel model gain
  - gain of the channel model only
- Channel width
  - the bandwidth of the device
- Cir control file
  - name of the channel model file (.IR or .TAP)
- Cir count
  - impulse response count of the channel
- Cir update speed
  - current update speed in Hz

- Mobile speed
  - current mobile speed in km/h
- Channel gain
  - total gain of the channel; from input to output

8.3.3.2 Output properties

The output parameters shown in Figure 180 can be monitored.

Output properties	1
Interface type	RF
Output gain	-15.0 dB
Average output level	-41.3 dBm
Expected output level	-41.3 dBm
Signal	ON
---	---
Interference generator 1	CW
Interference generator 2	---
Interference generator 3	---
Interference generator 4	---
Interference generator 5	---
Interference generator 6	---
Shadowing filename	---

Figure 180. Output properties

- Interface type
  - type of the output interface (RF interface supported)
- Output gain
  - gain of the output stage
- Average output level
  - setting value of the average output level
- Expected output level
  - expected average output level based on measured input level
- Signal
  - ON or OFF. Indicates if the output is currently on or off
- Interference generator
  - AWGN, CW or nothing. If nothing, interference is not present on the output.
- Signal to noise ratio
  - indicates the set signal to noise ratio
- Noise bandwidth
  - indicates noise bandwidth
- Noise power density
  - indicates noise power density. Is alternative setting to signal to noise ratio and noise bandwidth
- Shadowing filename
  - if shadowing is enabled and model has shadowing file it will be shown here

8.3.3.3 Input properties

The input parameters shown in Figure 181 can be monitored.

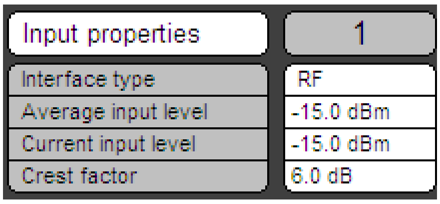


Figure 181. Input properties.

- Interface type
  - type of the input interface (RF interface supported)
- Average input level
  - set value of the average input level
- Current input level
  - measured input level
- Crest factor
  - set value of the crest factor

8.3.3.4 Emulator properties

The emulator parameters shown in Figure 182 can be monitored.

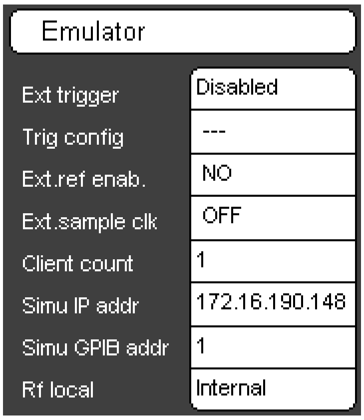


Figure 182. Emulator properties

- Ext trigger
  - indicates whether external trigger is enabled or not
- Trig config
  - can be one of the following:
    - Rising edge
    - Falling edge
    - High level
    - Low level
- Ext.ref enab.
  - indicates if the external reference clock is enabled
- Ext.sample clk
  - indicates if the external sample clock is enabled
- Client count
  - indicates how many Ate monitor interfaces are currently connected to the same emulator
- Simu IP addr
  - shows the emulator IP-address

8.3.3.5 Emulation properties

The emulation parameters shown in Figure 183 can be monitored.

Figure 183. Emulation properties.

- Run state
  - can be RUNNING or STOPPED
- SMU file
  - name of the loaded emulation file

8.3.3.6 Cir position and error count

The positions of channel input responses and the error count are listed in the right side of the ATE monitor, see Figure 184.

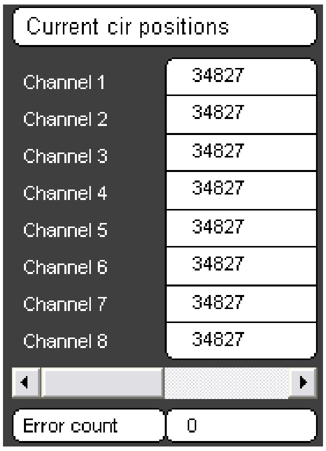


Figure 184. Current Cir positions

- Current cir positions
  - indicates current cir position when the emulation is running. Information is update once every few seconds. Using slider user can see channels 9 to 128.
- Error count
  - indicates how many errors have been received. Can be cleared with the **Clear log window** menu entry

8.4 ATE command syntax

8.4.1 Common syntax

ATE commands have common syntax as follows:

```
<COMMAND> <PARAMETERS>
```
or
```
<COMMAND>
```

where

- <COMMAND> includes a string of characters with no spaces in between. If the last character is ‘?’ then <COMMAND> is a query.
- <PARAMETERS> include a list of parameters separated with commas (no spaces). Note that there is a space between <COMMAND> and <PARAMETERS>.

General notes:

- Strings do not include surrounding hyphens (“...“).
- There can be space characters in the middle of the string (e.g. when defining model name).
- Decimal numbers include at their midst one dot (‘.’).
- Integers should not include other characters than digits.
- Successive commands sent at the same time should be separated with semicolons (‘;’).

8.4.2 Interface specific syntax

Additionally the ATE commands have specific features according to the interface type:

- GPIB strings do not necessarily need any end character.
- LAN strings should have ‘\n’ as an end character.

8.5 ATE command interface

The EB Propsim ATE commands described are divided in groups by their basic functionality and part of emulation they control:

- Common commands, as specified by IEEE 488.2 specification (chapter **8.5.1**).
- System commands for general system control (chapter **8.5.2**)
- Emulation commands (chapter **8.5.3**), for opening, closing and running the emulation.
- Input commands (chapter **8.5.4**) for controlling the input parameters of signal, such as input level and crest factor.
- Output commands (chapter **8.5.5**) for controlling the output parameters of signal, such as output gain.
- Channel commands (chapters **8.5.6**) for controlling the fading channel parameters, such as mobile speed. These also include commands that provide more information of channels parameters.
- Internal Interference Generator option commands (chapter **8.5.7**).
- Channel model information (chapter **8.5.8**) and Signal routing commands (chapter **8.5.9**) provide information of
  - channel model
  - control of reference signal
  - usage of front panel signal connectors of the emulator
- RF Local control commands for Internal Local Oscillator option (chapter **8.5.10**).
- Commands for controlling external trigger (chapter **8.5.11**).
- Multiple emulation loading commands (chapter **8.5.12**).

The ATE commands are presented in form of (for example) “SYSTem:ERRor?”. Commands are not case sensitive; capital letters denote the short version of command parts EB Propsim accepts. That is, for above example commands, all forms “system:error?”, “syst:err?”, “system:err?” and “syst:error?” are acceptable alternatives.

Examples presented here contain comments for user (lines beginning with double slashes, “//”), the command and optional reply from the emulator.

Most EB Propsim ATE commands are only available when emulation has been opened. Thus, the normal usage of command follows the basic sequence below:

- Open emulation (*calculate:filter:file* –command, chapter **8.5.3.1**)
- Emulation parameter adjustments
- Running emulation, doing measurements
- Closing emulation

8.5.1 Common commands

8.5.1.1 *CLS

Clear status
Syntax:
`*CLS`
Description:
This command clears/reset to default partly the status data structures. The following registers are cleared or reset:

- Error/Event Queue
- Operation Status Event Register
- Questionable Status Event Register
- Service Request Status Event Register
- Standard Event Status Register

- Status Byte Event Register
- Status Byte Condition Register: MAV bit
- Status Byte Condition Register: Error/Event Queue bit
- Device goes to operation command idle state.
- Device goes to operation complete query idle state.

Example:
```
*cls
```

8.5.1.2 *ESE

Set Standard Event Status Enable Register
Syntax:
```
*ESE <register value>
```
Description:
This query allows the user to determine the contents of the Standard Event Status Enable Register.
Example:
```
// Set Standard Event Status Enable Register so that
// query error is not shown in status or put
// to error/event queue.
*ese 253
```

8.5.1.3 *ESE?

Get Standard Event Status Enable Register
Syntax:
```
*ESE?
```
Description:
Query returns the contents of the Standard Event Status Enable Register and clears it.
Example:
```
*ese?
0
```

8.5.1.4 *ESR?

Get Standard Event Status Register
Syntax:
```
*ESR?
```
Description:
Query returns the contents of the Standard Event Status Register and clears it.
Example:
```
*esr?
0
```

8.5.1.5 *IDN?

Get identification
Syntax:
```
*IDN?
```
Description:
Query returns the identification of the ATE device.
Example:
```
*idn?
Company Name,Device Name,Serial Number,Firmaware Version Number
```

8.5.1.6 *OPC

Operation complete
Syntax:
`*OPC`
Description:
This command causes the device to set the Operation Complete bit in the Standard Event Status Event Register to 1 after there are no more pending operations.
Example:
```
*opc
```

8.5.1.7 *OPC?

Get operation complete
Syntax:
`*OPC?`
Description:
This query causes the device to return ASCII "1" after there are no more pending operations.
Example:
```
*opc?
1
```

8.5.1.8 *RST

Reset command
Syntax:
`*RST`
Description:
This command performs device reset:

- Disconnects local user, if connected (from Release 1.2.1 onwards)
- Closes the emulation.
- Device goes to operation command idle state.
- Device goes to operation complete query idle state.

Example:
```
*rst
```

8.5.1.9 *SRE

Set Service Request Enable Register
Syntax:
`*SRE <register value>`
Description:
This command sets the contents of the Service Request Enable Register.
Example:
```
// Set Service Request Enable Register so that neither standard
// events or errors cause service request for the ATE client
*sre 219
```

8.5.1.10 *SRE?

Get Service Request Enable Register
Syntax:
`*SRE?`
Description:
This query returns the contents of the Service Request Enable Register.

Example:
`*sre?`
191

8.5.1.11 *STB?

Get Status Byte Condition Register
Syntax:
`*STB?`
Description:
This query returns the contents of the Status Byte Condition Register.
Example:
`*stb?`
0

8.5.1.12 *TST?

Get self test results
Syntax:
`*TST?`
Description:
This query returns the self test results of the device.
Possible results:
0, Self test completed with no errors
1, Self test not completed
2, Self test completed with errors
Example:
`*tst?`
0

8.5.1.13 *WAI

Wait to continue
Syntax:
`*WAI`
Description:
This command shall prevent the device from executing any further commands or queries until the no-operation-pending flag is TRUE.
Example:
`*wai`

8.5.2 System commands

8.5.2.1 SYSTem:ERRor?

Get first error from queue
Syntax:
`SYSTem:ERRor?`
Description:
This query returns the oldest error/event message from the error/event queue. Errors and events are described in chapter **8.6**.
Example:
```
syst:err?
0,"No error"
```

8.5.2.2 SYSTem:VERSion?

Get SCPI version number
Syntax:
```
SYSTem:VERSion?
```
Description:
With this command the SCPI version number can be queried. Used version number in the EB Propsim should be 1999.0.
Example:
```
syst:vers?
1999.0
```

8.5.2.3 SYSTem:RESet

System reset
Syntax:
```
SYSTem:RESet
```
Description:
This command performs system reset:
Closes the emulation.
Example:
```
syst:res
```

8.5.3 Emulation Control

8.5.3.1 CALCulate:FILTer:FILE

Open emulation
Syntax:
```
CALCulate:FILTer:FILE  <filename>
```
Description:
This command opens emulation defined in file <filename>. Some communication interface tools require the folder name separators to be duplicated. For example: e:\\test\\my emulation.smu.
Example:
```
// Open emulation e:\1to1 class.smu.
calc:filt:file e:\1to1 class.smu
```

8.5.3.2 DIAGnostic:SIMUlation:MODel:CONTinuous?

Check if emulation is continuous
Syntax:
```
DIAGnostic:SIMUlation:MODel:CONTinuous?
```
Description:
This request checks whether emulation is continuous, i.e. all the channel models in emulation are continuous. It should be noted that even though emulation is not continuous, currently it is run as if it would be. In other words: Emulation is not automatically stopped at the end of the emulation, but rather continued from the start after that. Possible return values are:

0 Emulation is not continuous
1 Emulation is continuous

Example:
```
// Check if emulation is continuous.
diag:simu:mod:cont?
1
```

8.5.3.3 DIAGnostic:SIMUlation:MODel:STATE?

Get emulation state information
Syntax:
`DIAGnostic:SIMUlation:MODel:STATE?`
Description:
This request returns information on currently run emulation. The format of the answer is as follows:

< i >,<cir number of channel i+1>,<total emulation time>,
< i+1 >,<cir number of channel i+1>,<total emulation time>, …
…, < n >,<cir number of channel n>,<total emulation time>

where

i is number of channel
N is last channel

Example:
```
// Get emulation information. Actual response from Propsim
// does not contain line breaks; added here for clarity.
diag:simu:model:state?
1,345,2.3,
2,345,2.3,
3,345,2.3,
4,345,2.3,
5,99,2.3,
6,99,2.3,
7,99,2.3
```

8.5.3.4 DIAGnostic:SIMUlation:MODel:INFO?

Get emulation model information
Syntax:
`DIAGnostic:SIMUlation:MODel:INFO?`
Description:
This request returns information on currently run emulation. The format of the answer is as follows:

<number of inputs>,<number of channels>,<number of outputs>

Example:
```
// Get emulation information (2 inputs, 4 channels, 2 outputs)
diag:simu:model:info?
```
2,4,2

8.5.3.5 DIAGnostic:SIMUlation:GO

Run emulation
Syntax:
`DIAGnostic:SIMUlation:GO`
Description:
This command runs emulation.
Example:
```
// Run emulation.
diag:simu:go
```

8.5.3.6 DIAGnostic:SIMUlation:GOTO

Goto a CIR on the specific channel
Syntax:
`DIAGnostic:SIMUlation:GOTO <channel number>,<cir>`
Description:
This command moves the emulation running point to a given CIR on the specific channel. Possible other channels are also run during the time. In other words, the synchronization between channels is kept.
Notes:

- The emulation must be stopped or paused in order to carry out this operation.
- Goto operation to very far in the emulation (i.e. millions of CIRs) may take long time.

Example:
```
// Goto cir 99 on channel 1.
diag:simu:goto 1,99
```

8.5.3.7 DIAGnostic:SIMUlation:STOP

Pause emulation
Syntax:
`DIAGnostic:SIMUlation:STOP`
Description:
This command stops emulation without rewinding to the start of the emulation, i.e. performs pause operation.
Example:
```
// Pause emulation.
diag:simu:stop
```

8.5.3.8 DIAGnostic:SIMUlation:GOStart

Stop emulation, rewinding to start
Syntax:
`DIAGnostic:SIMUlation:GOStart`
Description:
This command stops emulation and rewinds to the start of the emulation, i.e. performs stop operation.
Example:
```
// Stop emulation.
diag:simu:gos
```

8.5.3.9 DIAGnostic:SIMUlation:CONTinue

Continue paused emulation
Syntax:
`DIAGnostic:SIMUlation:CONTinue`
Description:
This command continues paused (see 8.5.3.7) emulation.
Example:
```
// Continue emulation.
diag:simu:cont
```

8.5.3.10 DIAGnostic:SIMUlation:STEP

Step emulation into next cir

Syntax:
`DIAGnostic:SIMUlation:STEP`
Description:
This command steps emulation (which is not running) to next channel impulse response change on any channel of the emulation.
Example:
```
// Step emulation.
diag:simu:step
```

8.5.3.11 DIAGnostic:SIMUlation:STATE?

Get emulation running state
Syntax:
`DIAGnostic:SIMUlation:STATE?`
Description:
This request returns the running state of the emulation. Emulation is being run after calling go or continue on emulation. Otherwise it is not being run. Possible return values are:

STOPPED	Emulation is not being run
RUNNING	Emulation is being run
CLOSED	Emulation has not been loaded

Example:
```
//Get emulation running state.
diag:simu:state?
RUNNING
```

8.5.3.12 DIAGnostic:SIMUlation:MODEL:KEEPPERiod

Keep emulation period
Syntax:
`DIAGnostic:SIMUlation:MODEL:KEEPPERiod <keep state>`
Description:
This setting will attempt to keep emulation period same when the center frequency of the channel changes, effectively changing the mobile speed instead of CIR update rate. Valid input values are 1 (to enable keeping period) and 0 (to disable setting). This setting is not saved when emulation is closed.
Example:
```
// Keep emulation period same on center frequency change
diag:simu:model:keepper 1
```

8.5.3.13 DIAGnostic:SIMUlation:MODEL:KEEPPERiod?

Request keep state
Syntax:
`DIAGnostic:SIMUlation:MODEL:KEEPPERiod?`
Description:
Returns whether system attempts to keep emulation period the same when center frequency is changed. Returns 1 if keeping period is enabled, and 0 if not. This setting is not saved when emulation is closed.
Example:
```
// Request whether emulation's period is kept or not
diag:simu:model:keepper?
1
```

8.5.3.14 DIAGnostic:SIMUlation:CLOSE

Close emulation

Syntax:

```
DIAGnostic:SIMUlation:CLOSE
```

Description:

This command closes open emulation.

Example:

```
// Close emulation.
diag:simu:close
```

8.5.4 Channel Input Settings

8.5.4.1 INPut:ENable

Set channel input state (enable / disable)

Syntax:

```
INPut:ENable <input number>,<set value>
```

Description:

This command enables or disables the channel input. Possible set values are:

0 Disable input
1 Enable input

Example:

```
// Disable the input of channel 2.
inp:en 2,0
// Enable the input of channel 2.
inp:en 2,1
```

8.5.4.2 INPut:ENable?

Get channel input state (enable / disable)

Syntax:

```
INPut:Enable? <input number>
```

Description:

This command queries the state of the channel input (enabled or disabled). Possible return values are:

0 Input is disabled
1 Input is enabled

Example:

```
// Query the input state of channel 2.
inp:en? 2
1
// Return status is 1 (input is enabled)
```

8.5.4.3 INPut:LEVel:AMPlitude:CH

Set average input level

Syntax:

```
INPut:LEVel:AMPlitude:CH <input number>,<amplitude value>
```

Description:

This command sets the average input level of the specific channel input in dBm.

Example:

```
// Set average input level of channel input 2 to –18.2 dBm.
inp:lev:amp:ch 2,-18.2
```

8.5.4.4 INPut:LEVel:AMPlitude:CH?

Get average input level
Syntax:
`INPut:LEVel:AMPlitude:CH? <input number>`
Description:
This command retrieves the average input level of the specific channel input in dBm.
Example:
```
// Get average input level of channel input 8.
inp:lev:amp:ch? 8
-15
```

8.5.4.5 INPut:LEVel:AMPlitude:LIMits?

Get average input level limits
Syntax:
`INPut:LEVel:AMPlitude:LIMits? <input number>`
Description:
This command retrieves the average input level limit values of the specific channel input in dBm. Level can not be set outside the limits.
Limit values are returned as follows:
`<lower limit>,<higher limit>`
Example:
```
// Get average input level limits of channel input 1.
inp:lev:amp:lim? 1
-23,0
```

8.5.4.6 INPut:LEVel:MEASure?

Get average input level and crest factor
Syntax:
`INPut:LEVel:MEASure? <input number>,<measurement time>`
Description:
This command measures and returns the average input level (in dBm) and crest factor (in dB) of the specific channel input. A failed measurement (no input signal or too strong output signal) will produce device specific error. Measurement time can be 1, 3, 10 or 30 seconds.
Example:
```
// Get average input level and crest factor of input 1 using 3 seconds
measurement time.
inp:lev:meas? 1,3
-21.4,4
```

8.5.4.7 INPut:LEVel:AUTOSET

Set average input level and crest factor values
Syntax:
`INPut:LEVel:AUTOSET <input number>,<measurement time>`
Description:
This command measures average input level and crest factor and sets the parameters of the input accordingly. Measurement time can be 1, 3, 10 or 30 seconds. A failed measurement (no input signal or too strong output signal) will not change previous settings but produces device-specific error. If input number is set to 0, all inputs will be autoset simultaneously.
Example:
```
// Set average input level and crest factor of input 1 using 3 seconds
measurement time.
inp:lev:autoset 1, 3
```

8.5.4.8 INPut:CREst:SET

Set crest factor for channel input
Syntax:
`INPut:CREst:SET <input number>,<crest factor value>`
Description:
This command sets the crest factor for channel input in dB.
Example:
```
// Set crest factor of channel input 1 to 4 dB.
inp:cre:set 1,4
```

8.5.4.9 INPut:CREst:GET?

Get crest factor of channel input
Syntax:
`INPut:CREst:GET? <input number>`
Description:
This command retrieves the crest factor for channel input in dB.
Example:
```
// Get crest factor of channel input 1.
inp:cre:get? 1
4
```

8.5.4.10 INPut:CREst:LIMits?

Get crest factor limits of channel input
Syntax:
`INPut:CREst:LIMits? <input number>`
Description:
This command retrieves the crest factor limit values of the specific channel input in dB. Crest factor can not be set outside the limits. If attempted, factor is automatically set to closest acceptable value.
Limit values are returned as follows:
`<lower limit>,<higher limit>`
Example:
```
// Get crest factor limits of channel input 6:
inp:cre:lim? 6
0,23
```

8.5.4.11 INPut:IF:TYPE?

Get interface type of channel input
Syntax:
`INPut:IF:TYPE? <input number>`
Description:
This command retrieves the interface type of channel input. Possible return types are:
`RF`
Example:
```
// Get interface type of channel input 1.
inp:if:type? 1
RF
```

8.5.4.12 INPut:PHAse:CH

Set channel input phase register value
Syntax:
`INPut:PHAse:CH <input number>,<phase register value>`

Description:
This command sets the phase of the specific channel input by defining the affecting register value. The adjustment range is 1200 … 3200. Each step represents 0.2 degrees.
Example:
```
// Set channel input 2 phase register value to 2200.
inp:pha:ch 2,2200
```

8.5.4.13 INPut:PHAse:CH?

Get channel input phase register value
Syntax:
```
INPut:PHAse:CH? <input number>
```
Description:
This request returns the phase register value of the specific channel input.
Example:
```
// Get channel input 1 phase register value.
inp:pha:ch? 1
1800
```

8.5.4.14 INPut:PHAse:LIMits?

Get channel input phase register value limits
Syntax:
```
INPut:PHAse:LIMits? <input number>
```
Description:
This command retrieves the phase register limit values of the specific channel input. Register value can not be set outside the limits. If attempted, value is automatically set to closest acceptable value.
Limit values are returned as follows:
```
<lower limit>,<higher limit>
```
Example:
```
// Get phase register value limits of channel input 6:
inp:pha:lim? 6
1200,3200
```

8.5.4.15 INPut:LOSS:SET

Set the input loss.

Syntax:
```
INPut:LOSS:SET <input number>,<loss>
```
Description:
This command sets the input loss in dB.
Example:
```
// Set input 1 loss to 2.5 dB
inp:loss:set 1,2.5
```

8.5.4.16 INPut:LOSS:GET?

Get the input loss
Syntax:
```
INPut:LOSS:GET? <input number>
```
Description:
This command gets the input loss in dB.
Example:
```
// Get input 1 loss
```

```
inp:loss:get? 1
2.5
```

8.5.4.17 INPut:LOSS:LIMits?

Get the input loss limits
Syntax:
`INPut:LOSS:LIMits? <input number>`
Description:
This command gets the input loss limits in dB.
Example:
```
// Get input 1 loss limits
inp:loss:lim? 1
-100,100
```

8.5.4.18 INPut:MEASure:MODE:SET

Set the measurement mode of an input
Syntax:
`INPut:MEASure:MODE:SET <input number>,<measurement mode>`
Description:
Measurement mode is a numeric value. Available measurement modes are:
0 DISABLED
1 BASIC
2 CONTINUOUS
3 BURST
Example:
```
// Set measurement mode to continuous on input 2
inp:meas:mode:set 2,2
```

8.5.4.19 INPut:MEASure:MODE:GET?

Get the measurement mode of an input
Syntax:
`INPut:MEASure:MODE:GET? <input number>`
Description:
The measurement modes are:
0 DISABLED
1 BASIC
2 CONTINUOUS
3 BURST
Example:
```
// Get measurement mode of input 2. Is in continuous mode.
inp:meas:mode:get? 2
2
```

8.5.4.20 INPut:MEASure:FREEZE

Freeze the measurement result
Syntax:
`INPut:MEASure:FREEZE <input number>,<mode>`
Description:
The measurement taken from an input can be “freezed” as the last measured value or the measurements can be continued.
0 MEASURE
1 FREEZE

Example:
// Set the measurement mode of input 1 to frozen
inp:meas:freeze 1,1

8.5.4.21 INPut:MEASure:FREEZE?

Query the measuring on an input
Syntax:
`INPut:MEASure:FREEZE? <input number>`
Description:
This command retrieves status information of the input measurements i.e. whether an input is “frozen” as the last measured value or the input is continuously measured.
0 MEASURE
1 FREEZE
Example:
```
// Query the mode of input 1. It is measuring.
inp:meas:freeze? 1
0
```

8.5.4.22 INPut:MEASure:BURST:TRIGger:SET

Set the input trigger level in burst measurement mode.
Syntax:
`INPut:MEASure:BURST:TRIGger:SET <input number>,<trigger value>`
Description:
Set the trigger level of an input. This command is available only in burst measurement mode.
Example:
```
// Set the trigger level of input 1 to -10.1 dBm.
inp:meas:burst:trig:set 1,-10.1
```

8.5.4.23 INPut:MEASure:BURST:TRIGger:GET?

Get the input trigger level in burst measurement mode.
Syntax:
`INPut:MEASure:BURST:TRIGger:GET? <input number>`
Description:
Get the current trigger level of an input. This command is available only in burst measurement mode.
Example:
```
// Query the trigger level setting of input 1.
inp:meas:burst:trig:get? 1
-10.1
```

8.5.4.24 INPut:MEASure:BURST:TRIGger:LIMits?

Get the input trigger level limits in burst measurement mode.
Syntax:
`INPut:MEASure:BURST:TRIGger:LIMits? <input number>`
Description:
Query the trigger limits of an input. This command is available only in burst mode measurements.
Example:
```
// Query the trigger level limits of input 1. They are -50 and 0 dB.
inp:meas:burst:trig:lim? 1
-50,0
```

8.5.4.25 INPut:MEASure:BURST:AVERages:SET

Set the burst measurement length in samples

Syntax:

`INPut:MEASure:BURST:AVERages:SET <input number>,<sample count>`

Description:

Set the burst measurement length in samples. This command is available only in burst measurement mode.

Example:

```
// Set the burst measurement length of input 1 to 234567 samples.
```
inp:meas:burst:aver:set 1,234567

8.5.4.26 INPut:MEASure:BURST:AVERages:GET?

Get the burst measurement length in samples.

Syntax:

`INPut:MEASure:BURST:AVERages:GET? <input number>`

Description:

Get the current burst measurement length of an input. This command is available only in burst measurement mode.

Example:

```
// Query the measurement length of input 1. It is 234567 samples.
inp:meas:burst:aver:get? 1
234567
```

8.5.4.27 INPut:MEASure:BURST:AVERages:LIMits?

Get the burst measurement length limits in samples.

Syntax:

`INPut:MEASure:BURST:AVERages:LIMits? <input number>`

Description:

Get the measurement length limits in samples. This command is available only in burst measurement mode.

Example:

```
// Query the measurement length limits of input 1. The limits are 4 and
134217728.
inp:meas:burst:aver:lim? 1
4,134217728
```

8.5.4.28 INPut:MEASure:RESult:GET?

Returns the latest measurement of the input.

Syntax:

`INPut:MEASure:RESult:GET? <input number>`

Description:

Get the latest measurement result of an input.

Example:

```
// Query the last measured value of input 1. It is -10.1 dBm.
inp:meas:res:get? 1
-10.1
```

8.5.4.29 INPut:MEASure:STATus:GET?

Returns the measurement status of the input.

Syntax:
```
INPut:MEASure:STATus:GET? <input number>
```
Description:
Get the current measurement status of an input. Returned values:

0 Idle
1 No Result
2 Disabled
3 Freezed
4 Measuring
5 New Result
6 Result Not Ready

Example:
```
// Query the status of input 1. It is "New result".
inp:meas:stat:get? 1
5
```

8.5.4.30 INPut:MEASure:RESult:OFFSET:SET

Set the offset for input measurements
Syntax:
```
INPut:MEASure:RESult:OFFSET:SET <input number>,<offset>
```
Description:
This control sets the offset to an input. The resolution for offset is one decimal place.
Example:
```
// Set 0.7 dB offset to measurements on input 1
inp:meas:res:offset:set 1,0.7
```

8.5.4.31 INPut:MEASure:RESult:OFFSET:GET?

Query the offset value of measurements on an input
Syntax:
```
INPut:MEASure:RESult:OFFSET:GET? <input number>
```
Description:
Get the measurement offset of an input. The resolution for offset is one decimal place.
Example:
```
// Set an offset to measurements on input 1. It is 0.7 dB.
inp:meas:res:offset:get? 1
0.7
```

8.5.4.32 INPut:MEASure:RESult:OFFSET:LIMits?

Get the offset limits for input measurements
Syntax:
```
INPut:MEASure:RESult:OFFSET:LIM <input number>
```
Description:
Get the offset limits of an input. The resolution for offset is one decimal place.
Example:
```
// Get the offset limits of input 1. They are -3.0 and +3.0 dB.
inp:meas:res:offset:lim? 1
-3.0,3.0
```

8.5.4.33 INPut:ATTenuator:ENable

Set channel input attenuator state (enable / disable)

Syntax:
```
INPut:ATTenuator:ENable <input number>,<set value>
```
Description:
This command enables or disables the channel input attenuator. Possible set values are:
0 Disable attenuator
1 Enable attenuator
Example:
```
// Disable the input 2 attenuator
inp:att:en 2,0
// Enable the input 2 attenuator
inp:att:en 2,1
```

8.5.4.34 INPut:ATTenuator:ENable?

Get channel input attenuator state (enable / disable)
Syntax:
```
INPut:ATTenuator Enable? <input number>
```
Description:
This command queries the state of the channel input attenuator (enabled or disabled). Possible return values are:
0 Input attenuator is disabled
1 Input attenuator is enabled
Example:
```
// Query the input state of channel 2.
inp:att:en? 2
1
// Return status is 1 (input attenuator is enabled)
```

8.5.5 Channel Output Settings

8.5.5.1 OUTPut:ENable

Set channel output state (enable / disable)
Syntax:
```
OUTPut:ENable <output number>,<set value>
```
Description:
This command enables or disables the channel output. Possible set values are:
0, Disable output
1, Enable output
Example:
```
// Disable the output of channel 2.
outp:en 2,0
// Enable the output of channel 2.
outp:en 2,1
```

8.5.5.2 OUTPut:ENable?

Get channel output state (enable / disable)
Syntax:
```
OUTPut:Enable? <output number>
```
Description:
This command queries the state of the channel output state (enable or disable). Possible return values are:
0, Output is disabled
1, Output is enabled
Example:

```
// Query the output state of channel 2.
outp:en? 2
1
// return status is 1 (output is enabled)
```

8.5.5.3 OUTPut:LEVel:AMPlitude:CH

Set average output level
Syntax:
`OUTPut:LEVel:AMPlitude:CH <output number>,<amplitude value>`
Description:
This command sets the average output level of the specific channel output in dBm.
Example:

```
// Set average output level of channel output 2 to -40 dBm.
outp:lev:amp:ch 2,-40
```

8.5.5.4 OUTPut:LEVel:AMPlitude:CH?

Get average output level
Syntax:
`OUTPut:LEVel:AMPlitude:CH? <output number>`
Description:
This command retrieves the average output level of the specific channel output in dBm.
Example:

```
// Get average output level of channel output 7.
outp:lev:amp:ch? 7
-40
```

8.5.5.5 OUTPut:LEVel:AMPlitude:LIMits?

Get average output level limits
Syntax:
`OUTPut:LEVel:AMPlitude:LIMits? <output number>`
Description:
This command retrieves the average output level limit values of the specific channel output in dBm. Level can not be set outside the limits.
Limit values are returned as follows:
`<lower limit>,<higher limit>`
Example:

```
// Get average output level limits of channel output 1.
outp:lev:amp:lim? 1
-68.8401,-23.8401
```

8.5.5.6 OUTPut:GAIN:CH

Set output gain
Syntax:
`OUTPut:GAIN:CH <output number>,<gain value>`
Description:
This command sets the gain of the specific channel output in dB.
Example:

```
// Set gain of channel output 2 to -5 dB.
outp:gain:ch 2,-5
```

8.5.5.7 OUTPut:GAIN:CH?

Get output gain

Syntax:

```
OUTPut:GAIN:CH? <output number>
```

Description:

This command retrieves the gain of the specific channel output in dB.

Example:

```
// Get gain of channel output 2.
outp:gain:ch? 2
-5
```

8.5.5.8 OUTPut:GAIN:LIMits?

Get output gain limits

Syntax:

```
OUTPut:GAIN:LIMits? <output number>
```

Description:

This command retrieves the gain limit values of the specific channel output in dB. Gain can not be set outside the limits. If attempted, gain is automatically set to closest acceptable value.

Limit values are returned as follows:

```
<lower limit>,<higher limit>
```

Example:

```
// Get gain limits of channel output 1.
outp:gain:lim? 1
-45,0
```

8.5.5.9 OUTPut:PHAse:CH

Set channel output phase register value

Syntax:

```
OUTPut:PHAse:CH <output number>,<phase register value>
```

Description:

This command sets the phase of the specific channel output by defining the affecting register value. The adjustment range is 1200 … 3200. Each step represents 0.2 degrees.

Example:

```
// Set channel output 6 phase register value to 2200.
outp:pha:ch 6,2200
```

8.5.5.10 OUTPut:PHAse:CH?

Get channel output phase register value

Syntax:

```
OUTPut:PHAse:CH? <output number>
```

Description:

This request returns the phase register value of the specific channel output. The adjustment range is 1200 … 3200. Each step represents 0.2 degrees.

Example:

```
// Get channel output 1 phase register value.
outp:pha:ch? 1
1800
```

8.5.5.11 OUTPut:PHAse:LIMits?

Get channel output phase register value limits

Syntax:
`OUTPut:PHAse:LIMits? <output number>`
Description:
This command retrieves the phase register limit values of the specific channel output. Register value can not be set outside the limits. If attempted, value is automatically set to closest acceptable value.
Limit values are returned as follows:
`<lower limit>,<higher limit>`
Example:
```
// Get phase register value limits of channel output 2.
outp:pha:lim? 2
1200,3200
```

8.5.5.12 OUTPut:SIGnal:STatus

Set the status of signal path to output
Syntax:
`OUTPut:SIGnal:STatus <output number>,<signal status>`
Description:
This command sets the status of signal path to specific channel output. Possible settings are:
```
0,  Signal disabled
1,  Signal enabled
```
Example:
```
// Disable signal to channel output 1.
outp:sig:st 1,0
```

8.5.5.13 OUTPut:SIGnal:STatus?

Get the status of signal path to output
Syntax:
`OUTPut:SIGnal:STatus? <output number>`
Description:
This command retrieves the status of signal path to specific channel output. Possible return values are:
```
0,  Signal disabled
1,  Signal enabled
```
Example:
```
// Get the status of signal to channel output 1.
outp:sig:st? 1
0
```

8.5.5.14 OUTPut:IF:TYPE?

Get interface type of channel output
Syntax:
`OUTPut:IF:TYPE? <output number>`
Description:
This command retrieves the interface type of channel output. Possible return types are:
`RF`
Example:
```
// Get interface type of channel output 1.
outp:if:type? 1
RF
```

8.5.5.15 OUTPut:LOSS:SET

Set the output loss.

Syntax:
```
OUTPut:LOSS:SET <output number>,<loss>
```
Backward compatible syntax:
```
OUTPut:CABLELoss:SET <output number>,<loss>
```
Description:
This command sets the output loss in dB.
Example:
```
// Set output 1 loss to 2.5 dB
outp:loss:set 1,2.5
```

8.5.5.16 OUTPut:LOSS:GET?

Get the output loss
Syntax:
```
OUTPut:LOSS:GET? <output number>
```
Backward compatible syntax:
```
OUTPut:CABLELoss:GET? <output number>
```
Description:
This command gets the output loss in dB.
Example:
```
// Get output 1 loss
outp:loss:get? 1
2.5
```

8.5.5.17 OUTPut:LOSS:LIMits?

Get the output loss limits
Syntax:
```
OUTPut:LOSS:LIMits? <output number>
```
Backward compatible syntax:
```
OUTPut:CABLELoss:LIMits? <output number>
```
Description:
This command gets the output loss limits in dB.
Example:
```
// Get output 1 loss limits
outp:loss:lim? 1
-30,80
```

8.5.5.18 OUTPut:MEASure:RESult:GET?

Get the latest output measurement.
Syntax:
```
OUTPut:MEASure:RESult:GET? <output number>
```
Description:
Get the latest output level measurement.
Example:
```
// Query the last measured value of output 1. It is -30.1 dBm.
outp:meas:res:get? 1
-30.1
```

8.5.5.19 OUTPut:SHADowing:GAIN:GET?

Get the latest shadowing gain on an output.

Syntax:
```
OUTPut:SHADowing:GAIN:GET? <output number>
```
Description:
Get the latest shadowing gain on an output in dB.
Example:
```
// Query the last shadowing gain on output 1. It is -5.0 dB.
outp:shad:gain:get? 1
-5.0
```

8.5.6 Channel Settings

8.5.6.1 CALCulate:FILTer:CENTer:CH

Set center frequency for channel group
Syntax:
```
CALCulate:FILTer:CENTer:CH <channel number>,<frequency>
```
Description:
This command sets the center frequency of the specific channel group in MHz. It should be noted that frequency is set according to the given **channel** number. Frequency is set for given channel and for all the other channels belonging to the same group. Channels belong to same group if at least one of the following is true:

- They have same input
- They have same output

Example:
```
// Set center frequency of channel 1 (and all the channels
// belonging to same channel group) to 900 MHz.
calc:filt:cent:ch 1,900
```

8.5.6.2 CALCulate:FILTer:CENTer:CH?

Get center frequency of channel group
Syntax:
```
CALCulate:FILTer:CENTer:CH? <channel number>
```
Description:
This request retrieves the center frequency of the specific channel group in MHz. It should be noted that channel group is specified according to the given **channel** number. Frequency is retrieved from the channel group the channel belongs to. Channels belong to the same group if at least one of the following is true:

- They have same input
- They have same output

Example:
```
// Get center frequency of the channel group of channel 1
calc:filt:cent:ch? 1
1800
```

8.5.6.3 CALCulate:FILTer:CENTer:LIMits?

Get center frequency limits for group
Syntax:
```
CALCulate:FILTer:CENTer:LIMits? <channel number>
```
Description:
This retrieves the center frequency limits of the specific channel group in MHz. Frequency can not be set outside the limits. If tried, frequency is automatically set to closest acceptable value.

It should be noted that channel group is specified according to the given **channel** number. Limits are retrieved from the channel group the channel belongs to. Channels belong to the same group if at least one of the following is true:

- They have same input
- They have same output

Limit values are returned as follows:

```
<lower limit>,<higher limit>
```

Example:

```
// Get center frequency limits of the channel group of channel 1
calc:filt:cent:lim? 1
350,6000
```

8.5.6.4 DIAGnostic:SIMUlation:FIRUPDatespeed:MANual:CH

Set cir update rate

Syntax:

DIAGnostic:SIMUlation:FIRUPDatespeed:MANual:CH <channel number>,<cir update rate>

Description:

This command sets the channel impulse response update rate of the specific channel in 1/s.

Example:

```
// Set cir update rate of channel 1 to 1000.
diag:simu:firupd:man:ch 1,1000
```

8.5.6.5 DIAGnostic:SIMUlation:FIRUPDatespeed:CH?

Get cir update rate

Syntax:

DIAGnostic:SIMUlation:FIRUPDatespeed:CH? <channel number>

Description:

This request retrieves the channel impulse response update rate of the specific channel in 1/s.

Example:

```
// Get cir update rate of channel 8.
diag:simu:firupd:ch? 8
135000
```

8.5.6.6 DIAGnostic:SIMUlation:FIRUPDatespeed:LIMits?

Get cir update rate limits

Syntax:

DIAGnostic:SIMUlation:FIRUPDatespeed:LIMits? <channel number>

Description:

This request retrieves the channel impulse response update rate limit values of the specific channel in 1/s. Update rate can not be set outside the limits. If attempted, rate is automatically set to closest acceptable value.

Limit values are returned as follows:

```
<lower limit>,<higher limit>
```

Example:

```
// Get cir update rate limits of channel 1.
diag:simu:firupd:lim? 1
0.01,3.90533e+006
```

8.5.6.7 DIAGnostic:SIMUlation:FIRUPDatespeed:LOCKed?

Query whether cir update rate is locked
Syntax:
DIAGnostic:SIMUlation:FIRUPDatespeed:LOCKed? <channel number>
Description:
This request checks whether the channel impulse response update rate is locked (cannot be changed) in the specific channel. Possible return values are:

```
0,  Cir update rate not locked
1,  Cir update rate locked
```

Example:

```
// Check if cir update rate is locked in channel 1.
diag:simu:firupd:lock? 1
0
```

8.5.6.8 DIAGnostic:SIMUlation:MOBilespeed:MANual:CH

Set mobile speed
Syntax:
DIAGnostic:SIMUlation:MOBilespeed:MANual:CH <channel number>,<mobile speed> <possible unit description>
Description:
This command sets the mobile speed of the specific channel. If there is no unit description after the mobile speed setting value the speed is set as km/h. It is possible to define the setting to done as m/s by setting one of the following allowed unit descriptions. Note that unit descriptions are case sensitive.
MS
M/S
m/s

Example:

```
// Set mobile speed of channel 1 to 80 m/s.
diag:simu:mob:man:ch 1,80 m/s
```

8.5.6.9 DIAGnostic:SIMUlation:MOBilespeed:MANual:CHG

Set mobile speed for a channel group
Syntax:
DIAGnostic:SIMUlation:MOBilespeed:MANual:CHG <channel number>,<mobile speed> <possible unit description>
Description:
This command sets the mobile speed for a channel group containing the given channel. A channel group consists of channels which share common components. For example two channels with a same output create one channel group.

If there is no unit description after the mobile speed setting value the speed is set as km/h. It is possible to define the setting to done as m/s by setting one of the following allowed unit descriptions. Note that unit descriptions are case sensitive.
MS
M/S
m/s

Example:

```
// Set mobile speed 80 m/s for a channel group containing channel 1.
diag:simu:mob:man:chg 1,80 m/s
```

8.5.6.10 DIAGnostic:SIMUlation:MOBilespeed:CH?

Get mobile speed in km/h
Syntax:
DIAGnostic:SIMUlation:MOBilespeed:CH? <channel number>
Description:
This request retrieves the mobile speed of the specific channel in km/h.
Example:

```
// Get mobile speed of channel 8.
diag:simu:mob:ch? 8
200
```

8.5.6.11 DIAGnostic:SIMUlation:MOBilespeed:LIMits?

Get mobile speed limits in km/h
Syntax:
DIAGnostic:SIMUlation:MOBilespeed:LIMits? <channel number>
Description:
This request retrieves the mobile speed limit values of the specific channel in km/h. Mobile speed can not be set outside the limits. If tried, speed is automatically set to closest acceptable value.
Limit values are returned as follows:

```
<lower limit>,<higher limit>
```

Example:

```
// Get mobile speed limits of channel 1.
diag:simu:mob:lim? 1
0.00229954,14967.4
```

8.5.6.12 DIAGnostic:SIMUlation:MOBilespeed:LOCKed?

Is mobile speed locked?
Syntax:
DIAGnostic:SIMUlation:MOBilespeed:LOCKed? <channel number>
Description:
This request checks whether the mobile speed is locked (i.e. cannot be changed) in the specific channel. Possible return values are:
0, Mobile speed not locked
1, Mobile speed locked
Example:

```
// Check if mobile speed is locked in channel 1.
diag:simu:mob:lock? 1
0
```

8.5.6.13 DIAGnostic:SIMUlation:MOBilespeed:VALid?

Is mobile speed valid?
Syntax:
DIAGnostic:SIMUlation:MOBilespeed:VALid? <channel number>
Description:
This request checks whether the mobile speed is valid in the specific channel i.e. sample density of the model is not 0. Possible return values are:
0, Mobile speed not valid
1, Mobile speed is valid
Example:

```
// Check if mobile speed is valid in channel 1.
diag:simu:mob:val? 1
1
```

8.5.6.14 DIAGnostic:SIMUlation:MOBilespeed:MPS:CH?

Get mobile speed in m/s

Syntax:

DIAGnostic:SIMUlation:MOBilespeed:MPS:CH? <channel number>

Description:

This request retrieves the mobile speed of the specific channel in m/s.

Example:

```
//Get mobile speed of channel 1.
diag:simu:mob:mps:ch? 1
158.094
```

8.5.6.15 DIAGnostic:SIMUlation:MOBilespeed:MPS:LIMits?

Get mobile speed limits in m/s

Syntax:

DIAGnostic:SIMUlation:MOBilespeed:MPS:LIMits? <channel number>

Description:

This request retrieves the mobile speed limit values of the specific channel in m/s. Mobile speed can not be set outside the limits. If tried, speed is automatically set to closest acceptable value.

Limit values are returned as follows:

```
<lower limit>,<higher limit>
```

Example:

```
// Get mobile speed limits of channel 5.
diag:simu:mob:mps:lim? 5
0.000638762,4573.39
```

8.5.6.16 DIAGnostic:SIMUlation:MODEL:STATIC?

Get current static model state (Disabled / Bypass / Butler bypass)

Syntax:

DIAGnostic:SIMUlation:MODEL:STATIC?

Description:

This request retrieves the static state of the emulation, i.e. channel model bypass or Butler bypass. When static state is enabled (Bypass or Butler bypass), emulation is stopped and all channels are set to one-path constant with same gain as fading model.

Return values are:

0, Bypass disabled (channel model applied)
1, Bypass enabled
2, Butler bypass enabled

Example:

```
// Query static state
diag:simu:model:static?
1
```

8.5.6.17 DIAGnostic:SIMUlation:MODEL:STATIC

Set static model state (Bypass / Butler bypass / disabled)

Syntax:

DIAGnostic:SIMUlation:MODEL:STATIC <state>

Description:

This request sets the static state of the emulation, i.e. channel model bypass, specified by <state>. Possible states are:

0, Bypass disabled (channel model applied)
1, Bypass enabled

2, Butler bypass enabled

Example:

```
// Set static state to Butler bypass
diag:simu:model:static 2
```

8.5.7 Internal Interference Generator control (Optional)

Internal Interference Generator commands are available only if EB Propsim has optional interference sources installed. AWGN and CW interference sources are available.

8.5.7.1 OUTPut:INTERFerence:ADD

Add an interference source to the channel output

Syntax:

OUTPut:INTERFerence:ADD <output number>,<interference identification>,<interference type>

Description:

This command adds an interference source to specific channel output. Interference identification is user-given identifier to the interference; either a plain number (012, leading 0 is discarded) or a string (such as TX2_INTERF). Strings are case-sensitive (i.e. tx1_noise and TX1_NOISE are not same identifiers). Each identifier shall be unique, identifiers already in use can not be given. Possible settings for the interference type are:

1, AWGN
2, Carrier Wave (CW)
7, Filtered AWGN

Example:

```
// Add an AWGN noise source to output 2.
outp :interf:add 2,tx2_interf,1
```

8.5.7.2 OUTPut:INTERFerence:REMove

Remove the interference source from output.

Syntax:

OUTPut:INTERFerence:REMove <interference identification>

Description:

This command removes the interference source from the channel output. The removed interference is defined by the interference identification (see 8.5.7.1).

Example:

```
// Remove the AWGN noise source with identification
// tx2_interf.
outp: interf:rem tx2_interf
```

8.5.7.3 OUTPut:INTERFerence:STatus

Set the status of the specified interference source.

Syntax:

OUTPut:INTERFerence:STatus <interference identification>,<interference status>

Description:

This command sets the status of the interference source. The interference, which status is being set is defined by the interference identification (see 8.5.7.1). Possible status settings are:

0, Interference disabled
1, Interference enabled

Example:

```
// Disable the AWGN noise source with identification
```

```
// tx2_noise.
outp:interf:st tx2_noise,0
```

8.5.7.4 OUTPut:INTERFerence:STatus?

Get the status of the specified interference source.
Syntax:
OUTPut:INTERFerence:STatus? <interference identification>
Description:
This command retrieves the status of the interference source defined by the interference identification. Possible return values are:
0, Interference disabled
1, Interference enabled
Example:

```
// Get the status of the AWGN noise source with
// identification tx2_noise.
outp:interf:st? tx2_noise
1
```

8.5.7.5 OUTPut:INTERFerence:GET?

List all the interferers in emulation.
Syntax:
`OUTPut:INTERFerence:GET?`
Description:
This command gets and lists all the interference sources. The returned list is in the following form “output number, interference identification, type of the interferer, output number, interference identification, type of the interference…” where “interference identification” is the user defined interference identification given by the user when the interference has been added (see Chapter 8.5.7.1) and “type of the interference” tells the type of the interferer in numeric. Possible values for the interference type are:
1, AWGN
2, Carrier Wave (CW)
7, Filtered AWGN
Zero is returned if there are no interference sources in use.
Example:

```
// Get the status of the interference sources. Output 1 has Filtered AWGN
type of interference with identification TX1_AWGN and output 2 CW type or
interference with identification TX2_CW.
outp:interf:get?
1, TX1_AWGN, 7, 2, TX2_CW, 2
```

8.5.7.6 OUTPut:INTERFerence:STRATegy:SET

Set interference adjustment strategy for specified interference.
Syntax:
OUTPut:INTERFerence:STRATegy:SET <interference identification>, <adjustment strategy>
Description:
This command sets the adjustment strategy of the specified interference source. Possible settings are:
0, Constant carrier-to-interference ratio (C/I)
1, Constant interference power
`2, Constant C/I where the interference level is also user-given and fixed (AWGN only).`
This setting can currently not be used when multiple channels are combined to one output (i.e. MISO/MIMO type connections)
Example:

```
// Choose constant carrier-to-interference ratio to
// interference signal with identification tx1_cw.
outp:interf:strat:set tx1_cw,0
```

8.5.7.7 OUTPut:INTERFerence:STRATegy:GET?

Get the interference adjustment strategy of the specified interference.

Syntax:

OUTPut:INTERFEerence:STRATegy:GET? <interference identification>

Description:

This request retrieves the adjustment strategy of the specified interference source. Possible return values are:

0, Constant carrier-to-interference ratio (C/I)

1, Constant interference power

2, Constant C/I where the interference level is also user-given and fixed (AWGN only)

Example:

```
// Get the chosen interference adjustmentstrategy of the
// interference with identification tx1_cw
outp:interf:strat:get? tx1_cw
1
```

8.5.7.8 OUTPut:INTERFerence:POWer:SET

Set the power level of a specific interference signal.

Syntax:

OUTPut:INTERFerence:POWer:SET <interference identification>,<power level>

Description:

This command sets the output power level of a specified interference signal in dBm.

Example:

```
// Set power level of interference with identification
// tx1_cw to -30 dBm
outp:interf:pow:set tx1_cw,-30
```

8.5.7.9 OUTPut:INTERFerence:POWer:GET?

Get the power level of the specified interference signal.

Syntax:

OUTPut:INTERFerence:POWer:GET? <interference identification>

Description:

This request retrieves the output power level of the specified interference signal in dBm.

Example:

```
// Get the power level of the interference signal with
// identification tx1_cw.
outp:interf:pow:get? tx1_cw
-30
```

8.5.7.10 OUTPut:INTERFerence:POWer:LIMits?

Get the limits of the specified interference signal power at output.

Syntax:

OUTPut:INTERFerence:POWer:LIMits? <interference identification>

Description:

This command retrieves power level limits of the specified interference signal in dBm. The power level cannot be set outside these limits. If this is attempted the value is automatically set to the closest acceptable value.

Limit values are returned as follows:

```
<lower limit>,<higher limit>
```

Example:

```
// Get power level limits of the interference signal
// with identification tx1_cw.
outp:interf:pow:lim? tx1_cw
-50,-15
```

8.5.7.11 OUTPut:INTERFerence:BANDWidth:SET

Set the bandwidth of the specified interference (AWGN) at output.

Syntax:

```
OUTPut:INTERFerence:BANDWidth:SET <interference identification>,<bandwidth>
```

Description:

This command sets the AWGN interference bandwidth at channel output in MHz.

Example:

```
// Set noise bandwidth to 30 MHz for the interference with
// identification tx1_awgn
outp:interf:bandw:set tx1_awgn,30
```

8.5.7.12 OUTPut:INTERFerence:BANDWidth:GET?

Get the bandwidth of the interferer (AWGN).

Syntax:

```
OUTPut:INTERFerence:BANDWidth:GET? <interference identification>
```

Description:

This request retrieves the AWGN bandwidth setting of the specified interference signal in MHz.

Example:

```
// Get the bandwidth of the interference signal with
// identification tx1_awgn.
outp:interf:band:get? tx1_awgn
3.84
```

8.5.7.13 OUTPut:INTERFerence:BANDWidth:LIMits?

Get the bandwidth limits of the specified interference.

Syntax:

```
OUTPut:INTERFerence:BANDWidth:LIMits? <interference identification>
```

Description:

This command retrieves the AWGN bandwidth limit values of the specified interference signal in MHz. The bandwidth value cannot be set outside the se limits. If this is attempted the value is automatically set to the closest acceptable value.

The limit values are returned as follows:

```
<lower limit>,<higher limit>
```

Example:

```
// Get the bandwidth limits of the interference signal
// with identification tx1_awgn
outp:interf:bandw:lim? tx1_awgn
0.02,70
```

8.5.7.14 OUTPut:INTERFerence:LEVel:SET

Set output level of a specific interferer (AWGN).

Syntax:

```
OUTPut:INTERFerence:LEVel:SET <interference identification>,<level>
```

Description:

This command sets the output level of the specified (AWGN) interference in dBm/Hz.

Example:

```
// Set the noise level of interferer tx1_awgn to
// -100dBm/Hz
outp:interf:lev:set tx1_awgn,-100
```

8.5.7.15 OUTPut:INTERFerence:LEVel:GET?

Get output level of a specific interferer (AWGN).

Syntax:

```
OUTPut:INTERFerence:LEVel:GET? <interference identification>
```

Description:

This request retrieves the output level of the specified (AWGN) interference in dBm/Hz.

Example:

```
// Get the noise level of interferer tx1_awgn
outp:interf:lev:get? tx1_awgn
-100
```

8.5.7.16 OUTPut:INTERFerence:LEVel:LIMits?

Get limits of the of a specific interferer (AWGN).

Syntax:

```
OUTPut:INTERFerence:LEVel:LIMits? <interference identification>
```

Description:

This command retrieves the limits of the specified interference (AWGN) in dBm/Hz. The power level value cannot be set outside these limits. If this is attempted the value is automatically set to the closest acceptable value.

Limit values are returned as follows:

```
<lower limit>,<higher limit>
```

Example:

```
// Get the power level limits of the interferer
// tx1_awgn
outp:interf:level:lim? tx1_awgn
-150,-100
```

8.5.7.17 OUTPut:INTERFerence:DATARate:SET

Set the data rate in kbps of the information signal for Eb/N0 setting (AWGN)

Syntax:

```
OUTPut:INTERFerence:DATARate:SET <interference identification>,<data
rate>
```

Description:

This command sets the data rate of the information signal in kbps to which the identified AWGN noise is summed. This is needed if the signal-to-noise ratio is defined as Eb/N0.

Example:

```
// Set the data rate of the information signal to
// 200kbps. Noise level is adjusted to meet the user's
// Eb/N0 setting.
outp:interf:datar:set tx1_awgn,200
```

8.5.7.18 OUTPut:INTERFerence:DATARate:GET?

Get the data rate in kbps of the information signal set by the user

Syntax:

```
OUTPut:INTERFerence:DATARate:GET? <interference identification>
```

Description:

This command retrieves the data rate of the information signal in kbps to which the identified AWGN noise is summed. This is needed if the signal-to-noise ratio is defined as Eb/N0.

Example:

```
// Get the user set data rate of the information
// signal. Noise level is adjusted to meet the user's
// Eb/N0 setting.
outp:interf:datar:get? tx1_awgn
200
```

8.5.7.19 OUTPut:INTERFerence:DATARate:LIMits?

Get the data rate in kbps limits of the information signal

Syntax:

```
OUTPut:INTERFerence:DATARate:LIMits? <interference identification>
```

Description:

This command retrieves the limits of the specified interferer in kbps. This is needed if the signal-to-noise ratio is defined as Eb/N0. The data rate value cannot be set outside these limits. If this is attempted the value is automatically set to the closest acceptable value. Limit values are returned as follows:

```
<lower limit>,<higher limit>
```

Example:

```
// Get the data rate limits of the information signal.
// Noise level is adjusted to meet the Eb/N0 setting.
outp:interf:datar:lim? tx1_awgn
1,5000
```

8.5.7.20 OUTPut:INTERFerence:EBN0:SET

Set the Eb/N0 value.

Syntax:

```
OUTPut:INTERFerence:EBN0:SET <interference identification>,<Eb/N0>
```

Description:

With this command the user is able to define the needed Eb/N0 value. The interference has to be AWG noise.

Example:

```
// Set the Eb/N0 value to 5.4dB. The AWG noise has been
// given identification tx1_awgn
outp:interf:ebn0:set tx1_awgn,5.4
```

8.5.7.21 OUTPut:INTERFerence:EBN0:GET?

Get the Eb/N0 value for interferer (AWGN).

Syntax:

```
OUTPut:INTERFerence:EBN0:GET? <interference identification>
```

Description:

This command gets the Eb/N0 value set to the specified AWG noise.

Example:

```
// Get the Eb/N0 value related to AWG noise with
// identification tx1_awgn
outp:interf:ebn0:get? tx1_awgn
5.4
```

8.5.7.22 OUTPut:INTERFerence:EBN0:LIMits?

Get the Eb/N0 limits for interferer (AWGN).

Syntax:

`OUTPut:INTERFerence:EBN0:LIMits? <interference identification>`

Description:

This command retrieves the limits of the Eb/N0 value for the specified AWGN noise. The Eb/N0 value cannot be set outside these limits. If this is attempted the value is automatically set to the closest acceptable value.

The limit values are returned as follows:

`<lower limit>,<higher limit>`

Example:

```
// Get the Eb/N0 value limits related to AWG noise with
// identification tx1_awgn.
outp:interf:ebn0:lim? tx1_awgn
0,47
```

8.5.7.23 OUTPut:INTERFerence:FREQuency:SET

Set the center frequency of the specific interferer.

Syntax:

`OUTPut:INTERFerence:FREQuency:SET <interference identification>,<center` frequency>

Description:

With this command the user is able to set the center frequency of the specified interference in MHz.

Example:

```
// Set the center frequency of interference tx1_cw
// to 1877.5 MHz
outp:interf:freq:set tx1_cw,1877.5
```

8.5.7.24 OUTPut:INTERFerence:FREQuency:GET?

Get the center frequency of the specific interferer.

Syntax:

`OUTPut:INTERFerence:FREQuency:GET? <interference identification>`

Description:

This request retrieves the center frequency of the specified interference signal in MHz.

Example:

```
// Get the center frequency of the interference
// tx1_cw.
outp:interf:freq:get? tx1_cw
2000
```

8.5.7.25 OUTPut:INTERFerence:FREQuency:LIMits?

Get the center frequency limits of the interferer.

Syntax:

`OUTPut:INTERFerence:FREQuency:LIMits? <interference identification>`

Description:

This command retrieves the center frequency limit values of the specified interference signal. The center frequency value cannot be set outside these limits. If this is attempted the value is automatically set to the closest acceptable value.

The limit values are returned as follows:

`<lower limit>,<higher limit>`

Example:

```
// Get the center frequency limit values of the
// tx1_awgn interferer.
outp:interf:freq:lim? tx1_awgn
2170,2230
```

8.5.7.26 OUTPut:INTERFerence:RATio:SET

Set the output carrier-to-interference ratio

Syntax:

```
OUTPut:INTERFerence:RATio:SET <interference identification>,<carrier-to-interference ratio>
```

Description:

This command sets the output carrier-to-interference or signal-to-interference ratio related to the specified interferer in dB.

Example:

```
// Set the carrier-to-interference ratio of the
// interferer tx1_awgn to -10dB
outp:interf:rat:set tx1_awgn,-10
```

8.5.7.27 OUTPut:INTERFerence:RATio:GET?

Get the output carrier-to-interference ratio

Syntax:

```
OUTPut:INTERFerence:RATio:GET? <interference identification>
```

Description:

This command gets the output carrier-to-interference or signal-to-interference ratio related to the specified interferer in dB.

Example:

```
// Get the carrier-to-interference ratio of the
// interferer tx1_awgn.
outp:interf:rat:get? tx1_awgn
-10
```

8.5.7.28 OUTPut:INTERFerence:RATio:LIMits?

Get the output carrier-to-interference ratio limits

Syntax:

```
OUTPut:INTERFerence:RATio:LIMits? <interference identification>
```

Description:

This command gets the output carrier-to-interference ratio limits related to the specified interferer in dB. The output carrier-to-interference ratio cannot be set outside these limits. If this is attempted the value is automatically set to the closest acceptable value.

Limit values are returned as follows:

```
<lower limit>,<higher limit>
```

Example:

```
// Get the carrier-to-interference ratio limits of the
// interferer tx1_awgn.
outp:interf:rat:lim? tx1_awgn
-20,20
```

8.5.7.29 OUTPut:INTERFerence:RATio:MODE:SET

Set measurement mode of interferer

Syntax:

```
OUTPut:INTERFerence:RATio:MODE:SET <interference identification>,<mode>
```

Description:

Set measurement mode of interferer for C/I ratio adjustment. Mode is:

0, Measure all input signals connected to output and adjust C/I ratio based on combined power of signals. Note that high correlation between input signals may distort measurement.

1, Measure only input signal of the first channel and adjust C/I ratio based on that

Example:

```
// Set measurement mode so that all input signals connected to
```

```
// output are measured
outp:interf:rat:mode:set tx1_awgn,0
```

8.5.7.30 OUTPut:INTERFerence:RATio:MODE:GET?

Get measurement mode of interferer
Syntax:
`OUTPut:INTERFerence:RATio:MODE:GET? <interference identification>`
Description:
Query returns measurement mode of specified interferer:
0, Measure all input signals connected to output and adjust C/I ratio based on combined power of signals. Note that high correlation between input signals may distort measurement.
1, Measure only input signal of the first channel and adjust C/I ratio based on that
Example:

```
// Get measurement mode
outp:interf:rat:mode:get? tx1_awgn
1
```

8.5.8 Channel Model Information

8.5.8.1 CHannel:MODel:GAIN:MODel?

Get channel model gain of channel
Syntax:
`CHannel:MODel:GAIN:MODel? <channel number>`
Description:
This command retrieves the channel model gain of the specific channel in dB.
Example:

```
//Get channel model gain of channel 1.
ch:mod:gain:mod? 1
```

-19.8401

8.5.8.2 CHannel:MODel:GAIN:TOTal?

Get total gain of channel
Syntax:
`CHannel:MODel:GAIN:TOTal? <channel number>`
Description:
This command retrieves the total gain of the specific channel in dB. Total gain consists of channel model gain and input and output gain/level settings.
Example:

```
// Get total channel gain of channel 1.
ch:mod:gain:tot? 1
```

-15.8401

8.5.8.3 CHannel:MODel:GAIN:ADJust:SET

Adjust gain imbalance for channel
Syntax:
`CHannel:MODel:GAIN:ADJust:SET <channel number>,<adjustment value>`
Description:
This command can be used to adjust gain imbalance of the specific channel in dB.
Example:

```
// Adjust gain imbalance of channel 1.
```

```
ch:mod:gain:adj:set 1,-15.0
```

8.5.8.4 CHannel:MODel:GAIN:ADJust:GET?

Get gain imbalance adjustment setting for channel
Syntax:
```
CHannel:MODel:GAIN:ADJust:GET? <channel number>
```
Description:
This command retrieves gain imbalance adjustment setting value of the specific channel in dB.
Example:
```
// Get imbalance adjustment setting of channel 1.
ch:mod:gain:adj:get? 1
-10
```

8.5.8.5 CHannel:MODel:GAIN:ADJust:LIMits?

Get gain imbalance adjustment setting limits for channel
Syntax:
```
CHannel:MODel:GAIN:ADJust:LIMits? <channel number>
```
Description:
This command retrieves gain imbalance adjustment setting limits of the specific channel in dB.
Example:
```
// Get imbalance adjustment setting limits of channel 1.
ch:mod:gain:adj:lim? 1
-100,0
```

8.5.8.6 CHannel:MODel:GAIN:ANAlog?

Get analog part of channel model gain.
Syntax:
```
CHannel:MODel:GAIN:ANAlog? <channel number>
```
Description:
This command retrieves the analog part of the channel model gain of the specific channel in dB. This command is useful only with models processed with “EB MultiEmulatorScaler” –tool or models created with “EB MIMO OTA” -tool. Other models return the value 0 dB.
Example:
```
// Get analog channel model gain of channel 1.
ch:mod:gain:ana? 1
```
-15.8401

8.5.8.7 CHannel:MODel:ENABLE

Enable or disable channel
Syntax:
```
CHannel:MODel:ENABLE <channel number>,<state>
```
Description:
This command either enables or disables of the specific channel.
1 channel enabled
0 channel disabled
Example:
```
// Disable channel 1.
ch:mod:enable 1,0
```

8.5.8.8 CHannel:MODel:ENABLE?

Query whether channel is enabled or disabled
Syntax:
```
CHannel:MODel:ENABLE? <channel number>
```
Description:
This command retrieves the state of specific channel.
1 channel enabled
0 channel disabled
Example:
```
// Get status of channel 1 (disabled)
ch:mod:enable? 1
0
```

8.5.8.9 CHannel:MODel:PHASE:ADJust:SET

Adjust phase imbalance for channel
Syntax:
```
CHannel:MODel:PHASE:ADJust:SET <channel number>,<adjustment value>
```
Description:
This command can be used to adjust phase imbalance of the specific channel in degrees.
Example:
```
// Adjust phase imbalance of channel 1.
ch:mod:phase:adj:set 1,25.0
```

8.5.8.10 CHannel:MODel:PHASE:ADJust:GET?

Get phase imbalance adjustment setting for channel
Syntax:
```
CHannel:MODel:PHASE:ADJust:GET? <channel number>
```
Description:
This command retrieves phase imbalance adjustment setting value of the specific channel in degrees.
Example:
```
// Get phase imbalance adjustment setting of channel 1.
ch:mod:phase:adj:get? 1
25.0
```

8.5.8.11 CHannel:MODel:PHASE:ADJust:LIMits?

Get phase imbalance adjustment setting limits for channel
Syntax:
```
CHannel:MODel:PHASE:ADJust:LIMits? <channel number>
```
Description:
This command retrieves phase imbalance adjustment setting limits of the specific channel in degrees.
Example:
```
// Get phase imbalance adjustment setting limits of channel 1.
ch:mod:gain:adj:lim? 1
-180,180
```

8.5.8.12 CHannel:MODel:FILE:CIR?

Get channel model control file name
Syntax:
```
CHannel:MODel:FILE:CIR? <channel number>
```
Description:

This command retrieves the channel model control file name (*.sim).

Example:

```
// Get the control file name of the channel model of channel 2.
ch:mod:file:cir? 2
e:\class_0.sim
```

8.5.8.13 CHannel:MODel:CIR?

Get number of impulse responses of channel model

Syntax:

`CHannel:MODel:CIR? <channel number>`

Description:

This command retrieves the amount of impulse responses on the specific channel model.

Example:

```
// Get the number of impulse responses on the channel model
// of channel 2.
ch:mod:cir? 2
```

128000

8.5.8.14 CHannel:MODel:SD?

Get sample density of channel model

Syntax:

`CHannel:MODel:SD? <channel number>`

Description:

This command retrieves the sample density on the specific channel model.

Example:

```
//Get the sample density of the channel model of channel 2.
ch:mod:sd? 2
```

64

8.5.8.15 CHannel:MODel:TIME?

Get emulation time of channel model

Syntax:

`CHannel:MODel:TIME? <channel number>`

Description:

This command retrieves the emulation time (length) of the specific channel model in seconds. Note that the length of the channel model is valid for settings defined in channel model design.

Example:

```
// Get the emulation time of the channel model of channel 2.
ch:mod:time? 2
```

0.948148

8.5.8.16 CHannel:MODel:CORRelating?

Check if channel model is correlating

Syntax:

`CHannel:MODel:CORRelating? <channel number>`

Description:

This command checks if the specific channel model is correlating. Possible return values are:

0, Channel model is not correlating

1, Channel model is correlating

Example:

```
// Check if the channel model of channel 1 is correlating.
ch:mod:corr? 1
0
```

8.5.9 Signal routing

8.5.9.1 ROUTe:PATH:REFerence

Set reference clock source
Syntax:
ROUTe:PATH:REFerence <source>
Description:
This command sets the source of the reference clock. Possible sources are:
EXT, External reference clock
INT, Internal reference clock
Example1:

```
//Set external reference clock
rout:path:ref EXT
```

Example2:

```
//Set internal reference clock
rout:path:ref INT
```

8.5.9.2 ROUTe:PATH:REFerence?

Get current reference clock source
Syntax:
ROUTe:PATH:REFerence?
Description:
This commands gets the source of the reference clock. Possible sources are:
EXT External reference clock
INT Internal reference clock
Example1:

```
//Get reference clock source
rout:path:ref?
EXT
```

8.5.9.3 ROUTe:PATH:CONNector?

Get channel's physical connectors
Syntax:
ROUTe:PATH:CONNector? <channel id>
Description:
This query retrieves information of input, output and RF local connectors that are allocated to specified channel. Propsim will reply with following information:

```
sim,in,out,inlo,outlo
```

"`sim`" specifies emulator; 1 being master, 2 second emulator (first slave) and so on.
"`in`" specifies type and physical input number for channel.
"`out`" specifies type and physical output number for channel.
"`inlo`" specifies channel's RFLO connector for input signal.
"`outlo`" specifies channel's RFLO connector for output signal.
Note that in most cases "inlo" and "outlo" specify the same connector. If channel is baseband channel or internal local oscillator is used, local oscillator connectors are returned as "-" (dash).

Example1:
```
ROUT:PATH:CONN? 1
1,RF-1,RF-1,1,1
```

Example2:
```
ROUT:PATH:CONN? 2
1,RF-2,RF-2,-,-
```

8.5.9.4 ROUTe:PATH:ID?

Get channels input and output id
Syntax:
ROUTe:PATH:ID? <channel id>
Description:
This query retrieves the input and output numbers where the given channel is connected.

Example:
// Get intput and output of the channel 3 (from input 1 to output 2)
```
ROUT:PATH:ID? 3
1,2
```

8.5.10 RF Local Source (Internal Local Oscillator) control

One to four Internal Local Oscillators can be installed on EB Propsim.

8.5.10.1 ROUTe:PATH:RFLO:CH

Set RF local source to channel
Syntax:
```
ROUTe:PATH:RFLO:CH <channel number>,<RF local source>
```
Description:
This command sets a RF local source to the specific channel. Possible values for the <RF local source> are:
```
INT
DIRECT   (not supported in EB Propsim F32 and FS8)
```
Example:
```
// Set a RF local source DIRECT to channel 2.
rout:path:rflo:ch 2,DIRECT
```

8.5.10.2 ROUTe:PATH:RFLO:CH?

Get the RF local source of channel
Syntax:
```
ROUTe:PATH:RFLO:CH? <channel number>
```
Description:
This request retrieves the RF local source of the specific channel. Possible return values are:
```
INT
DIRECT
NONE
INVALID
```
Example:
```
// Get the RF local source of channel 3.
rout:path:rflo:ch? 3
INT
```

8.5.10.3 ROUTe:PATH:RFLO:AUTO

Set optimal RF local sources automatically
Syntax:
```
ROUTe:PATH:RFLO:AUTO
```
Description:
This command sets optimal RF local sources for the emulation. The command will attempt to use internal local sources if possible.
Example:
```
// Set optimal RF local sources.
rout:path:rflo:auto
```

8.5.11 External Trigger

8.5.11.1 DIAGnostic:SIMUlation:TRIG:SET

Enable or disable external trigger
Syntax:
```
DIAGnostic:SIMUlation:TRIG:SET <ON/OFF>
```
Description:
This command enables or disables external trigger.
Off, Disable external trigger
On, Enable external trigger
Example:
```
// Enable external trigger.
diag:simu:trig:set on
```

8.5.11.2 DIAGnostic:SIMUlation:TRIG:SET?

Query if external trigger is enabled
Syntax:
```
DIAGnostic:SIMUlation:TRIG:SET?
```
Description:
This request checks whether the trigger is enabled or disabled
Example:
```
//trigger enabled.
diag:simu:trig:set?
External trigger is on
```

8.5.11.3 DIAGnostic:SIMUlation:TRIG:CONFiguration

Set triggering configuration
Syntax:
```
DIAGnostic:SIMUlation:TRIG:CONF <triggering mode>,<mode parameter>
```
Description:
This command sets triggering configuration. Possible source values are:
Triggering Mode:

LEVEL,	Level triggering
EDGE,	Edge triggering

Mode Parameter:

LOW,	Low level triggering
HIGH,	High Level triggering
RISING,	Trigger at rising edge
FALLING,	Trigger at falling edge

Example1:

```
// Set triggering on low level.
diag:simu:trig:conf level,low
```

Example2:

```
// Set triggering on high level.
diag:simu:trig:conf level,high
```

Example3:

```
// Set triggering on falling edge.
diag:simu:trig:conf edge, falling
```

Example4:

```
// Set triggering on rising edge.
diag:simu:trig:conf edge, rising
```

8.5.11.4 DIAGnostic:SIMUlation:TRIG:CONFiguration?

Get current triggering configuration
Syntax:
`DIAGnostic:SIMUlation:TRIG:CONF?`
Description:
This command gets triggering configuration.
Look at 8.5.11.3 for possible return values.
Example:

```
// triggering on low level
diag:simu:trig:conf?
Trigger at LEVEL,LOW
```

8.5.11.5 DIAGnostic:SIMUlation:TRIG:FORCEOFF

Force triggering off
Syntax:
`DIAGnostic:SIMUlation:TRIG:FORCEOFF`
Description:
This command disables trigger. Command is parallel, so command is executed even if there are other ATE-commands pending.
Force off command is useful in situations where the emulator is waiting for external trigger and for some reason could not get trigger signal. Executing force off command disable external trigger, external trigger signal waiting stops and ATE command execution continues normally.
Example:

```
// Force trigger off
diag:simu:trig:forceoff
```

8.5.12 Multiple emulations loading

By using multiple emulations loading user can proactively load emulations to emulator's memory, even when an emulation is already running. This will significantly speed up the process of changing emulations.

8.5.12.1 MMEM:LOAD

Proactive emulation loading
Syntax:
`MMEM:LOAD <filename>`
Description:

This command loads emulation (defined in file <filename>) to emulator's memory. Some communication interface tools require the folder name separators to be duplicated. For example: e:\\test\\my emulation.smu.
NOTE! If there isn't enough free space for new emulation, MMEM:LOAD-command deletes oldest proactively loaded emulation.
Example:
```
// Load e:\1to1 class.smu emulation to emulator
mmem:load e:\1to1 class.smu
```

8.5.12.2 MEM:ALL?

List proactively loaded emulations in emulator
Syntax:
```
MEM:ALL?
```
Description:
This request returns all proactively loaded emulations in emulator that are up to date.
Example:
```
// Request all proactive loaded emulations in emulator
mem:all?
e:\1to1 class.smu
// Only e:\1to1 class.smu emulation was loaded to emulator
```

8.5.12.3 MEM:CURRENT?

Get the current emulation's name
Syntax:
```
MEM:CURRENT?
```
Description:
This request returns the current emulation name
Example:
```
// Query current emulation name
mem:current?
e:\1to1 class.smu
```

8.5.12.4 MEM:DEL:ALL

Remove proactively loaded emulations from memory
Syntax:
```
MEM:DEL:ALL
```
Description:
This command deletes all proactively loaded emulations in emulator.
Example:
```
// Removes all emulations from emulator
mem:del:all
```

8.5.12.5 MEM:DEL:NAME

Remove single loaded emulation from emulator's memory
Syntax:
```
MEM:DEL:NAME <filename>
```
Description:
This command deletes proactively loaded emulation (defined in file <filename>) in emulator. Some communication interface tools require the folder name separators (backslashes) to be duplicated. For example: e:\\test\\my emulation.smu.
Example:
```
// Remove e:\1to1 class.smu emulation from emulator
mem:del:name e:\1to1 class.smu
```

8.5.12.6 MEM:FREE?

Returns emulators' free space
Syntax:
```
MEM:FREE?
```
Description:
This request returns known emulators' names and free space (in megabytes).
Example:
```
// Retrieves free space in emulator
mem:free?
propsim-202452:2103
```

8.5.12.7 MEM:FILE:SIZE?

Returns emulation's size
Syntax:
```
MEM:FILE:SIZE? <filename>
```
Description:
This request returns emulation's size (in megabytes). Some communication interface tools require the folder name separators (backslashes) to be duplicated. For example: e:\\test\\my emulation.smu.
Example:
```
//request e:\1to1 class.smu emulation's size
mem:file:size? e:\1to1 class.smu
19
```

8.5.13 Multi-emulator synchronization

8.5.13.1 SYSTem:MSIMulator:CONFig

Syntax:
```
SYSTem:MSIMulator:CONFig <emulator position>,<emulator total amount>
```
Description:
This command defines the position of an emulator in multi-emulator configuration.
Accepted values for emulator total amount: 1-36
Accepted values for emulator position: 1- emulator total amount.
Note: max. synchronization of max 6 devices supported in current release.
Example:
```
// Example to set the device to be number 2 in chain of 4 emulators.
SYSTem:MSIMulator:CONFig 2,4
```

8.5.13.2 SYSTem:MSIMulator:CONFig?

Syntax:
```
SYSTem:MSIMulator:CONFig?
```
Description:
This command can be used to query position of emulator in multi-emulator configuration.
Configuration information is returned as follows:
```
<emulator position>,<emulator total amount>
```
Example:
```
// In the example, the device is number 2 in chain of 3 emulators
SYSTem:MSIMulator:CONFig?
2,3
```

8.5.13.3 SYSTem:MSIMulator:CABle

Syntax:

```
SYSTem:MSIMulator:CABle <cable length>
```

Description:

This command can be used to define the synchronization cable length used in multi-emulator configuration. Possible length definitions are: 2.0 m, 4.0 m and 6.0 m.

Example:

```
// Example to set the synchronization cable length to 4.0 m
SYSTem:MSIMulator:CABle 4.0 m
```

8.5.13.4 SYSTem:MSIMulator:CABle?

Syntax:

```
SYSTem:MSIMulator:CABle <cable length>
```

Description:

This command can be used to query the synchronization cable length used in multi-emulator configuration. Possible lengths are: 2.0 m, 4.0 m and 6.0 m.

Example:

```
// Example to query the synchronization cable length (4.0 m).
SYSTem:MSIMulator:CABle?
2.0 m
```

8.5.14 Example script

```
//script is designed to run through LAN-interface
//opening first emulation
calc:filt:file e:\emulation1.smu
syst:err?

//some setting commands

//opened emulation running
diag:simu:go
syst:err?

//loading second emulation to emulator's memory
mmem:load e:\emulation2.smu
syst:err?

//some measurement

//closing old emulation
diag:simu:close
syst:err?

//opening already loaded emulation
calc:filt:file e:\emulation2.smu
syst:err?

//some setting commands

//opened emulation running
diag:simu:go
syst:err?

//checking third emulation size
mem:file:size? e:\emulation3.smu
syst:err?

//checking free space in emulator
mem:free?
syst:err?
```

```
//loading third emulation to emulator's memory
mmem:load e:\emulation3.smu
syst:err?

//some measurement

//closing old emulation
diag:simu:close
syst:err?

…

//opening already loaded emulation
calc:filt:file e:\emulation10.smu
syst:err?

//some setting commands

//opened emulation running
diag:simu:go
syst:err?

//checking 11th emulation size
mem:file:size? e:\emulation11.smu
syst:err?

//checking free space in emulator
mem:free?
syst:err?

//maybe there isn't enough free space
//deleting old loaded emulation
mem:del:name e:\emulation2.smu
syst:err?

//loading 11th emulation to emulator's memory
mmem:load e:\emulation11.smu
syst:err?

//some measurement

//closing old emulation
diag:simu:close
syst:err?

//opening already loaded emulation
calc:filt:file e:\emulation11.smu
syst:err?
//some setting commands

//opened emulation running
diag:simu:go
syst:err?

//next we are going to run emulation3.smu so check if //it's up to date
in emulator's memory

mem:all?
Syst:err?
//emulation3.smu exists, no need to load it again

//some measurement

//closing old emulation
diag:simu:close
syst:err?

…
```

8.6 Errors and events

There is a number of error / event messages that are inserted to the error / event queue (if enabled) and can be read by the user. The error event messages have the following syntax:

```
<error code>,<error string>;<device specific error string>
```

The error codes and related error strings are listed in the following chapters.

8.6.1 Command error

Error code:
-100

Reason:
Received user command is not supported by ATE server:

- Command itself is not supported or is misspelled
- Command contains parameter data when it should not, or vice versa

Example:

```
//Command is misspelled
dag:simu:go
```

-100,"Command error;ATE command not supported"

8.6.2 Execution error

Error code:
-200

Reason:
Command execution failed:

- Command parameter data did not contain required parameters
- Command could not and would not be executed due to the device internal state
- Command execution returned failure

Example:

```
// Emulation is tried to be run eventhough no
// Emulation has been opened.
diag:simu:go
-200,"Execution error;Wrong device state for command"
```

8.6.3 Device-specific error

Error code:
-300

Reason:
Unexpected device-specific error or event occurred.

Example:

```
//Temperature warning.
-300,"Device-specific error;HW temperature over limit"
```

8.6.4 Error/Event queue overflow

Error code:
-350

Reason:
Error/event queue overflow occurred. Queue can contain up to 100 error messages. If the user does not read the messages or clear the queue, it is possible that overflow occurs.

Example:
```
//Error/event queue overflow.
-350,"Queue overflow"
```

8.6.5 Communication error

Error code:
-360

Reason:
Communication error occurred. Possible reasons:
- GPIB -board or TCP/IP socket error
- GPIB send buffer got full
- LAN receive buffer got full
- LAN receive was interrupted by remote host
- Another user interface interrupted the connection

It should be noted that this error occurs also while closing the remote LAN connection at the end of emulation run, which is not necessarily an error condition. See the following example.

Example:
```
// User closes TCP/IP connection after emulation was
// closed
-360,"Communication error;LAN receive error"
```

8.6.6 Query error

Error code:
-400

Reason:
This error occurs typically in ATE GPIB connection while the user has not read the response to a query before issuing a next command. In ATE LAN connection the response is automatically sent back to the user without any user read request operation.

Example:
```
// User sends a command without reading the response to
// the previous query.
*idn?
calc:filt:file e:\1to1 class.smu
-400,"Query error"
```

8.7 Troubleshooting

Reliability of ATE communication is dependent of the used interface. While ATE LAN interface (TCP/IP) includes higher level protocol in itself, taking care of the possible data communication failures, the matter is not the same with ATE GPIB connection. Even if the test system has been set up with care, communication failures tend to occur. The responsibility to take care of these failure occasions lies on GPIB ATE client application, which is controller-in-charge of the test system.

EB Propsim ATE GPIB interface provides several supporting functions for the client, which should be used in the client side design. The purpose of this chapter is not to list all the provided functionality, but rather to describe the most typical problem situations, offer help on noticing those cases and give some suggestions on how to recover. The failure handling and recovery of the test system depends heavily on the test system itself i.e. what is being done. Please refer to IEEE 488.2 standard for additional information.

8.7.1 Failure handling

The failure handling support of EB Propsim ATE interfaces include following main areas:

- Emulation status
- Operation state
- ATE device status
- GPIB interface

8.7.1.1 Emulation status

ATE interfaces provide almost all the functionality of the GUI Running view with few exceptions (e.g. CIR graphics is not available). User can follow the status of the emulation via ATE Monitor interface in order to ensure the correct settings. It is recommend to design and set up the emulation as ready as possible by using Editing view in order to avoid unnecessary GPIB data traffic. For instance the setting of used center frequencies can be done beforehand and saved to the emulation instead of separate initialization at the start of every test session.

All of the emulation settings are not available in ATE Monitor. It is possible to check the status of the settings by using query -commands related to setting command. This method is valuable in order to detect the cases where data communication error has not caused an error in the device, e.g. if the mobile speed setting command misses the last digit. Queries can also be used for retrieving information on those settings that are set as default and have not been changed by the user. This can be further on be used for documentation purposes.

Example:

```
// Opening and running of ready made emulation
calc:filt:file d:\My Emulations\SN\emulation.smu
     diag:simu:go

// Emulation is predefined, but we still need to make
// measurements with different output gain settings

// Set gain of channel output 1 to -5 dB
outp:gain:ch 1,-5

// Get gain of channel output 1
outp:gain:ch? 1
-5

// Set gain of channel output 1 to -10 dB
```

```
outp:gain:ch 1,-10

// Get gain of channel output 1
outp:gain:ch? 1
-10

// Set gain of channel output 1 to –15 dB
outp:gain:ch 1,-15

// Get gain of channel output 1
outp:gain:ch? 1
-1

// Error! Digit 5 did not go through! Do something!
// Let' s send again
outp:gain:ch 1,-15

// Get gain of channel output 1
outp:gain:ch? 1
-15
```

8.7.1.2 Operation state

EB Propsim ATE interfaces execute commands sequentially. This means that there is no need to follow operation execution status before sending the next command (for instance using *wai command). Instead, operation state query (*opc?) can be used for synchronization purposes in test system. Note however that the result of the *opc? query indicates only that previous operation has been executed, not that there has not been any errors during the execution.

Example:

```
// We wish to load emulation, run it and measure output
// level when emulation is running

calc:filt:file d:\My Emulations\SN\Emulation.smu
diag:simu:go
*opc?
1

// We received "1" from emulator. Emulation is
// running. Now we can measure with our measurement
// device
abor
calc:mark:y?
```

8.7.1.3 Device status

ATE interfaces provide status information on different kind of errors, which might occur during the test session. These errors include command errors, execution errors, device specific errors and interface errors. Refer to chapter 8.6 of this document.

It is feasible to check for these errors after every ATE command. Most of the error situations can be detected by this way. One way to do this is to check for status byte of the ATE device by using *stb? query. Depending on SCPI register settings, status byte indicates any existing error conditions. Furthermore the status byte can give indication on readable data, which is typically a result of a query command.

Example:

```
// Initialise SCPI registers
```

```
// Set Standard Event Status register to indicate all
// errors/events
// By default this is 0: SESR information is not enabled
*ese 255

// Check Service Request Enable Register
// By default this is 191. SESR summed output,
// error/event queue status and message available bit as
// are shown in status byte. RQS is always summed from
// other status byte conditions and therefore not
// editable
*sre?
191

// Send identification query
*idn?

// Send again. This causes an error because response
// data from a query has to be always read
*idn?

// Read status byte
*stb?
100

// The result indicates:
// Error/event queue is not empty
// SESR indicates an error
// RQS is set
// Note that due to error, MAV –bit is not set
```

Normally it is not required to use SCPI register information in order to perform failure handling on device state. By default all the errors are added to the error/event queue and all the errors in the queue can be read by using syst:err? query. Query responds with text string indicating latest error. Note that there might be multiple errors in the queue, so it is recommended to read all the errors. Errors are destroyed from the queue as they are read. Note that error can also occur on the error checking query itself.

Example:

```
// Set center frequency of channel 1 (and all the
// channels belonging to same channel group) to 2200 MHz
calc:filt:cent:ch 1,2200

// Check error
// Whole message does not go through
sy

// Read return value
// GPIB read operation causes timeout because
// emulator will not respond

//Resend
syst:err?

// Read return value
-100,"Command error;ATE command not supported"

// That was caused by failure on error check
// Check for possible previous error
syst:err?

// Read return value
0,"No Error."

// Initial message went through correctly
```

8.7.1.4 GPIB interface

In addition to higher lever support there is a possibility to detect errors indicated by GPIB lines. When emulator updates status byte it also asserts these lines accordingly. For instance setting of the status byte indicating RQS set (Device requesting service) causes SRQI line to be asserted. Line status itself does not give any information on error and it is feasible to get the error information by for example using syst:err? query. Line will be unasserted when error condition (status byte RQS bit is not set) disappears.

8.7.1.5 GPIB example

```
// Open and run the emulation. Measure and close
*rst
*opc?
1
syst:err?
0,"No Error."

*cls
*opc?
1
syst:err?
0,"No Error."

calc:filt:file d:\My Emulations\Test\My Emulation.smu
*opc?
1
syst:err?
0,"No Error."

diag:simu:go
*opc?
1
syst:err?
-300,"Device-specific error;Input cut-off warning"
syst:err?
0,"No Error."

// Oops! Something is wrong. Close the test

diag:simu:close
*opc?
1
syst:err?
0,"No Error."

// Check RF input signal level and start from the
// beginning

*rst
*opc?
1
syst:err?
0,"No Error."

*cls
*opc?
1
syst:err?
0,"No Error."

calc:filt:file d:\My Emulations\Test\My Emulation.smu
*opc?
1
syst:err?
```

```
0,“No Error.”

diag:simu:go
*opc?
1
syst:err?
0,“No Error.”

// Measurements here

diag:simu:close
*opc?
1
syst:err?
-100,”Command error;ATE command not supported”
syst:err?
0,“No Error.”

//Last command did not go through. Send again
diag:simu:close
*opc?
1
syst:err?
0,“No Error.”
```

// Continue with new test

9 File Formats

This section describes the file formats supported by EB Propsim for exporting and importing channel impulse response data.

9.1 .ASC File

An ASC-file is a simple text file format for representing channel impulse response data. EB Propsim software supports conversions between textual ASC-files and binary IR-files. An ASC-file can be created, for example, in Microsoft Excel spreadsheet application by saving the document as tab delimited text file.

A short ASC-file is presented below. It, like all ASC-files, consists of two main parts: header and tap data. Each line in the tap data corresponds to a single CIR. Entries on each line are separated by tabulator characters.

```
*****  Header   *****
1008       CIRs
2          Taps/CIR
2200000000 Carrier_Frequency
Route_Closed
CIRUpdateRate_Unlocked
CarrierFrequency_Unlocked
5          Delay_Resolution
2          Sample_Density
26092.1    CIR_Update_Rate
*****  Tap data *****
Delay      Re         Im         Delay      Re         Im
0.00000    -0.27080   -0.92316   50.00000   0.86202    -0.12606
0.00000    -0.29589   -0.93519   50.00000   0.78361    -0.05152
0.00000    -0.32334   -0.95466   50.00000   0.62736    0.08362
0.00000    -0.34575   -0.95903   50.00000   0.55293    0.16272
```

9.1.1 Header

The header contains the following information

- Number of CIRs (usually in range of 4 - 1,000,000)
- Number of taps per CIR (1 to 48, though system limitations with tap placement apply)
- Carrier frequency (in Hz)
- Model continuity
  - "Route_Closed" for continuous models
  - "Route_Open" for non-continuous models
- Lock / Unlock of CIR update rate
  - In case CIRUpdateRate_Unlocked, values can be changed in run-time
  - In case CIRUpdateRate_Locked, values can not be changed run-time
- Lock / Unlock carrier frequency
  - In case CarrierFrequency_Unlocked, values can be changed in run-time
  - In case CarrierFrequency_Locked, values can not be changed run-time
- Delay resolution (nanoseconds, use always 5ns)
- Sample density i.e. the number of CIRs calculated for one wavelength
- CIR update rate

9.1.2 Tap data

The Tap data contains information about complex taps. Tap data for each tap data contains three values (”triplet”):

- Delay value (measured in nanoseconds)
- Real value
- Imaginary value

For each CIR, equal number of these triplets must exist on the row as specified in the header. For example, if the header specifies 4 taps/CIR, there must be 4 triples, or 12 values (delay, real, imaginary) per row.

9.2 .ASO File (EB Propsim F8, release 1.4)

An ASO-file is a simple text file format for representing channel impulse response data for use with Aerospace and Satellite channel modeling Option. EB Propsim software supports conversions between textual ASO-files and binary ASO SIM-files.

The ASO file format describes either periodical **Function based**, **Coordinate based** (N-type) or **Arbitrary model** (A-type) parameters. Only one model type can be specified in a file at same time.

9.2.1 Function based .ASO file

Function based model creates model where Doppler is defined based on linear, sinusoidal or triangular function. Delay is calculated from Doppler. User can define minimum and maximum values for gain.

Example of function based model

```
; Propsim Aerospace Model file, version 1.0
[Model]
SimulationCenterFrequency=27000000000 Hz
RFCenterFrequency=370000000 Hz

[Sinusoidal 1]
Period = 2.0 s
MinDoppler = -1.0 MHz
MaxDoppler = 1.0 MHz
MinDelay = 100 us
MinGain = 0 dB
MaxGain = -6 dB
```

9.2.2 Coordinate based .ASO file

Coordinate based model contains a point-to-point link between three-dimensionally moving transmitter and receiver. Link is assumed to be line-of-sight, with possibility to have up to three reflectors somewhere in the area causing multi-path effects. Reflectors will cause reflected signal to arrive to receiver delayed and attenuated by user-given attenuation factor.

In coordinate based model trajectories of the objects are defined by their coordinates in 3D space. The coordinates are then used to calculate delay and Doppler values.

Example of coordinate based model

```
; Propsim Aerospace Model file, version 1.0
[Model]
SimulationCenterFrequency=27000000000 Hz
RFCenterFrequency=370000000 Hz

[LOS]
N=0.0, 10.0,0.0,0.0, 1.0,0.0,0.0, 0.0
N=1.0, 10.0,0.0,0.0, 2.0,0.0,0.0, -1.0
N=2.0, 10.0,0.0,0.0, 3.0,0.0,0.0, -2.0
[Reflector 1]
N=0.0, 3.14,2.71,0, -5.0
```

9.2.3 Arbitrary .ASO file

Arbitrary model contains a point-to-point link between moving transmitter and receiver. Link is assumed to be line-of-sight, with possibility to have up to three reflectors causing multi-path effects. Delay, Doppler and gain values for transmitter, receiver and reflectors are defined directly by the user.

Example Arbitrary model

; Propsim Aerospace Model file, version 1.0

[Model]
SimulationCenterFrequency = 27000000000 Hz
RFCenterFrequency = 370123400 Hz

[LOS]
A = 0, 0.005, 0, 0
A = 10, 0.002, 100, -10

[Reflector 1]
A = 0, 0.001, 100, -10
A = 10, 0.002, 200, -20

9.2.4 File syntax

.ASO file itself is a plain text file. It consists of comment header and multiple sections of data. Lines beginning with semicolon (;) are comment lines and ignored, with exception of header command line, which is required.

Section’s name is enclosed in “[ ]” characters. The data itself is in “keyword=value” lines, each keyword-value pair being on its own line. The type of all numeric variables is double.

ASO file can contain either N- or A-keywords. Any mixing of A- and N-keywords in a single file is not allowed.

Comment	Propsim Aerospace Model file, version 1.0
Description	Normal comment header in .ASO file. Required field.

9.2.4.1 [Model] section

Section	Model
Description	General model parameters.

Key	SimulationCenterFrequency
Value	*Frequency* Hz
Description	Center frequency of model, in Hz. Doppler and channel update rates are calculated by using this value

Key	RFCenterFrequency
Value	*Frequency* Hz
Description	EB Propsim center frequency, given in Hz’s.

Key	RangeStart
Value	Time
Description	Optional field describing start of the range. Range is used to crop model at the time of emulation creation. If field is missing, range is not enabled.

Key	RangeEnd
Value	Time
Description	Optional field describing end of the range. Range is used to crop model at the time of emulation creation. If field is missing, range is not enabled.

Key	AutomaticStop
Value	*True/False*
Description	Optional field. If set to true, compiler adds CIR's for 0.5 seconds at the end of the emulation. Compiler also sets delta delay to zero and doppler to last doppler value in the model. If the field is missing, default value is true. If model is continuous (function based model), no extra CIR's are added.

9.2.4.2 [LOS] section

Section	LOS
Description	Line-of-sight component definition.

Key	N
Value	*Time,X1,Y1,Z1,X2,Y2,Z2,Gain*
Description	Positions of devices and link gain on given time index. Time specifies time index, in seconds. Xn,Yn, Zn specifies device position in 3D space in meters. Gain specifies link gain, in dB. Whitespace characters are allowed between values and commas. Note that using "N" keys in file disallows usage of "A" keys in LOS or in Reflector sections.

Key	A
Value	*Time,delay,Doppler,gain*
Description	Delay, Doppler and gain of path on given time. Time specifies time index, in seconds. Delay specifies current LOS path delay, in seconds. Doppler specifies current Doppler of link, in Hz. Gain specifies link gain, in dB. Whitespace characters are allowed between values and commas. Note that using "A" keys in file disallows usage of "N" keys in LOS or in Reflector sections.

9.2.4.3 [Reflector X] section

Section	Reflector *x*
Description	Data for reflector number *x* (1, 2 or 3)

Key	N
Value	*Time,X,Y,Z,Gain*
Description	Position of reflector and link gain on given time index. Time specifies time index, in seconds. X,Y,Z specify reflector position in 3D space, in meters. Gain specifies link gain in dB. Whitespace characters are allowed between values and commas. Note that using "N" keys in file disallows usage of "A" keys in LOS or in Reflector sections.

Key	A
Value	*Time,delay,Doppler,gain*
Description	Delay, Doppler and gain of path on given time. Time specifies time index, in seconds. Delay specifies current LOS path delay, in seconds. Doppler specifies current Doppler of link, in Hz. Gain specifies link gain, in dB. Whitespace characters are allowed between values and commas. Note that using "A" keys in file disallows usage of "N" keys in LOS or in Reflector sections.

9.2.4.4 [Sinusoidal X] section

Section	Sinusoidal *x*
Description	Data for sinusoidal function number *x* (1)

Key	Period
Value	*Time* s
Description	Period of the function in seconds.

Key	MinDoppler
Value	*Frequency* Hz
Description	Minimum Doppler value. Usually equals to MaxDoppler with difference of the sign.

Key	MaxDoppler
Value	*Frequency* Hz
Description	Maximum Doppler value. Usually equals to MinDoppler with difference of the sign.

Key	MinDelay
Value	*Time* s
Description	Minimum delay for model. Used to calculate position offset for fictional moving object in model.

Key	MinGain
Value	*Gain* dB
Description	Minimum gain value for the model. Gain is following used function.

Key	MaxGain
Value	*Gain* dB
Description	Maximum gain value for the model. Gain is following used function.

9.2.4.5 [Triangle X] section

Section	Triangle *x*
Description	Data for triangle function number *x* (1)

Key	Period
Value	*Time* s
Description	Period of the function in seconds.

Key	MinDoppler
Value	*Frequency* Hz
Description	Minimum Doppler value. Usually equals to MaxDoppler with difference of the sign.

Key	MaxDoppler
Value	*Frequency* Hz
Description	Maximum Doppler value. Usually equals to MinDoppler with difference of the sign.

Key	MinDelay
Value	*Time* s
Description	Minimum delay for model. Used to calculate position offset for fictional moving object in model.

Key	MinGain
Value	*Gain* dB
Description	Minimum gain value for the model. Gain is following used function.

Key	MaxGain
Value	*Gain* Db
Description	Maximum gain value for the model. Gain is following used function.

9.2.4.6 [LinearDoppler X] section

Section	LinearDoppler *x*
Description	Data for Linear Doppler function number *x* (1)

Key	Period
Value	*Time* s
Description	Period of the function in seconds.

Key	MinDoppler
Value	*Frequency* Hz
Description	Minimum Doppler value. Usually equals to MaxDoppler with difference of the sign.

Key	MaxDoppler
Value	*Frequency* Hz
Description	Maximum Doppler value. Usually equals to MinDoppler with difference of the sign.

Key	MinDelay
Value	*Time* s
Description	Minimum delay for model. Used to calculate position offset for fictional moving object in model.

Key	MinGain
Value	*Gain* dB
Description	Minimum gain value for the model. Gain is following used function.

Key	MaxGain
Value	*Gain* dB
Description	Maximum gain value for the model. Gain is following used function.

9.3 .ICS File

ICS (Intermediate CSS) files are used to run multiple channel models in sequence. In practice, continuous sequences reflect changing conditions when a signal source moves. For example, when a car moves from flat rural area into a tunnel, then comes back onto the rural scenery, and finally goes into a city. The taps simulate incoming signals perceived by the moving signal source. Those signals are usually either reflections of the original signal from mountain sides, buildings, etc or other signals interpreted as interference.

The ICS File Format consists of one Model section and multiple Sequence sections. In addition, each Sequence may contain any number of Tap sections. The basic file layout is (three dots represent key-value pairs of that section):

```
; Propsim Intermediate CSS File, version 1.0

[Model]
…


[Sequence 0]
…

[Tap 0]
…

[Tap 1]
…

[Sequence 1]
…

[Tap 0]
…

[Tap 1]
…
```

9.3.1 [Model] Section

There can be only one Model section per ICS file. The following key-value pairs are possible in a Model section (some must always be there and some only when certain conditions are met):

```
[Model]                              ‘ Mandatory
CenterFrequency = 2200000000 Hz      ‘ Mandatory
DistributionSeed = 10                ‘ Mandatory
SampleDensity = 2                    ‘ Mandatory
MobileSpeed = 50 km/h                ‘ Mandatory this or DopplerShift
DopplerShift = 101.922 Hz            ‘ Mandatory this or MobileSpeed
InputCrestFactor = 6 dB              ‘ Mandatory
SignalBandwidth = 125 Hz             ‘ AWGN: mandatory, CW/None: discard
Continuousity = None                 ‘ Mandatory
```

CenterFrequency

- This is the Center frequency of model used for distance calculations. You must provide this value.

- Valid range: 30000000 – 6000000000 Hz, default is 2200000000 Hz

DistributionSeed

- The EB Propsim internal software logic uses this seed value to initialize randomization generators. Using the same seed with the same parameters should lead to same end-result. You must provide this value.
- Valid range: 0 – 2147483647, default is 10

SampleDensity

- This is the amount of samples per half wave. Only 2 is accepted and supported at the moment. You must provide this value.
- Valid range: 2, default is 2

MobileSpeed

- Speed either in m/s, km/h, or mph. You must provide either this or the DopplerShift.
- Valid range: 0,001 – 100 000 m/s,
- Valid range: 0,004 – 360 000 km/h, or
- Valid range: 0,002 – 223 694 mph
- Default 50 km/h

DopplerShift

- This is the Doppler shift of a fading signal in Hz. You must provide either this or the MobileSpeed. The formula to calculate this is: DopplerShift = (mobile speed m/s * center frequency Hz) / speed of light m/s
- Valid range: 0,007 – 10 000 Hz, default is 101,922 Hz (= 50 km/h @ 2.2GHz)

InputCrestFactor

- This defines the signal crest factor (space reserved for signal modulation) and is used for level calculations during building. You must provide this value.
- Valid range: 0 – 20 dB, default is 6 dB

SignalBandwidth

- This defines the signal bandwidth. Only used when AWGN interference is defined, and is merely an informative field. Discard if using any other interference or none at all.
- This has the following relation to InterferenceOutputLevel and InterferenceOutputPowerDensity: level = density dBm/Hz + (10 * log10(bandwidth Hz))
- density = level dBm - (10 * log10(bandwidth Hz))
- Valid range: 2000 – 125000000 Hz, default is 30000000 Hz

Continuousity

- Defines how model should behave when it reaches the end of simulation.
- Value **None**: the simulation “wraps around” and immediately continues from the beginning when model reaches its end. Only this is currently supported.

9.3.2 [Sequence] Section

There must be at least one Sequence section per ICS file. Each Sequence must be uniquely numbered in ascending order starting with either zero or one (typically with zero). For example, 0, 1, 2, 3, 4..., N. The following key-value pairs are possible in a Sequence section (some must always be there and some only when certain conditions are met):

```
[Sequence 0]                          ‘ Mandatory
Length = 10 m                         ‘ Mandatory
Time  = 0.1 s                         ‘ Mandatory this or Length
CenterFrequency = 2200000000 Hz       ‘ Optional or same as in Model
```

DistributionSeed = 10	‘ Mandatory
SampleDensity = 2	‘ Mandatory
MobileSpeed = 50 km/h	‘ Mandatory this or DopplerShift
DopplerShift = 101.922 Hz	‘ Mandatory this or MobileSpeed
OutputLevel = -30 dBm	‘ Mandatory
InterferenceType = none	‘ Mandatory

‘ Mandatory if InterferenceType = awgn or cw
InterferenceOutputLevel = -50 dBm

‘ Mandatory if InterferenceType = cw
InterferenceFrequency = 2202000000 Hz

‘ Mandatory if InterferenceType = awgn
InterferenceOutputPowerDensity = -110 dBm/Hz

ChannelGainOffset = 0, 0, -1.0 dB	‘Optional, defines digital gain offset applied for certain channel (here -1.0 dB offset to channel between input index 0 and output index 0)
IR = filename.ir	‘ Optional, discard all taps if defined
CIRStart = 0	‘ Mandatory if IR is included
CIREnd = 200	‘ Mandatory if IR is included

Length

- This is the length of a sequence in meters.
- Valid range: 1 – 200000 meters, default is 200m

Time

- This is the duration of a sequence in seconds (optional way to define sequence length. Note: all sequences must be defined with using the same method - Time or Length)
- Minimum duration depends on Center Frequency, Mobile Speed and Sample Density. When using Length-parameter, modeling accuracy with very short sequence lengths depends on EB Propsim hardware and firmware version. Parameter low limit can be calculated with following formula:
  Length(minimum) = speed_of_light/(sample_density * 2.0 * center_frequency) / mobile_speed

  As an example:
  center_frequency = 2200 MHz (2200000000 Hz)
  sample_density = 2
  mobile_speed = 10 km/h (2.777 m/s)
  speed_of_light = 299792458 m/s

  Length(minimum) = 0,0122 s

CenterFrequency

- This is the Center frequency for the sequence and is used for distance calculations. You may omit this value, but if you define this, then you must set it to same value as the CenterFrequency in Model section.
- Valid range: 30000000 – 6000000000 Hz, default is 2200000000 MHz

DistributionSeed

- The EB Propsim internal software logic uses this seed value to initialize randomization generators. Using the same seed with the same parameters should lead to same end-result. We highly recommended using distinct values for each sequence. You must provide this value.
- Valid range: 0 – 2147483647, default is 10

SampleDensity

- This is the amount of samples per half wave. Only 2 is accepted and supported at the moment. You must provide this value, and must be set to the same value in the whole simulation.
- Valid range: 2, default is 2

MobileSpeed

- Speed of the sequence either in m/s, km/h, or mph. You must provide either this or the DopplerShift.
- Valid range: 0,001 – 292636091,386 m/s,
- Valid range: 0,004 – 1053489928,989 km/h, or
- Valid range: 0,002 - 654608293,186 mph
- Default is 50 km/h

DopplerShift

- This is the Doppler shift of a fading signal of the sequence in Hz. You must provide either this or the MobileSpeed. The formula to calculate this is:
- DopplerShift = (mobile speed m/s * center frequency Hz) / speed of light m/s
- Valid range: 0,007 - 2147483647 Hz, default is 101,922 Hz (= 50 km/h)

OutputLevel

- This is the signal level at output.
- Valid range: -125 - -25 dBm, default is -25 dBm

InterferenceType

- This defines the interference type (none, awgn, or cw).
- If **none**: discard all other interference fields for this sequence.
- If **awgn**: set InterferenceOutputLevel and InterferenceOutputPowerDensity.
- If **cw**: set InterferenceOutputLevel and InterferenceFrequency.
- You may use one interference type (**awgn** or **cw)** and **none** in the same .ics file.
- Valid range: none, awgn, or cw, default is none

InterferenceOutputLevel

- This is the interference output level in dBm. Mandatory if InterferenceType is awgn or cw.
- For **awgn**, calculate the output level with formula: level = density + (10 * log10(bandwidth))
- Valid range for **cw**: -125 - -25 dBm, default is -25 dBm
- Valid range for **awgn**: -201,99 - -14,031 dBm, default is -40 dBm (depends of SignalBandwidth and InterferenceOutputPowerDensity)

InterferenceFrequency

- This is the frequency of interferer. Only used in **cw** interference. Valid range is CenterFrequency +/- half of the device bandwidth (40, 80, or 125 MHz in half). For example, if CenterFrequency is 2202000000 Hz and device bandwidth is 80000000 Hz then range is 2162000000 – 2242000000 Hz.
- Valid range: 10000000 – 6062500000 Hz, default is 2202000000 Hz

InterferenceOutputPowerDensity

- This is the interference output power density in dBm/Hz. Mandatory if InterferenceType is awgn.
- Valid range: -235 - -95 dBm/Hz, default is -110 dBm/Hz

ChannelGainOffset (InputIndex, OutputIndex, GainOffset in dB)

- This is digital channel gain offset applied to any channel defined by ics file (MIMO).
- InputIndex valid range: 0 – (Max MIMO Inputs - 1)
- OutputIndex valid range: 0 – (Max MIMO Outputs - 1)
- GainOffset valid range: -60 – 0 dB

As an example,
ChannelGainOffset 0,2,-10.0 dB
sets a channel gain offset between input 0 and output 2 to -10.0 dB.
Note that input and output indexing starts from 0.

If channel gain offset for a channel is not defined in sequence, channel gain offset for the channel in the sequence is 0 dB.

IR

- This field controls whether you want to use an already built tap configuration or to manually configure all taps of a sequence. Type a valid .ir file name with path to use an already built configuration. Discard this field, CIRStart, and CIREnd to manually define tap parameters. Notice that the path must be given as a path inside the EB Propsim device, not as a path inside your PC. CIRStart and CIREnd must be provided too if you choose to use this value.
- Valid range: any valid .ir file

CIRStart

- This is the starting cir number in the given .ir file for a sequence. You must provide this value if IR is defined and included.
- Valid range: depends of the simulation, but 0 – 2147483647 are the boundaries

CIREnd

- This is the last cir number in the given .ir file for a sequence. You must provide this value if IR is defined and included.
- Valid range: depends of the simulation, but 0 – 2147483647 are the boundaries

9.3.3 [Tap] Section

There must be at least one and no more than 40 Tap sections per Sequence section. The number of taps may vary between sequences. Do not add any Tap sections under a Sequence section, in which an IR file has been defined. Each Sequence must be uniquely numbered in ascending order starting with either zero or one (typically with zero). For example, 0, 1, 2, 3, 4..., N. The following key-value pairs are possible in a Tap section (some must always be there and some only when certain conditions are met):

```
[Tap 0]                         ‘ No IR field Sequence: mandatory. Otherwise: discard all
StandardModel = Classical       ‘ Mandatory
Amplitude = -10 dB              ‘ Mandatory
RiceParameters = -90, 4.5       ‘ Mandatory if StandardModel is rice
Angle = 180                     ‘ Mandatory if StandardModel is pure
Delay = constant, 1             ‘ Mandatory
Correlation = filename.cor      ‘ Optional

CorrelationInputs = 2           ‘ Optional way to define correlation inside .ics -file
CorrelationOutputs = 2          ‘ Optional way to define correlation inside .ics -file
CorrelationCoefficients = 0.01-0.01j, 0.02-0.02j, 0.03-0.03j, 0.04-0.04j, 0.05-0.05j, 0.06-
0.6j                            ‘ Optional way to define correlation inside .ics -file
```

StandardModel

- This value defines the standard model of the tap. The “pure” refers to Pure Doppler. You must provide this value.
- Valid range: classical, constant, pure, or rice, no default value

Amplitude

- This is the amplitude of the tap. You must provide this value.
- Valid range: -60 dB – 0 dB, no default value

RiceParameters
- You must provide this field only when StandardModel is **rice**. The first value is the angle between the direct ray and the mobile station. The second value is the power ratio direct/scattered (k-factor).
- Valid range for the first parameter: -360 – 360 degrees of angle, no default value
- Valid range for the second parameter: -60 – 60 dB, no default value

Angle
- You must provide this field only when StandardModel is **pure** (Pure Doppler). The value is the angle between the direct ray and the mobile station.
- Valid range: -360 – 360 degrees of angle, no default value

Delay
- This is the delay of this tap in seconds. The type of the delay must always be “constant”. The syntax is thus always: “Delay = constant, time”.
- Valid range: 0 – 3 ms, no default value

Correlation
- This field defines the correlation of this tap to other taps. Notice that the path must be given as a path inside the EB Propsim device, not as a path inside your PC.
- Valid range: any valid .cor file

CorrelationInputs
- Defines correlation input count in case of correlative .ics –file. Input count must remain the same throughout the whole .ics file.

CorrelationOutputs
- Defines correlation output count in case of correlative .ics –file. Output count must remain the same throughout the whole .ics file.

CorrelationCoefficients
- Defines complex correlation coefficients for the fading tap (path). Because of the symmetry of the matrix, only values of the upper triangle are defined (lower triangle contains always same values as complex conjugates). As an example, following row defines the 2x2 MIMO result matrix shown below:

  CorrelationCoefficients = 0.01-0.01j, 0.02-0.02j, 0.03-0.03j, 0.04-0.04j, 0.05-0.05j, 0.06-0.6j

  The same values as a complete matrix:

	1	2	3	4
1	1	0,0100-0,0100i	0,0200-0,0200i	0,0400-0,0400i
2	0,0100+0,0100i	1	0,0300-0,0300i	0,0500-0,0500i
3	0,0200+0,0200i	0,0300+0,0300i	1	0,0600-0,6000i
4	0,0400+0,0400i	0,0500+0,0500i	0,0600+0,6000i	1

9.4 .IR File

EB Propsim IR-files are backwards compatible with the IR-files of earlier EB Propsim products.

The .IR-file consists of the header and impulse response data (CIR). Header contains ASCII type description about measurement and essential parameters like carrier wave center frequency, impulse response sample density and delay resolution. IR-data is a table of impulse responses. Each IR consists of many taps. Taps are described as their delay and complex strength. .IR-data is handled as a table.

DATA TYPE	EXPLANATION
uint8	8-bit unsigned two's complement integer
uint16	16-bit little-endian unsigned two's complement integer
uint32	32-bit little-endian unsigned two's complement integer
float32	32-bit little-endian IEEE floating point number
float64	64-bit little-endian IEEE floating point number
String	String using 8-bit wide characters and usually ending in character '\0' (it is possible that when string length is known otherwise, tailing zero may be absent.) Used character set may be for example ISO-8859-1, SJIS or similar. Use of anything else than ASCII characters is not recommended if files are used in different locales.

Table 1. Data types used in IR-file syntax description

9.4.1 Syntax

```
IR_FILE {
       INDEX_TABLE
       HEADER_DATA
       TAP_DATA
}

INDEX_TABLE {
       HEADER_DATA_INDEX     : uint32    = offset of HEADER_DATA
       TAP_DATA_INDEX        : uint32    = offset of TAP_DATA
       RESERVED_INDEX        : uint32    = 0xFFFFFFFF
       RESERVED_INDEX        : uint32    = 0xFFFFFFFF
       RESERVED_INDEX        : uint32    = 0xFFFFFFFF
       NULL_INDEX            : uint32    = 0x00000000
}

HEADER_DATA {
       COMMON_HEADER_DATA
       CREATOR_SPECIFIC_HEADER_DATA
       NULL_BLOCK
}

NULL_BLOCK : block {
       fields    : uint16    = 0
}

COMMON_HEADER_DATA : block {
       fields    : uint16    = 8 + 3
       TITLE        : field {
              size           : uint16    = size of data
              data           : string    = user comment of current file
       }
       SOURCE_TIME_STAMP     : field {
              size           : uint16    = size of data
```

```
        data            : string    = date & time from source file. a
        free-form string.
    }
    SOURCE_FILE_NAME      : field {
        size            : uint16    = size of data
        data            : string    = the source file name
    }
    CARRIER_FRQ : field {
        size            : uint16    = size of data
        data            : uint32    = carrier frequency in Hz or
                 float64            = carrier frequency in Hz
    }
    IR_SAMPLE_DENSITY     : field {
        size            : uint16    = size of data
        data            : float32   = samples/half-wave or
                 float64            = samples/half-wave. Used for
                 mobile speed calculation. If SD == 0, mobile speed
                 is not meaningful.
    }
    DELAY_RESOLUTION      : field {
        size            : uint16    = size of data
        data            : uint16    = nanoseconds or
             float64    = seconds. If 0, not available.
    }
    CLOSED_ROUTE: field {
        size            : uint16    = size of data
        data            : uint8     = 0 = open, 1 = closed
    }
    INTERPOLATION         : field {
        size            : uint16    = size of data
        data            : uint8     = 0 - round to nearest
                                1 - default interpolation
                                4 - four tap interpolator
                                14 - fourteen tap interpolator
    }
    CIR_UPDATE_RATE       : field {
        size            : uint16    = size of data
        data            : float64   = CIR update rate in Hz.
    }
    CIR_UPDATE_RATE_LOCKED            : field {
        size            : uint16    = size of data
        data            : uint8     = 0 : unlocked
                                      1 : locked
    }
    CARRIER_FRQ_LOCKED    : field {
        size            : uint16    = size of data
        data            : uint8     = 0 : unlocked
                                      1 : locked
    }

}

CREATOR_SPECIFIC_HEADER_DATA : block {
    fields    : uint16    = field count
    CREATOR     : field {
        size            : uint16    = size of data
        data            : string    = name of creator (no spaces
        allowed)
    }
}

PROPSOFT_HEADER_DATA : CREATOR_SPECIFIC_HEADER_DATA {
    fields    : uint16    = 3
    CREATOR     : field {
        size            : uint16    = size of data
        data            : string    = "PropSoft"
    }
    CELL_TYPE   : field {
```

```
                size            : uint16    = size of data
                data            : string    = CELL type
         }
         SEED        : field {
                size            : uint16    = size of data
                data            : float32   = seed for random generator
         }
    }

    GEN_IR_GEN_HEADER_DATA : CREATOR_SPECIFIC_HEADER_DATA {
        fields    : uint16    = 4
        CREATOR     : field {
                size            : uint16    = size of data
                data            : string    = "General-IR-Generator"
        }
        CHANNEL_TYPE: field {
                size            : uint16    = size of data
                data            : uint8     = 0 = channel 1, 1 = channel 2, 2
                = correlative channel 2
        }
        CORRELATION_FACTOR    : field {
                size            : uint16    = size of data
                data            : float32   = correlation factor (0, when not
                correlative channel)
        }
        OTHER_CHN_FILE_NAME   : field {
                size            : uint16    = size of data
                data            : string    = other channel file name (NUL
                when one channel only)
        }
}
TOOLBOX_HEADER_DATA : CREATOR_SPECIFIC_HEADER_DATA {
        fields    : uint16    = amount of fields
        CREATOR     : field {
             size: uint16    = size of data
             data: string    = "ToolBox"
        }
}

TAP_DATA {
        IRS       : uint32    = number of impulse responses
        TAPS      : uint16    = amount of TAP on each IR. If TAPS = 0,
        amount of taps is specified separately at the beginning of each
        CIR.
        IR : table of IRS members {
                OPTIONAL_IR_TAPS            : uint16    = amount of taps in
                            this impulse response (present only if TAPS=0)
                TAP : table of TAPS or OPTIONAL_IR_TAPS members {
                        DELAY   : float32   = delay value in nanoseconds
                        RE      : float32   = real part
                        IM      : float32   = imaginary part[1]
                     }
             }
     }
```

[1] If IRs have different number of taps and TAPs is not set to 0, the number of taps per IR is selected according to longest one. These "shorter" IRs must then be filled with dummy taps having `Re` and `Im` values zero. Note that delays are required to be in increasing order.

With non zero taps, `Re` and `Im` may have any values because values are re-scaled when converted to .SIM emulator format.

.IR to .SIM conversion keeps relative gains between channels and therefore each .IR file used at the same emulation must be equally normalised.

9.5 .MAT file

MAT file is a MathWorks Inc's MatLab® internal file format, usually generated with MatLab. Importing .MAT files is supported as either directly in simulation, or via conversion to .IR file format. Imported .MAT file must have four variables, and may additionally have nine more. Any variables not explicitly mentioned here are discarded during conversion. All variable names are case sensitive.

Please note that .mat file must be compatible with Matlab version 6. If you are using newer version of Matlab, use the save-command from Matlab command line with parameter –V6 in order to create version 6 compatible mat-file.

For example (typed to Matlab shell to save .mat –file):
save 'c:\my_model.mat' –V6

Reading .mat files uses Matlab Component Runtime (MCR), provided by MATLAB. 1984 – 2008, The MathWorks Inc. It is delivered under licensing terms defined by *"The Mathworks, inc. Software license agreement - Deployment Addendum".*

9.5.1 Required variables

NAME	EXPLANATION
CIRs	Number of channel impulse responses in model, min. 1000
Taps	Number of taps (paths) in model
Coeff	Matrix of complex impulse response coefficients, with "Taps" rows and "CIRs" columns. Each coefficient must have a delay value in delay matrix.
Delay	Matrix of coefficient delay values, with "Taps" rows and "CIRs" columns. Delays are expressed in nanoseconds.

9.5.2 Optional variables

These variables may be omitted from .MAT file. If any of these is not found, the default value is used.

NAME	DEFAULT	EXPLANATION
CIRUpdateRate	10000 Hz	Update rate of model, in Hertz. Valid range is 0.01 to 1000000 Hz.
Carrier_Frequency	2.2 GHz	Carrier center frequency. Valid range is 30MHz to 6 GHz.
Tap_Spacing	5 ns	Tap spacing
Route_Closed	1	Is route closed i.e. is model continuous. 0 = Non-continuous 1 = Continuous This parameter is only informative.
Sample_Density	2	Samples per half-wave; 2 to 1000.
Hardware_Usage	0	For backwards compatibility, value is ignored
Description	"Mat2Ir generated file"	Description of model.
CIRUpdateRateLocked	0	Is CIR update rate locked or changeable run-time 0 = Not locked (changeable)

NAME	DEFAULT	EXPLANATION
		1 = Locked
CarrierFrequencyLocked	0	Is carrier frequency of model locked or changeable run-time 0 = Not locked (changeable) 1 = Locked

9.5.3 MAT file example

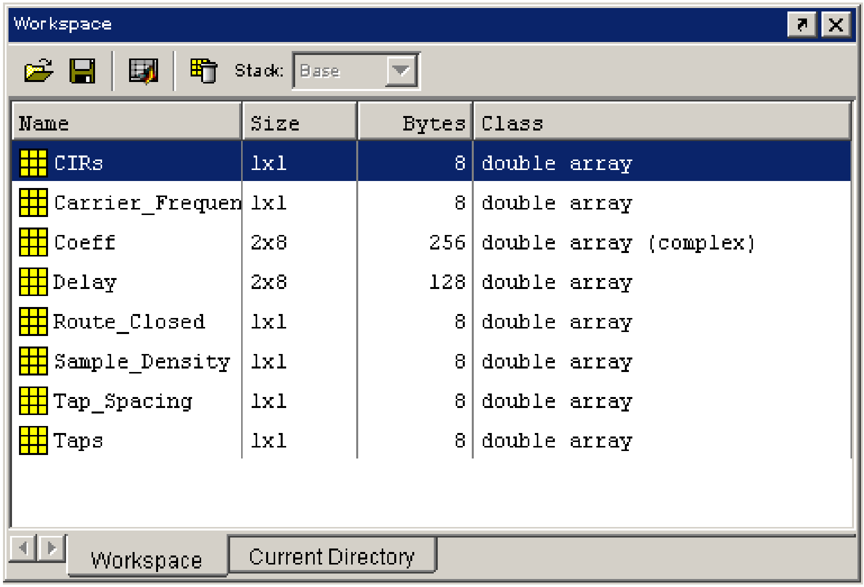


Figure 185. Example of workspace (as shown in MatLab) stored in .MAT file

Variables listed have following values;

```
CIRs                    8
Taps                    2
Coeff                   [1.0 0.8 0.6 0.4 0.2 0.4 0.6 0.8;
                        0.2i 0.4i 0.6i 0.8i 1.0i 0.8i 0.6i 0.4i]
Delay                   [0 0 0 0 0 0 0 0;
                        50 50 50 50 50 50 50 50]

Carrier_Frequency       2.1e+9
Route_Closed            1
Sample_Density          8
Tap_Spacing             5
```

9.6 Multi Emulator Scaler calibration file

The file format for calibration file is described in the example below.

```
<?xml version="1.0" ?>
<Calibration>
<Date>19.05.2010 (10:17:27)</Date>
<VNA_ID>Network analyzer device ID</VNA_ID>
<Fader_ID>EB Propsim device ID</Fader_ID>
<Frequency_MHz>2450</Frequency_MHz>
<Power_dBm>-15</Power_dBm>
<Output>
     <CH1>
    <!--gain and phase measurement for output 1-->
                            <Gain_dB>-2.10</Gain_dB>
                            <Phase_val>1175</Phase_val>
     </CH1>
     <CH2>
    <!--gain and phase measurement for output 2-->
                            <Gain_dB>-3.15</Gain_dB>
                            <Phase_val>1175</Phase_val>
     </CH2>
     <CH3>
    <!--gain and phase measurement for output 3-->
                            <Gain_dB>-1.05</Gain_dB>
                            <Phase_val>1175</Phase_val>
     </CH3>
     <CH4>
    <!--gain and phase measurement for output 4-->
                            <Gain_dB>-1.05</Gain_dB>
                            <Phase_val>1175</Phase_val>
     </CH4>
     <CH5>
    <!--gain and phase measurement for output 5-->
                            <Gain_dB>0.00</Gain_dB>
                            <Phase_val>1175</Phase_val>
     </CH5>
     <CH6>
    <!--gain and phase measurement for output 6-->
                            <Gain_dB>-3.15</Gain_dB>
                            <Phase_val>1175</Phase_val>
     </CH6>
     <CH7>
    <!--gain and phase measurement for output 7-->
                            <Gain_dB>-4.20</Gain_dB>
                            <Phase_val>1175</Phase_val>
     </CH7>
     <CH8>
    <!--gain and phase measurement for output 8-->
                            <Gain_dB>-2.10</Gain_dB>
                            <Phase_val>1175</Phase_val>
     </CH8>
</Output>
<Input>
     <CH1>
    <!--level and phase measurement for input 1-->
                            <Level_dBm>-19.95</Level_dBm>
                            <Phase_val>1919</Phase_val>
     </CH1>
     <CH2>
    <!--level and phase measurement for input 2-->
                            <Level_dBm>-19.45</Level_dBm>
                            <Phase_val>0</Phase_val>
     </CH2>
     </Input>
</Calibration>
```

10 Maintenance

EB Propsim is designed to require minimum maintenance. The main points of maintenance are discussed below.

10.1 Warranty

Standard warranty for EB Propsim emulators is one year from the date of the shipment. During the warranty period, EB will, at its option, either repair or replace products that prove to be defective.

If the emulator needs repair or calibration, either during or after the warranty period, please contact your EB Propsim sales representative or customer support through Elektrobit website www.elektrobit.com before sending the unit to maintenance.

Note: Unauthorized opening of the EB Propsim emulator enclosure voids the warranty.

Elektrobit System Test Ltd’s terms for warranty can be found in Introduction and Quick Guide parts of the EB Propsim product documentation.

Note: The unit shall be shipped to the service center in the original transmit case designed for EB Propsim.

10.2 Calibration and service

The recommended calibration period is one year. Calibration and service is provided by EB. Calibration is recommended to be ordered at least six weeks before the operation.

For calibration, service or repair, EB Propsim must be returned to a service facility designated by EB. The shipment to the service center shall be paid by the customer and the shipping charges for the return by EB.

Note: The unit shall be shipped to the service center in the original transmit case designed for EB Propsim.

10.3 Connectors and cables

To maintain the high performance of the emulator, proper care must be taken when connecting cables to the emulator.

When removing the RF local jumper cables, both connectors should be opened at the same time and bending the cable avoided. It is recommended to use a torque wrench to open and tighten the cables. Maximum torque is 1 Nm.

The connectors shouldn’t be touched, especially not the center conductor to prevent dust and dirt going into the connector. Also possible damages caused by static discharges are avoided.

10.4 Cleaning

The instrument front and rear panels should be cleaned using a soft and damp cloth with water or a mild soap and water mixture only. EB Propsim should be kept in a dust-free environment.

10.5 Over current protector

The main power switch of the emulator doubles as an over current protector. The emulator has no fuse that the user can change. If over current occurs, the emulator turns itself off and the switch returns to the OFF position.

If the over current protection turns the emulator off repeatedly, it is an indication of some fault in the emulator. The device should be sent to a facility designated by Elektrobit System Test Ltd. for appropriate service.

Note: The power switch must not be forced to the ON position if the over current protection shuts down the emulator. It will not help keep the emulator operational.

10.6 Assistance

Product maintenance agreements and other customer assistance agreements are available for EB Propsim products.

The most convenient way of contacting EB on all matters related to EB Propsim, is through Elektrobit website www.elektrobit.com or email wct.support@elektrobit.com.

11 Troubleshooting

This document explains the error messages that the EB Propsim emulator may produce and advises the proper troubleshooting actions.

The error indicators of EB Propsim consist of LEDs and error messages in the Graphical User Interface (GUI).

Most of the typical problems can be analyzed by using the basic one path constant channel model and a test setup with signal generators and a spectrum analyzer.

11.1 Typical problems

This chapter lists the typical errors that may be encountered and the corresponding procedures to solve the problems. If the problem still appears after the actions have been taken, please contact EB.

11.1.1 Emulator does not start

- Check that the power cord is connected to the rear panel of the emulator.
- Check that the power switch is turned on.
- Check that EB logo (FS8: standby switch) is alighted when standby switch in front panel is pressed.
- Check that the fans are running.

11.1.2 Self test fails

- Turn off the emulator using the standby switch in front panel of the emulator.
- Turn off the power using the power switch in rear panel of the emulator.
- Wait at least 15 seconds and turn the switch back on.
- If the reboot doesn’t help, contact EB.

11.1.3 No signal at the emulator output

- Check the frequency and the level of the input signal.
- Check that the test setup connections are according to the Active Connector View (in Running View of GUI).

11.1.4 Status led is red

Red color of the control unit status led indicates a warning due to
- overheating,
- over voltage or
- self test failure.

The reason of the warning is reported in the Status window of the Running view after emulation is loaded.

11.1.5 Power led is red

- Red color of the power led indicates power failure. Note: during initialization the led is blinking red

11.1.6 RF channel status led is red

Red color of the RF channel status led indicates a warning due to
- overheating,
- over voltage,
- input cut-off,
- RF local missing or
- self test failure.

The reason of the warning is reported in the Status window of the Running view after emulation is loaded.

11.1.7 Incorrect signal level

- Connect the emulator to a test setup with signal generator and spectrum analyzer.
- Load and run the constant one tap channel model
- Adjust the levels on the generators and the spectrum analyzer according to the figures in the GUI emulation settings.
- Measure the output signal level. The level should be the same that can be read as the expected output level in the GUI.

11.1.8 Incorrect test results

Incorrect DUT performance results arise usually from incorrect signal levels either at the input or output of the emulator. This can be avoided by going through the following steps:

- Open one channel static emulation in Running view from D:\Standard Emulations\Emulation Examples\Static Emulation\1to1 Static.smu
- Check the input signal level setting in the GUI. It should be -15 dBm.
- Average level of input signal must be the same i.e. -15 dBm.
- Enter the input signal crest factor in the GUI.
- Check with the spectrum analyzer that the signal level in the emulator input is -15 dBm.
- Connect test signal generator to EB Propsim RF IN 1 connector.
- Start emulation by clicking RUN button in the Running view.
- Click output box in GUI and set Output gain to 0 dB.
- Check with the spectrum analyzer that the output signal level value corresponds to the expected average level of the output in the GUI. Note the attenuation caused by the cable.

11.2 Error and warning messages

In this chapter, the error and warning messages of EB Propsim are listed and their meaning explained. The actions to be taken are also described for each error case.

11.2.1 Running view errors

The warning and errors of Running view are reported in Status window at the bottom of the Running view along with other status information. The Status window can be enabled or disabled from Window menu -> Show Status Window.

ERROR MESSAGE	DESCRIPTION / ACTIONS
Cannot find emulation file	Check that the emulation is generated and re-build if needed.
Cannot open emulation file	Check that the emulation is generated and re-build if needed.
Channel allocation failed	Emulation requires more HW resources than is available. It is necessary to simplify the channel models (reduce number of taps) or reduce the number of channels in emulation.
Connection to emulators failed	Emulation is already open either through GPIB or LAN. If not, restart the emulator.
Connection to emulator failed	Connection to emulator failed. Check that there is no open emulation through ATE interface.
Could not open <filename>	Check that file exists and is readable.
Calibration data corrupted in slot <number> unit	Contact EB.
Current warning	Contact EB.
Direct RF local level too high (RF <number>), adjust the level	Adjust the level or RF local.
Direct RF local missing (RF <number>), check the connection and level	Check the connection and level of RF local.
Emulation file copy failed. The destination cannot be reached	Re-build the emulation.
Emulation file missing, rebuild emulation	Re-build the emulation.
Emulation open failed	Re-build the emulation.
Emulation run failed	Re-build the emulation.
Emulator cannot be reached	Reboot the device. If the problem still occurs, contact EB.
Emulator control file missing, rebuild emulation	Re-build the emulation.
Emulator is out of extended delay area license	Emulation cannot be opened because of missing license.
Emulator is out of interference license	Emulation cannot be opened because of missing license.
Emulator is out of MIMO license	Emulation cannot be opened because of missing license.
Emulator is out of noise features	Emulation cannot be opened because of missing license.

ERROR MESSAGE	DESCRIPTION / ACTIONS
Emulator is out of noise licenses	Emulation cannot be opened because of missing license.
Error changing state	Reboot the device. If the problem still occurs, contact EB.
External reference clock not present	Check the external reference clock connection.
Failed to configure emulator	Reboot the device. If the problem still occurs, contact EB.
Go error	Reboot the device. If the problem still occurs, contact EB.
HW delay compensation failed	Emulator internal delay compensation failed. Restart emulator and try to load some simpler model without summing or splitting.
Input cut off warning	Peak input level is too high. Check Average Input Level and Crest Factor settings.
Internal RF local level too high (RF <number>)	Contact EB.
Internal RF local missing (RF <number>), check the RFLO jumper cable	Check the RFLO jumper cable.
Internal RF local unlocked.	Check the external reference clock signal level. Specified signal level > 0 dBm, max. + 20 dBm.
Missing emulation file	Check that emulation is build.
Network client closed error	Reboot the device. If the problem still occurs, contact EB.
Network connection error	Reboot the device. If the problem still occurs, contact EB.
Network receive error	Reboot the device. If the problem still occurs, contact EB.
Network send error	Reboot the device. If the problem still occurs, contact EB.
No AWGN option license available	Emulation cannot be opened because of missing license.
Not enough emulators	Emulation is too big for one emulator, reduce channel count.
No shadowing license available	Emulation cannot be opened because of missing license.
- no shadowing option license	Emulation cannot be opened because of missing license.
- not enough noise sources available	Emulation cannot be opened because of missing license.
Parity error	Contact EB.
Self test failed	Contact EB.
Setting of RF local sources failed	Reboot the device. If the problem still occurs, contact EB.
Shadowing profile file not found	Check that shadowing file (.SHD) exists.
SMU file closing failed	Reboot the device. If the problem still occurs, contact EB.
Source file missing, check emulation	Check that channel model file exists (.TAP/.IR/.MAT/.ASC)

ERROR MESSAGE	DESCRIPTION / ACTIONS
Stopping emulation failed	Reboot the device. If the problem still occurs, contact EB.
System clock unlocked. Check External 10MHz reference clock frequency	External reference clock frequency is not in specified range (10 MHz ±5 Hz) or signal level is too low (< 0 dBm).
System clock unlocked (oscillator warming up).	After changing from external reference clock to internal reference clock, internal oscillator warming up time is up to 10 minutes before it set up to proper frequency. If the message persists, contact EB.
Unknown network client error	Reboot the device. If the problem still occurs, contact EB.
Update rate file missing, rebuild emulation	Re-build the emulation.
Voltage warning	Emulator has an internal voltage level error. If restart does not help, the device requires service. Please contact EB.
Warning: High device temperature. Ensure free airflow to prevent device shutdown.	Internal temperature is too high. Turn the emulator off and make sure that there is enough airflow through the emulator. When CPU temperature is over limit, the CPU clock frequency is decreased.

11.2.2 Editing view errors

The warning and errors of Editing view are reported in Status window at the bottom of the Running view along with other status information.

ERROR MESSAGE	DESCRIPTION / ACTIONS
Error: Antenna array specification is invalid	Antenna element count in ARR file and channel count in TAP file do not match.
Error: Can't mix taps with and without DoA model	Specify the direction of arrival either for all taps or none in the channel model.
Error: Channel count specified doesn't match with correlation	Correlation matrix rank in COR file and channel count in TAP file do not match.
Error: Channels have incompatible correlations	Check the correlation file.
Error: Correlating channels must have same tap counts	Check the correlation file.
Error: Dynamic AoA not supported	Contact EB.
Error: No tap file specified or tap file not found	Check that you have specified a TAP file. This error message might also occur if the filename includes some special characters. Change the name to ASCII characters.
Error: Invalid antenna type found in antenna array	Contact EB.
Error: unable to create IR –file <filename>	Check that the same emulation is not open from Running view or via remote interface (LAN or GPIB). Check that the directory exists, you have writing permissions to it and that there is no read-only file with the same name. Finally, check that you have free space in the drive.

ERROR MESSAGE	DESCRIPTION / ACTIONS
Error: Unable to reach specified directory	TAP file directory was invalid. This error message might also occur if the directory includes some special characters. Change the directory to ASCII characters.
Error: Correlation data is invalid	Correlation matrix is missing or not valid. Check the COR file.
Error: Invalid channel count specified (possibly due to TAP files)	Contact EB.
Error: Internal IR sequence generation error	Contact EB.
ERROR: IrToSimCompiler: IR-file opening failed.	Check that you have the file in question.
ERROR: IrToSimCompiler: SIM-file writing failed.	Check that the directory exists, you have writing permissions to it and that no read-only file with the same name exists. Check also that you have free space in the drive.
ERROR: IrToSimCompiler: unhandled exception.	Reboot the device. If the problem still occurs, contact EB.
ERROR: Missing field in SIM-file common header.	Corrupted IR file.
Error: No such correlation file	Check that COR files specified in TAP file exist.
Error: Unsupported amplitude distribution specified	Specified amplitude distribution is currently unsupported.
Error: Unsupported delay function specified	Specified delay function is currently unsupported.
Error: unsupported Doppler spectrum specified	Specified Doppler spectrum is currently unsupported.
Error: Unable to write impulse response -- disk full?	Check that you have free space in the disk.
Error: Invalid emulation parameters; DOA unspecified?	Check that DOA and ARR files defined in TAP file are valid.
ERROR: No null index.	Invalid data format.
ERROR: Missing fields in common header.	Invalid data format.
ERROR: Missing creator field in creator specific header.	Invalid data format.
Internal compiler error: Invalid parameter	Contact EB.
Not an .IR file.	Input file did not have IR extension.
Not a .SIM file.	Input file did not have SIM extension.
Nothing to be done!	No taps specified in TAP file.
Unable to open file: <filename>	Check that the file exists.
Undefined error.	Contact EB.
Undefined fatal error.	Contact EB.
Steering vector error: please check AOA/ARR files	Check that AOA and ARR files defined in TAP file are valid.
WARNING: Defaulting cirUpdateRate to 10e3!	Invalid data format, cirUpdateRate missing

ERROR MESSAGE	DESCRIPTION / ACTIONS
WARNING: Missing fields in General-IR-Generator creator specific header.	Invalid data format.
WARNING: Old SIM-file, defaulting carrierFreqLocked to false.	Invalid data format.
WARNING: Old SIM-file, defaulting cirUpdateRateLocked to false.	Invalid data format.
WARNING: Unknown creator field in common header.	Invalid data format.
WARNING: Sequence #n signal level is saturated to xx dBm (target yy dBm)	Check levels in ICS-file.

11.2.3 Channel model view errors

ERROR MESSAGE	DESCRIPTION / ACTIONS
ERROR: .cor file channel count not matching!	Number of selected channels in the New Model Generation Wizard must match to number of channels selected in the Correlation Editor.

11.2.4 Correlation editor errors

ERROR MESSAGE	DESCRIPTION / ACTIONS
ERROR: Invalid matrix declared.	Check the matrix status. The message appears if the status of the matrix is not valid.

11.2.5 ATE Command specific warnings

ERROR MESSAGE	DESCRIPTION / ACTIONS
ATE command not supported	Invalid command. Check the command syntax. If the LAN server does not accept commands even though the syntax seem to be correct, check that the End Of String mark character is '\n' (line feed). When using for example TELNET the user must use Ctrl-ENTER instead of plain ENTER.

12 Wireless propagation environment

Understanding the radio channel behavior is a key factor in performing meaningful emulations with the EB Propsim. The mechanisms of electromagnetic wave propagation are complex and diverse. The following sections explain the most important concepts of radio channel propagation.

The wireless environment imposes many constraints and limitations on the performance of wireless telecommunication systems. If the radio propagation were to take place in ideal free space, the propagation model would be fairly simple. In this ideal free-space model, the RF energy attenuates between the transmitter and receiver according to a simple mathematical formula. The attenuation factor is called the path loss or free space loss. For this case of idealized propagation, received signal power is very predictable. However, to get such a simple model quite radical assumptions of the propagation environment have to be made. The ideal free space model assumes that the region between the transmitter and the receiver is considered as being free of all objects that might absorb or reflect RF energy. It also assumes that within this region the atmosphere behaves as a perfectly uniform and non-absorbing medium. Furthermore, the model considers earth as being infinitely far away from the propagating signal.

In a real world wireless communication systems signal propagation takes place in the atmosphere and near the ground with most practical radio channels. For these channels the free space propagation model is inadequate to describe the channel and predict system performance. When a wireless signal is sent from the transmitter to the receiver it traverses a complex radio channel that can distort the signal transmission in many ways. The transmission path between the transmitter and the receiver can vary from a simple line-of-sight (LOS) to one that is severely obstructed by both man made objects, such as buildings and vehicles, and geographical obstacles, such as mountains and hills, trees and foliage.

12.1 Radio wave propagation

Wireless telecommunication systems transfer data by modulating the data on top of a radio wave. The data stream “piggybacks” on a carrier wave at a certain chosen frequency. Modulating the carrier wave causes also changes in the frequency of the wave. In general, the more data is carried on top of the carrier wave, the more **bandwidth** (BW) the transmission requires. Today’s communication systems generate very complex radio wave signals to maximize throughput in a bandwidth limited communication channel. In other words, spectral efficiency needs to be high. To understand the radio communication it is best to analyze the most basic form of a radio signal, the sine wave. Figure 186 illustrates two sine waves.

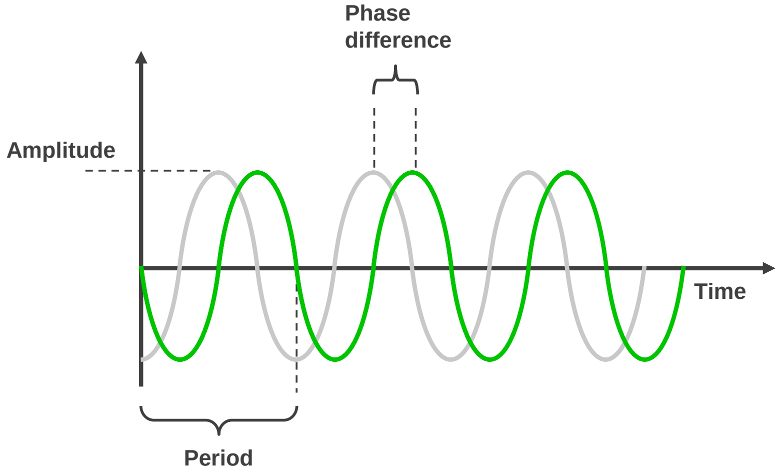


Figure 186. Sine wave

The sine wave can be unambiguously presented using three parameters; amplitude, frequency and phase. The sine wave is a waveform that oscillates between two peak values. The value of the peak is called the **amplitude** of the waveform. The sine wave has a period, during which the wave makes one full oscillation and after which the waveform repeats itself. The amount of full oscillations per second is called the **frequency** of the waveform. The distance that the wave travels during a full oscillation is called the **wavelength** of the waveform, which can be calculated from its frequency, and the other way around.

Thirdly, the wave has an offset from a specified reference point in time, known as the **phase** of the wave. As the phase of a signal changes the entire waveform is moved on the time axis. In Figure 186 there are two sine waves that are identical in all other aspects, except having a different phase. Phase is usually represented as degrees or radians. 360° degrees (2π radians) equals to moving the waveform a full period in time, so that the wave moves back to its starting position.

Radio waves can interact with objects in their propagation path in many ways. First of all, radio waves can travel through solid objects. This phenomenon is evident to anyone who owns a mobile phone, since the phones can be used even when staying indoors. However, the dense objects do **absorb** considerable amount of the radio wave energy, as illustrated in Figure 187.

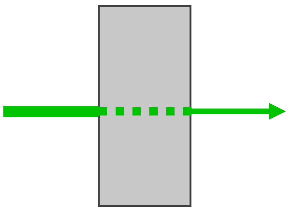


Figure 187. Absorption

Obstructions in the radio wave propagation environment can **reflect** radio signals, much in the same way than a mirror reflects light. This phenomenon is illustrated in Figure 188. Reflection occurs when the radio wave meets a large object that has a smooth surface and large dimensions compared to the RF signal wavelength.

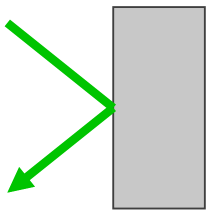


Figure 188. Reflection

Another common phenomenon that affects radio wave propagation is **scattering**. Scattering means the spreading of the signal in all directions due to rough objects. Scattering occurs when a radio wave meets either a large, rough surface, or any surface whose dimensions are on the order of the RF signal wavelength or less. Scattering causes the reflected energy to spread out, or scatter, in all directions. Scattering is illustrated in Figure 189. In an urban environment typical signal obstructions that create scattering are lampposts, street signs and foliage, for example.

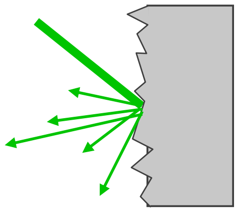


Figure 189. Scattering

Diffraction is a phenomenon where radio waves bend and spread when they meet a large obstacle that has large dimensions compared to the RF signal wavelength. This causes secondary waves to be formed behind the obstructing body, as illustrated in Figure 190.

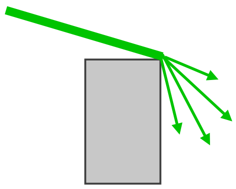


Figure 190. Diffraction

The radio waves can “bend” around the obstruction. Diffraction is a phenomenon that makes it possible for RF energy traveling from the transmitter to the receiver even without a line-of-sight path between the two.

12.2 Shadowing

Shadowing is a phenomenon in which a large obstruction such as a hill or large building obscures the main signal path between a transmitter and a receiver, causing attenuation in the received signal power. This phenomenon is termed shadowing because the receiver is being “shadowed” by an impenetrable obstruction, as illustrated in Figure 191.

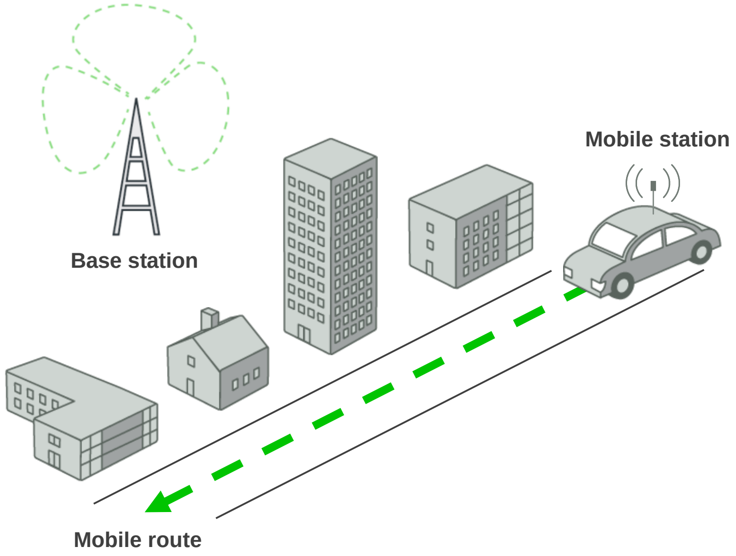


Figure 191. Shadowing

With shadowing the received average signal power varies as mobile moves over a distance corresponding to several tens of wavelengths or more. Because the shadowing effect causes variation of the signal power over relatively large distances it is also called large scale fading. The red line in Figure 192 illustrates the effect of shadowing in received signal power.

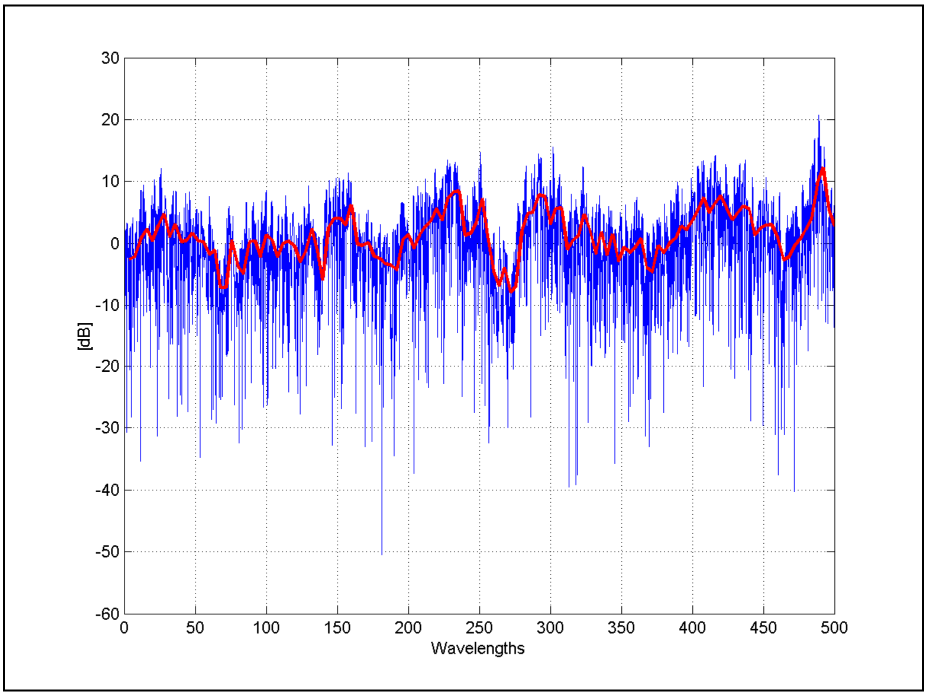


Figure 192. Shadowing graph

The amplitude change caused by shadowing is often modeled using a log-normal distribution.

12.3 Multipath propagation

The wireless communication systems lack a guiding medium for the signal. The transmission propagates to all directions from the transmitter. Most wireless communication systems operate in urban areas, where there rarely is a direct line-of-sight (LOS) path between the base station and the mobile terminal. In this environment the presence of high buildings and other obstructions create several different reflective paths between the transmitter and the receiver. The result is that a transmitted signal propagates to the receiver antenna through many different spatial paths of varying lengths. This phenomenon is called **multipath propagation** and it is illustrated in Figure 193.

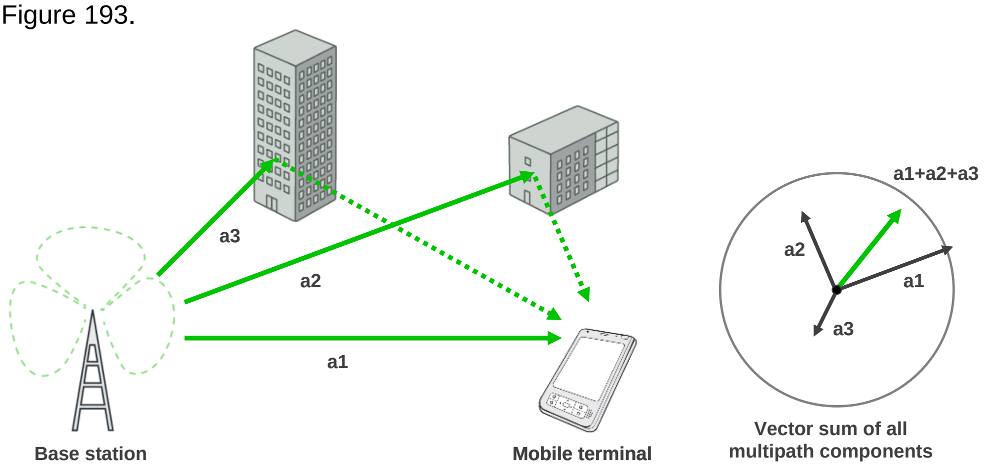


Figure 193. Multipath propagation

The received signal “captured” by a mobile stations antenna at any point in space consists from a number of **multipath components** (MPC). Each MPC has random amplitude, phase and direction of arrival (DoA), as well as different propagation delay. These signals combine at the receiver antenna, so that the resulting signal is the vector sum of all the MPCs. Multipath results in distortion of the signal due to the summation of multiple delayed copies of the signal with different phase relationships. In channel modeling the MPCs are usually referred to as **taps** or paths.

The MPCs can sum up constructively or destructively. When the mobile terminal is moving the vector sum of all received MPCs changes as well. As a result the received signal power fluctuates rapidly. This power fluctuation phenomenon is known as **fast fading**, or small scale fading. As a concept a communication channel that experiences fast fading is called a fading channel. An example of the received signal power in a fading channel is illustrated in Figure 194.

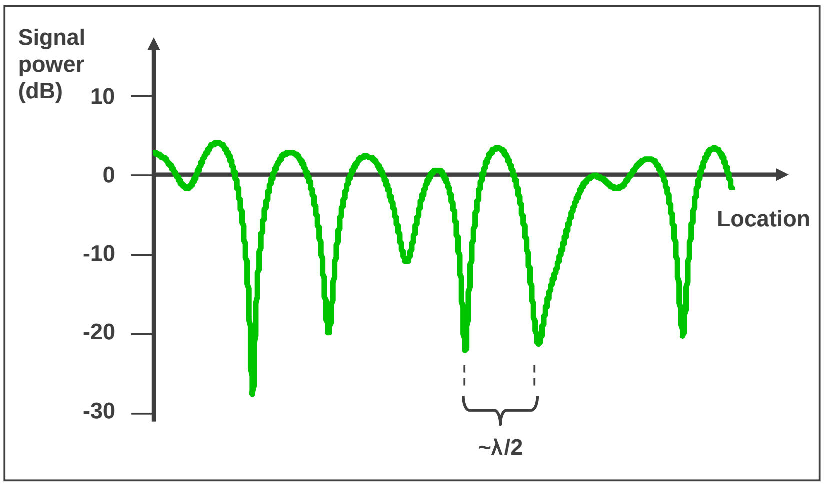


Figure 194. Fading channel

The destructive summing of MPCs in a fast fading environment causes deep fades in the received signal power. At these fading instances error free signal reception is effectively impossible. The interval at which the deep fades occur in the spatial domain is closely related to the carrier frequency of the signal. The deep fades are approximately a half carrier wavelength apart. The interval at which a mobile station experiences the deep fades depends on the mobile station speed.

12.4 Factors affecting overall attenuation

The instantaneous received signal power in wireless communication systems is affected by many factors. As has been shown in the previous sections, the three main phenomena causing signal power attenuation are path loss due to distance from transmitter, shadowing caused by buildings and other obstacles, and fast fading due to multipath propagation environment. These phenomena affect the signal propagation together, simultaneously. Figure 195 illustrates the overall signal attenuation in respect to distance.

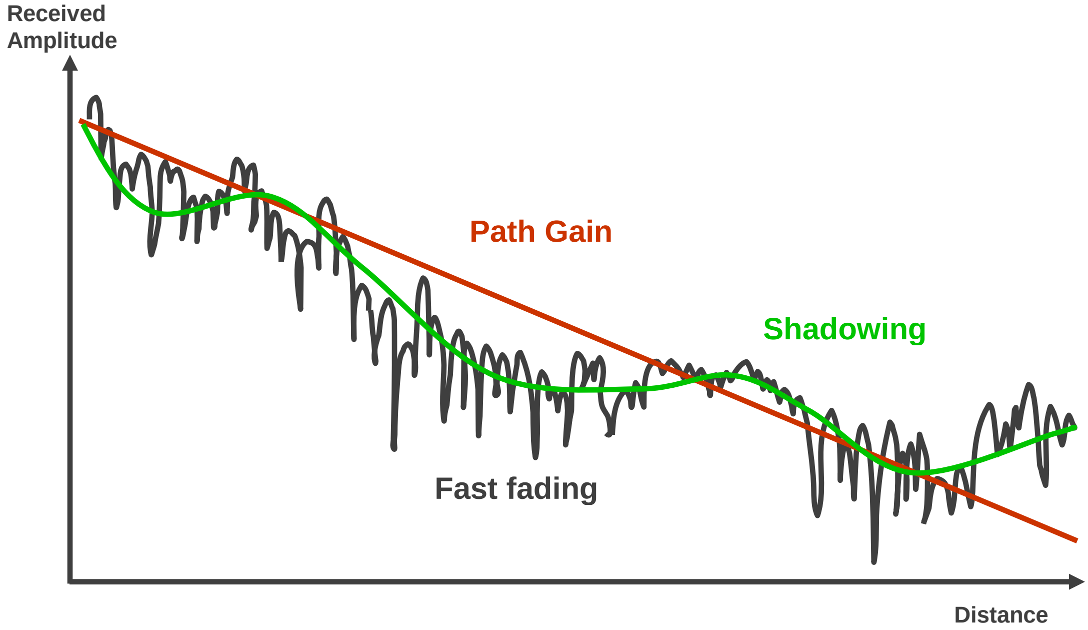


Figure 195. Overall signal gain

12.5 Doppler effect

When an emergency vehicle passes by, a clear change in the pitch of the siren can be observed. This is a very good example of the **Doppler effect** from everyday life. The change in the pitch results from a shift in the frequency of the sound waves. As the emergency vehicle approaches an observer, the sound waves from its siren are compressed relative to the observer. The interval between successive waves becomes smaller. In other words the frequency of the sound increases. As the ambulance passes and starts moving away from the observer, the sound waves are stretched relative to the observer. In other words the frequency of the sound decreases. The severity of the Doppler effect depends on the speed of the emergency vehicle, as well as on the angle of the incoming wave.

The Doppler effect also affects mobile communication systems, where the mobile terminals can be moving at vehicular speeds. As the receiver is moving relative to the base station, the Doppler effect causes a change in frequency and wavelength of the received signal.

Figure 196 illustrates the signal distortion in a telecommunication system caused by the Doppler effect. The radio waves arriving against the direction of the mobile experience a positive frequency shift, and the radio waves arriving to the direction of the mobile experience a negative frequency shift.

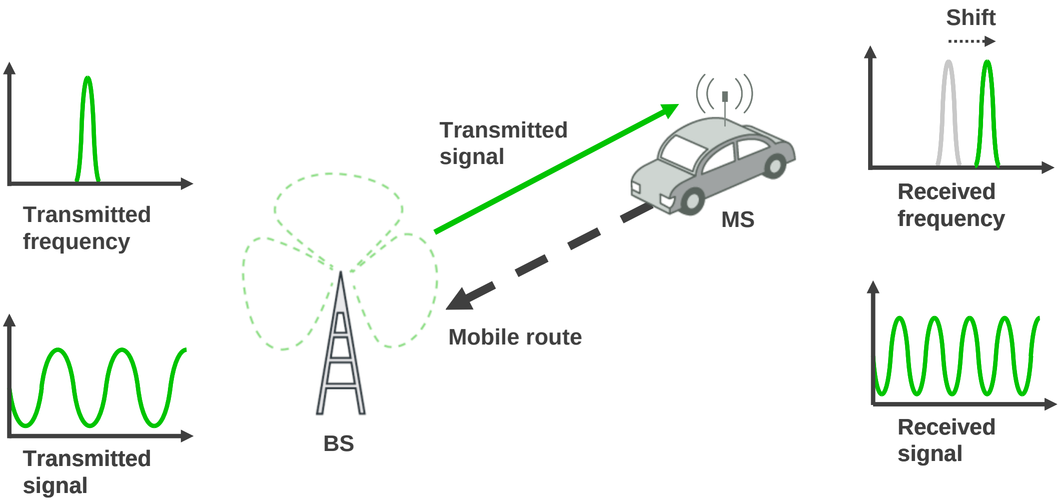


Figure 196. Doppler effect

Because of the multipath propagation environment the Doppler effect does not have a constant value. Because the different MPCs arrive from various directions the signal received by the mobile station includes varying frequency shifts, or Doppler shifts. The width of the frequency distortion is called the **Doppler spread**. The Doppler effect manifests itself as random frequency modulation within the received signal, as illustrated in Figure 197.

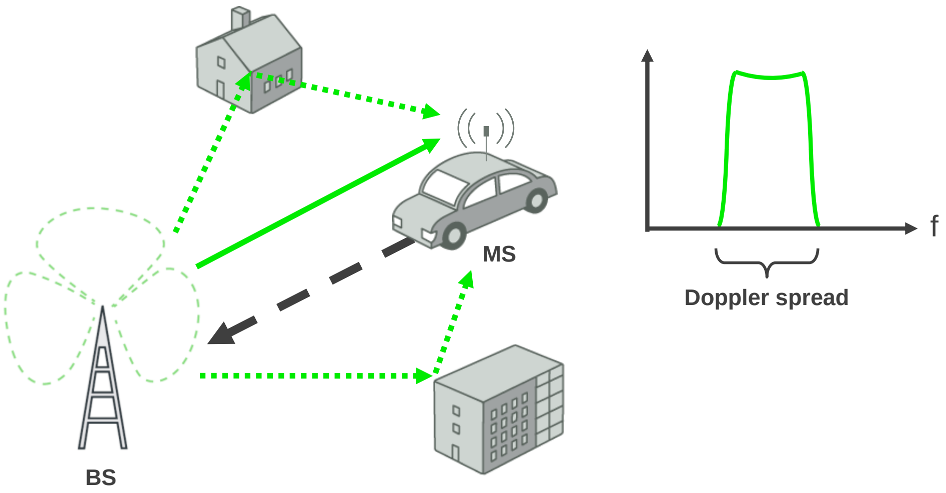


Figure 197. Doppler spread

The distribution of the Doppler spread depends on the type of environment. When the mobile is moving at a constant velocity the Doppler frequency for each MPC can be calculated from the carrier wavelength, mobile velocity, and the angle between the mobile direction and arriving signal. The signal arriving from the front of the mobile direction experiences the maximum frequency shift, called the **maximum Doppler frequency**.

12.6 Noise and interference

In real-life wireless communication the receivers are subjected to signals from multiple sources. The signals in the wireless propagation channel include many unwanted signals in addition to the wanted signal. These unwanted signals complicate correct detection and processing of the wanted signal at the receiver, thus deteriorating the receiver performance. The unwanted signals can roughly be divided into noise and interference.

An omnipresent problem with wireless communication is noise in the communication channel. Noise adds to the wanted signal, distorting the signal amplitude and phase. This increases the **Error Vector Magnitude** (EVM) and **Bit Error Rate** (BER) of the wanted signal. The most common type of noise experienced in real communication systems is **Additive White Gaussian Noise** (AWGN). AWGN is noise that has a constant spectral density, meaning it’s equally strong in all frequencies. The magnitude of AWGN is Gaussian distributed.

Another way to look at noise is to look at how it distorts the received symbol on the real and imaginary axes, as illustrated in Figure 198.

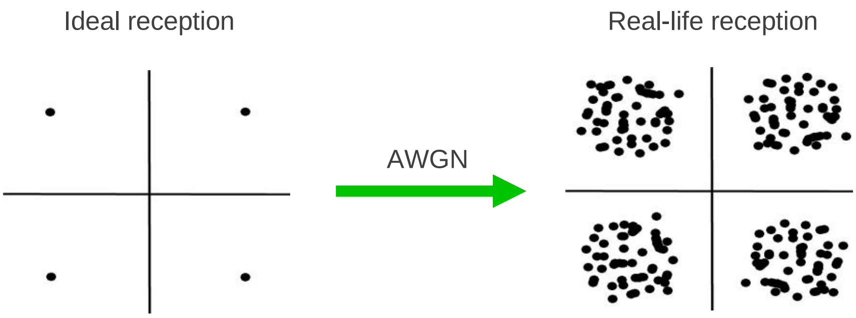


Figure 198. AWGN distorting reception

A signal sent by an ideal transmitter and received by an ideal receiver would have all constellation points precisely at the ideal locations, as shown in the left side picture of Figure 186. However, the addition of AWGN causes the actual constellation points to deviate from their ideal locations, as shown in the right side picture of Figure 198. If the constellation points deviate too far from their ideal locations they are interpreted incorrectly, causing bit errors in reception.

Another problem in wireless communication is signals coming from other wireless transmitters. If proper precautions are not taken in system design, these unwanted signals can interfere with the wanted signal. Several different types of interference exist, as illustrated in Figure 199.

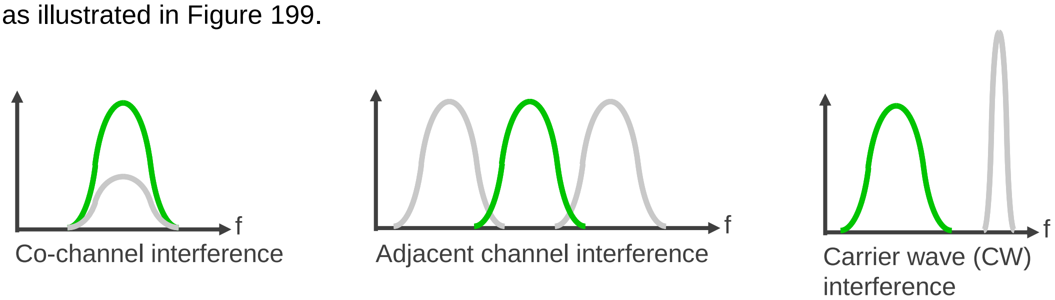


Figure 199. Interference types

12.7 Multi-antenna solutions

Multi-antenna or **Adaptive Antenna Solutions** (AAS) are a key tool in enhancing the performance of wireless telecommunication systems. Current systems are beginning to reach the limits of the achievable maximum bit rate and signal quality in a bandwidth-limited radio channel. With the evolution of antenna and digital signal processor (DSP) technology the new telecommunication systems can take advantage of using multiple antennas. Multiple antennas can be implemented either at the transmitter, at the receiver, or in both. AAS systems can improve a telecommunication system coverage, capacity or peak data rate. Several implementations of AAS systems exist. Generally, the AAS systems can be divided into the following categories:

- Receive diversity
- Transmit diversity
- Spatial multiplexing
- Beamforming

Even though the AAS implementations differ, they are all based on the same fundamental concept of finding parallel uncorrelating communication channels between multi-antenna transceivers. AAS systems take advantage of the radio propagation channel characteristics in the spatial domain. In a rich scattering environment, the fast fading phenomenon is unique between a certain transmitter-receiver channel in the spatial domain. Thus, each channel between two specific points in space has its own spatial "signature" that differentiates it from other propagation channels.

Receive diversity, also known as **Single-Input-Multiple-Output** (SIMO) increases the reliability of the radio link, meaning the signal-to-noise-and-interference (SNR) ratio. In SIMO systems the transmitter has one antenna but the receiver employs multiple, spatially separated antennas, as illustrated in Figure 200.

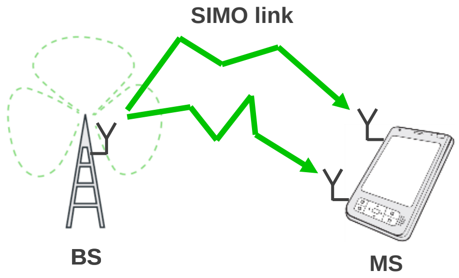


Figure 200. SIMO system

Transmit diversity, also known as **Multiple-Input-Single-Output** (MISO) is another way of increasing the radio link reliability. In MISO systems the transmitter employs multiple antennas, whereas the receiver only has a single antenna, as illustrated in Figure 201. Typically MISO systems pre-code the signals before transmission, using techniques such as Space Time Coding (STC). STC provides time diversity to the transmission.

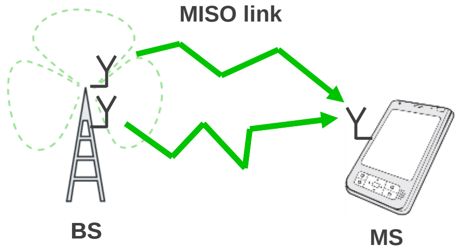


Figure 201. MISO system

In **Multiple-Input-Multiple-Output** (MIMO) systems both the transmitter and the receiver have multiple antennas. The generic notation for the MIMO antenna array size is “*N* X *M*”. Number “*N*” tells the number of transmitting antennas and number “*M*” tells the number of receiving antennas. An example of a 2 X 2 MIMO system is illustrated in Figure 202.

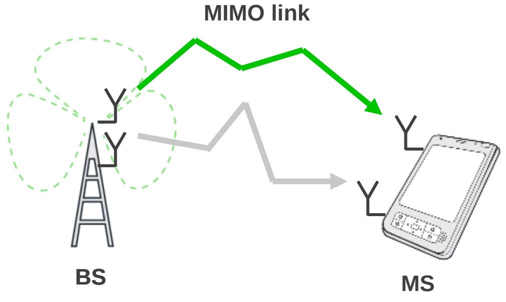


Figure 202. MIMO system

To give the reader a better understanding of AAS systems, the following sections introduce two different implementations of AAS.

12.7.1 Selection diversity

A simple implementation of a SIMO is a system that uses **selection diversity**. In its simplest form selection diversity utilizes two antennas at the receiver, as illustrated in Figure 203. The transmitter does not need to be modified in any way when selection diversity is used. The transmission propagates to the two receiver antennas through separate spatial paths. The signals propagating through the two spatial channels experience uncorrelated fast fading. The receiver has an antenna switch that selects the antenna that is receiving the highest signal power at any given time to be used for the reception of the signal.

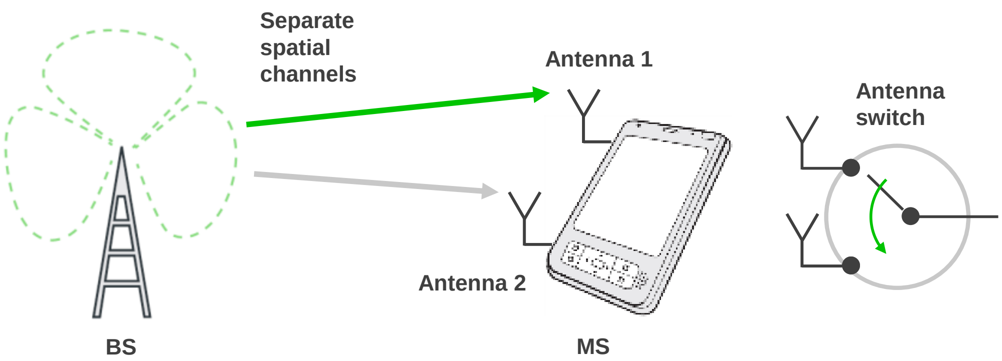


Figure 203. Selection diversity

Because the multipath fading for the signal depends on the receiver antenna location, the antennas receive different summed amplitude and phase of the signal. Thus the received signals in different antennas have only a small probability of being in a deep fade at the exact same time, as illustrated in Figure 204.

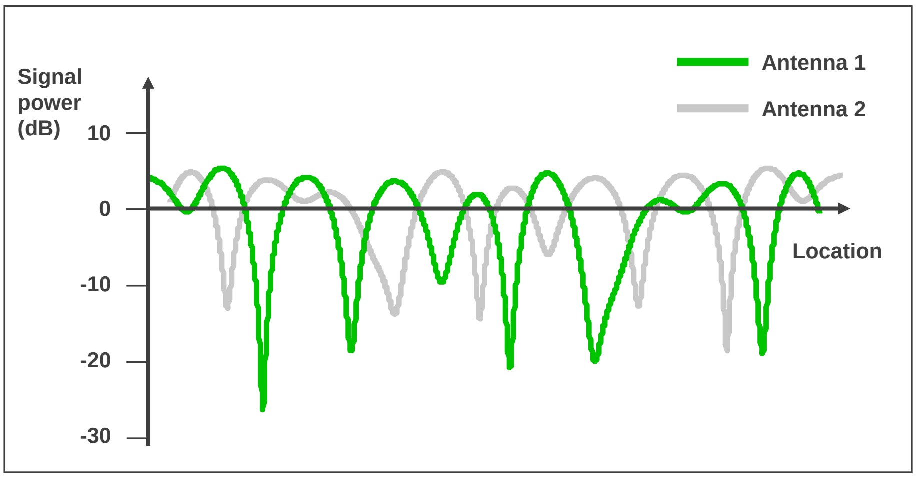


Figure 204. Reception diversity graph

The graph in Figure 204 shows that even a simple implementation of selection diversity can overcome the deep fades and greatly enhance the received signal quality. To obtain good results selection diversity requires that the spatial channels for the different receiving antennas are completely **uncorrelated**. This means that the channel characteristics, such as the location of the deep fades do not have any relationship to each other. A strong multipath environment and adequate spatial separation between the receiver antennas are prerequisites for uncorrelated spatial channels. In real-life communication systems it may be that the received signals have some correlation, in which case the performance of selection diversity becomes worse.

12.7.2 Spatial multiplexing

MIMO systems can increase the peak data rate of transmission by implementing a technology known as **spatial multiplexing** (SM). In MIMO SM implementations the transmitter sends independent data streams through different antennas. The data streams are transmitted through the same frequency channel at the same time, but through different transmitter-receiver chains. In theory, the capacity of a system employing spatial multiplexing increases linearly with the number of antennas, if the number of antennas at both transmitter and receiver is the same. This means that a 2X2 MIMO system has double the capacity than a system with no MIMO, and a 3X3 MIMO system has triple the capacity, and so on.

Because all the transmissions propagate through the same frequency channel, each of the MIMO receivers receives the sum of all independent transmissions. However, using sophisticated MIMO algorithms, the independent data streams can be separated within the receiver. This means that each transmitter-receiver pair can transmit an independent data stream without interfering with the others, effectively multiplying the achievable data rate for free. A 2x2 MIMO system employing spatial multiplexing is illustrated in Figure 205.

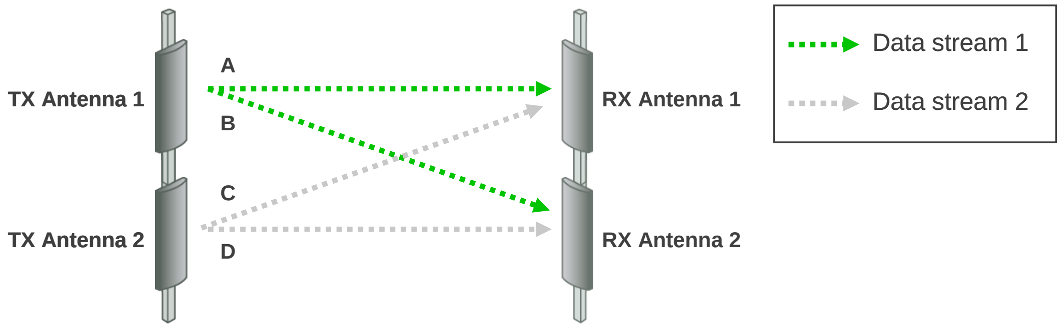


Figure 205. MIMO system

The MIMO algorithms use periodic **channel estimates** between each transmitter-receiver pair. The fading properties of the spatial channel between each pair are unique, each spatial channel has its own “signature”. Even though the “RX Antenna 1” in Figure 205 simultaneously receives the transmissions from both “TX Antenna 1” through propagation channel “A” and “TX Antenna 2” through propagation channel “C” it can use the different spatial “signatures” of these propagation channels to distinguish between the two transmissions.

As MIMO gain is based on the differences in the propagation channels, it requires a rich multipath environment to work properly. If the independent data streams do not arrive at the receiver with large enough spatial “signature” difference the MIMO processing is unable to distinguish the different streams. In other words, the “signatures” of the propagation channels have to be uncorrelated for the distinction to be possible. MIMO SM implementations also require a high signal-to-noise ratio (SNR). This limits MIMO SM implementations to use cases, where the receiver is close to the base station. If the receiver moves far away from the base station, only diversity technologies can be used to improve the communication link. Many communication systems that employ MIMO include a functionality to dynamically change the MIMO implementation from SM to TX/RX diversity, based on received signal quality.

12.8 Radio channel measurements

12.8.1 Channel impulse response

In order for the transmitters and receivers of a wireless communication system to operate optimally, the system needs to have knowledge about the propagation channel characteristics. As discussed above, the MPCs cause fading and Doppler effects. Additionally, it is necessary to study the impact of the different **delays** of the MPCs. Basically, the propagation channel acts as a filter that varies in time and space and distorts the signal. This filter's characteristics can be determined by "sounding" the channel. The result of this sounding process is known as the **channel impulse response** (CIR).

The channel impulse response is a measurable response of the radio communication channel that quantifies the varying filter characteristics of the channel in both the time and spatial dimensions. The CIR can be obtained by transmitting an extremely short impulse over the air. As the impulse propagates through the channel it gets filtered according to the channel characteristics. Thus the received, filtered signal then contains a "signature" of the current channel characteristics. The channel impulse response concept is illustrated in Figure 206.

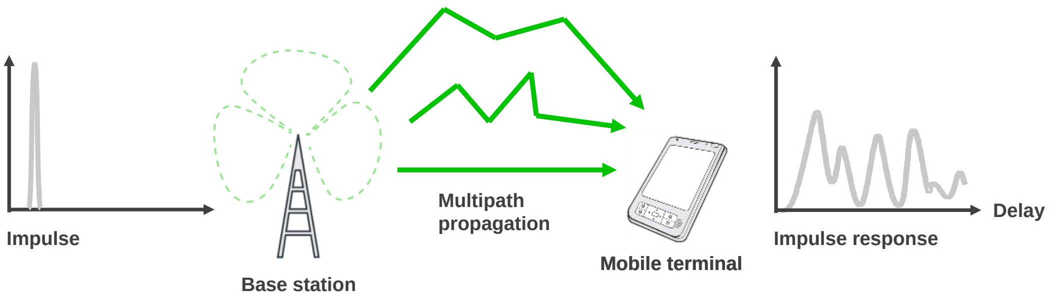


Figure 206. Channel impulse response

The impulse response (IR) measurements of a radio propagation channel provide important information for the design, development and planning of radio communication systems. This is easily understandable, since nearly all aspects of a radio communication system – from digital modulation techniques to channel coding and network aspects – are determined by the propagation characteristics of the channel. The statistical analysis of IR can be used for channel modeling and emulation. The IR of a signal that has propagated through the communication channel contains information, such as:

- Propagation delays for different taps, due to reflections from various surfaces.
- Doppler effects due to the movement of the transmitter and receiver, as well as the movement of other reflective surfaces in the propagation path.
- Path loss or attenuation between the transmitter and the receiver.

12.8.2 Frequency response

Whether the multipath components combine constructively or destructively depend on the relative phases of the multipath components. The relative phases of the components in a specific location depend on the signal frequency. Thus, the fading channel is different for different frequencies. In other words, the severity of the fading phenomenon in these channels is frequency selective. The characteristics of the frequency selective fading in a certain channel can be obtained by determining **frequency response** of the channel.

The frequency response can be obtained by transmitting a signal that includes all possible frequencies in equal proportions (having constant amplitude) through the channel. Incidentally, an ideal impulse consists of all possible frequencies. As the signal propagates through the channel different frequency components attenuate differently, according to the frequency selective fading characteristics of the channel. Figure 207 illustrates a time varying frequency response graph.

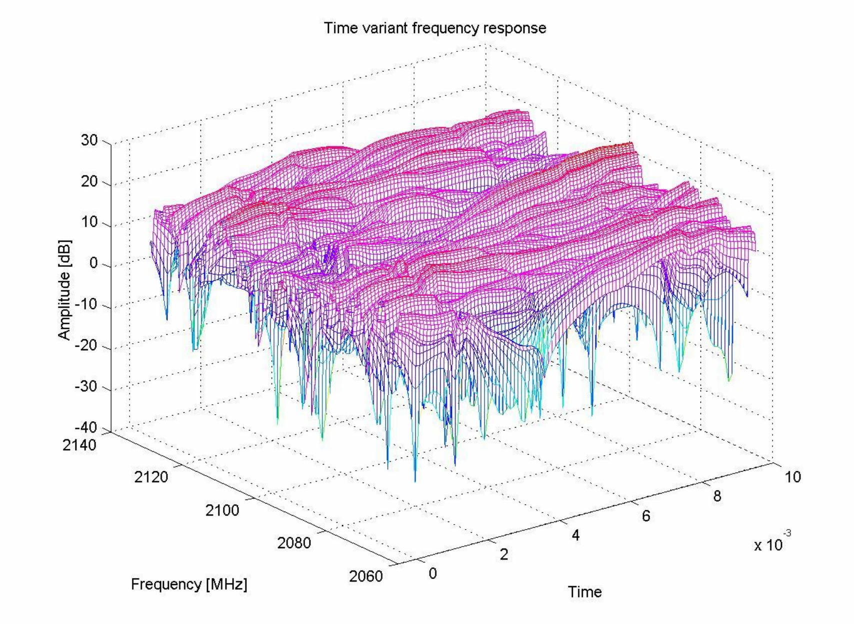


Figure 207. Frequency response graph

Whether or not the frequency selective fading causes problems for telecommunication systems depends largely on the system bandwidth. A wireless communication system that has a narrow channel bandwidth, such as the GSM (200 kHz) system is not affected nearly as much as a wideband system, such as the WCDMA (5 MHz) system.

The **coherence bandwidth** indicates the minimum separation in frequency, after which two signals will experience uncorrelated fading. A transmitted signal will experience **flat fading** if the coherence bandwidth is larger than the signal bandwidth. In this case all frequency components of the signal experience the same magnitude of fading. A transmitted signal will experience **frequency selective fading** if the coherence bandwidth is smaller than the signal bandwidth. In this case different frequency components of the signal experience different magnitudes fading.

12.9 Channel modeling

Radio channel modeling has a long history and is currently a very active area of research. This is especially true with terrestrial mobile radio communication systems, such as 3G and 4G cellular networks and WiMAX, due to commercial interest. Many of the telecommunication standardization organizations have their own standard channel models that can be used in testing the transmitter and receiver equipment.

Several different types of radio channel models exist today. One group is the **statistical channel models**. These models rely on modeling the channel characteristics, such as fading, through statistical analysis. As many of the radio channel properties are random, it is convenient to model the channel with appropriate random processes that produce numerical outcomes. Statistical channel models are widely used as commonly agreed reference test cases with different wireless standards.

Radio channel is modeled by time variant impulse response $h(t,\tau)$. If the receiver is moving, the impulse responses are different in every position of the receiver. Equation 1 shows the impulse response $h(t,\tau)$.

Equation 1

$$h(t,\tau)=\sum_{i=1}^{L}\beta_i(t)e^{j\phi_i(t)}\delta[\tau-\tau_i(t)],$$

where $\beta_i(t)$ and $\phi_i(t)$ represent the amplitude and phase of the i^{th} path arriving at delay τ_i and at time t. This equation is widely used for statistical modeling of both indoor and outdoor radio propagation. In so-called **Tapped Delay Line (TDL)** models, delays are fixed, but amplitudes and phases are fluctuating according to some statistical distributions.

Received signal $y(t)$ can be calculated by transmitted signal $x(t)$ and system's impulse response $h(t,\tau)$. Equation 2 shows that received signal $y(t)$ is convolution of the $x(t)$ and $h(t,\tau)$.

Equation 2

$$y(t)=x(t)*h(t,\tau).$$

12.9.1 Standard models

Available fading distributions in EB Propsim are shown in
Table 71. Standard models are widely used fading processes with certain amplitude distributions and Doppler spectra. For example model called Classical has Rayleigh amplitude distribution and Jakes Doppler spectrum, which both can be tuned with parameters. The custom models that can be created are listed in
Table 72 and described more detailed in the following subchapters.

MODEL NAME	AMPLITUDE DISTRIBUTION	DOPPLER SPECTRUM
Constant	Constant	None
Classical	Rayleigh	Jakes
Rice	Rice	Jakes + Pure Doppler
Flat	Rayleigh	Flat
Pure Doppler	Constant	Pure Doppler
Nakagami	Nakagami	Typical spectrum for Nakagami model
Lognormal	Lognormal	Butterworth
Gaussian	Rayleigh	Gaussian
Suzuki	Rayleigh + Lognormal	Jakes + Butterworth

Table 71. Available standard models.

AMPLITUDE DISTRIBUTION	DOPPLER SPECTRUM
Constant	Pure Doppler
Rayleigh	Jakes
	Flat
	Rounded
	Gaussian
	Butterworth
Rice	Jakes
	Flat
	Rounded
	Gaussian
	Butterworth

Table 72. Custom models.

12.9.1.1 Constant

Constant distribution has no fading and no Doppler; complex tap coefficients are constant. The I and Q components of the complex tap are related to amplitude and phase via Euler formula (Equation 3).

Equation 3

$$c = Ae^{j\theta} = A(\cos\theta + j\cdot\sin\theta) = I + jQ,$$

where A is the amplitude and θ is phase shift.

One tap constant model (Figure 208) can be used for calibrating a test system (emulator insertion loss measurement) and for tests without fading.

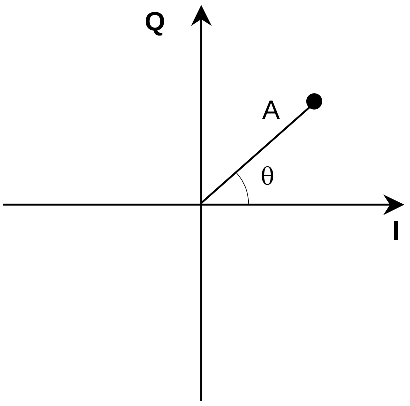


Figure 208. Constant model with constant phase and amplitude.

Parameters of the constant model:

- Delay [ns]
- Strength [dB]
- Phase shift [deg].

12.9.1.2 Pure Doppler

Pure Doppler distribution has no fading, but includes Doppler frequency shift. Doppler frequency shift is implemented by rotation of the coefficients phase. The rate of the phase rotation is determined by the mobile velocity, carrier center frequency and angle between the incident ray and the direction of the movement on horizontal plane. The Pure Doppler model simulates the line of sight (LOS) propagation without fading. Doppler frequency shift is

Equation 4

$$f_d = \frac{v}{c} f_c \cdot \cos\alpha \,,$$

where

v is vehicle speed
c is velocity of light
f_c is carrier frequency
α is the angle between mobile motion and incoming radio wave, see Figure 209.

Figure 209. Angle between mobile motion and incoming radio wave

Parameters of the Pure Doppler model:

- Delay [ns]
- Strength [dB]
- Angle α [deg] between mobile motion and incoming radio wave on horizontal plane, see
- Figure 209
- Vehicle speed [m/s]
- Carrier frequency [MHz]
- Phase shift [deg].

12.9.1.3 Classical (Rayleigh) fading

Classical is a statistical channel model often used to model the effect of fading in an urban area, where there are many objects in the environment that reflect and scatter the radio signal before it arrives at the receiver. Rayleigh fading models assume that there is no dominant line-of-sight (LOS) path between the transmitter and the receiver. The received signal is composed of many reflections coming from all directions in a horizontal plane (azimuth plane), as illustrated in Figure 210.

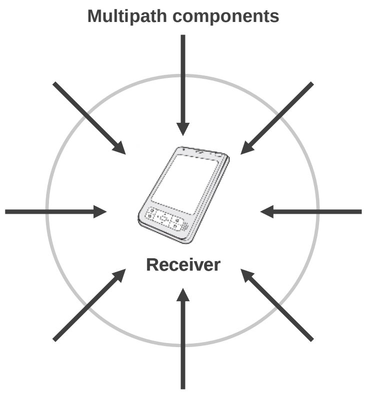


Figure 210. Rayleigh fading environment

In this environment the received signal power will vary, or fade, randomly according to a **Rayleigh distribution**. The received multipath components have random amplitude. Thus, the combined received signal amplitude at any given moment is also random. However, receiving certain amplitude values is more probable than others. For example, the deep fades in a fading channel occur only in a small percentage of the time, so their probability is rather small. Most of the time the signal is experiencing small to moderate amount of fading, so those amplitude values have a higher probability. Often the probability of receiving different amplitudes is represented as a **probability density function** (PDF) graph. A Rayleigh amplitude PDF graph is illustrated in Figure 211.

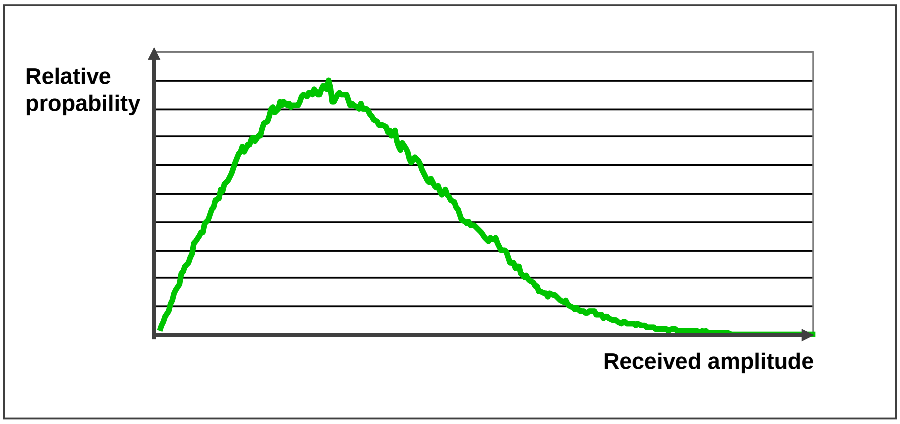


Figure 211. Rayleigh PDF

The PDF is a statistical graph. It shows the received amplitude with regard to the relative probability of receiving that value. As the amplitude can take any value between a certain range of amplitudes, rather than being restricted to a countable amount of discrete amplitudes, the PDF has a continuous distribution. Since a continuous distribution has an uncountable number of possible values, the probability of receiving a certain discrete value is diminishingly small. This means that in theory the probability for receiving any specific value is zero. However, receiving a signal in a certain range of amplitudes has a nonzero probability that can be calculated.

Classical model simulates a multipath environment where all multipath components have the same delay and there is no line of sight signal.

Rayleigh fading coefficients can be generated the following way. Complex white noise with Gaussian distributed real and imaginary parts are filtered with a digital spectrum shaping filter to obtain the desired Doppler spectrum. This approach yields desired temporal correlation and Rayleigh amplitude distribution performing steep and deep fades in average every half wavelength, see Figure 211 and Equation 5.

The Rayleigh probability density function of amplitude *r* is given by

Equation 5

$$p_{Ra}(r) = \frac{r}{\sigma^2}\exp\left(-\frac{r^2}{2\sigma^2}\right),$$

where *r* is amplitude and σ^2 is the variance of both the real and imaginary components alone.

The classical model assumes equal occurrence for all incident angles, resulting in the following Doppler power spectrum formula (Equation 6) and plot (Figure 212).

Equation 6

$$S(f) = \frac{1}{\pi f_d \sqrt{1-\left(\frac{f}{f_d}\right)^2}},$$

where f_d is the maximum Doppler frequency shift, $f_d = f_c \cdot (v/c)$.

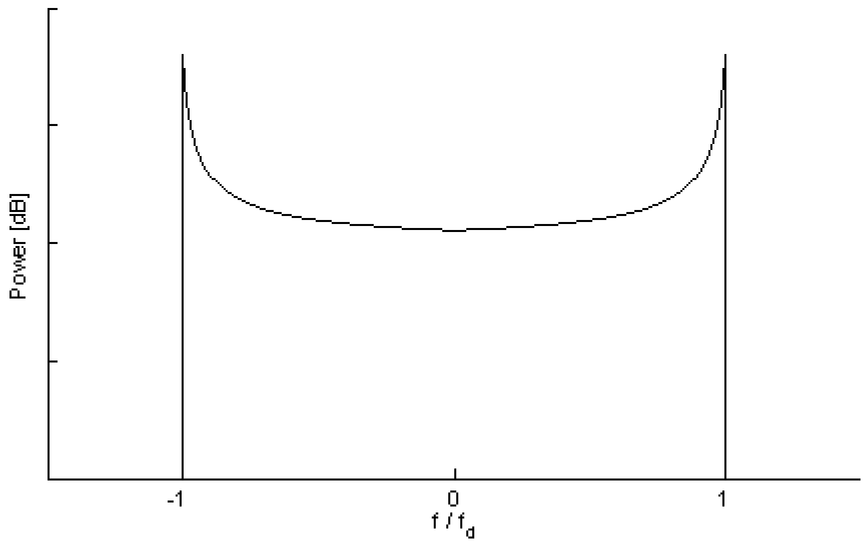


Figure 212. Classical Doppler spectrum

Parameters of the Classical model:

- Delay [ns]
- Strength [dB]
- Vehicle speed [m/s]
- Carrier frequency [MHz]
- Phase shift [deg].

12.9.1.4 Flat

The Flat model is equivalent to the Classical model except for the shape of the Doppler power spectrum. The amplitude distribution remains the Rayleigh distribution. In the case of the Classical model it was assumed that waves arrived with uniform probability from all horizontal angles. By contrast, a more reasonable assumption for the indoor environment, particularly when propagation occurs between floors, is that waves arrive with uniform probability from all horizontal and vertical angles.

The Flat model employs a rectangular spectrum shaping filter with a flat amplitude response from $-f_d$ to f_d and zero response elsewhere. The resulting Doppler power spectrum has a flat magnitude characteristic from $-f_d$ to f_d. The maximum Doppler shift is denoted by f_d.

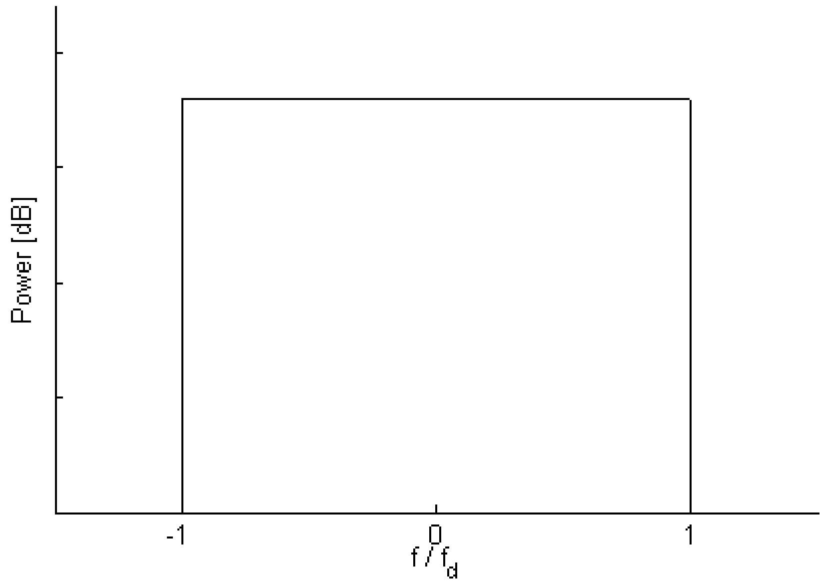


Figure 213. Flat Doppler spectrum.

Parameters of the Flat model:

- Delay [ns]
- Strength [dB]
- Vehicle speed [m/s]
- Carrier frequency [MHz]
- Phase shift [deg].

12.9.1.5 Rician fading

In some wireless propagation environments, such as in rural areas where there are no high buildings, a line-of-sight (LOS) communication path between the transmitter and receiver exists. In this case a strong direct signal component can be received together with multiple multipath signals, as illustrated in Figure 214.

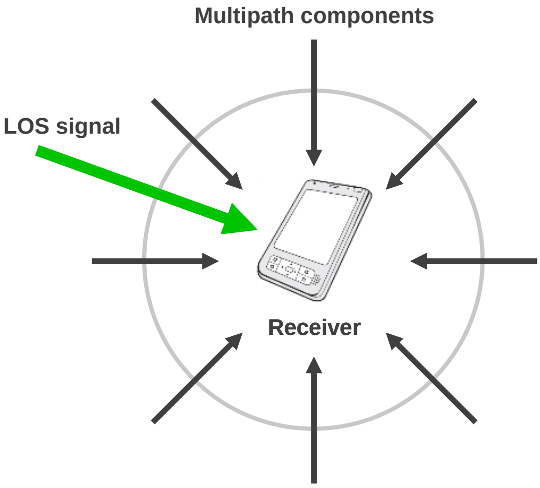


Figure 214. Rician fading environment

In this kind of environment a **Rician fading** model is better suited to model the channel characteristics. The Rician fading model includes a so called ***K*-factor**, which tells the relative strength of the direct signal compared to the multipath components. The received signal probability distribution is called Rician distributed PDF. Figure 215 illustrates a Rician amplitude PDF graph with different K-factor values.

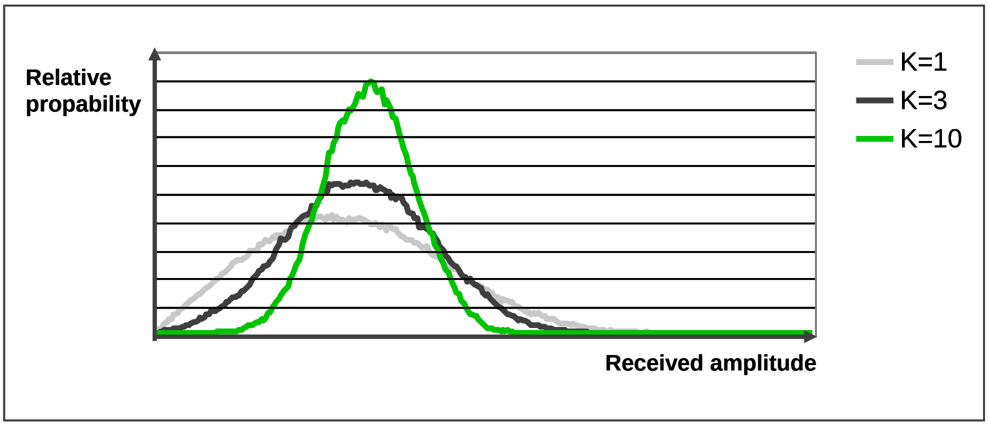


Figure 215. Rician PDF

The Rician model is a combination of the Classical (non line of sight, NLOS) model and the Pure Doppler (line of sight, LOS) model. In other words, there is a dominant path in addition to the scattered paths in a Rician fading channel. The power ratio between LOS (direct wave) and NLOS (scattered waves) is the Rician *K* factor. The Rician factor is defined as

Equation 7

$$K = \frac{r_s^2/2}{\sigma^2} = \frac{r_s^2}{2\sigma^2},$$

where r_s is magnitude of the LOS component and σ^2 is as defined in Equation 5.

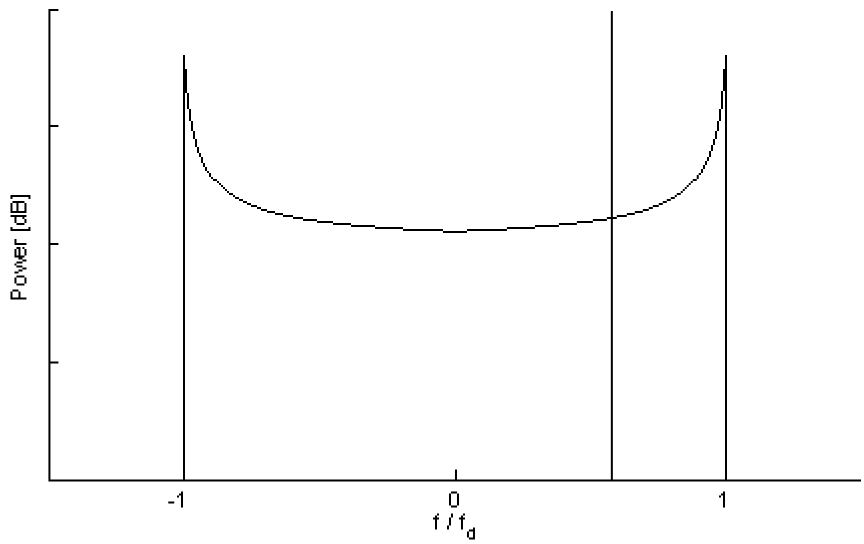


Figure 216. Power spectrum of Rician model (Classical spectrum + specular component)

When the dominant path becomes weak (*K* factor is low), the Rician distribution degenerates into the Rayleigh distribution.

The Rician probability density function of amplitude *r* is given by

Equation 8
$$p_{Ri}(r) = \frac{r}{\sigma^2} \exp\left(-\frac{r^2 + r_s^2}{2\sigma^2}\right) I_0\left(\frac{r r_s}{\sigma^2}\right),$$

where r_s is the magnitude of the LOS component, σ^2 is the variance of either real or imaginary components of the multipath part alone and $I_0(\cdot)$ is the modified Bessel function of the first kind and zero order.

The Rician distribution can be written in terms of *K* as

Equation 9
$$p_{Ri}(r) = \frac{2rK}{r_s^2} \exp\left(\frac{-K(r^2 + r_s^2)}{r_s^2}\right) I_0\left(\frac{2rK}{r_s}\right).$$

Rician *K* factor in linear scale can be calculated from

Equation 10
$$K_{lin} = 10^{K_{dB}/10}.$$

Parameters of the Rice model:

- Delay [ns]
- Strength [dB]
- Angle between mobile speed and direct ray [deg]
- Rice K-parameter (power ratio direct / scattered [dB])
- Vehicle speed [m/s]
- Carrier frequency [MHz]
- Phase shift [deg].

12.9.1.6 Nakagami

In Nakagami model a scattering object is simulated as specified in the CoDiT project [1]. The complex tap coefficient at time index *k* is calculated according to the following formula (Equation 11):

Equation 11
$$E_k = A_0 e^{j(\Phi_0 + k\frac{\pi}{SD}\cos(\alpha_0))} + \sum_{i=1}^{N} A_i e^{j(\Phi_i + k\frac{\pi}{SD}\cos(\alpha_i))},$$

where

- E_k complex tap coefficient at time index k,
- A_0 amplitude of the specular component,
- A_i amplitudes of the diffused partial waves,
- Φ_0 initial phase of the specular component,
- Φ_i initial phases of the diffused partial waves,
- α_0 mean incident angle from the scatterer with respect to the mobile route,
- α_i incident angles of the diffused partial waves,
- *SD* response sample density as number of impulse responses per half wavelength
- *N* number of partial waves = 105.

The time index k is incremented by the time step determined by the sample density SD.

The amplitude values depend on the m parameter as follows (Equation 12 and Equation 13):

Equation 12
$$A_0 = \sqrt[4]{1 - \frac{1}{m}}$$

Equation 13
$$\sum_{i=0}^{N} E\left[A_i^{\,2}\right] = 1,$$

where E[x] is the expectation value.

The partial wave amplitudes are random, taken from a truncated Gaussian distribution with zero mean and standard deviation conforming to the previous equations. All partial wave amplitudes have equal standard deviations.

The initial phase of the specular component Φ_0 and the initial phases of the partial waves Φ_i are evenly distributed.

The incident angles α_i are also taken from a truncated Gaussian distribution with mean value α_0 and standard deviation s = 0.15 rad = 8.59° [1].

This model yields the Nakagami amplitude distribution where probability density function can be stated as (Equation 14):

Equation 14
$$p_R(r, \mathrm{m}) = \frac{2\mathrm{m}^{\mathrm{m}} r^{2\mathrm{m}-1}}{\Gamma(\mathrm{m})} e^{-\mathrm{m}r^2},$$

where

Γ(m)	is Gamma function of m and
r	is amplitude.

With *m* = 1 the amplitude distribution equals the Rayleigh distribution. The greater the value of the m parameter the less deep fading and less amplitude fluctuation occurs around the mean value. When $m \rightarrow \infty$, tap coefficients become constant (Figure 217).

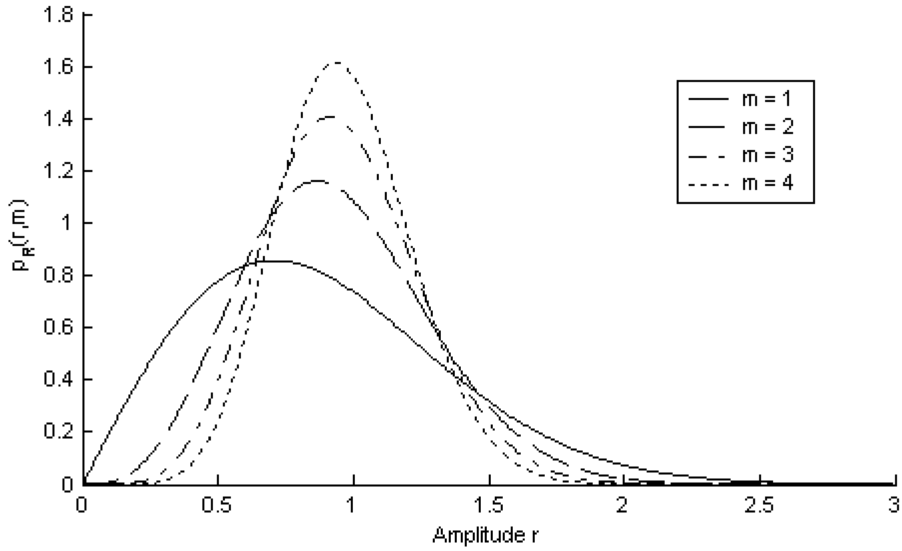


Figure 217. Nakagami distributions with *m* = 1 … 4, mean power = 1

The Doppler spectrum is generated in the vicinity of the pure Doppler shift due to the specular component. Figure 218 shows an example with m = 2 and α_0 = 15º.

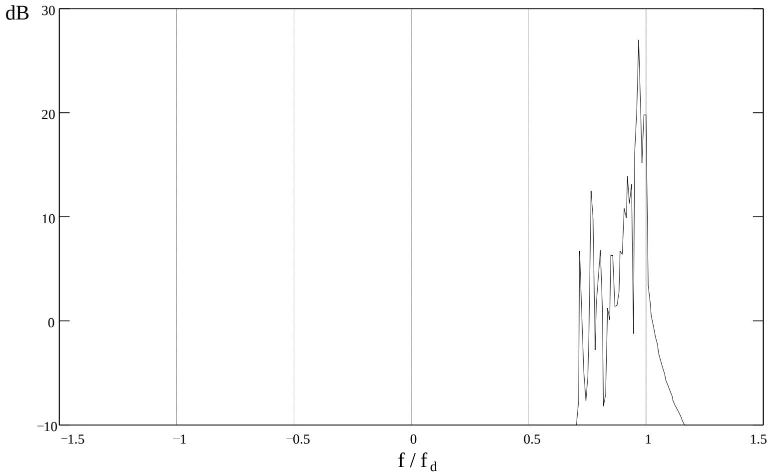


Figure 218. Nakagami Doppler spectrum

Parameters of the Nakagami model:

- Delay [ns]
- Strength [dB]
- Angle between mobile speed and specular radio wave [deg]
- m parameter
- Standard deviation of scatterer visual angle [deg]
- Vehicle speed [m/s]
- Carrier frequency [MHz]
- Phase shift [deg].

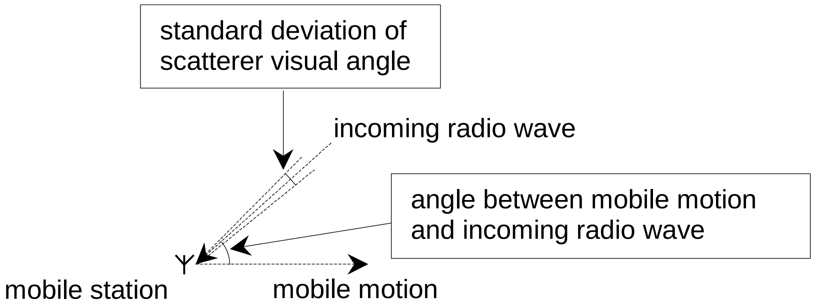


Figure 219. Nakagami parameters.

The scatterer visual angle is the same as the incident angle between the direction of the ray coming from a scatterer and the motion vector of the mobile.

12.9.1.7 Lognormal

Lognormal distribution describes the slow fading of the channel. It is a distribution of a random variable whose decibel value follows the normal distribution. The lognormal distribution is assumed to describe the slow fading.

The lognormal probability density function can be written as

Equation 15
$$p_{\log}(r) = \frac{1}{r\sigma\sqrt{2\pi}} \exp\left(-\frac{(\ln r - \mu)^2}{2\sigma^2}\right), \quad r > 0 .$$

Here σ is the standard deviation of the logarithm of the amplitudes r (the shape parameter) and μ is their mean value.

A lognormally distributed random process is formed by generating a zero mean normally distributed sequence of real valued samples. This sequence is filtered with the butterworth filter and resulting sequence $\{y_i\}$ is transformed to lognormal distribution with function

Equation 16
$$r_i = 10^{y_i/20} = e^{y_i \ln(10)/20},$$

To produce the desired slowly fading process, the sequence of independent lognormally distributed random numbers is filtered by a Butterworth filter. The relation between the filter cut-off frequency and the correlation length (L) is

Equation 17
$$f_s = \frac{v}{2L},$$

where v is the vehicle speed

Parameters of the lognormal model:

- Strength [dB]
- Standard deviation of the tap coefficients [dB]
- Correlation distance i.e. the distance between the origin of the first zero of the autocorrelation function (see Equation 17)
- Mobile speed [m/s]

12.9.1.8 Gaussian

The Gaussian spectrum models effect of confined scatterer clusters on far distance (see ref. [1]). Two scatterer clusters yield a Doppler spectrum that comprises of two normal distributions (beams). The center frequency and standard deviation for both distributions are parameters, as well as the power ratio of the two beams. The following figure (Figure 220) shows an example of Doppler spectrum for Gaussian fading model.

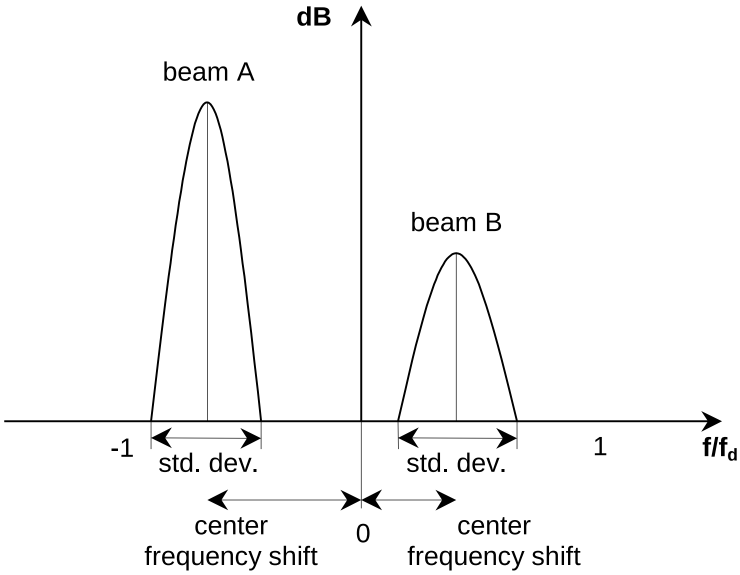


Figure 220. Gaussian Doppler spectrum

Parameters of the Gaussian model:

- Delay [ns]
- Strength [dB]
- Center frequency shift of beam A
- Frequency shift standard deviation of beam A
- Center frequency shift of beam B
- Frequency shift standard deviation of beam B
- Power ratio of beams A / B [dB]
- Vehicle speed [m/s]
- Carrier frequency [MHz]
- Phase shift [deg].

The center frequency shifts and frequency shift standard deviations are given as relative numbers with respect to the maximum Doppler shift f_d.

12.9.1.9 Suzuki

Suzuki distribution is generated by a combination of the lognormal distribution and the Rayleigh distribution. The lognormal distribution describes the slow fading and the Rayleigh distribution the fast fading of the electromagnetic field. They are combined so that the local rms-value of the fast fading signal is fading slowly according to the lognormal distribution. The Suzuki-fading process may be regarded as a product of the two processes.

A fundamental part in the forming of the Suzuki-distributed random process is the control of the time-behaviour of the respective partial processes. The tap coefficient process may be described by the formula:

Equation 18

$$z(t) = x(t) \cdot y(t)\,,$$

where $x(t)$ is a fast fading complex Gaussian process, with the Rayleigh-distributed amplitude, and $y(t)$ is a lognormally distributed slowly fading process. The partial processes are formed as described earlier (see the sections on Classical and Lognormal models).

Parameters of the Suzuki model:
- Delay [ns]
- Strength [dB]
- Sigma [dB], standard deviation of Lognormal process
- Correlation distance [m]
- Vehicle speed [m/s]
- Phase shift [deg].

12.9.2 Custom models

Custom models are fading processes obtained by combining available amplitude distributions to Doppler spectrums. Especially Rayleigh and Rice distributions can be combined to variety of Doppler spectra.

Constant & Pure Doppler
Same as standard model "Pure Doppler" see on page 327.

Rayleigh & Jakes
Same as standard model "Classical" see page 327.

Rayleigh & Gaussian
Same as standard model "Gaussian" see page 335.

Rayleigh & Flat
Same as standard model "Flat" see page 329.

Rayleigh & Butterworth

Butterworth filters are characterized by a magnitude response that is maximally flat in the passband and monotonic overall. The transfer function for a first degree Butterworth filter is given by

Equation 19

$$H(f) = \frac{1}{1 + j\left(2\pi fT\right)},$$

where T is the correlation length in time (s). Filter is designed in the time domain and the normalized (unity voltage gain) impulse response is

Equation 20

$$h(t) = \sqrt{\frac{2}{T}}e^{-t/T}u(t),$$

where $u(t)$ is unit step function.

Complex white noise with Gaussian distribution in both real and imaginary parts is filtered with a digital spectrum shaping filter (Butterworth) to obtain the desired Doppler spectrum.

Parameters of the Rayleigh & Butterworth model:
- Delay [ns]
- Strength [dB]
- Correlation distance [m]
- Vehicle speed [m/s]
- Phase shift [deg].

Rayleigh & Rounded

In wireless channels with fixed transmitter and receiver positions a very small Doppler spread is caused by movement of the environment. In such a case the power in Doppler power spectrum is concentrated in the center of the frequency range as in Figure 221. This spectrum is called rounded in [3] and is given as the following function:

Equation 21
$$S(f)=\begin{cases}1-1.72f_0^2+0.785f_0^4, & |f_0|\le 1\\ 0 & |f_0|>1\end{cases},$$

where $f_0=\dfrac{f}{f_d}$, f_d is the maximum Doppler shift.

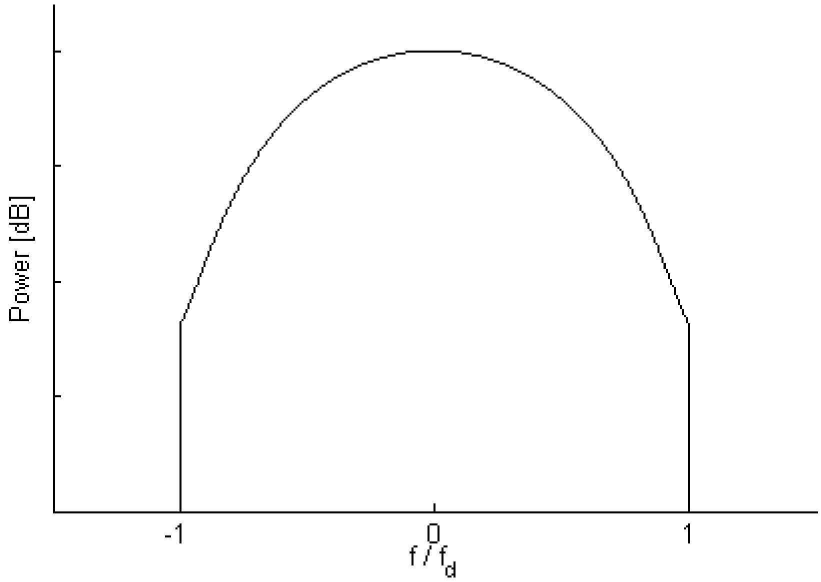


Figure 221. Rounded Doppler spectrum.

Rice & Jakes
Same as standard model “Rice” see on page 330.

Rice & Gaussian
Model is a combination of Rice and Gaussian models. Zero mean complex Gaussian number is filtered with Gaussian spectrum shaping filter. Addition to this model includes LOS component.

Rice & Flat
Model is same as standard model Rice, but spectrum shaping filter is flat.

Rice & Butterworth
Same as custom model Rayleigh & Butterworth, but includes LOS component.

Rice & Rounded
Same as custom model Rayleigh & Rounded, but includes LOS component.

12.9.3 Multichannel models

Channel Model view of EB Propsim can be used to create independent channel models for each channel. In general the channel realizations exhibit zero mutual correlation. Channel model view can be used also to apply correlated multichannel models to generate correlated channels. These can be used for example in testing of antenna array and diversity schemes. Also MIMO (Multiple-Input-Multiple-Output) models can be generated by using the Channel model view.

There are two different ways to create models for SIMO (Single-Input-Multiple-Output) and MISO (Multiple-Input-Single-Output) scenarios. In the first method the user defines the correlation coefficients between the channels by means of the correlation matrix. The other way is use geometric modeling of MIMO with only 1 element at transmitting or receiving array.

MIMO models can also be generated by two different methods. In the correlation matrix based method the user defines correlation matrices either separately for the both ends of the link or for the full MIMO setup. In the former case the spatial MIMO correlation matrix is then calculated as a Kronecker product of the spatial correlation matrix at MS and BS. In the geometry based method the user defines antenna configurations, arrival and departure nominal angles and angle spreads. Correlated MIMO channels are generated as a superposition of discrete rays with defined directions. Fading is caused by phase variation of rays.

12.9.3.1 SIMO and MISO models

12.9.3.1.1 Generation of correlated sequences

The Channel model view is used to set correlation coefficients for MIMO channels (Tx/Rx antenna pairs). The correlation can be defined either for complex path coefficients or for amplitudes of path coefficients. Unique correlation matrix is given for each path. In multichannel models all the channels have same delay power spectrum i.e. delays and average powers are same. There are means to convert the amplitude correlation matrix to another correlation matrix (matrix C) between the complex path coefficients [2]. In the process of creating correlated sequences a factorization M of the correlation matrix C is needed. Matrix C can be factorized by applying e.g. Cholesky decomposition. When the vector composed of the uncorrelated complex impulse responses is multiplied by matrix M, a new set of complex impulse responses are generated that exhibits the wanted mutual correlations [4].

The method based on correlation matrix is applicable to all the fading models with Rayleigh or Rician amplitude distribution or to non-fading constant amplitude model. There are however some constraints with non-Rayleigh fading models, e.g. high Ricean *K*-factor enables only higher correlation values and with non-fading models correlation may affect only the relative phases of impulse responses. Desired correlation can be either complex correlation or simple magnitude correlation. In the former one both phase and magnitude are correlated, in the latter one only magnitude.

The principle of the multichannel emulation is shown in Figure 222.

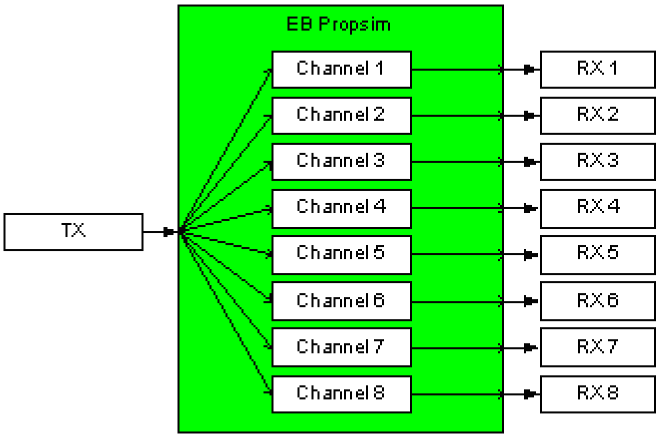


Figure 222. Principle of the multichannel emulation.

The signals coupled to the receivers in Figure 223 represent the signals obtained from the elements of an antenna array.

Let us denote the output signals representing the k[th] path of the emulator by

Equation 22 $$X^k(t) = \left[X_1^k(t), X_2^k(t), ..., X_N^k(t)\right]^T,$$

where N is the number of antenna elements in the array and $()^T$ means transpose operator. Equal average power is assumed for all output signals.

A correlation matrix C^k can be defined in the following way:

Equation 23 $$C^k = \frac{E\left\{X^k(t) \cdot \left[X^k(t)\right]^H\right\}}{P},$$

where $E(\ )$ is the expectation operation of x and $P = E\{\left|X_i^k(t)\right|^2\}$ is the power of the output signal of the emulator, i = 1,2,…,N. Transpose and complex conjugate of a vector or a matrix is denoted by $(\ )^H$. The definition of correlation matrix C yields to the following immediate conclusions

Equation 24 $$C_{ii} = 1 \text{ and } C_{ij} = C_{ji}^*,$$

where * denotes complex conjugate. Therefore, it is sufficient to set correlation values to the lower triangular matrix only.

The path coefficient sequences $\alpha^k(t) = \left[\alpha_1^k(t), \alpha_2^k(t), ..., \alpha_N^k(t)\right]^T$ exhibiting the desired correlation properties are generated from independent zero mean complex Gaussian distributed sequences $\beta^k(t) = \left[\beta_1^k(t), \beta_2^k(t), ..., \beta_N^k(t)\right]^T$ as

Equation 25 $$\alpha^k(t) = M \cdot \beta^k(t).$$

The matrix M is time invariant. Relation between matrix M and matrix C is

Equation 26 $$C = M \cdot M^H.$$

The eigenvalue decomposition can be used to determine M, as

Equation 27 $$M = U \cdot \Lambda^{1/2},$$

where Λ are eigenvalues of matrix C and U are the corresponding eigenvectors. The principle of correlation matrix method is shown in Figure 223.

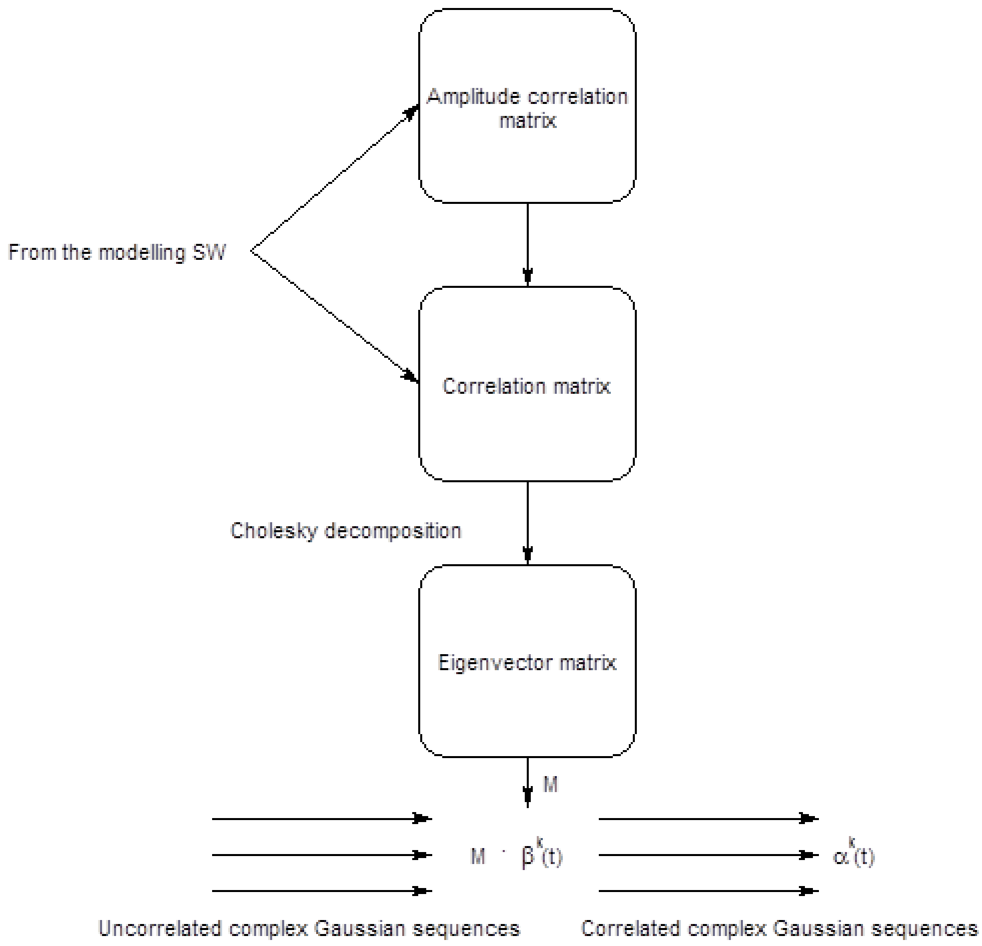


Figure 223. Principle of the generating correlated impulse responses.

12.9.3.2 MIMO models

12.9.3.2.1 Correlation matrix based method

MIMO channel model is generated by following the basic principle described in section 12.9.3.1.1. MIMO spatial correlation matrix can be composed of spatial correlation matrices at MS and BS applying the Kronecker product

Equation 28 $$C_{MIMO} = C_{MS} \otimes C_{BS}\ .$$

By applying eigenvalue decomposition the C_{MIMO} matrix can be written as

Equation 29 $$C_{MIMO} = MM^T\ ,$$

and MIMO channel coefficients can be calculated similarly to Equation 25. It is required that C_{MIMO} matrix is a positive definite matrix.

12.9.3.2.2 Geometry based method

Geometry based method can be applied to generate correlative channel coefficients for two dimensional MIMO channel on horizontal plane (azimuth, no elevation). Antenna array at MS receives signal transmitted by BS antenna array and scattered from *S* scatterers. MS is in motion with velocity $\bar{v}$. Directions of arrival (DoA) and direction of departure (DoD) have same distributions with certain variances and average directions. Directions of scatterers seen by BS and MS are independent on each others. Far field condition is assumed, thus DoA and DoD does not vary from element to element.

Geometry of the model is presented in Figure 224. There angles θ refer to BS and angles α to MS. All the angles are given with respect to normal of corresponding antenna array broad side. Definitions of the symbols are

θ_i is the DoD to i-th scatterer
θ_{av} is the average DoD to the scatterer cluster
θ_{LOS} is the line of sight direction from BS to MS
α_i is the DoA from ith scatterer
α_{av} is the average DoA from the scatterer cluster
α_{LOS} is the line of sight direction from MS to BS
$\alpha_{\bar{v}}$ is the direction of MS motion

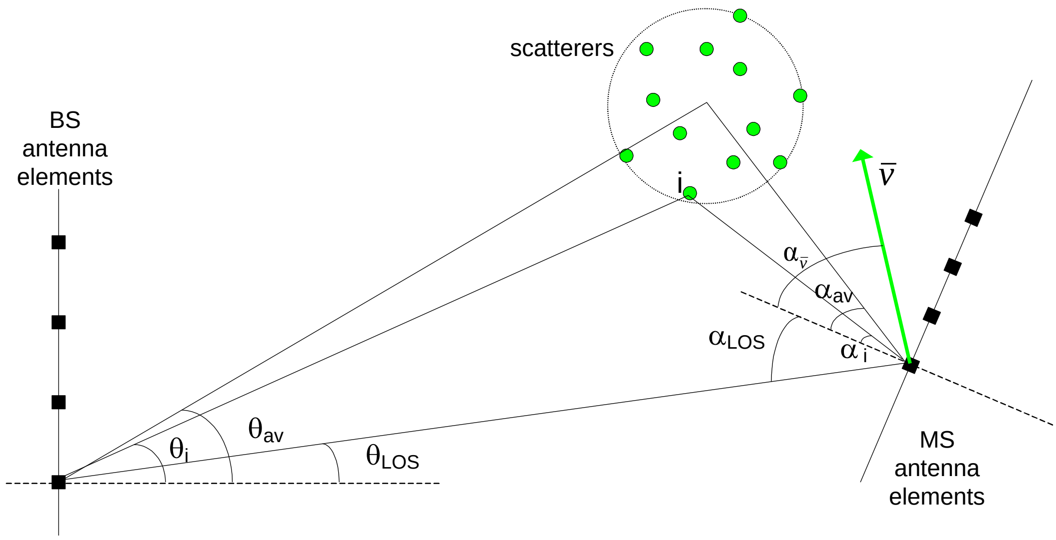


Figure 224. Model geometry.

All the scatterers have equal power *P/S* and random initial phase φ_i. Random initial phase is needed to model random time of travel (distance) from each scatterer to antenna array. Antenna elements have some gain pattern $G(\theta)$. The complex channel gain from *m*th BS antenna element to *n*th MS antenna element on time instant *t* is a superposition of phase rotating rays:

Equation 30

$$g_m(t) = \sqrt{P/S}\sum_{i=1}^{S}\begin{pmatrix} G_m(\theta_i)\exp\left(jkd_m \sin\theta_i\right)\cdot \\ G_n(\alpha_i)\exp\left(jkd_n \sin\alpha_i\right)\cdot \\ \exp\left(j(\varphi_i + kvt\cos\alpha_i)\right) \end{pmatrix},$$

where k = 2π/λ is the wave number (λ=carrier wave length), d_m is the distance between antenna element *m*th BS antenna element and reference element in meters and d_n is the distance between antenna element nth MS antenna element and reference element.

Scatterer cluster in Figure 224 generates a single resolvable multipath (delay, direction and tap coefficient) for the channel.

12.10 References

[1] CoDiT, *"Final Propagation Model",* UMTS CODE DIVISION TESTBED, R2020/TDE/DS/P/040b1, 1994.

[2] B. Natarajan, C.R. Nassar and V. Chandrasekhar, *"Generation of Correlated Rayleigh Fading Envelopes for Spread Spectrum Applications"*, IEEE Communications Letters, Vol. 4, No. 1, January 2000.

[3] IEEE 802.16 Broadband Wireless Access Working Group, *"Channel Models for Fixed Wireless Applications"*, 17.07.2001. <http://iee802.org/16>

[4] Stephenne A., Champagne B.; "Effective Multi-Path Vector Channel Simulator for Antenna Array Systems"; IEEE Transactions on Vehicular Technology; vol. 49; No. 6; Nov. 2000; pp. 2370-2381.

www.elektrobit.com