UNIVERSAL RADIO EMULATOR FOR SDR

Datasheet and User Manual

Creative Engineering

Contents

Scope of the Document

This document describes the principals of operation of the Universal Radio Emulator Software and gives a detailed user manual for users at all level.

Overview of Channel Emulator Operation

The Universal Radio Emulator for Software Defined Radio reproduces the changes to an actual RF signal according to the various radio wave propagation models using the different types of Software Defined Radio. It emulates a real-time radio channel present in wireless communications.

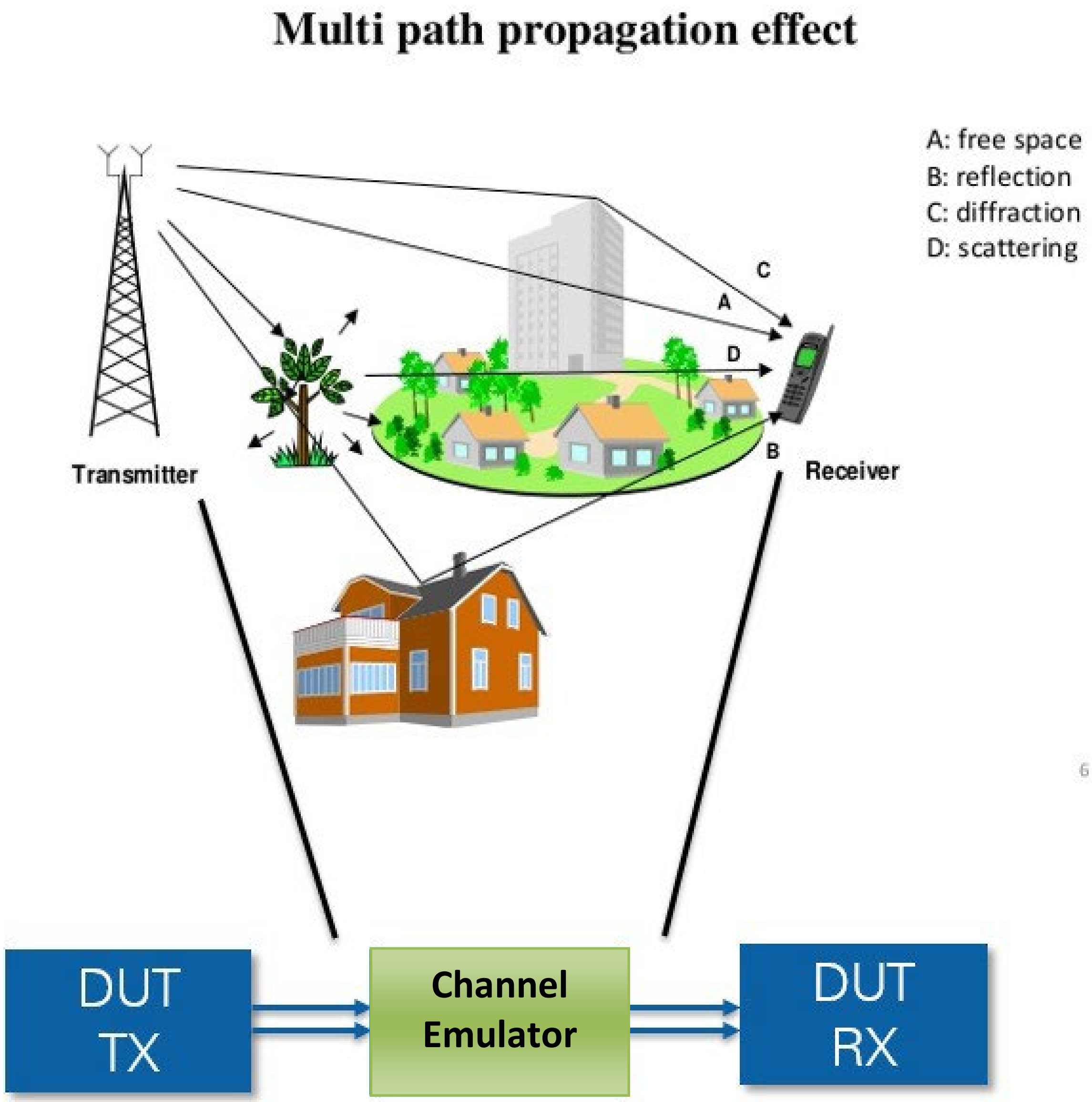

Figure 1: Radio Emulator Functionality

Main Features

- Supported hardware: NI VST, USRP RIO series SDR
- Enabling 18 paths for SISO operation.
- Enabling Different channel models for each path.
- Channel Models: Deterministic, Pure Doppler, Rice, Rayleigh
- Frequency Ratio settings to simulate different speeds for paths
- Advanced operation and correlation matrices.
- Support for Complex antenna correlation
- FPGA based AWGN support
- Power normalization of all paths
- Scalable Host code through configurable fader parameters
- Save and Load channel, this enables storing channel profiles compatible with 3GPP and other standards.
- Industry standard user interface

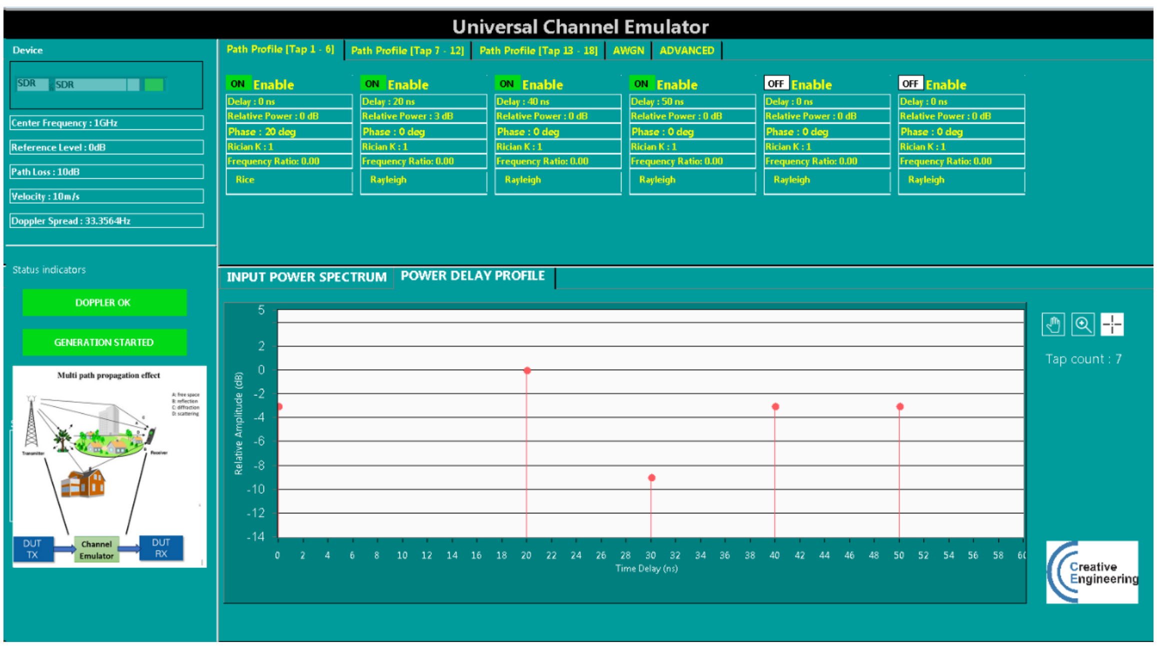

Figure 2: Industry standard user interface

Hardware and Software Requirements

HW Requirements:

- Intel Core I5 or higher-grade Processor
- NI PXIe-5646R, USRP RIO x410,

SW Requirements (compatibility):

- LabVIEW, LabVIEW FPGA
- GNU Radio, UHD (open source), RF NOC

System level specifications

Parameter	Value	Comments
Configurations Supported	1x1	MIMI Configuration is possible in further versions
Supported fading models	Deterministic, Pure Doppler, Rayleigh, Rice	Other models may be added but host side calculation time is a key issue.
Maximum RF Bandwidth	80/160/320 MHz	SDR HW Dependents
Theoretical Maximum relative power dynamics	84 dB	Theoretical maximum power difference in the path profile. Due to technical reasons 14 bits are used.
Taped-delay-line sampling rate	100/200/400 MSPS	HW dependent.
Delay resolution	10 ns	This is currently implemented as a sample-based tapped-delay-line. Fractional Delay filters would not fit the FPGA, better resolution is not possible.
Minimum In-Out delay	Min 8.9 us	Delay measured between RF in and RF Out, Settings: 1 deterministic path; 0 ns.
Maximum Doppler shift	4.65 kHz	Limited by the interpolation filter. (1dB point 4.65 kHz)
Fading channel interpolation rate	8192/16384	This is fixed. It may be modified later to 4096 to enable 9.3 kHz Doppler shift, but it results in a higher load for the Host CPU. Same number of paths with the same quality will not be possible.
Fading channel sampling rate	12.207 kHz	This is the rate at which the fading samples are generated on the host
Maximum no. of taps per channel	18	Host fader generation is the bottleneck
Maximum delay	20.6 µs	
Correlation modes	Vector, Correlation Matrix, Kroenecker	See details in user manual.

Figure 4: System level specifications

System Architecture

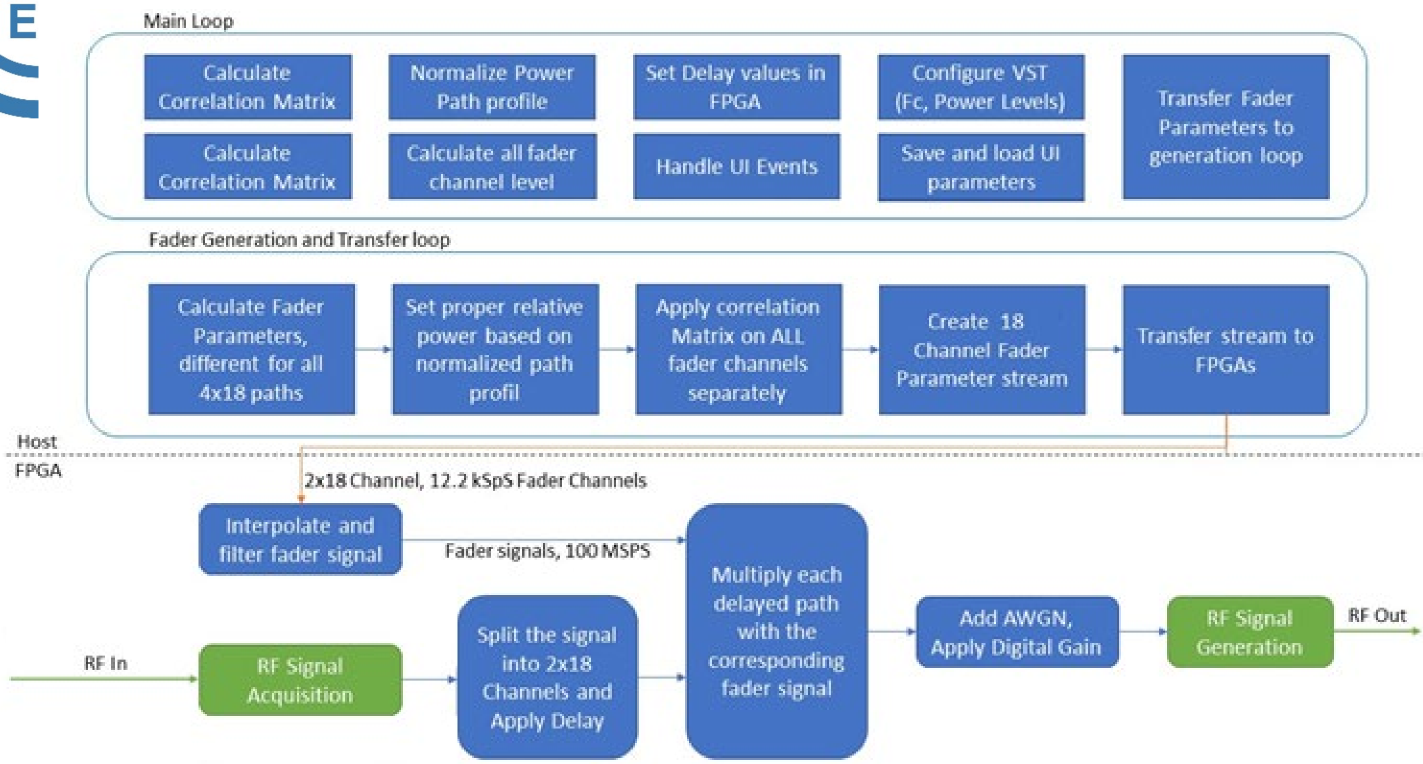

Figure 5: Software architecture

Host Part

The main goal of the host part is to generate the fader signals for the FPGAs, separately for each path. In every channel the software enables 18 paths.

Path profile

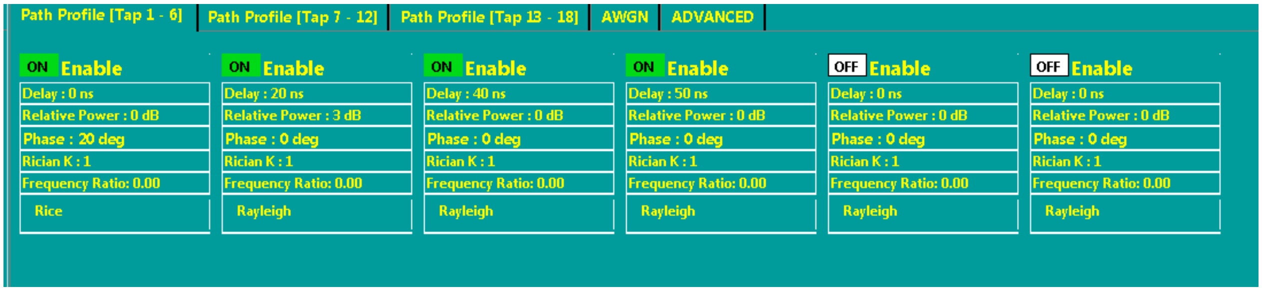

Figure 7: Path profile settings

The UI enables setting different path parameters to all 18 paths. User has the option to choose from the following options:

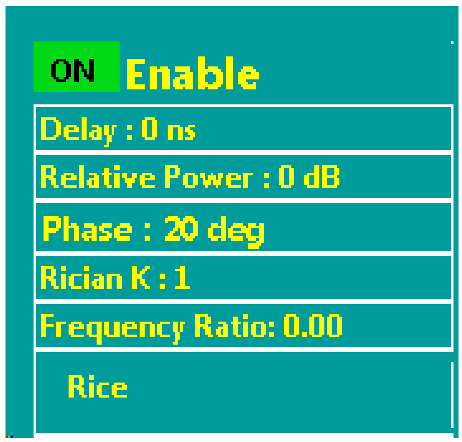

Figure 8: Settings for a given path

- ***Enable***: Enables the selected path.
- ***Relative Power***: The power of all paths is set relative to each other. The path with the largest power settings will be normalized to 0 dB. Absolute power is set using the path loss and reference level.
- ***Phase***: A constant phase can be set for each tap. This is also a relative setting as it is not possible to control the absolute phase from the input to the output through all RF components.
- ***Rician K***: Rice distribution specific K parameter. Ignored with other distributions.
- ***Frequency Ratio***: Used in Pure Doppler and Rician case to determine the actual Doppler shift of the direct path.

  Actual shift = Frequency Ratio * Doppler Spread

  This parameter is ignored in Rayleigh or Deterministic mode.
- ***Select path profile***: Possible options are deterministic, Pure Doppler, Rayleigh, Rician.

The selected power path profile is visualized on the tab below the settings.

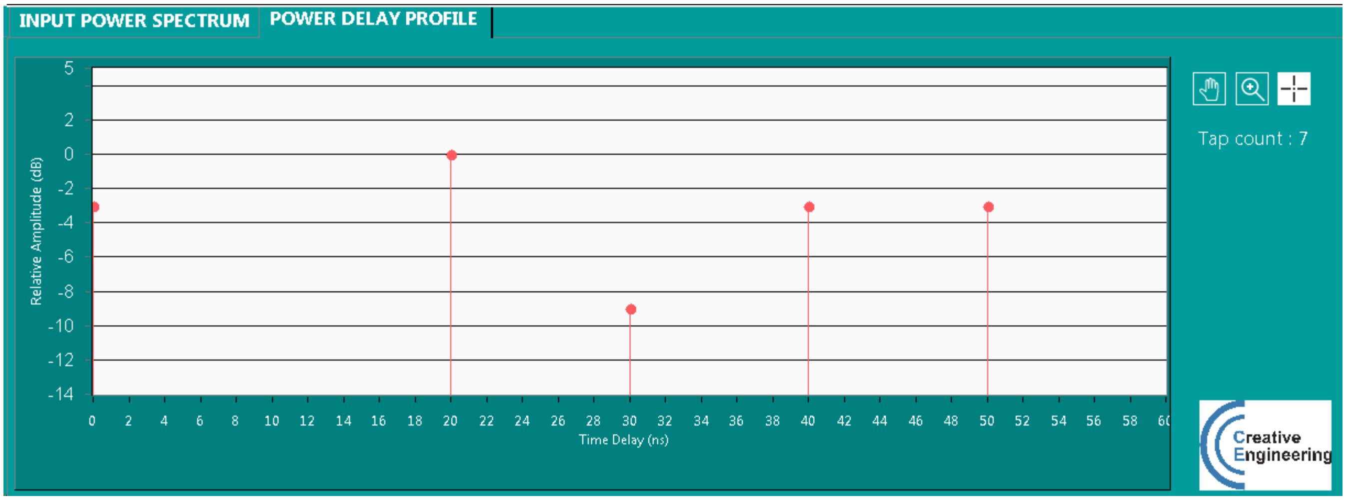

Figure 9: Normalized Power Path Profile displayed

Basic RF and instrument parameters

The application is planned to be run in LabVIEW or Open Source, not in standalone mode, compiled into an executable. The application wanted to focus on the Channel Emulator functionality.

This is also why user must set resources **before** running the application.

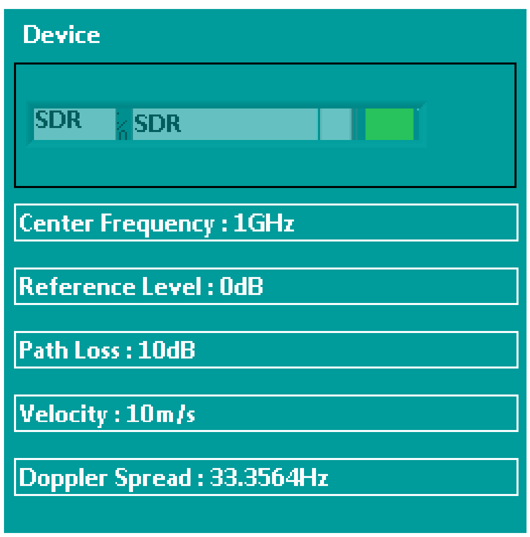

Figure 10: Basic parameters

- ***Green rectangles:*** enable a given SDR.
- ***Reference level:*** sets analyzer input level
- ***Path loss:*** determines the attenuation between the input and output, power is normalized so the number of paths enabled does not change the attenuation.
    - Based on the reference level the path loss may not be possible to achieve due to HW limitations. In this case we coerce to the max power and red indicator turns on.
- ***Velocity***: user control, sets the speed of the receiver.
    - Positive speeds: RX is moving away
    - Negative speeds: RX is approaching

    In case user sets a Velocity value, the Doppler Spread is also updated automatically.
- ***Doppler Spread:*** Frequency change based on the center frequency and the velocity.

    In case user sets the Doppler spread, the velocity gets updated automatically.

Maximum doppler Spread.

The Doppler Spread is limited by the host side fader parameter sampling rate. In the example it is 100MHz/8192 = 12.2 kHz. This would limit the maximum doppler frequency to 6.1 kHz, but additional bandwidth is needed for the interpolation filters.

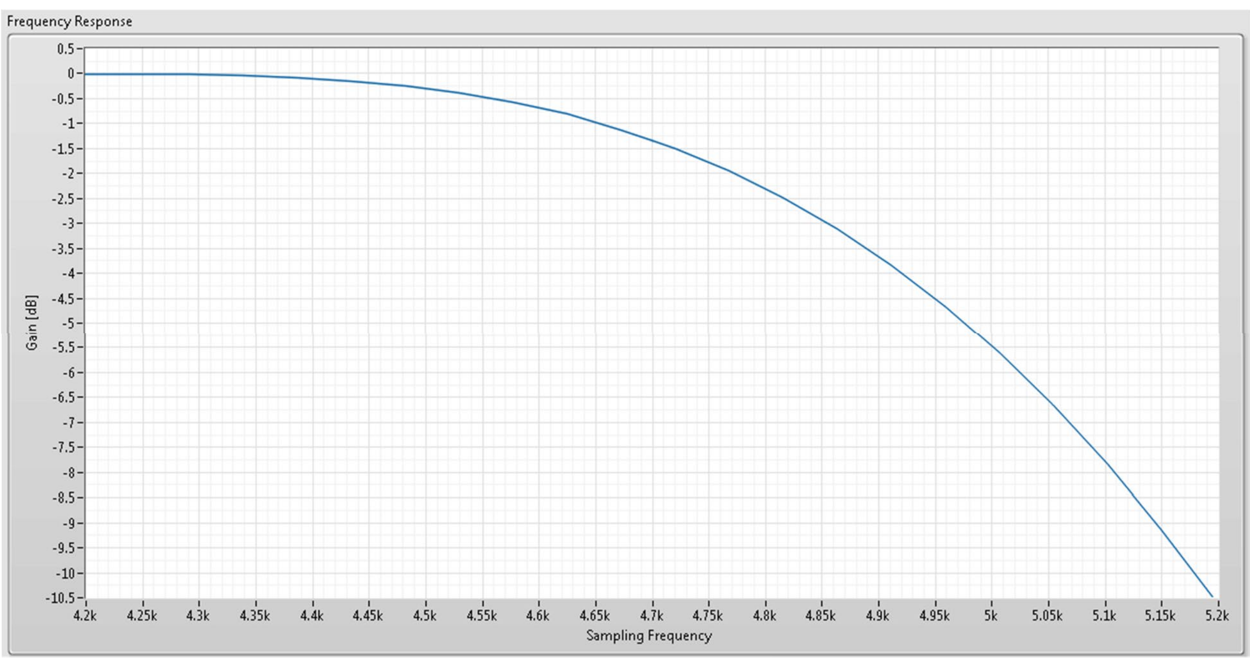

Figure 11: Interpolator filter characteristics, 4.65 kHz 1dB point

The maximum Doppler frequency is 4.65 kHz.

Status indicators

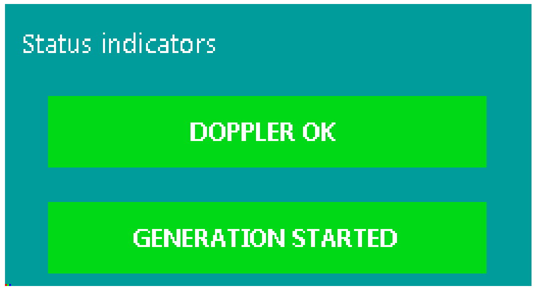

Figure 12: Status indicator panel

- ***Doppler OK / Doppler Coerced***: In case the Doppler Frequency setting is higher than 4.65 kHz this indicator signal coerced value.

  Generation Started/Stopped: when the SW is started, some time is needed to initiate generation. This indicator marks the start of generation.

Input power spectrum graph

For debug purposes the user may peak into the input spectra of each SDR.

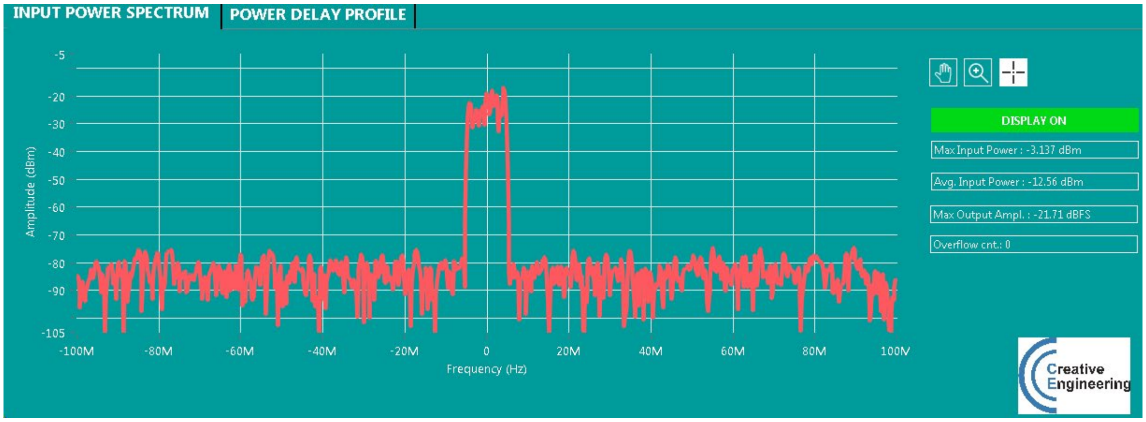

Figure 13: Input power spectrum graph

The maximum and average power is calculated on the FPGA.

The maximum output amplitude is displayed in dBFs in order to see if there is headroom for further digital gain or noise.

The display can be turned off in order to save CPU resource for fader parameter generation.

AWGN settings

AWGN is inserted on the FPGA at the end of the signal processing path. Parameter generation is controlled from the host.

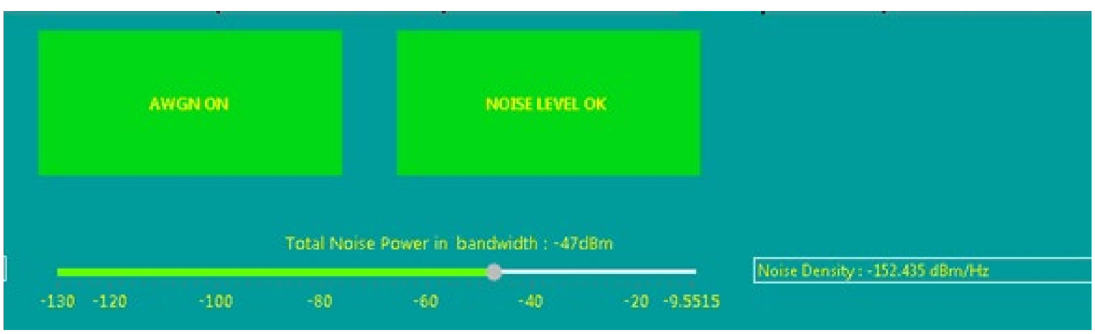

Figure 14: AWGN panel

- ***AWGN ON/OFF:*** Turning on and off the noise injection.
- ***Reached max. Noise /Noise Level OK:*** Host IP determines if the given noise level is possible to reach. This indicator shows if it is above the theoretical limit.
- ***Noise Power Level:*** The slider sets the total noise power in the entire bandwidth of the VST.
- ***Noise Density:*** Noise Density in the same bandwidth

Vector mode

In a simple case there is no correlation among antennas. In this mode user can set the complex gains on all Hxy paths above separately.

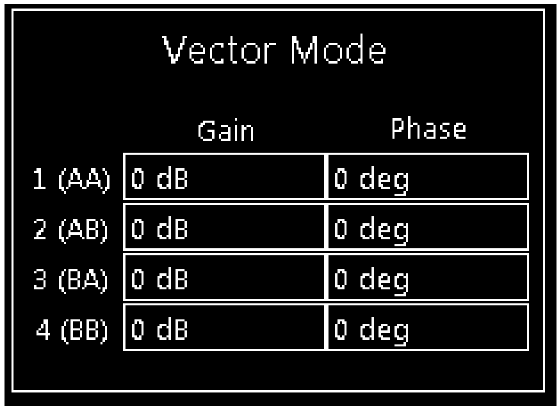

Figure 16: Vector mode

Correlation Matrix mode

As antennas are correlated at both ends all Hxy paths are correlated to each other. A 4x4 correlation matrix describes all possible interferences. In this mode user has the most control over MIMO propagation as each matrix element can be changed independently.

Correlation Matrix Mode

	AA Magn	Phase	AB Magn	Phase	BA Magn	Phase	BB Magn	Phase
1 (AA)	1 (lin)	0 deg	0 (lin)	0 deg	0 (lin)	0 deg	0 (lin)	0 deg
2 (AB)			1 (lin)	0 deg	0 (lin)	0 deg	0 (lin)	0 deg
3 (BA)					1 (lin)	0 deg	0 (lin)	0 deg
4 (BB)							1 (lin)	0 deg

Figure 17: Correlation matrix mode

The diagonal represents the same factors as described in Vector mode. The matrix is symmetric, so the lower triangle is calculated automatically.

Kronecker mode

Kronecker mode assumes that the overall correlation is built from the Hxy paths that exists independently with unit multiplier and a fixed complex correlation exists on both RX and TX ends.

User has to enter TX and RX correlation and the correlation matrix is calculated automatically using Kronecker product.

$$R_l = R_{\mathrm{Tx}}^{(l)} \otimes R_{\mathrm{Rx}}^{(l)}, \text{ where } R_{\mathrm{Tx}}^{(l)} = \begin{bmatrix} 1 & \rho_{\mathrm{Tx}}^{(l)} \\ \rho_{\mathrm{Tx}}^{(l)*} & 1 \end{bmatrix} \text{ and } \mathrm{R}_{\mathrm{Rx}}^{(l)} = \begin{bmatrix} 1 & \rho_{\mathrm{Rx}}^{(l)} \\ \rho_{\mathrm{Rx}}^{(l)*} & 1 \end{bmatrix}$$

Figure 18: Kronecker product calculation formula to get correlation matrix

R_{Tx} and R_{Rx} denotes the TX and RX antenna correlation respectively.

Kronecker Mode

TX Correlation	
Magnitude	Phase
0 (lin)	0 deg

RX Correlation	
Magnitude	Phase
0 (lin)	0 deg

Figure 19: Kronecker mode settings on UI

***Important**: Only one of the three mode can be selected using the radio button. Data displayed in the other fields are ignored.*

Per path correlation

Each path may have a different angle of arrival. Proper emulation of such a case in MIMO mode requires handling antenna correlation separately for all path.

UI supports this by having an array for MIMO correlation settings. Each element corresponds to a path. The red circle notes that the correlation panel is an array, each element represents the correlation for the given path.

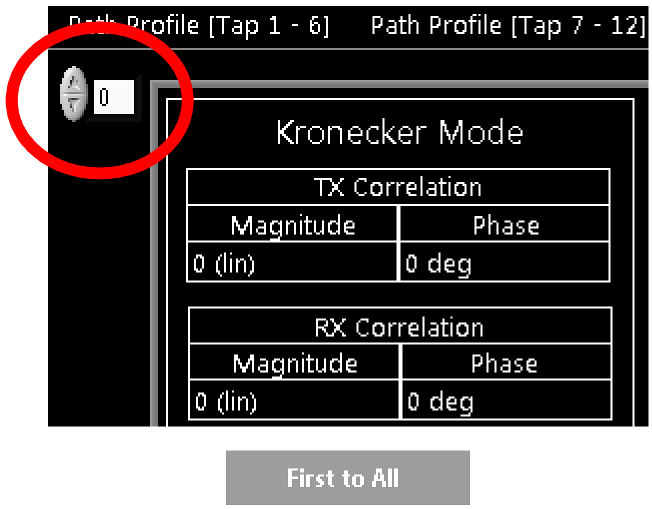

Figure 20: per path correlation settings and first to all button

First to All/Full Matrix: In case this feature is not needed the first element of the array is used for all paths.

Advanced settings

An extra tab is separated for debug indicators and advanced parameter settings.

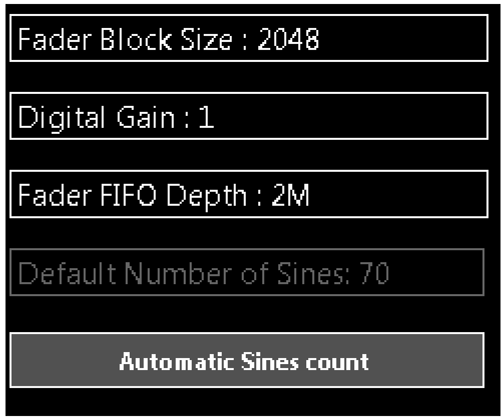

Figure 21: FIFO related settings

- ***__Fader block size:__*** number of fader samples generated at each iteration. Larger block size requires more memory and larger FIFO size. In return it is easier to distribute large matrix operations to multi core. 2k samples are recommended for PXIe – 8135.
- ***__Digital Gain:__*** Simple digital gain control that take effect on the FPGA at the end of the signal processing part. It can be used to fine tune attenuation. *WARNING: this value is not used in power normalization!*
- ***__FIFO depth:__*** Host to FPGA FIFO size. Larger FIFO may enable larger blocks and buffers for temporary change of CPU load, but it also makes flushing slower in case of parameter change.
- ***__FIFO space left:__*** Monitors the host to FPGA FIFO status. It helps tuning the no. of Sines per path for the given controller.

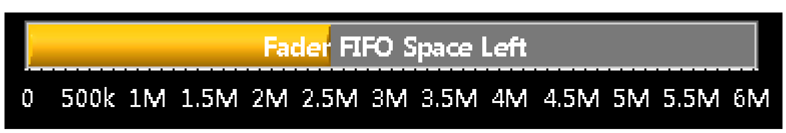

Figure 22: Indicator of the current FIFO status for each VST

Advanced settings – Sine count for Rayleigh generation

Rayleigh fader parameter generation is done based on Jakes model. For a simple but more detailed description visit the following Wiki page:

https://en.wikipedia.org/wiki/Rayleigh_fading#Jakes's_model

As described in the article, the bandwidth limited Gaussian noise is approximated by multiple, random phased sine waves approaching a moving receiver.

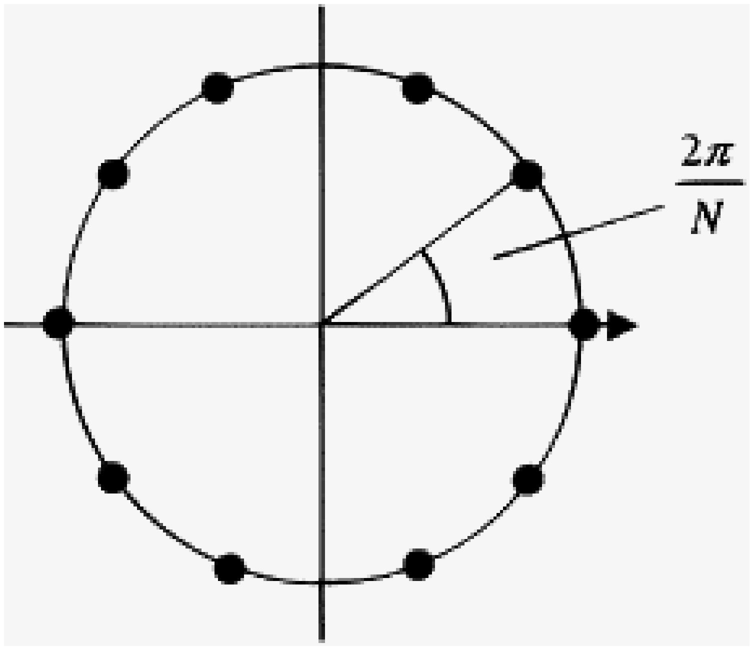

Figure 23: Multiple directions in Jakes model

In order to get a result closer to a BW limited Gaussian noise, more sine waves have to be added around the unit circle. The computation power increases with this number. In a maximum 18 channel, 2x2 MIMO case 72 uncorrelated fading profile has to be calculated.

To scale the code with the number of paths enables the software automatically selects the number of sines that the controller can calculate real time.

In case the user runs it on a different controller, the following advanced parameter setting enables manual sine count generally or per path level to properly scale the fader channels.

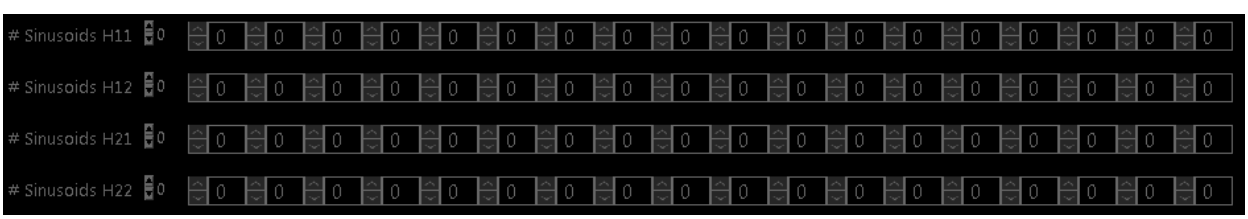

Figure 24: Manual sine count per channel per path.

More detailed description of the controls:

- ***Automatic Sines Count / Manual Sines:*** number of sines are determined automatically based on the number of enabled path and MIMO mode. This is optimized for PXIe-8135 controller running no other applications.
- ***Default no. of Sines:*** Enabled if manual mode is selected. It sets the default number of sines used for all paths.
- ***Sinusoids Hxy:*** Default number of sines may be overridden for each MIMO direction and tap. If no items in the array is active then default is used. Active array element overrides default settings for the given path. Use this mode if automatic mode returns FIFO underflow error. This way less significant paths may have less precision to reduce calculation time.