Agilent Technologies

Advanced Design System 2011.01

Feburary 2011
Ultra-Wideband DesignGuide

Acknowledgments

Mentor Graphics is a trademark of Mentor Graphics Corporation in the U.S. and other countries. Mentor products and processes are registered trademarks of Mentor Graphics Corporation. * Calibre is a trademark of Mentor Graphics Corporation in the US and other countries. "Microsoft®, Windows®, MS Windows®, Windows NT®, Windows 2000® and Windows Internet Explorer® are U.S. registered trademarks of Microsoft Corporation. Pentium® is a U.S. registered trademark of Intel Corporation. PostScript® and Acrobat® are trademarks of Adobe Systems Incorporated. UNIX® is a registered trademark of the Open Group. Oracle and Java and registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners. SystemC® is a registered trademark of Open SystemC Initiative, Inc. in the United States and other countries and is used with permission. MATLAB® is a U.S. registered trademark of The Math Works, Inc.. HiSIM2 source code, and all copyrights, trade secrets or other intellectual property rights in and to the source code in its entirety, is owned by Hiroshima University and STARC. FLEXlm is a trademark of Globetrotter Software, Incorporated. Layout Boolean Engine by Klaas Holwerda, v1.7 http://www.xs4all.nl/~kholwerd/bool.html . FreeType Project, Copyright (c) 1996-1999 by David Turner, Robert Wilhelm, and Werner Lemberg. QuestAgent search engine (c) 2000-2002, JObjects. Motif is a trademark of the Open Software Foundation. Netscape is a trademark of Netscape Communications Corporation. Netscape Portable Runtime (NSPR), Copyright (c) 1998-2003 The Mozilla Organization. A copy of the Mozilla Public License is at http://www.mozilla.org/MPL/ . FFTW, The Fastest Fourier Transform in the West, Copyright (c) 1997-1999 Massachusetts Institute of Technology. All rights reserved.

The following third-party libraries are used by the NlogN Momentum solver:

"This program includes Metis 4.0, Copyright © 1998, Regents of the University of Minnesota", http://www.cs.umn.edu/~metis , METIS was written by George Karypis (karypis@cs.umn.edu).

Intel@ Math Kernel Library, http://www.intel.com/software/products/mkl

SuperLU_MT version 2.0 - Copyright © 2003, The Regents of the University of California, through Lawrence Berkeley National Laboratory (subject to receipt of any required approvals from U.S. Dept. of Energy). All rights reserved. SuperLU Disclaimer: THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS

Errata The ADS product may contain references to "HP" or "HPEESOF" such as in file names and directory names. The business entity formerly known as "HP EEsof" is now part of Agilent Technologies and is known as "Agilent EEsof". To avoid broken functionality and to maintain backward compatibility for our customers, we did not change all the names and labels that contain "HP" or "HPEESOF" references.

Warranty The material contained in this document is provided "as is", and is subject to being changed, without notice, in future editions. Further, to the maximum extent permitted by applicable law, Agilent disclaims all warranties, either express or implied, with regard to this documentation and any information contained herein, including but not limited to the implied warranties of merchantability and fitness for a particular purpose. Agilent shall not be liable for errors or for incidental or consequential damages in connection with the furnishing, use, or performance of this document or of any information contained herein. Should Agilent and the user have a separate written agreement with warranty terms covering the material in this document that conflict with these terms, the warranty terms in the separate agreement shall control.

Technology Licenses The hardware and/or software described in this document are furnished under a license and may be used or copied only in accordance with the terms of such license. Portions of this product include the SystemC software licensed under Open Source terms, which are available for download at http://systemc.org/ . This software is redistributed by Agilent. The Contributors of the SystemC software provide this software "as is" and offer no warranty of any kind, express or implied, including without limitation warranties or conditions or title and non-infringement, and implied warranties or conditions merchantability and fitness for a particular purpose. Contributors shall not be liable for any damages of any kind including without limitation direct, indirect, special, incidental and consequential damages, such as lost profits. Any provisions that differ from this disclaimer are offered by Agilent only.

Restricted Rights Legend U.S. Government Restricted Rights. Software and technical data rights granted to the federal government include only those rights customarily provided to end user customers. Agilent provides this customary commercial license in Software and technical data pursuant to FAR 12.211 (Technical Data) and 12.212 (Computer Software) and, for the Department of Defense, DFARS 252.227-7015 (Technical Data - Commercial Items) and DFARS 227.7202-3 (Rights in Commercial Computer Software or Computer Software Documentation).

About Ultra-Wideband DesignGuide

The Ultra-Wideband DesignGuide test benches provide rapid setup, analysis, and simulation results to verify the most common performance characteristics of UWB transmitters and receivers. Bi-phase and pulse-position UWB modulation and formats are supported.

Simulation results provide information for spectral characteristics and basic bit-error-rate versus signal-to-noise ratios, as well as environmental effects such as multi-path and propagation loss, allowing the BER to be determined as a function of range. Simulations also evaluate rake receiver performance in a multi-path environment and synchronization of a UWB correlator. Effects of narrowband and wideband interference on UWB system performance can also be evaluated.

The *UWB DesignGuide* requires the ADS Ptolemy (signal processing simulation) and dg_ultrawideband licenses in addition to the ADS design environment license. The Analog/RF linear simulator can be used to generate S-parameter files for antennas, but is not required. S-parameter files can be obtained via network analyzer measurements or other sources.

Features and contents of the *UWB DesignGuide* are accessible from the **DesignGuide** menu in an ADS Schematic window. Selecting a test bench copies a schematic into the current Workspace and opens a Data Display window.

Hints regarding this DesignGuide
Information about items in a Data Display window that you would want to modify is outlined in red.

Equations that you typically do not need to modify are often included in a separate Equations page.

After selecting an item that opens Schematic and Data Display windows, if you re-name the schematic and run a simulation, to display your latest simulation results open the data display file that corresponded to the original schematic, change the default dataset name (typically the same as the new name of your schematic).

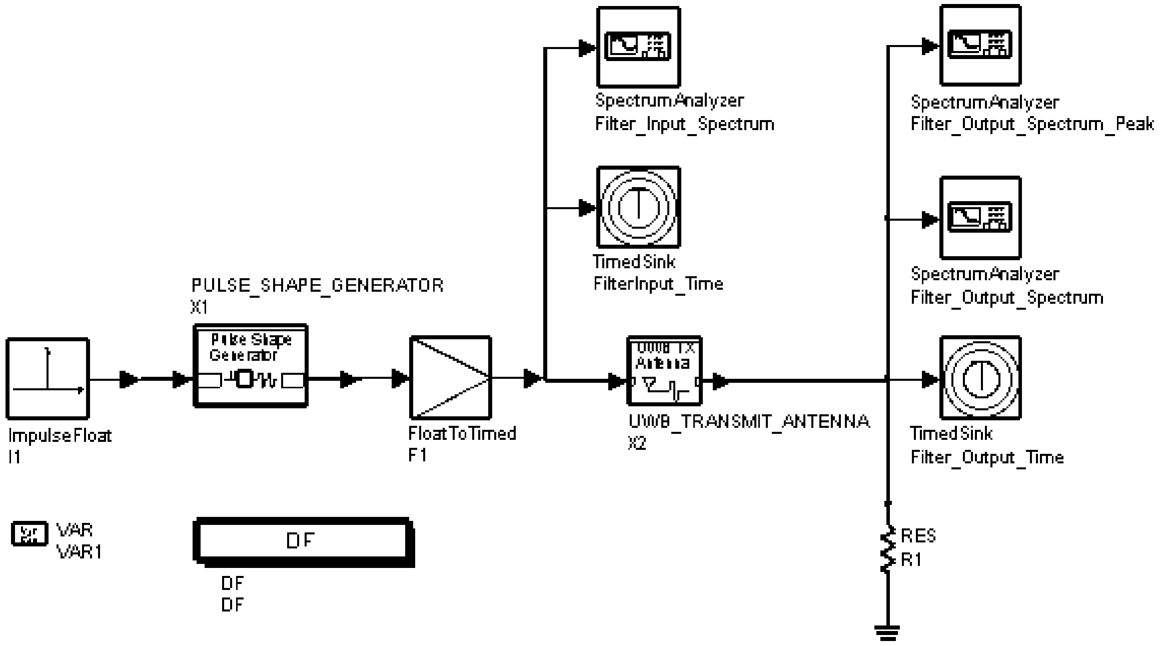
Topic Sequence

After selecting an item that opens Schematic and Data Display windows, if you re-name the schematic and run a simulation, to display your latest simulation results open the data display file that corresponded to the original schematic, change the default dataset name (typically the same as the new name of your schematic).

Pulse Mode Test Benches

Spectrum

The distribution of energy in the UWB spectrum is primarily determined by the spectrum corresponding to the shape of the individual pulses. The Spectrum Test Bench provides simulations of the spectrum of a single UWB pulse; the schematic for this test bench is shown in Spectrum Test Bench (_UWB_Pulse_Spectrum).



Spectrum Test Bench

(_UWB_Pulse_Spectrum)
Standard pulse shapes, as well as amplitudes and widths, can be selected; for details, refer to *PULSE_SHAPE_GENERATOR.* (dguwb)

The transmit filter represents the effects of transmitter front-end and antenna; for details, refer to *UWB_TRANSMIT_ANTENNA.* (dguwb)

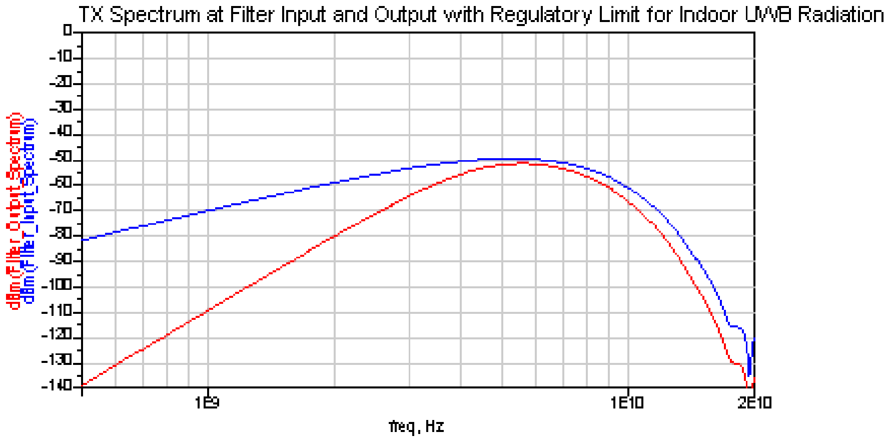
The SpectrumAnalyzer *Filter_Output_Spectrum* component is configured to have a resolution bandwidth of 1 MHz; *Filter_Output_Spectrum_Peak* is configured to have a resolution bandwidth of 50 MHz. The 50 MHz resolution bandwidth single pulse spectrum shows whether a UWB transmitter with a pulse rate less than 50 MHz complies with FCC regulations for peak radiated power.

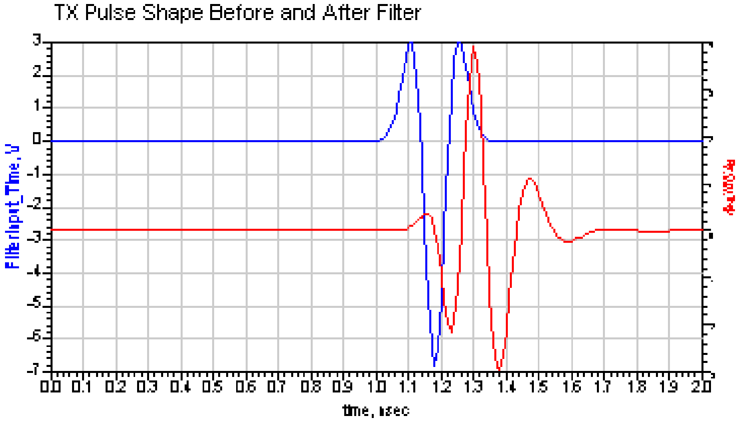
Spectrum Test Bench Design Parameters

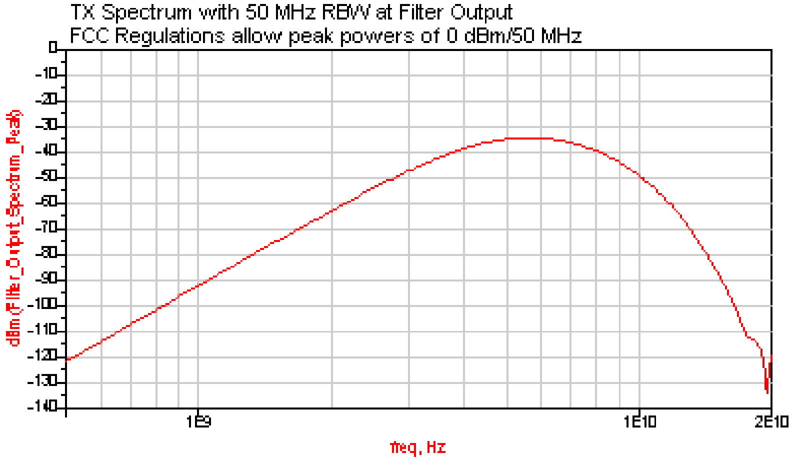
Name	Description
TStep	Time step of the simulation. It should be approximately PulseWidth/10.
PulseEnergy_joule	Total energy of the single pulse in Joules. For the doublet pulse shape, this is the energy of each individual monopulse making up the doublet. Total doublet energy is twice this value.
PulseWidth	Width of output pulse.
DoubletSeparation	Time between the positive and negative peaks of the waveform when the GAUSSIAN_DOUBLET_UWB_TRANSMITTER subnetwork of PULSE_SHAPE_GENERATOR is active.

Simulation Results

The Data Display window shows the spectrum before and after a transmit filter is applied; pulse shapes before and after filtering are also shown. The transmit spectrum with 50 MHz resolution bandwidth shows compliance with FCC regulatory limits for peak indoor UWB radiation.

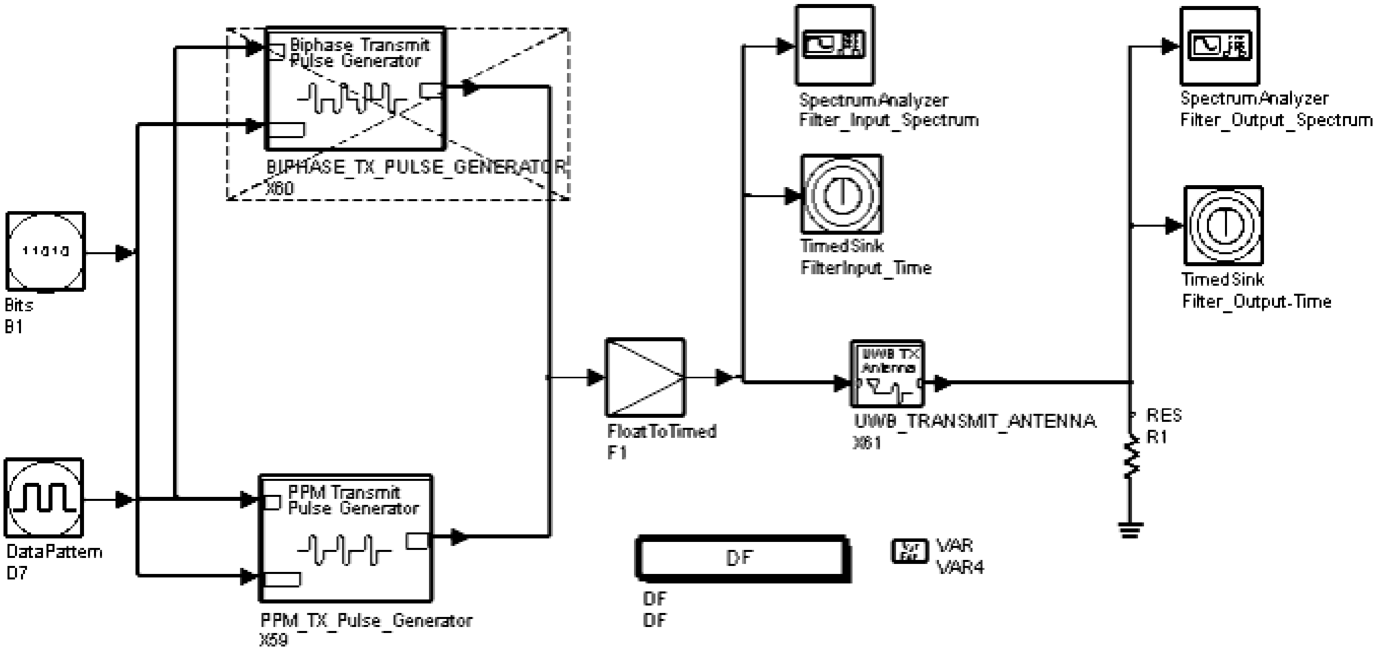

TX Spectrum at Filter Input and Output with Regulatory Limit for Indoor UWB Radiation


TX Pulse Shape Before and After Filter


TX Spectrum with 50 MHz RBW at Filter Output
FCC Regulations allow peak powers of 0 dBm/50 MHz

Modulated Transmit Spectrum

The shape of the spectrum is primarily determined by the spectrum corresponding to the shape of the UWB pulses. The distribution of energy into narrow bandwidth spectral lines is determined by the transmitter pulse rate and modulation. The Modulated Transmit Spectrum Test Bench simulates the spectrum of a basic UWB signal with pulse position or bi-phase modulation; the schematic for this test bench is shown in Modulated Transmit Spectrum Test Bench (_UWB_Modulated_Transmit_Spectrum).



Modulated Transmit Spectrum Test Bench

(*_UWB_Modulated_Transmit_Spectrum*)
To choose pulse position or bi-phase modulation, enable the PPM or bi-phase pulse generator component; deselect the unused component. Pulse shapes, as well as pulse rates, amplitudes, and widths can be selected. For details, refer to *BIPHASE_TX_PULSE_GENERATOR* (dguwb) or *PPM_TX_Pulse_Generator.* (dguwb)

The transmit filter represents the effects of transmitter front-end and antenna; for details, refer to *UWB_TRANSMIT_ANTENNA.* (dguwb)

A pseudorandom bit sequence is used as the spreading code for UWB modulation. For bi-phase modulation, one bit from the spreading code is consumed per transmitted pulse. The spreading code repeat time is equal to the pulse interval multiplied by the pseudorandom bit sequence length. For pulse position modulation, the value of DitherBits determines the number of spreading code bits consumed for each transmitted pulse. The spreading code repeat time for pulse position modulation is the pulse interval multiplied by the spreading code length divided by the number of value of DitherBits. The spreading code repeat time determines the spacing of spectral lines in the modulated transmit spectrum.

For the SpectrumAnalyzer *Filter_Output_Spectrum* component, the NPoints parameter is set equal to 1e-6/TStep; this corresponds to a sample time of 1 μ or a resolution bandwidth of 1 MHz. The spectral power density can be compared to the FCC regulations for indoor UWB radiation. For example, for 3.1 to 10.6 GHz the maximum allowed spectral power density is -41.3 dBm/MHz. The simulated average spectral power density in this frequency range must be less than this limit.

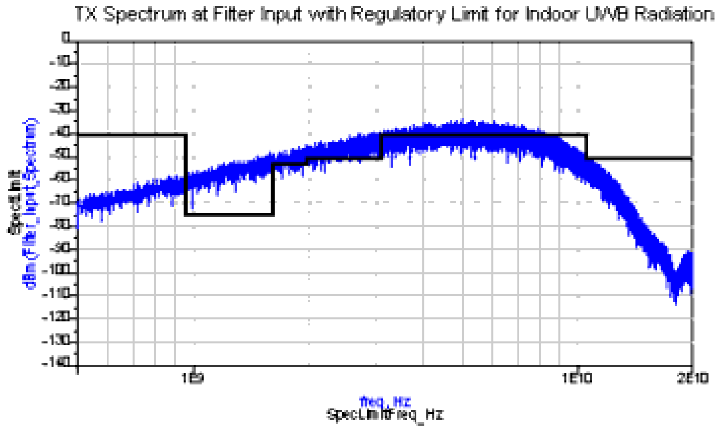
Simulation length is determined by the DefaultTimeStop parameter. When this value is larger, the output spectrum will have less deviation because it is averaged over more sample times.

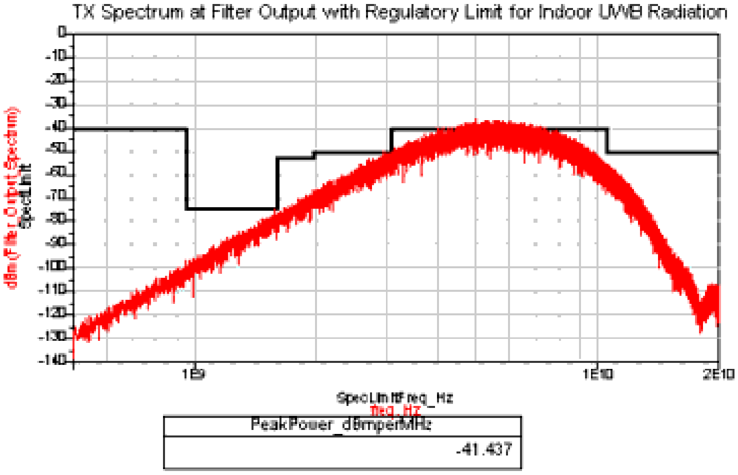
Modulated Transmit Spectrum Test Bench Design Parameters

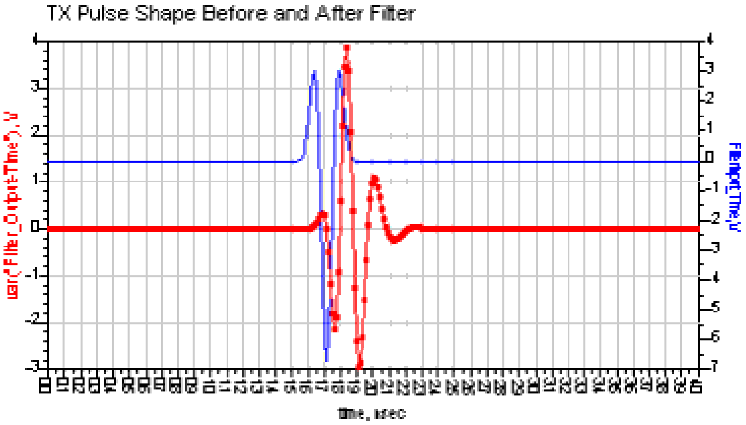
Name	**Description**
ChipInterval	Time between pulses. For PPM this is the nominal time between pulses.
ChipsPerBit	Number of pulses transmitted for each bit.
DitherBits	Number of bits used to the dither position of a PPM pulse within each pulse interval. $2^{DitherBits}$ pulse positions are possible within each PPM pulse interval.
TStepsPerPulseWidth	Approximate number of simulation time steps in one pulsewidth interval.
TStepsPerDither	Determines the number of simulation time steps between each possible position.
DithersPerPPMBitOffset	Number of dither positions offset between a value of 1 and 0 in a PPM modulated code. Length is given in the number of dither intervals in the offset time. This makes the offset between 0 and 1 an integer number of possible pulse positions. (For pulse position modulation.)
TStepsPerPPMBitOffset	Number of TStep offset between a value of 1 and a 0 in a PPM modulated code. Length is given in the number of TStep long intervals in the offset time. (For pulse position modulation.)
DitherTime	Time between possible dither positions in a PPM pulse interval. (For pulse position modulation.)
TStep	Time step of the simulation. It is an integer division of the ChipInterval. It is approximately PulseWidth/10
PulseWidth	1/2 amplitude pulsewidth of a Gaussian monopulse output. For the Gaussian derivative pulse shapes, this is the 1/2 amplitude pulsewidth of the Gaussian monopulse from which it is derived.
DoubletSeparation	Time between the positive and negative peaks of the waveform when doublet pulse is used. (Used when doublet waveform is selected.)
PulseEnergy_joule	Energy in Joules of a single pulse output from the pulse generator.

Simulation Results

The Data Display window shows the transmit spectrum before and after a transmit filter is applied. The transmit spectrum is compared to the FCC regulatory limit for indoor UWB radiation. Transmit pulse shapes before and after filtering are also displayed. Calculations of power in dBm/MHz over a user-selectable band are displayed.







Receiver Sensitivity Eb/No

The Bi-Phase and Pulse Position Modulation test benches simulate the bit-error-rate of a basic UWB transmitter and receiver as a function of receiver signal-to-noise ratio. Basic BER vs. signal-to-noise ratio can be investigated using this simulation (antenna and environmental effects are not included).

The time required for a simulation to run is dependent on TStep and the length of time simulated. TStep, determined by the PulseWidth value, is approximately 1/10th PulseWidth. The length of time simulated is the product of the number of bits simulated, the number of chips per bit, and the time interval per chip. Example simulation times using a 1500 MHz Pentium IV processor are:

- For a chip rate of 1 GHz using 5 chips per bit, 10000 bits can be simulated in 200 seconds when PulseWidth is 250 psec and TStep is 27.78 psec. Simulation time is 50 µ, so the simulation rate is 9000 TSteps per second.
- For a chip rate of 10 MHz using 5 chips per bit, 100 bits can be simulated in 285 seconds when PulseWidth is 250 psec and TStep is 20.13 psec. The simulation rate is 8700 TSteps per second.

Filters were not used in these simulations; adding filters with long impulse response times will increase the time for the simulation to complete. As time interval per chip increases, the simulation time for a fixed number of bits increases proportionally.

A time compression technique can be applied in order to maximize the number of bits that can be simulated in a given simulator run time. This is necessary because, to accurately simulate BERs, the simulation must run until approximately 10 times 1/BER bits have been output from the receiver. In addition, the time step size of the simulation is determined by the pulse width. This means that it takes longer to simulate a number of bits when the pulse rate is low; however, at low pulse rates, there is essentially no interaction between pulses. And, because the correlator reference signal is 0 during the time between pulses, the receive signal between pulses can be assumed to have insignificant contribution to the integrator output signal. This means that the chip interval used in simulation can be reduced as much as possible before interaction between pulses becomes a factor. This can significantly reduce the run time of low duty-cycle pulse simulations.

The StopBits variable determines the number of bits to be collected by the DataOutput sink. This controls the length of time required to run the simulation. To speed simulations that sweep the interfering noise power, StopBits can be defined in a VAR equation using the piecewise linear function so the number of bits simulated is 10 times the estimated 1/BER at that noise power. For high noise-power levels, the BER will be higher, and fewer bits are required to accurately determine the BER. At lower noise-power levels, simulation of more bits are required. The user can rely on information from previously completed simulations to configure the piecewise linear function. When the SpectrumAnalyzer and TimedSink components are active during a simulation sweep, the user can also use a piecewise-linear function to optimize the TimeStop variable for the sweep.

Receiver Sensitivity Eb/No Design Parameters

Name	**Description**
ChipInterval	Time between pulses. For PPM this is the nominal time between pulses.
ChipsPerBit	Number of pulses transmitted for each bit.
DitherBits	Number of bits used to determine the dither position of a PPM pulse within each pulse interval. There are $2^{DitherBits}$ possible pulse positions within each PPM pulse interval. (For pulse position modulation.)
TStepsPerDither	This determines the number of simulation time steps between each possible position.
DithersPerPPMBitOffset	Length of offset between 1 and 0 in a PPM modulated code. Length is given in the number of dither interval long intervals in the offset time. This make the offset between 0 and 1 an integer number of possible pulse positions. (For pulse position modulation.)
TStepsPerPPMBitOffset	Length of offset between 1 and 0 in a PPM modulated code. Length is given in the number of TStep long intervals in the offset time. This make the offset between 0 and 1 an integer number of possible pulse positions. (For pulse position modulation.)
DitherTime	Time between possible dither positions in a PPM pulse interval. (For pulse position modulation.)
TStep	Time step of the simulation. It is an integer division of the ChipInterval. It is approximately PulseWidth/10.
PulseWidth	1/2 amplitude pulsewidth of a Gaussian monopulse output. For the Gaussian derivative pulse shapes, this is the 1/2 amplitude pulsewidth of the Gaussian monopulse from which it is derived.
DoubletSeparation	time between the positive and negative peaks of the waveform when the GAUSSIAN_DOUBLET_UWB_TRANSMITTER subnetwork of PULSE_SHAPE_GENERATOR is active.
PulseEnergy_joule	Energy in Joules of a single pulse output from the pulse generator.
NoiseBandWidthRatio	Ratio of the simulation bandwidth to the noise bandwidth of the band limited noise source. The bandwidth is of the band limited noise source is 1/(2×NoiseBandWidthRatio×TStep).
NoisePower_dBm	Total power of bandwidth limited noise source in dBm.
StopBits	Number of bits simulated.
TimeStop	Length of time SpectrumAnalyzer components collect data.

Bi-Phase Modulation

The Bi-Phase Modulation Test Bench is shown in Bi-Phase Modulation Test Bench, Transmitter Section (_UWB_Biphase_Bench) (transmitter section) and Bi-Phase Modulation Test Bench, Receiver Section (_UWB_Biphase_Bench) (receiver section).

Representative pulse shapes can be selected. For component details, refer to *BIPHASE_TX_PULSE_GENERATOR.* (dguwb)

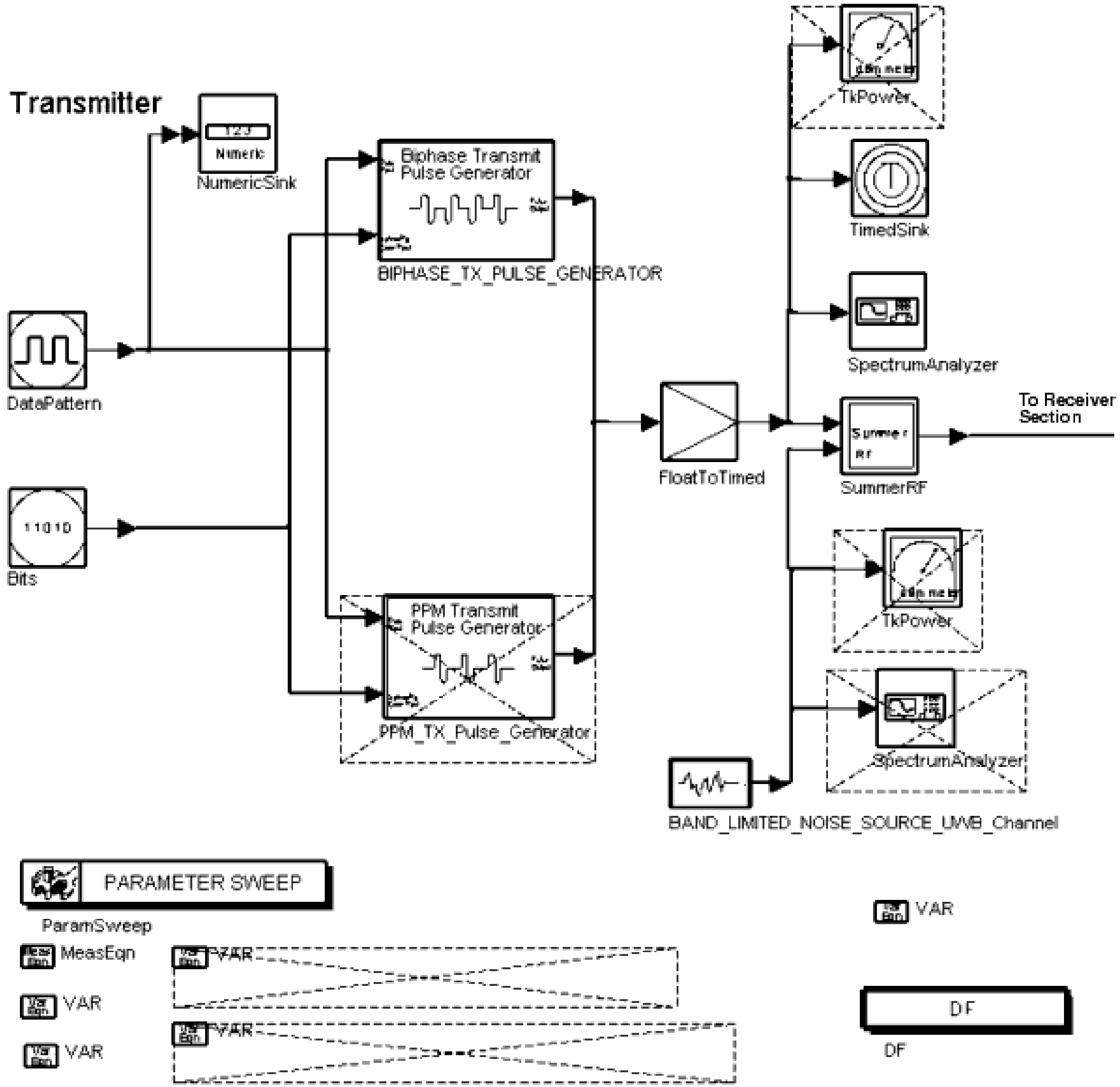
A pseudorandom code is used to spread the transmit data. Band-limited noise is added to the transmit signal before it enters the receiver. For component details, refer to *BAND_LIMITED_NOISE_SOURCE_UWB_Channel* (dguwb).

The BIPHASE_RX_REFERENCE_PULSER component outputs the same waveform as BIPHASE_TX_PULSE_GENERATOR when the data input is all 1s. The waveform output from this test block represents a bi-phase modulated UWB waveform. The input data bits are spread using a spreading code. For component details, refer to

BIPHASE_RX_REFERENCE_PULSER. (dguwb)

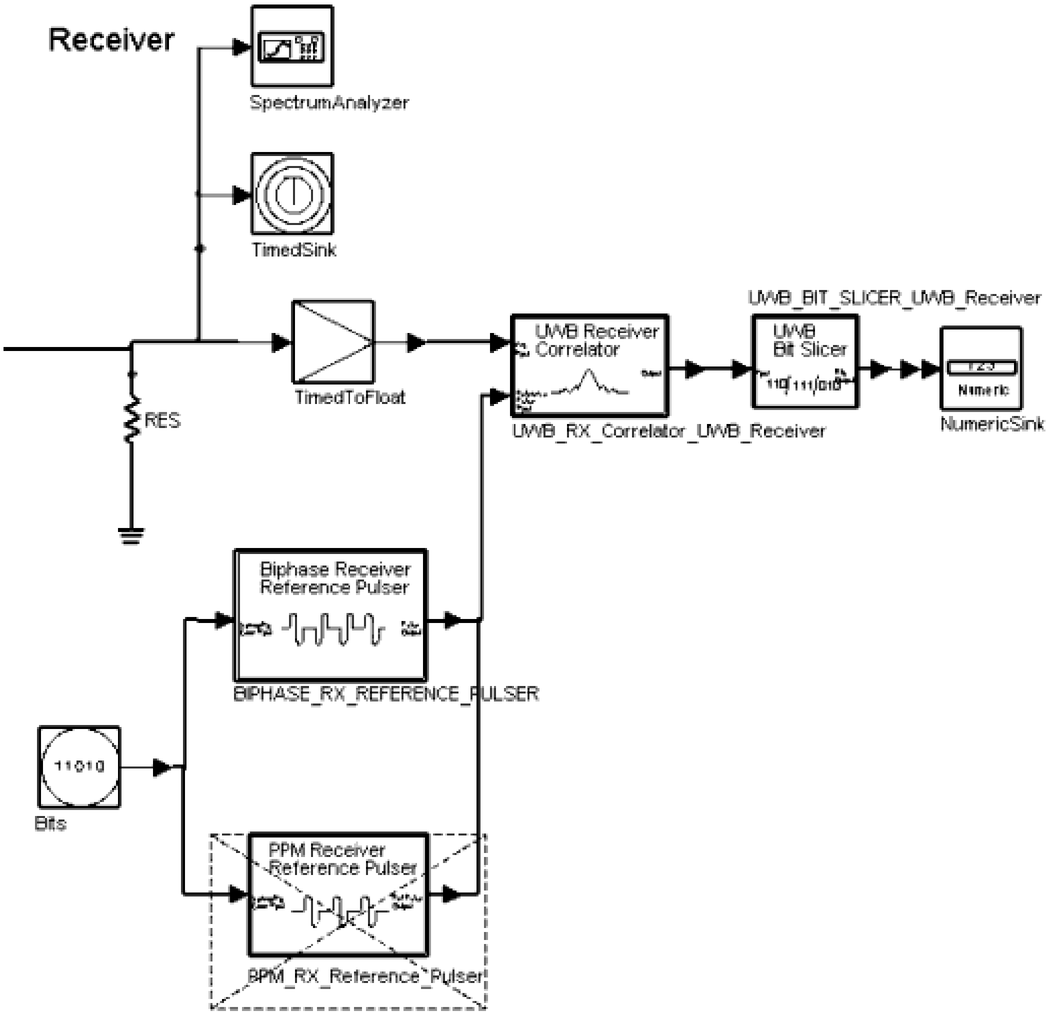
Simulations can sweep the noise power level relative to that of the transmit signal. The receiver correlator de-spreads the input UWB signal and the data bit stream is recovered from the correlator output. For component details, refer to *UWB_RX_Correlator_UWB_Receiver.* (dguwb)

The bit slicer captures the correlator integrator output value immediately before the integrator resets. This value is used to determine whether the output is a 0 or 1 bit. For component details, refer to *UWB_BIT_SLICER_UWB_Receiver.* (dguwb)



Bi-Phase Modulation Test Bench, Transmitter Section

(*_UWB_Biphase_Bench*)

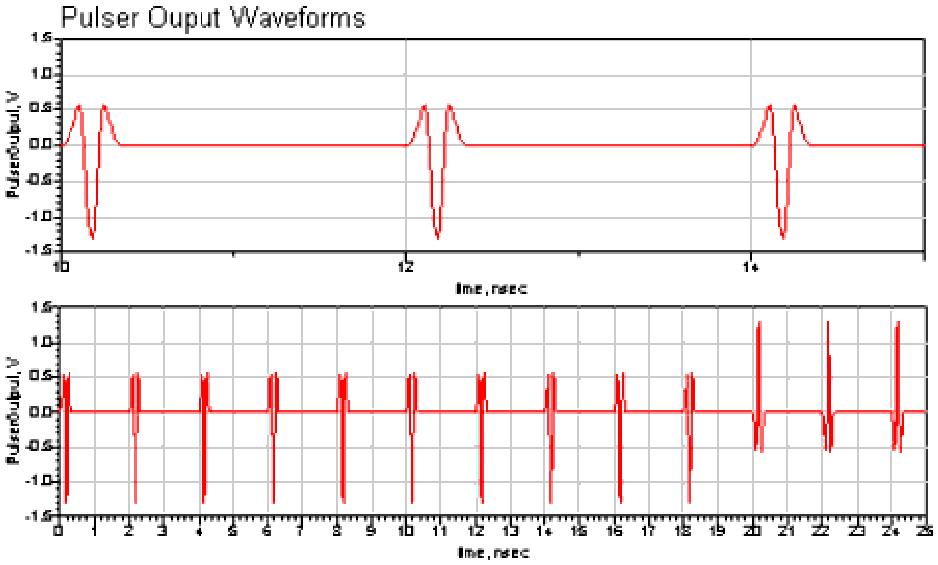

Receiver
SpectrumAnalyzer
TimedSink
UWB_BIT_SLICER_UWB_Receiver
UWB Receiver Correlator
UWB Bit Slicer
Numeric
NumericSink
TimedToFloat
RES
UWB_RX_Correlator_UWB_Receiver
Biphase Receiver Reference Pulser
BIPHASE_RX_REFERENCE_PULSER
11010
Bits
PPM Receiver Reference Pulser
PPM_RX_Reference_Pulser

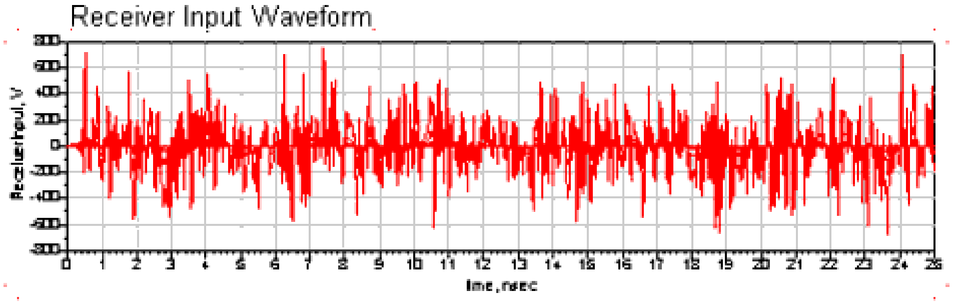
Bi-Phase Modulation Test Bench, Receiver Section

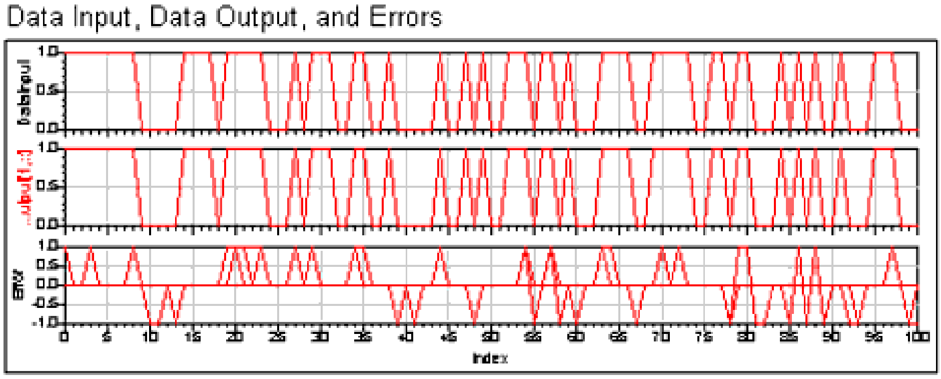
(*_UWB_Biphase_Bench*)

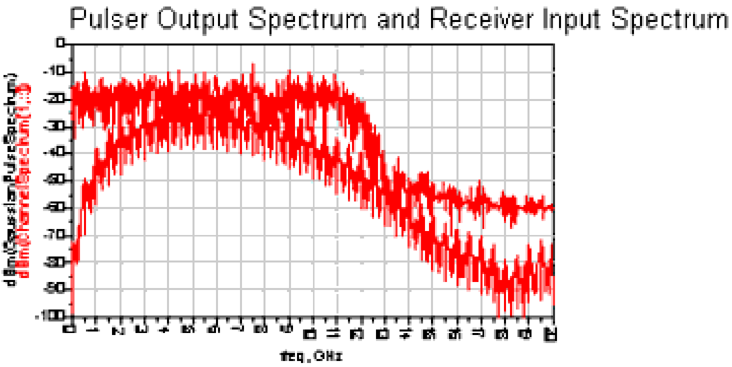
Simulation Results

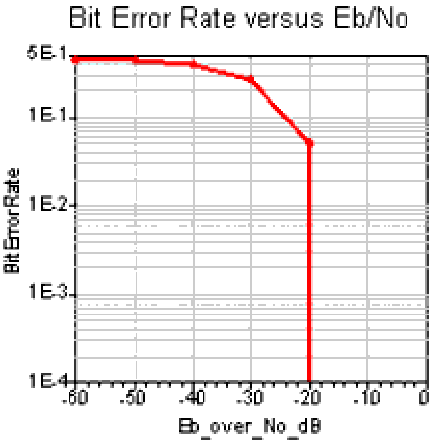
For bi-phase modulation, the Data Display window shows the transmit pulse train over two time scales. The receiver input signal plot shows the transmit signal combined with interfering noise. The spectra of the transmit signal and interfering noise are also shown. Bit errors are determined by comparing the data bits input to the transmitter to those output from the receiver. A BER vs. signal-to-noise ratio plot (labeled *Eb/No*) shows how BER is degraded by the interfering noise.

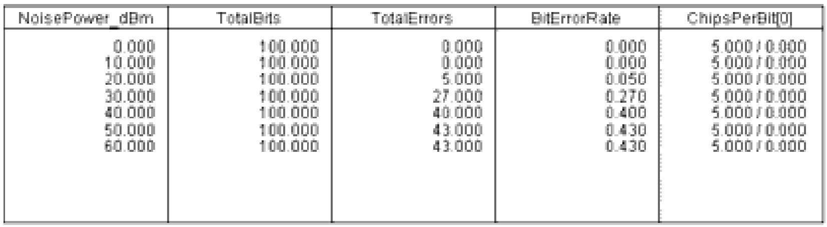

Pulser Ouput Waveforms
PulserOutput, V
1.5
1.0
0.5
0.0
-0.5
-1.0
-1.5
10
12
14
time, nsec
PulserOutput, V
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
time, nsec


Receiver Input Waveform
ReceiverInput, V
800
600
400
200
0
-200
-400
-600
-800
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
time, nsec

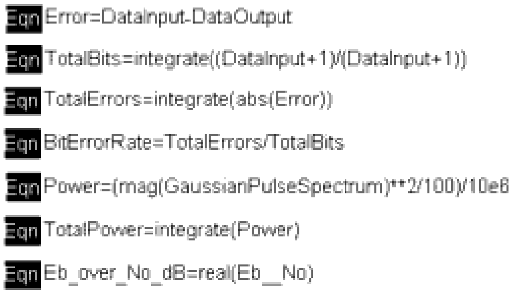

Data Input, Data Output, and Errors
DataInput
Output[1,::]
Error
1.0
0.5
0.0
-0.5
-1.0
0 5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80 85 90 95 100
Index







NoisePower_dBm	TotalBits	TotalErrors	BitErrorRate	ChipsPerBit[0]
0.000	100.000	0.000	0.000	5.000 / 0.000
10.000	100.000	0.000	0.000	5.000 / 0.000
20.000	100.000	5.000	0.050	5.000 / 0.000
30.000	100.000	27.000	0.270	5.000 / 0.000
40.000	100.000	40.000	0.400	5.000 / 0.000
50.000	100.000	43.000	0.430	5.000 / 0.000
60.000	100.000	43.000	0.430	5.000 / 0.000



Eqn Error=DataInput-DataOutput

Eqn TotalBits=integrate((DataInput+1)/(DataInput+1))

Eqn TotalErrors=integrate(abs(Error))

Eqn BitErrorRate=TotalErrors/TotalBits

Eqn Power=(mag(GaussianPulseSpectrum)**2/100)/10e6

Eqn TotalPower=integrate(Power)

Eqn Eb_over_No_dB=real(Eb__No)

Pulse Position Modulation

The Pulse Position Modulation Test Bench is shown in Pulse Position Modulation Test Bench, Transmitter Section (_UWB_PPM_Bench) (transmitter section) and Pulse Position Modulation Test Bench, Receiver Section (_UWB_PPM_Bench) (receiver section).

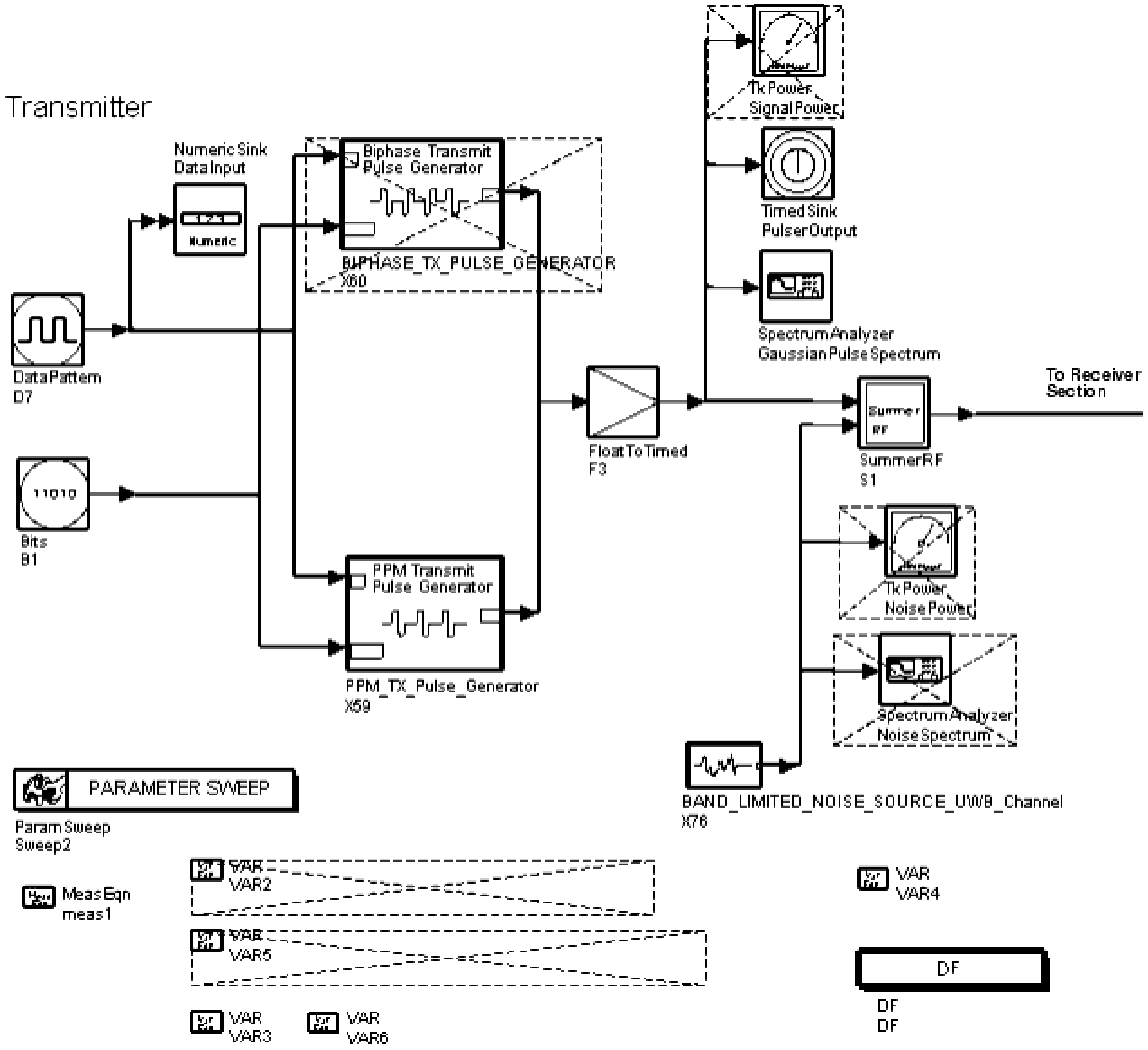
Representative pulse shapes can be selected. For component details, refer to *PPM_TX_Pulse_Generator.* (dguwb)

A pseudorandom code is used to spread the transmit data. Band-limited noise is added to the transmit signal before it enters the receiver. For component details, refer to *BAND_LIMITED_NOISE_SOURCE_UWB_Channel.* (dguwb)

The PPM_RX_Reference_Pulser component outputs a waveform that is the sum of the pulse waveform produced by PPM_TX_Pulse_Generator when the data input is 1 with the inverse of the output when the data input is 0. The output represents a pulse position modulated UWB waveform. The input data bits are spread using a spreading code. For component details, refer to *PPM_RX_Reference_Pulser.* (dguwb)

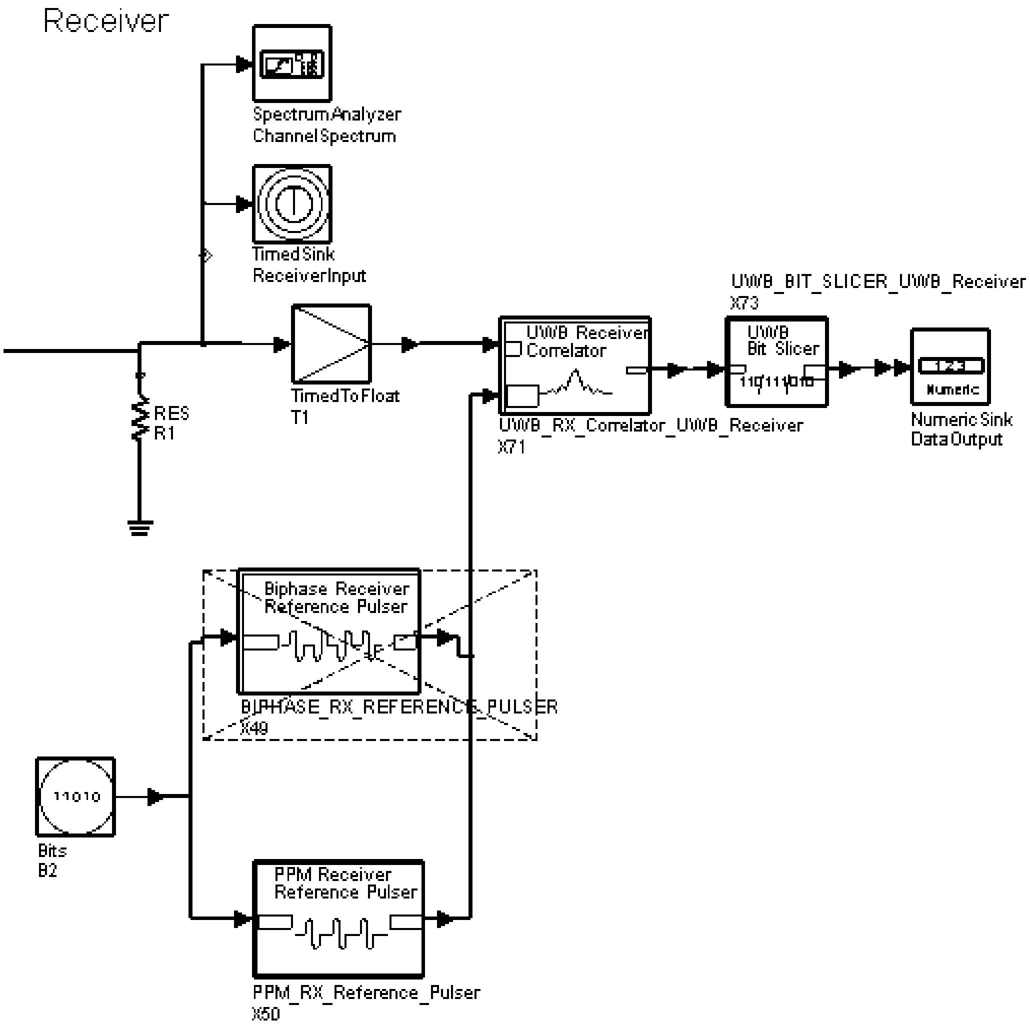
Simulations can sweep the noise power level relative to that of the transmit signal. The receiver correlator de-spreads the input UWB signal and the data bit stream is recovered from the correlator output. For component details, refer to *UWB_RX_Correlator_UWB_Receiver.* (dguwb)

The bit slicer captures the correlator integrator output value immediately before the integrator resets. This value is used to determine whether the output is a 0 or 1 bit. For component details, refer to *UWB_BIT_SLICER_UWB_Receiver.* (dguwb)



Pulse Position Modulation Test Bench, Transmitter Section

(_UWB_PPM_Bench)

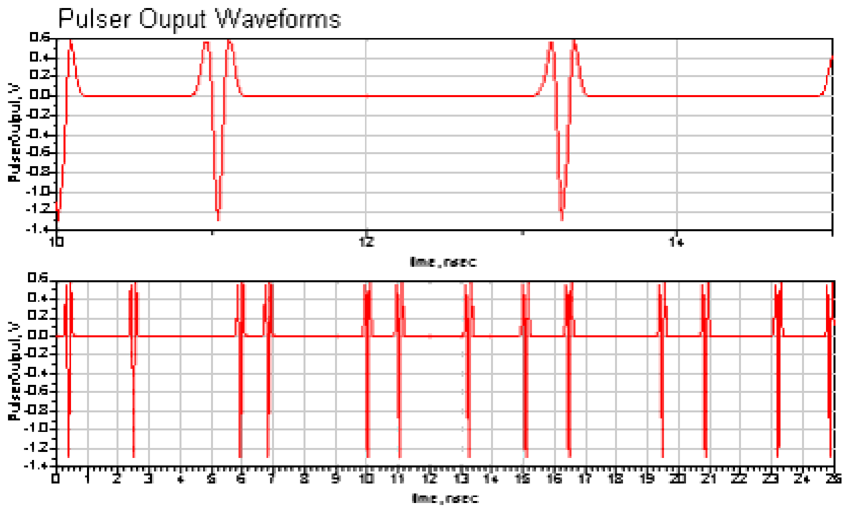


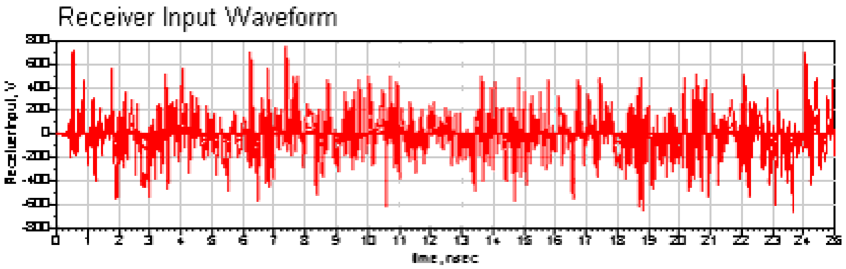
Pulse Position Modulation Test Bench, Receiver Section

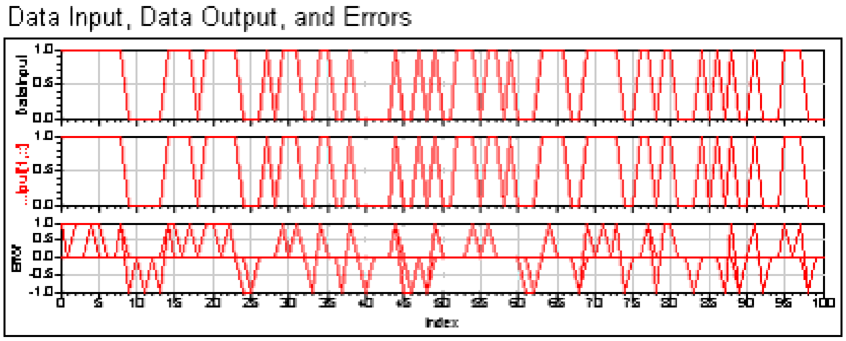
(*_UWB_PPM_Bench*)

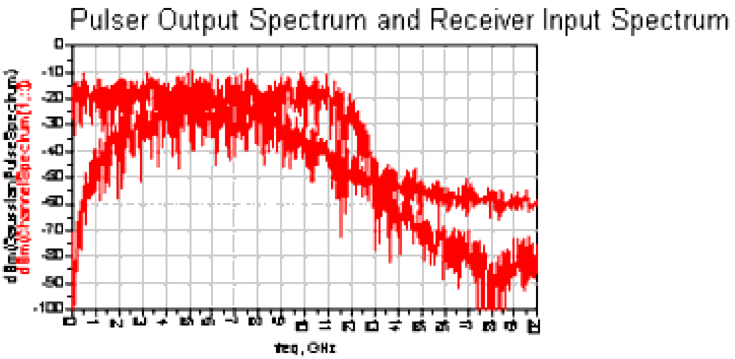
Simulation Results

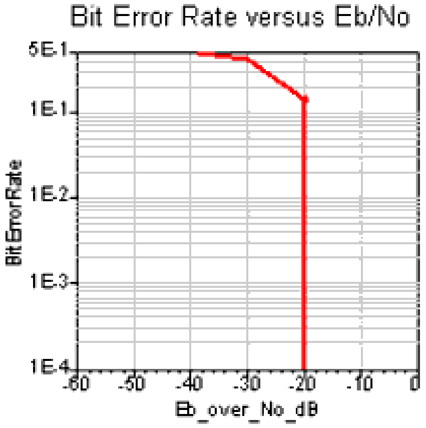
For pulse position modulation, the Data Display window shows the transmit pulse train over two time scales. The receiver input signal plot shows the transmit signal combined with interfering noise. The spectra of the transmit signal and interfering noise are also shown. Bit errors are determined by comparing the data bits input to the transmitter to those output from the receiver. A BER vs. signal-to-noise ratio plot (labeled *Eb/No*) shows how BER is degraded by the interfering noise.

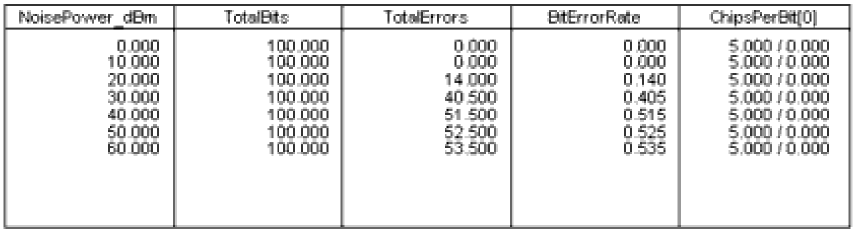

Pulser Ouput Waveforms
PulserOutput, V
0.6
0.4
0.2
0.0
-0.2
-0.4
-0.6
-0.8
-1.0
-1.2
-1.4
10
12
14
time, nsec
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
time, nsec


Receiver Input Waveform
ReceiverInput, V
800
600
400
200
0
-200
-400
-600
-800
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
time, nsec

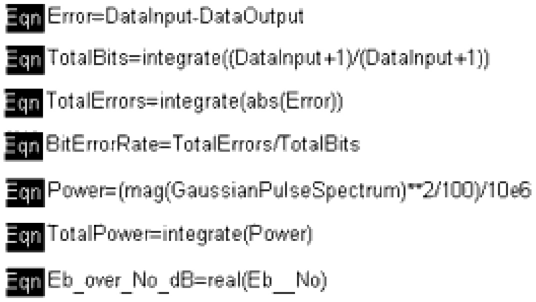

Data Input, Data Output, and Errors
DataInput
1.0
0.5
0.0
...tput[1,::]
Error
-0.5
-1.0
0 5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80 85 90 95 100
index







NoisePower_dBm	TotalBits	TotalErrors	BitErrorRate	ChipsPerBit[0]
0.000	100.000	0.000	0.000	5.000 / 0.000
10.000	100.000	0.000	0.000	5.000 / 0.000
20.000	100.000	14.000	0.140	5.000 / 0.000
30.000	100.000	40.500	0.405	5.000 / 0.000
40.000	100.000	51.500	0.515	5.000 / 0.000
50.000	100.000	52.500	0.525	5.000 / 0.000
60.000	100.000	53.500	0.535	5.000 / 0.000



Eqn Error=DataInput-DataOutput

Eqn TotalBits=integrate((DataInput+1)/(DataInput+1))

Eqn TotalErrors=integrate(abs(Error))

Eqn BitErrorRate=TotalErrors/TotalBits

Eqn Power=(mag(GaussianPulseSpectrum)**2/100)/10e6

Eqn TotalPower=integrate(Power)

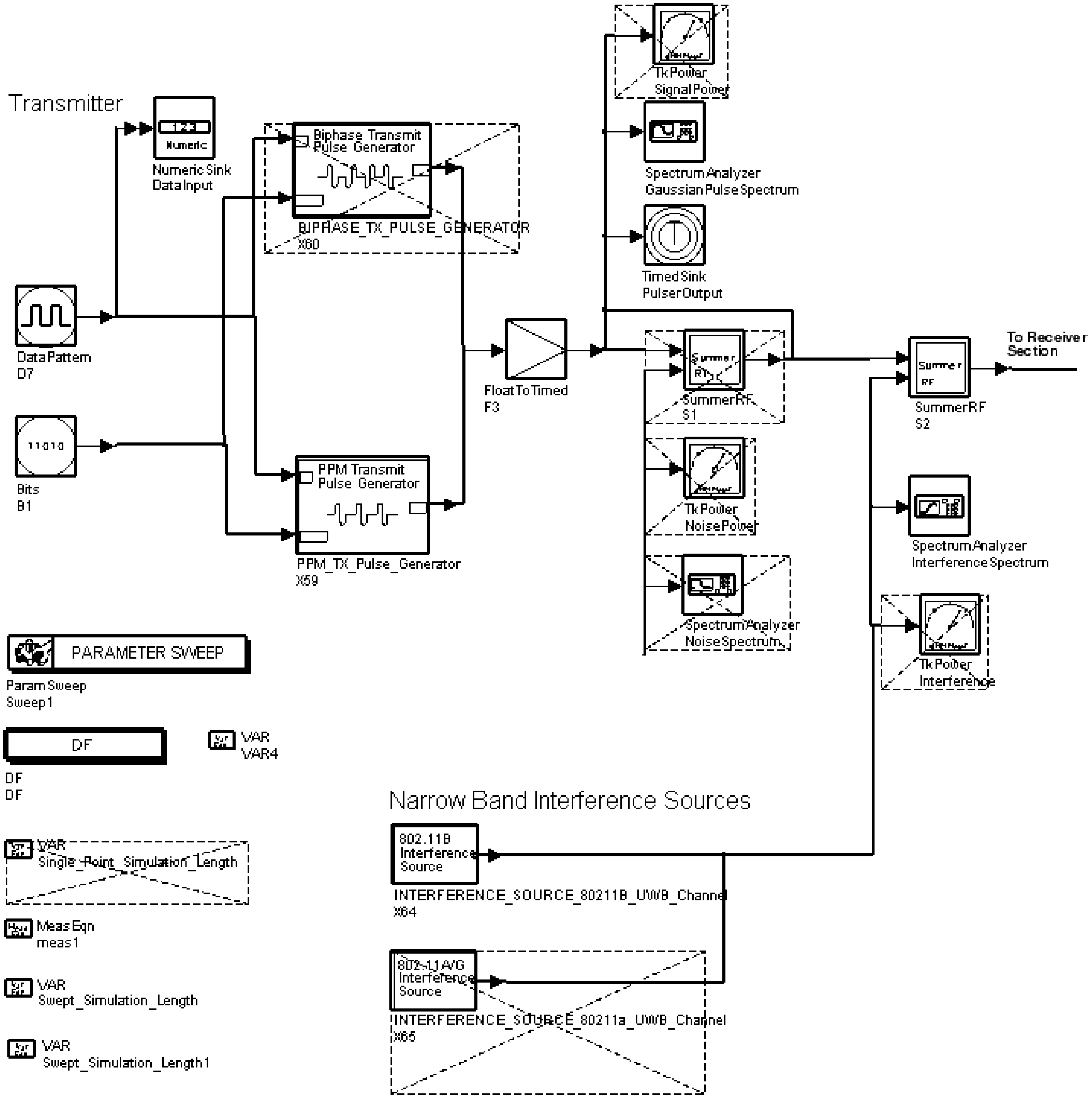
Eqn Eb_over_No_dB=real(Eb__No)

Receiver Sensitivity with Interference Sources

Narrow Band Interference

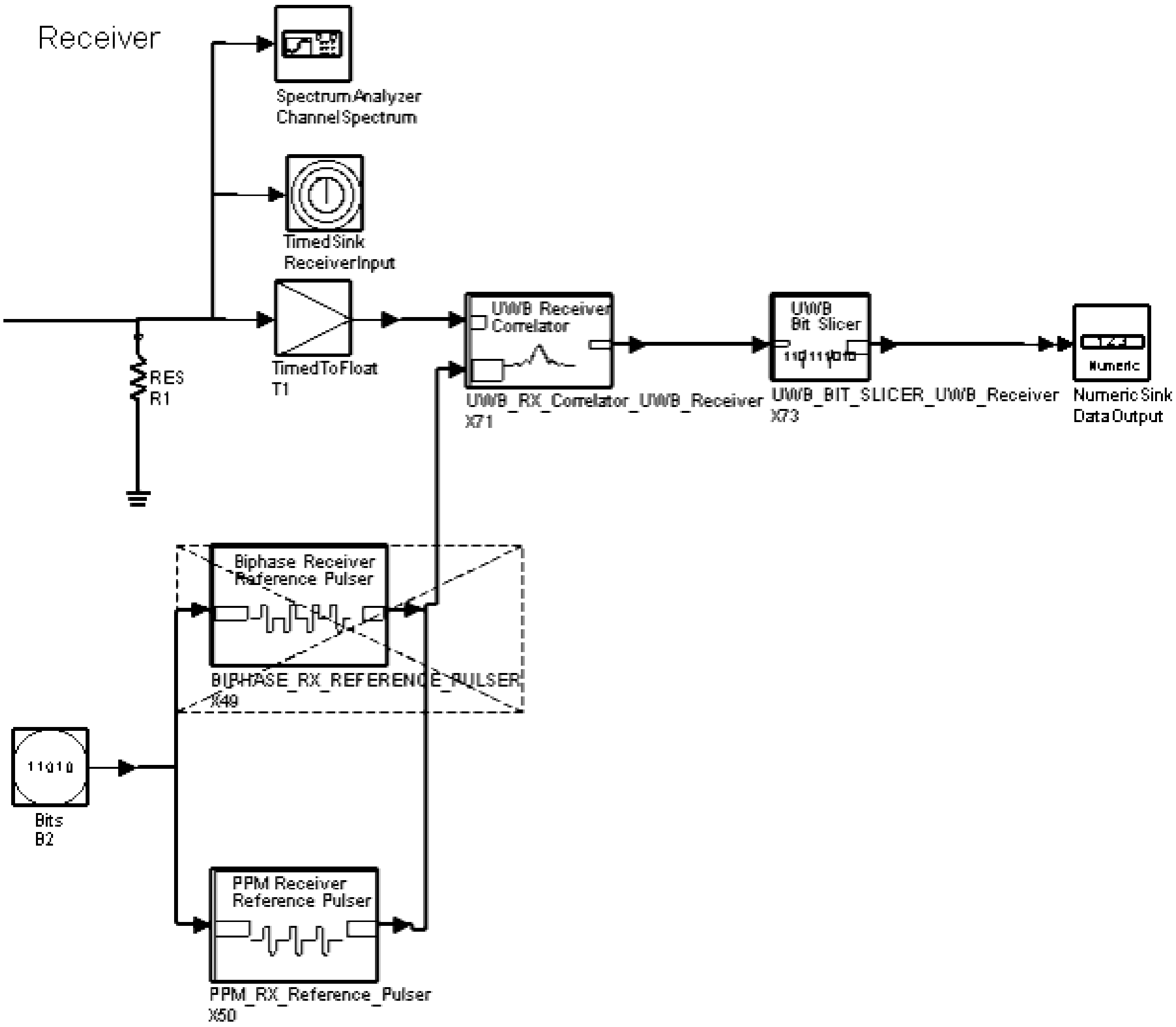
The Narrow Band Interference Test Bench simulates a UWB transmitter and receiver with interference from an 802.11a/g or 802.11b signal source. Simulations can sweep the interference power level relative to that of the transmit signal. A correlator de-spreads and bit slices the received UWB signal. The receiver outputs the demodulated bit stream.

The transmitter section of the test bench is shown in Narrow Band Test Bench, Transmitter Section (_UWB_Narrow_Band_Interference); the receiver section of the test bench is shown in Narrow Band Test Bench, Receiver Section (_UWB_Narrow_Band_Interference).



Narrow Band Test Bench, Transmitter Section

(*_UWB_Narrow_Band_Interference*)



Narrow Band Test Bench, Receiver Section

(*_UWB_Narrow_Band_Interference*)
A pseudorandom code is used to spread the transmit data. Pulse shapes, as well as pulse rates, amplitudes, and widths can be selected. To choose pulse position or bi-phase modulation, enable the PPM or bi-phase pulse generator component; deselect the unused component. For component details, refer to *BIPHASE_TX_PULSE_GENERATOR* (dguwb) or *PPM_TX_Pulse_Generator.* (dguwb)

To choose the interference source, enable the 802.11B or 802.11a source component; deselect the unused component.

- The waveform for the 802.11a source is read from dataset file *WLAN_80211a_Order11.ds* ; for component details, refer to *INTERFERENCE_SOURCE_80211a_UWB_Channel.* (dguwb)

- The waveform for the 802.11b source is read from dataset file *WLAN_80211b_8Xoversample.ds* ; for component details, refer to *INTERFERENCE_SOURCE_80211B_UWB_Channel.* (dguwb)

For bi-phase modulation, the BIPHASE_RX_REFERENCE_PULSER component outputs the same waveform as BIPHASE_TX_PULSE_GENERATOR when data input is all 1s. The output represents a bi-phase modulated UWB waveform. Input data bits are spread using a spreading code. For component details, refer to *BIPHASE_RX_REFERENCE_PULSER.* (dguwb)

For pulse position modulation, the PPM_RX_Reference_Pulser component outputs the same waveform as PPM_TX_Pulse_Generator when the data input is 1 with the inverse of the output when the data input is 0. The output represents a pulse position modulated UWB waveform. The input data bits are spread using a spreading code. For component details, refer to *PPM_RX_Reference_Pulser.* (dguwb)

UWB_RX_Correlator_UWB_Receiver provides multiple correlators for receiving arrivals of a multipath signal. Each correlator multiplies the receive signal by an appropriately delayed reference signal. The integrator in the correlator integrates the multiplier output signal over the period of ChipInterval×ChipsPerBit. It resets the integrator value to 0 and restarts the integration. The outputs of each correlator are scaled relative to its signal-to-noise ratio, and the outputs of all correlators are summed. For component details, refer to *UWB_RX_Correlator_UWB_Receiver.* (dguwb)

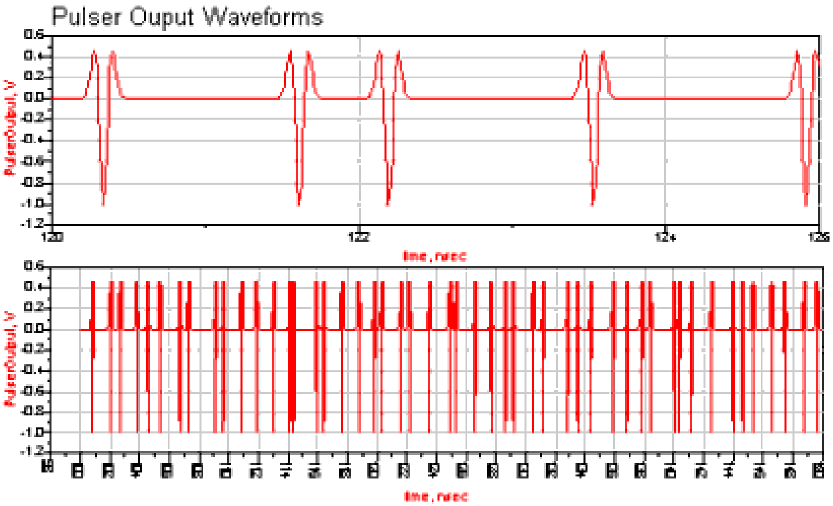
The bit slicer captures the correlator integrator output value immediately before the integrator resets. This value is used to determine whether the output is a 0 or 1 bit. For component details, refer to *UWB_BIT_SLICER_UWB_Receiver.* (dguwb)

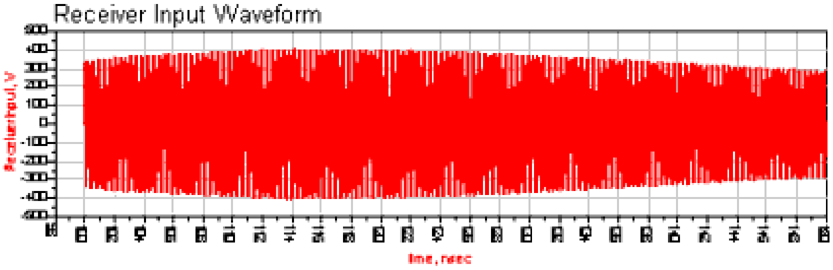
Narrow Band Interference Test Bench Design Parameters

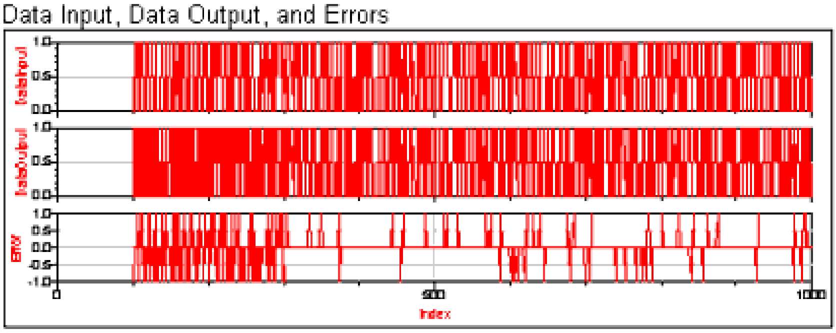
Name	**Description**
ChipInterval	Time between pulses. For PPM this is the nominal time between pulses.
ChipsPerBit	Number of pulses transmitted for each bit.
DitherBits	Number of bits used to determine the dither position of a PPM pulse within each pulse interval. There are $2^{DitherBits}$ possible pulse positions within each PPM pulse interval. (For pulse position modulation.)
TStepsPerDither	Determines the number of simulation time steps between each possible position.
DithersPerPPMBitOffset	Length of the offset between 1 and 0 in a PPM modulated code. Length is given in the number of dither interval long intervals in the offset time. This makes the offset between 0 and 1 an integer number of possible pulse positions. (For pulse position modulation.)
TStepsPerPPMBitOffset	Length of offset between 1 and 0 in a PPM modulated code. Length is given in the number of TStep long intervals in the offset time. This make the offset between 0 and 1 an integer number of possible pulse positions. (For pulse position modulation.)
DitherTime	Time between possible dither positions in a PPM pulse interval. (For pulse position modulation.).
TStep	Time step of the simulation. An integer division of ChipInterval; it is approximately PulseWidth/10.
PulseWidth	1/2 amplitude pulsewidth of a Gaussian monopulse output. For the Gaussian derivative pulse shapes, this is the 1/2 amplitude pulsewidth of the Gaussian monopulse from which it is derived.
DoubletSeparation	Time between the positive and negative peaks of the waveform when doublet pulse is used. (Used when doublet waveform is selected.)
PulseEnergy_joule	Energy in Joules of a single pulse output from the pulse generator.
NoiseBandWidthRatio	Ratio of simulation bandwidth to noise bandwidth of the band limited noise source. The bandwidth is of the band limited noise source is 1/(2×NoiseBandWidthRatio×TStep).
NoisePower_dBm	Total power of bandwidth limited noise source in dBm.
wlanPower_dBm	Center power of interfering WLAN source.
StopBits	Number of bits simulated.
TimeStop	Length of time SpectrumAnalyzer components collect data.

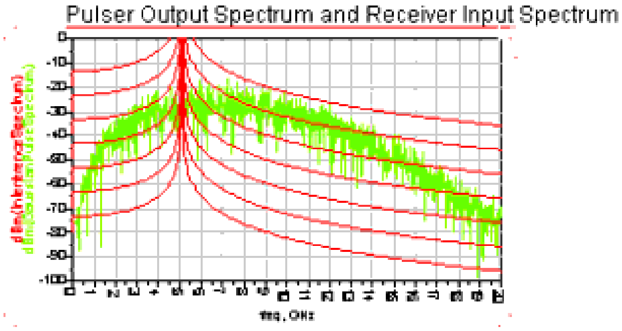
Simulation Results

The Data Display window shows the transmit pulse train over two time scales. The receiver input signal plot shows the transmit signal combined with the narrow band interference. The spectra of the transmit signal and the interfering noise are also shown.

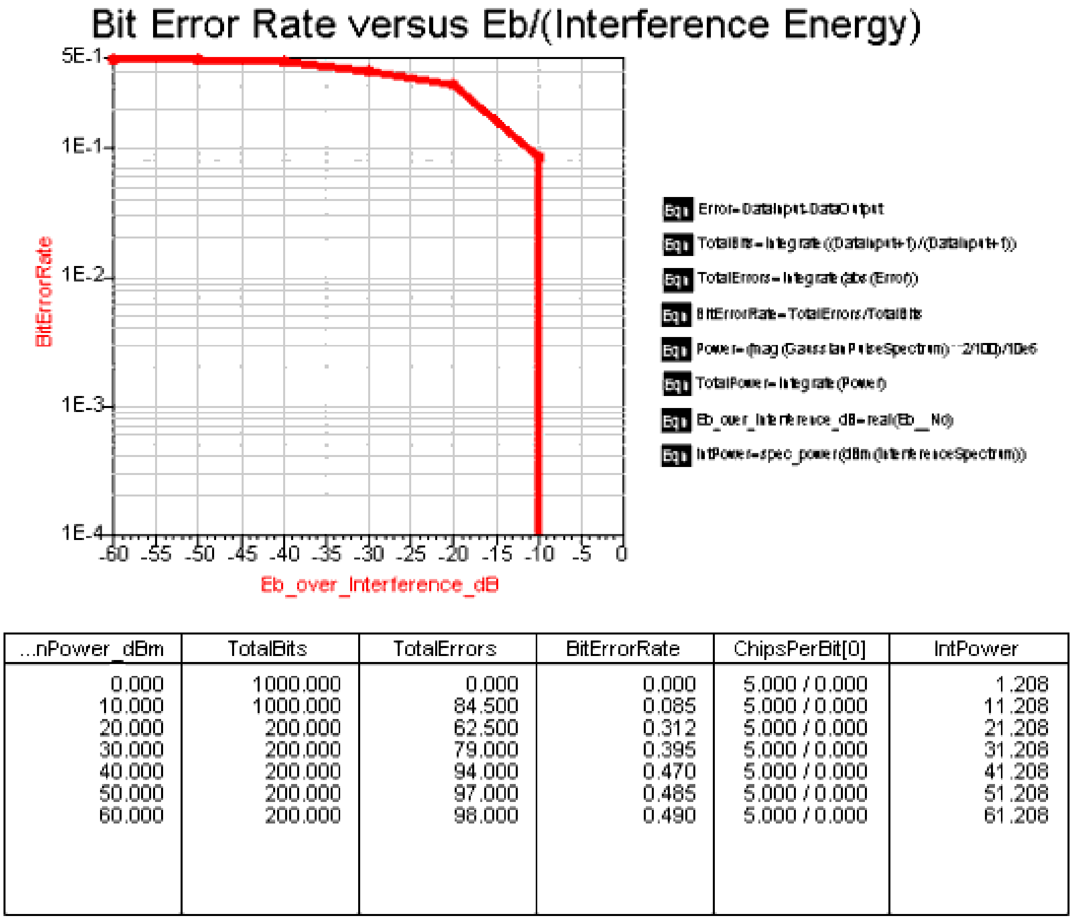








Bit errors are determined by comparing the data bits input to the transmitter to those output from the receiver. An Eb/No plot shows how the bit error rate is degraded by the narrow band interference power.

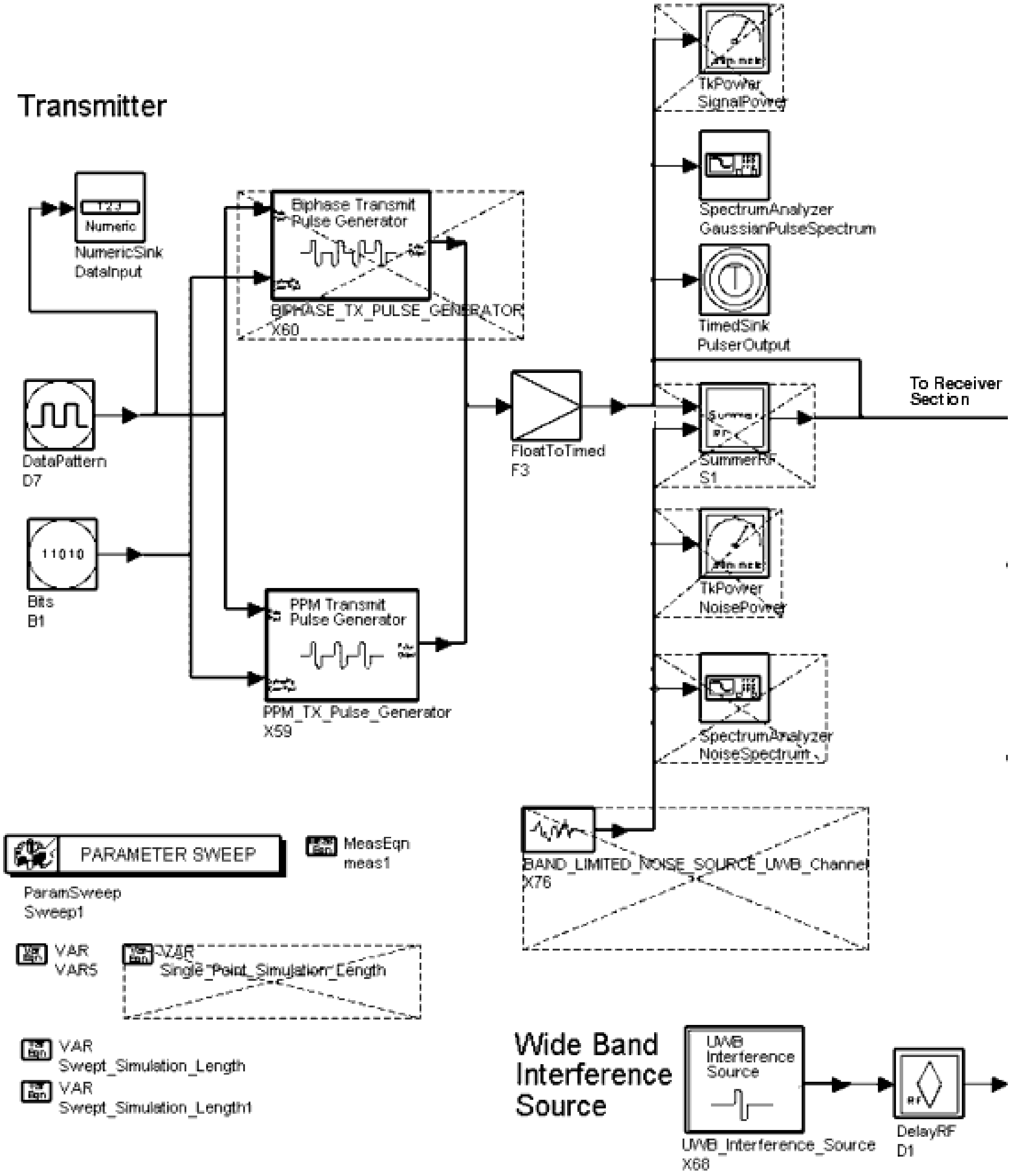


```
Error=DataInput-DataOutput
TotalBits=integrate((DataInput+1)/(DataInput+1))
TotalErrors=integrate(abs(Error))
BitErrorRate=TotalErrors/TotalBits
Power=(mag(GaussianPulseSpectrum)**2/100)/1e6
TotalPower=integrate(Power)
Eb_over_Interference_dB=real(Eb__No)
IntPower=spec_power(dBm(InterferenceSpectrum))
```

...nPower_dBm	TotalBits	TotalErrors	BitErrorRate	ChipsPerBit[0]	IntPower
0.000	1000.000	0.000	0.000	5.000 / 0.000	1.208
10.000	1000.000	84.500	0.085	5.000 / 0.000	11.208
20.000	200.000	62.500	0.312	5.000 / 0.000	21.208
30.000	200.000	79.000	0.395	5.000 / 0.000	31.208
40.000	200.000	94.000	0.470	5.000 / 0.000	41.208
50.000	200.000	97.000	0.485	5.000 / 0.000	51.208
60.000	200.000	98.000	0.490	5.000 / 0.000	61.208

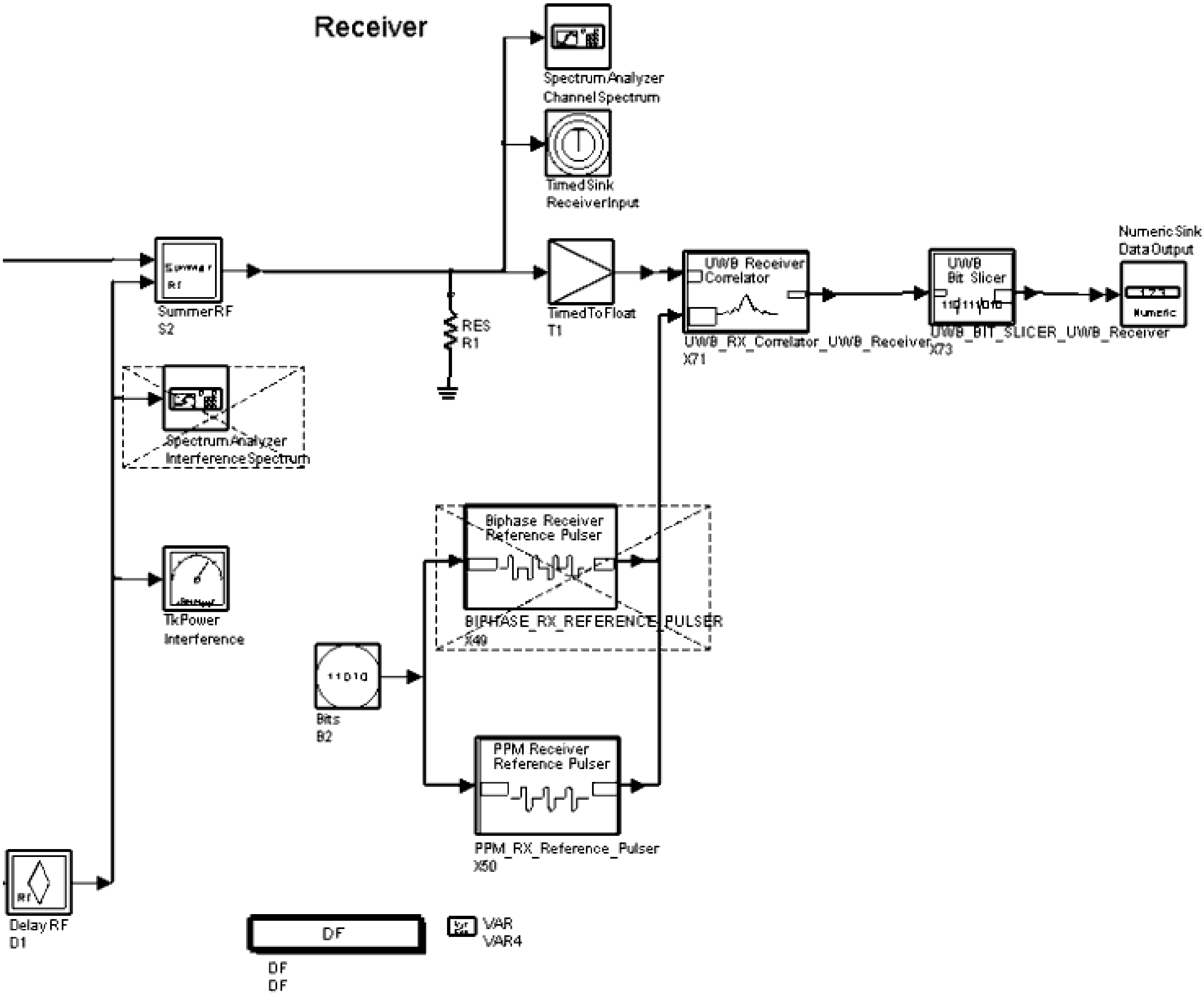
Wide Band Interference

The Wide Band Interference Test Bench simulates a UWB transmitter and receiver with interference from another UWB transmitter. Simulations can sweep the interference power level relative to that of the transmit signal. A correlator de-spreads and bit slices the received UWB signal. The receiver outputs the demodulated bit stream.
The transmitter section of the test bench is shown in Wide Band Interference Test Bench, Transmitter Section (_UWB_Wide_Band_Interference); the receiver section of the test bench is shown in Wide Band Interference Test Bench, Receiver Section (_UWB_Wide_Band_Interference).



Wide Band Interference Test Bench, Transmitter Section

(_UWB_Wide_Band_Interference)



Wide Band Interference Test Bench, Receiver

(*_UWB_Wide_Band_Interference*)
Pulse shapes, as well as pulse rates, amplitudes, and widths can be selected. To choose pulse position or bi-phase modulation, enable the PPM or bi-phase pulse generator component; deselect the unused component. For component details, refer to *BIPHASE_TX_PULSE_GENERATOR* (dguwb) or *PPM_TX_Pulse_Generator.* (dguwb)

The interference source is a second UWB transmitter. For component details, refer to *UWB_Interference_Source.* (dguwb)

For bi-phase modulation, the BIPHASE_RX_REFERENCE_PULSER component outputs the same waveform as BIPHASE_TX_PULSE_GENERATOR when the data input is all 1s. The output represents a bi-phase modulated UWB waveform. The input data bits are spread using a spreading code. For component details, refer to *BIPHASE_RX_REFERENCE_PULSER.* (dguwb)

For pulse position modulation, the PPM_RX_Reference_Pulser component outputs the same waveform as PPM_TX_Pulse_Generator when the data input is 1 with the inverse of the output when the data input is 0. The output represents a pulse position modulated UWB waveform. The input data bits are spread using a spreading code. For component details, refer to *PPM_RX_Reference_Pulser.* (dguwb)

UWB_RX_Correlator_UWB_Receiver provides multiple correlators for receiving arrivals of a multipath signal. Each correlator multiplies the receive signal by an appropriately delayed reference signal. The integrator in the correlator integrates the multiplier output signal over the period of ChipInterval×ChipsPerBit. It resets the integrator value to 0 and restarts the integration. The outputs of each correlator are scaled relative to its signal-to-noise ratio, and the outputs of all correlators are summed. For component details, refer to *UWB_RX_Correlator_UWB_Receiver.* (dguwb)

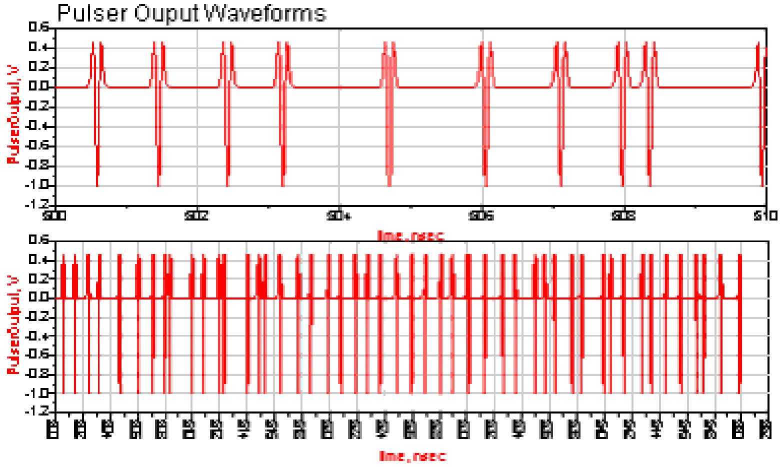
The bit slicer captures the correlator integrator output value immediately before the integrator resets. This value is used to determine whether the output is a 0 or 1 bit. For component details, refer to *UWB_BIT_SLICER_UWB_Receiver.* (dguwb)

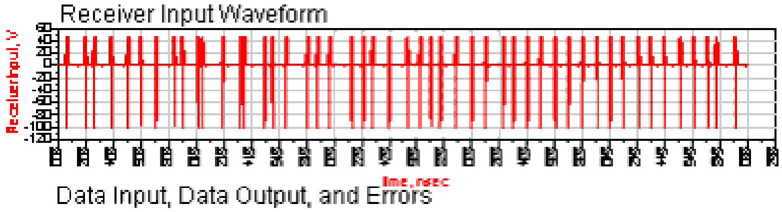
Wide Band Interference Test Bench Design Parameters

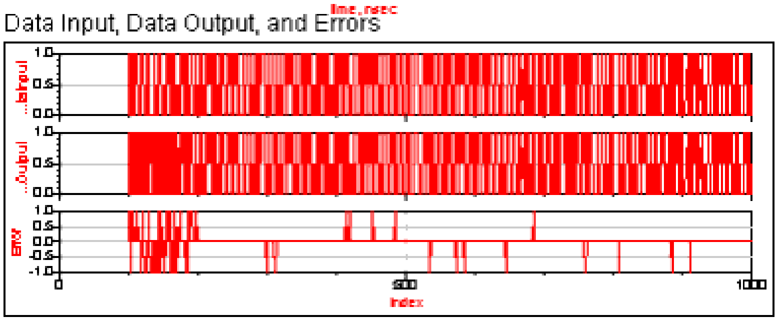
Name	Description
ChipInterval	Time between pulses. For PPM this is the nominal time between pulses.
ChipsPerBit	Number of pulses transmitted for each bit.
DitherBits	Number of bits used to determine the dither position of a PPM pulse within each pulse interval. There are $2^{DitherBits}$ possible pulse positions within each PPM pulse interval. (For pulse position modulation.)
TStepsPerDither	Determines the number of simulation time steps between each possible position.
DithersPerPPMBitOffset	Length of offset between 1 and 0 in a PPM modulated code. The length is given in the number of dither long intervals in the offset time. This makes the offset between 0 and 1 an integer number of possible pulse positions. (For pulse position modulation.)
TStepsPerPPMBitOffset	Length of offset between a value of 1 and 0 in a PPM modulated code. The length is given in the number of TStep long intervals in the offset time. This make the offset between 0 and 1 an integer number of possible pulse positions. (For pulse position modulation.)
DitherTime	Time between possible dither positions in a PPM pulse interval. (For pulse position modulation.)
TStep	Time step of the simulation. It is an integer division of the ChipInterval. It is approximately PulseWidth/10.
PulseWidth	1/2 amplitude pulsewidth of a Gaussian monopulse output. For the Gaussian derivative pulse shapes, this is the 1/2 amplitude pulsewidth of the Gaussian monopulse from which it is derived.
PulseEnergy_joule	Energy in Joules of a single pulse output from the pulse generator.
DoubletSeparation	Time between positive and negative peaks of the waveform when doublet pulse is used.
MaxFingerDelay	Maximum delay applied to a correlator finger in the rake receiver.
NoiseBandWidthRatio	Ratio of simulation bandwidth to noise bandwidth of band limited noise source. The bandwidth is of the band limited noise source is 1/(2×NoiseBandWidthRatio×TStep).
NoisePower_dBm	Total power of bandwidth limited noise source in dBm.
Interference_dBm	Power of interfering UWB source in dB relative to desired channel power.
StopBits	Number of bits simulated.
TimeStop	Length of time that SpectrumAnalyzer components collect data.

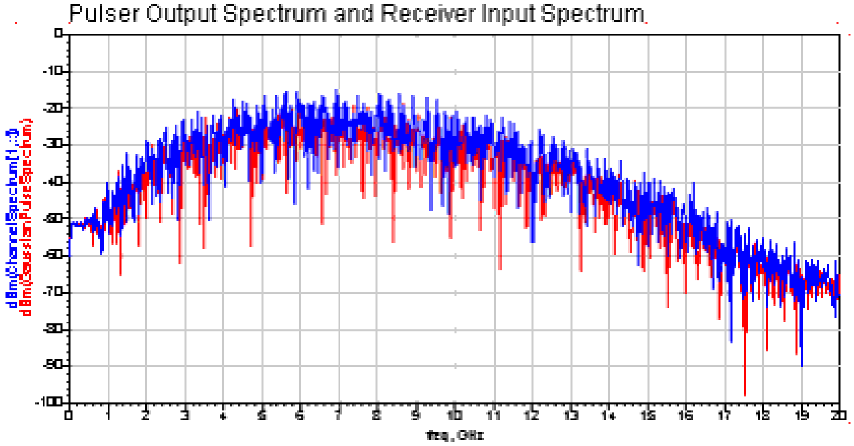
Simulation Results

The Data Display window shows the transmit pulse train over two time scales. The receiver input signal plot shows the transmit signal combined with the wide band interference. The spectra of the transmit signal and the interfering noise are also shown.

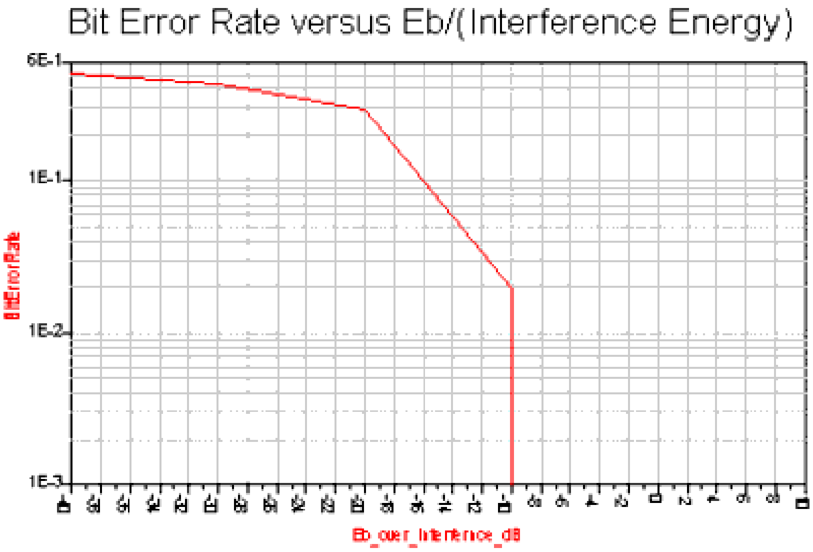








Bit errors are determined by comparing the data bits input to the transmitter to those output from the receiver. An Eb/No plot shows how the BER is degraded by the wideband interference power.



Eqn Error=DataInput-DataOutput

Eqn TotalBits=integrate((DataInput+1)/(DataInput+1))

Eqn TotalErrors=integrate(abs(Error))

Eqn BitErrorRate=TotalErrors/TotalBits

Eqn Power=(mag(GaussianPulseSpectrum)**2/100)/10e6

Eqn TotalPower=integrate(Power)

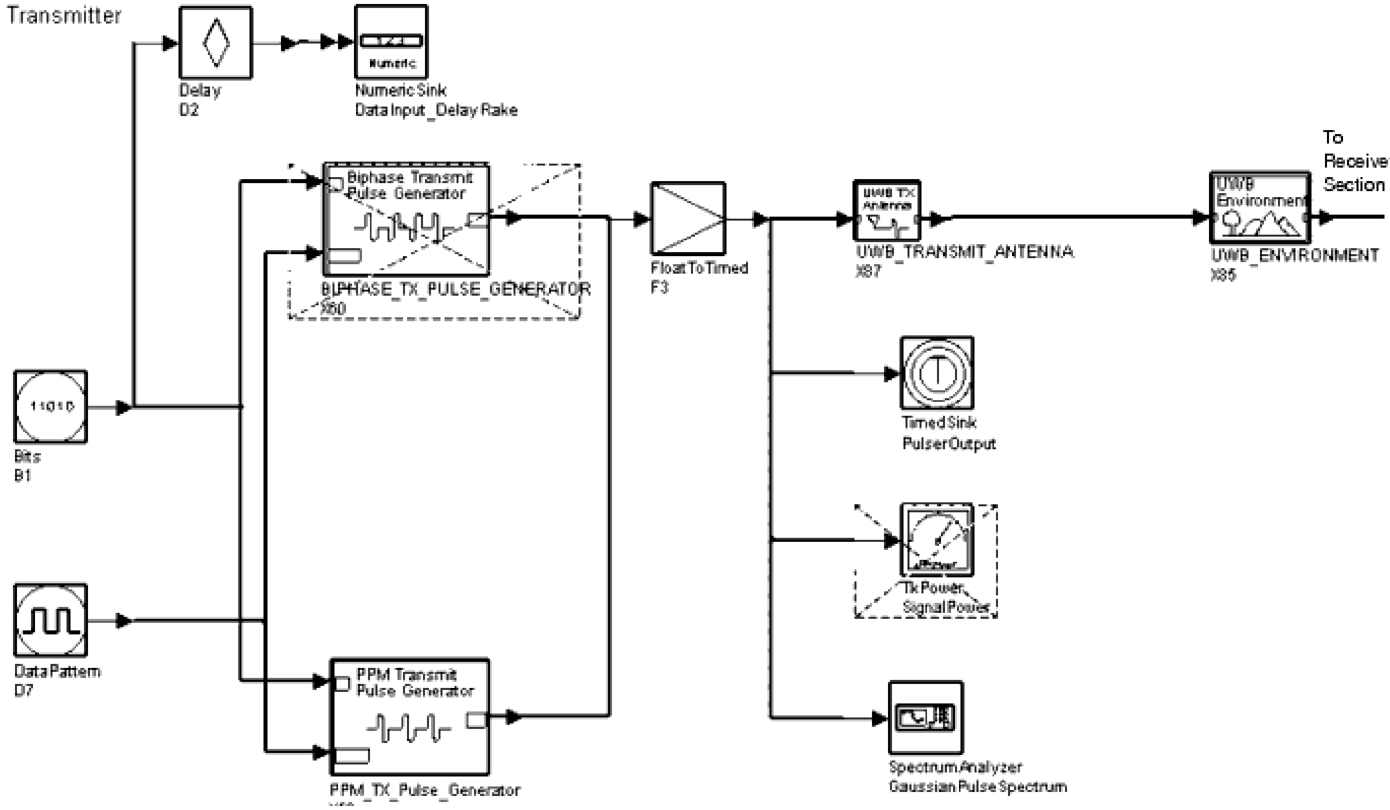
Eqn Eb_over_Interfernce_dB=real(Eb__No)

Interference_dB	TotalBits	TotalErrors	BitErrorRate	ChipsPerBit[]
-10.000	1000.000	0.000	0.000	5.000 / 0.000
0.000	1000.000	0.000	0.000	5.000 / 0.000
10.000	1000.000	19.500	0.019	5.000 / 0.000
20.000	100.000	29.000	0.290	5.000 / 0.000
30.000	100.000	44.000	0.440	5.000 / 0.000
40.000	100.000	51.000	0.510	5.000 / 0.000

BER versus Range

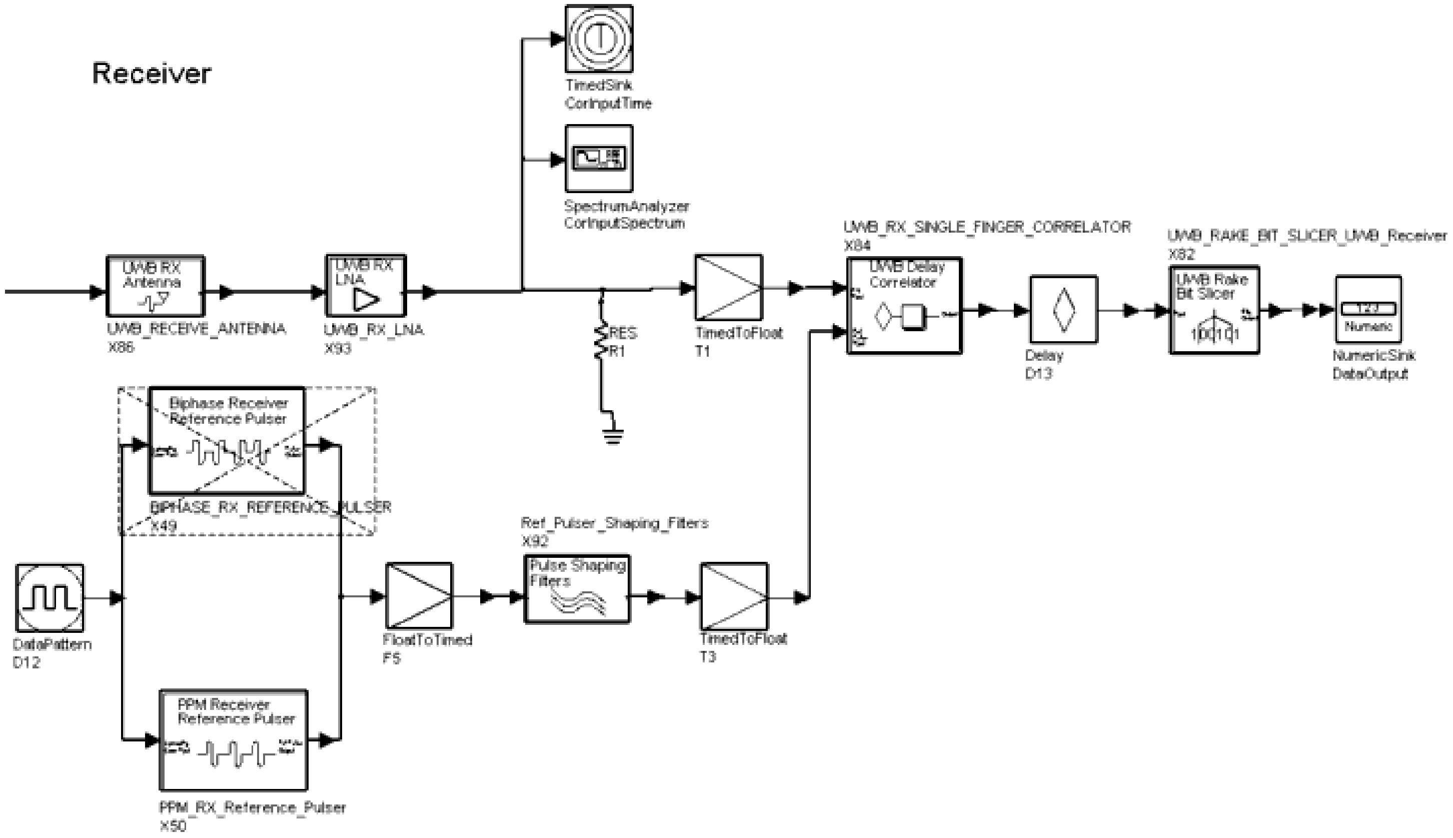
The BER versus Range Test Bench simulates a UWB system with environmental factors. The simulation determines BER performance as a function of distance between the transmit and receive antennas.
Antenna, propagation loss, and multipath models are provided. A single receive path correlates a reference waveform with an individual arrival of the multipath signal. The receiver front-end noise figure and bandwidth are selectable.

The transmitter section of the test bench schematic is shown in BER versus Range Test Bench Schematic, Transmitter Section (_UWB_BER_vs_Range); the receiver section of the test bench schematic is shown in BER versus Range Test Bench Schematic, Receiver Section (_UWB_BER_vs_Range).



BER versus Range Test Bench Schematic, Transmitter Section

(_UWB_BER_vs_Range)



BER versus Range Test Bench Schematic, Receiver Section

BER versus Range Test Bench Schematic, Receiver Section

(_UWB_BER_vs_Range)
Pulse shapes, as well as pulse rates, amplitudes, and widths can be selected.To choose pulse position or bi-phase modulation, enable the PPM or bi-phase pulse generator component; deselect the unused component. For details, refer to *BIPHASE_TX_PULSE_GENERATOR* (dguwb) or *PPM_TX_Pulse_Generator.* (dguwb)

A pseudorandom code is used to spread transmit data. The UWB_TRANSMIT_ANTENNA transmit filter represents the effects of transmit chain and antenna. For component details, refer to *UWB_TRANSMIT_ANTENNA.* (dguwb)

To simulate a UWB system that is compliant with FCC regulations for indoor communications, the transmitter pulse energy can be adjusted to produce a maximum average spectral power density of -41.3 dB/MHz in the 3.1 to 10.6 GHz band (-41.3 dBm/MHz is the total power radiated from an isotropic antenna). The pulse energy required to achieve the appropriate level can be determined using the Modulated Transmit Spectrum.

UWB_ENVIRONMENT contains the propagation loss and multi-path models. For component details, refer to *UWB_ENVIRONMENT.* (dguwb)

UWB_RECEIVE_ANTENNA consists of an SBlock component that reads a file of S-parameters representing the RF frontend and antenna of the receiver. The receive antenna also contains a loss component to allow scale of the overall loss of the antenna. For component details, refer to *UWB_RECEIVE_ANTENNA.* (dguwb)

UWB_RX_LNA is used to set the receiver low noise amplifier noise figure and bandwidth to simulate receiver RF frontend performance. For component details, refer to *UWB_RX_LNA.* (dguwb)

For bi-phase modulation, the BIPHASE_RX_REFERENCE_PULSER component outputs the same waveform as BIPHASE_TX_PULSE_GENERATOR when data input is all 1s. The output represents a bi-phase modulated UWB waveform. Input data bits are spread using a spreading code. For component details, refer to *BIPHASE_RX_REFERENCE_PULSER.* (dguwb)

For pulse position modulation, the PPM_RX_Reference_Pulser component outputs the same waveform as PPM_TX_Pulse_Generator when the data input is 1 with the inverse of the output when the data input is 0. The output represents a pulse position modulated UWB waveform. The input data bits are spread using a spreading code. For component details, refer to *PPM_RX_Reference_Pulser.* (dguwb)

The reference pulser shaping filters apply the same filtering to the reference pulse as is applied to the transmit signal by the transmitter and receiver. For component details, refer to *Ref_Pulser_Shaping_Filters.* (dguwb)

The correlator multiplies the receive signal with a reference signal and integrates the results over a period of time. The integrator in the correlator integrates the multiplier output signal over the period of ChipInterval×ChipsPerBit. It resets the integrator value to

0 and restarts the integration. For component details, refer to *UWB_RX_SINGLE_FINGER_CORRELATOR.* (dguwb)

UWB_RAKE_BIT_SLICER_UWB_Receiver serves as a bit slicer for use with a rake receiver. For component details, refer to *UWB_RAKE_BIT_SLICER_UWB_Receiver.* (dguwb)

The StopBits variable determines the number of bits to be collected by the DataOutput sink; this controls the run time of the simulation. To speed simulations that sweep the Range (distance from transmitter to receiver parameter), StopBits can be defined in a VAR equation using the piecewise linear function, so the number of bits simulated is 10 times the estimated BER at each Range. For large Range values, the BER will be higher and fewer bits will be required to determine the BER; for short Range values, simulation of more bits will be required. The user can rely on information from previously completed simulations to configure the piecewise linear function. If the SpectrumAnalyzer and TimedSink components are to be active during a simulation sweep, the user can also use a piecewise line function to optimize the TimeStop variable for the sweep. This allows collection of enough data at points of interest without producing excessively large datasets.

BER versus Range Test Bench Design Parameters

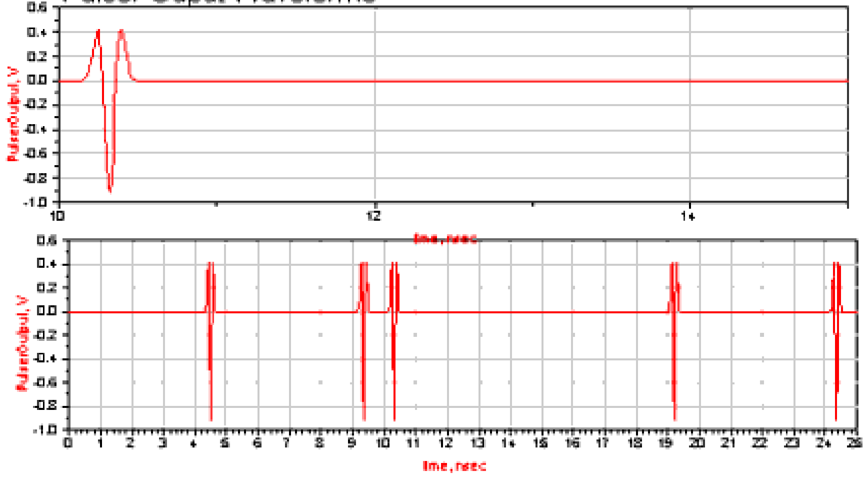
Name	**Description**
UWB Configuration Parameters.	
ChipInterval	Time between pulses. For PPM this is the nominal time between pulses.
ChipsPerBit	Number of pulses transmitted for each bit.
DitherBits	Number of bits used to determine the dither position of a PPM pulse within each pulse interval. There are 2 ^{DitherBits} possible pulse positions within each PPM pulse interval. (For pulse position modulation.)
TStepsPerPulseWidth	Approximate number of simulation time steps in one pulsewidth interval.
TStepsPerDither	Determines the number of simulation time steps between each possible position. Not used for bi-phase modulation simulations, but calculation is used to determine TStep.
DithersPerPPMBitOffset	Number of dither positions offset between a value of 1 and 0 in a PPM modulated code. The length is given in the number of dither intervals in the offset time. This makes the offset between a 0 and 1 an integer number of possible pulse positions. Not used for bi-phase modulation simulations, but calculation is used to determine TStep.
TStepsPerPPMBitOffset	Number of TStep offset between a value of 1 and 0 in a PPM modulated code. The length is given in the number of TStep long intervals in the offset time. Not used for bi-phase modulation simulations, but calculation is used to determine TStep.
DitherTime	Time between possible dither positions in a PPM pulse interval. Parameters must be selected such that DitherTime is greater than TStep or a divide by zero error will occur. Not used for Biphase modulation simulations, but calculation is used to determine TStep.
TStep	Time step of the simulation. It is an integer division of the ChipInterval. It is approximately PulseWidth/TStepsPerPulseWidth.
PulseWidth	1/2 amplitude pulsewidth of a Gaussian monopulse output. For the Gaussian

	derivative pulse shapes, this is the 1/2 amplitude pulsewidth of the Gaussian monopulse from which it is derived.
PulseEnergy_joule	Energy in Joules of a single pulse output from the pulse generator.
DoubletSeparation	Time between the positive and negative peaks of the waveform when doublet pulse is used. (Used when doublet waveform is selected.)
MaxFingerDelay	Maximum delay applied to a correlator finger in the rake receiver.
NoiseFigure_dB	Noise figure of receiver frontend.
RXNoiseBandwidth	Noise passband of the receiver frontend.
FilterDelay	Delay applied to the integrator reset and bit slicer to allow for signal delay through filters.
gamma	Power of distance from the source at which signal amplitude decays. For an isotropic antenna in free space, gamma equals 2.
Range	Distance from transmitter to receiver.
RangePower	A base distance is multiplied by $2^{RangePower}$ in simulation sweeps of distance.
Multi-Path Parameters	
DelayTime1, ... , DelayTime6	Time of arrival of each multipath component relative to the first arrival time.
Delay1, ... , Delay6	Number of TSteps of delay applied to each series delay in the multipath component. This is calculated such that round off errors do not accumulate.
Mag0, ... , Mag5	Relative magnitude of each multi-path arrival.
SumDelay1, ... , SumDelay6	Calculates the series delays applied by the multipath component in a way that round-off errors do not accumulate.

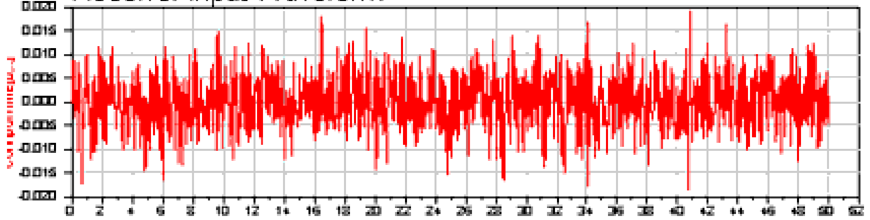
Simulation Results

The Data Display window shows the transmit signal and the receiver signal inputs. Input data bits and the receiver outputs are shown.

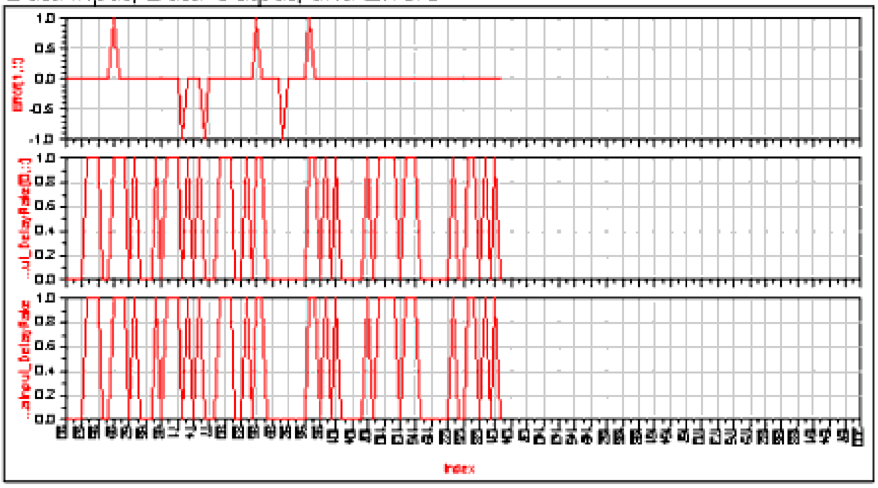
Pulser Ouput Waveforms



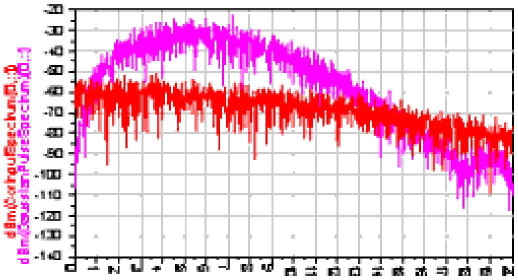
Receiver Input Waveform



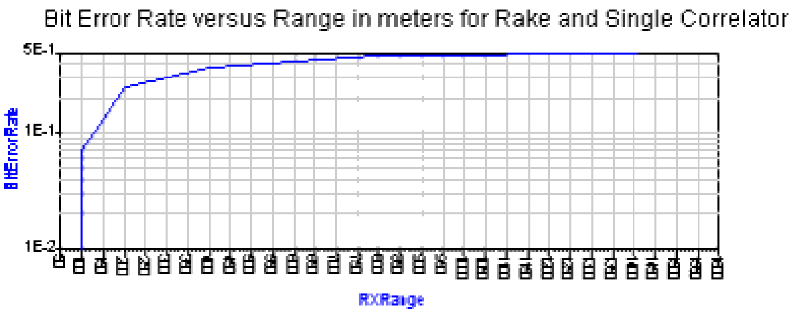
Data Input, Data Output, and Errors

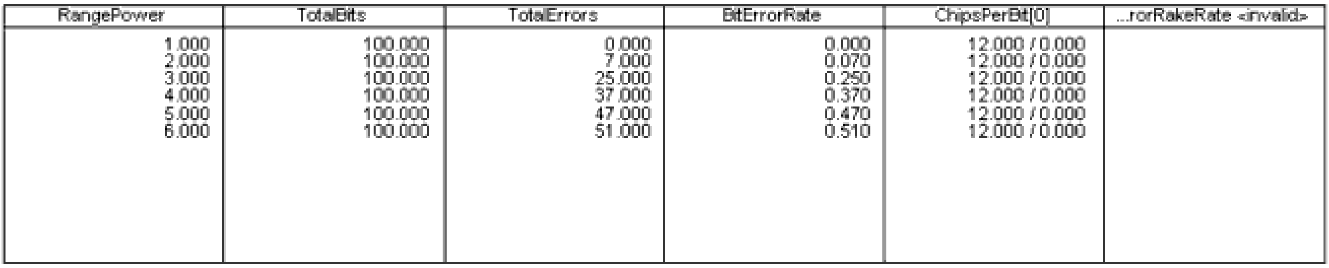


Pulser Output Spectrum and Receiver Input Spectrum

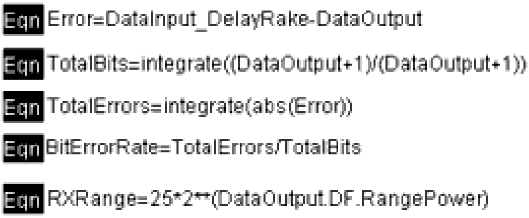


The Range plot shows the BER of both receivers as a function of range.





RangePower	TotalBits	TotalErrors	BitErrorRate	ChipsPerBit[0]	...rorRakeRate <invalid>
1.000	100.000	0.000	0.000	12.000 / 0.000	
2.000	100.000	7.000	0.070	12.000 / 0.000	
3.000	100.000	25.000	0.250	12.000 / 0.000	
4.000	100.000	37.000	0.370	12.000 / 0.000	
5.000	100.000	47.000	0.470	12.000 / 0.000	
6.000	100.000	51.000	0.510	12.000 / 0.000	



Eqn Error=DataInput_DelayRake-DataOutput

Eqn TotalBits=integrate((DataOutput+1)/(DataOutput+1))

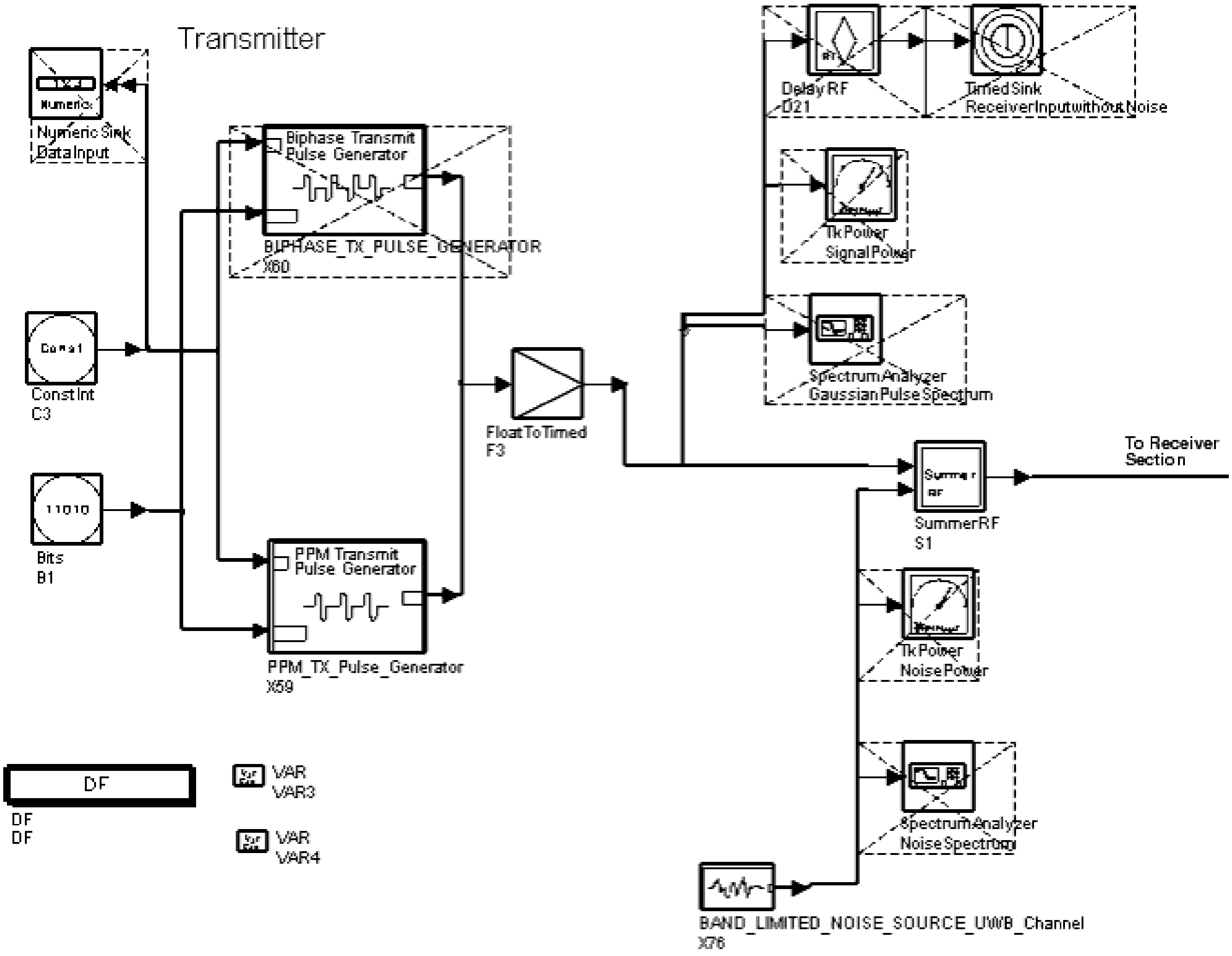
Eqn TotalErrors=integrate(abs(Error))

Eqn BitErrorRate=TotalErrors/TotalBits

Eqn RXRange=25*2**(DataOutput.DF.RangePower)

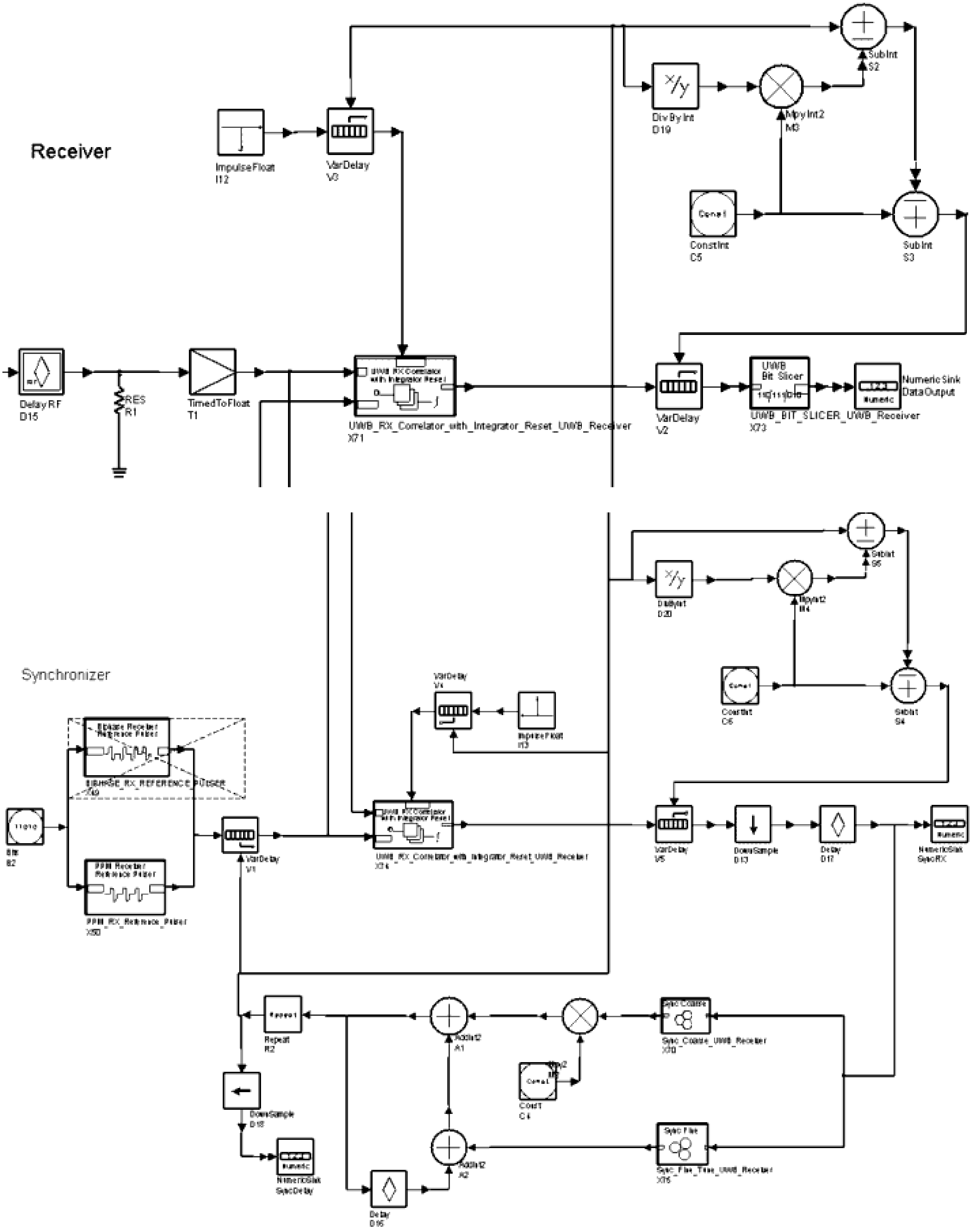
Synchronization

The Synchronization Test Bench simulates a UWB receiver obtaining synchronization with a received signal.



Synchronization Test Bench

(_UWB_Synchronization_Bench)



Pulse shapes, as well as pulse rates, amplitudes, and widths can be selected.To choose pulse position or bi-phase modulation, enable the PPM or bi-phase pulse generator component; deselect the unused component. For details, refer to *BIPHASE_TX_PULSE_GENERATOR* (dguwb) or *PPM_TX_Pulse_Generator.* (dguwb)

A pseudorandom code is used to spread the transmit data. Band-limited noise is added to the transmit signal before it enters the receiver (for component details, refer to *BAND_LIMITED_NOISE_SOURCE_UWB_Channel* (dguwb)).

For bi-phase modulation, the BIPHASE_RX_REFERENCE_PULSER component outputs the same waveform as BIPHASE_TX_PULSE_GENERATOR when the data input is all 1s. The output represents a bi-phase modulated UWB waveform. The input data bits are spread using a spreading code. (For component details, refer to *BIPHASE_RX_REFERENCE_PULSER.* (dguwb))

For pulse position modulation, the PPM_RX_Reference_Pulser component outputs the same waveform as PPM_TX_Pulse_Generator when the data input is 1 with the inverse of the output when the data input is 0. (For component details, refer to *PPM_RX_Reference_Pulser.* (dguwb)) The output represents a pulse position modulated UWB waveform. The input data bits are spread using a spreading code.

A delay is applied to the received signal to cause the receiver to be out of synchronization with the receiver. When the reference signal to the correlator is not synchronized with the receive signal, distribution of the correlator output will be centered about 0V. When the correlator is synchronized, distribution of the correlator output signals will be centered about a positive offset voltage. The synchronization algorithm applied in this simulation adjusts the correlator timing to maximize the amplitude to the correlator output signal. The correlator output is used as feedback for coarse and fine synchronization algorithms that adjust the delay applied to the de-spreading code; this brings it into alignment with the spreading code of the receive signal. These simulations demonstrate the ability of a correlator to obtain and maintain synchronization under user-defined conditions of signal-to-noise ratio and correlator integration time.

For typical simulation, the delay applied to the received signal will be a few spreading code positions. This delay is determined by the CodeOffset variable; it could be set to a very large value, but DefaultNumericStop must also be set to a large value in order for the correlator to achieve synchronization during simulation (this could require excessive time for the simulation to complete). This simulation is designed to focus on the critical time period when the correlator is a few code positions out of synchronization through the time that synchronization is achieved.

This simulation uses two correlators: the *synchronization* correlator provides synchronization feedback; the *receive* correlator is used to decode data after synchronization is achieved. Timing is controlled by a feedback loop around the synchronization correlator. The data transmitted is all 1s. The feedback loop adjusts the timing of the receiver reference pulse train as well as the timing of the correlator integration interval and the bit slicer. For correlator component details, refer to *UWB_RX_Correlator_with_Integrator_Reset_UWB_Receiver.* (dguwb)

The coarse synchronization block measures the average amplitude of negative polarity outputs from the correlator. If the ratio of current correlator output value to the absolute value of the average negative correlator output value is less than the value of RelSyncAmplitude, the coarse synchronization algorithm increases the delay of the correlator reference signal by the value of CoarseTimeStep. When the ratio is greater than RelSyncAmplitude, no coarse adjustment in synchronizer timing is applied. For component details, refer to *Sync_Coarse_UWB_Receiver* (dguwb).

The fine synchronization block measures the average correlator output over a given time interval. After each averaging time interval the fine synchronization algorithm adjusts the correlator delay by plus or minus one TStep (typically, approximately 1/10th a pulse width). If the most recent averaged output value is greater than the previous value, the polarity of the delay adjustment is the same as the previous adjustment. If the most recent value recorded averaged output value is less than the previous value, the polarity of the delay adjustment is the opposite of the previous adjustment. This algorithm will optimally align the correlator reference signal with the receive signal after the coarse synchronization is achieved. For component details, refer to *Sync_Fine_Tune_UWB_Receiver* (dguwb).

For this algorithm to achieve optimal synchronization depends on the shape of the pulse waveform. For the Gaussian monopulse waveform, the synchronization algorithm optimally aligns the received and reference waveforms. However, if the pulse shape has several oscillations (such as Gaussian Second Derivative pulse shape), there will be local correlator output maxima at offsets from optimal synchronization. There will be a range of signal-to-noise ratios for which this algorithm may synchronize one of the local maxima rather than on the optimal alignment. Simulations showing synchronization on non-optimal local maxima indicate the need to implement a higher level search algorithm for the optimal synchronization offset within a time window about the synchronization point achieved by the low-level algorithm of this simulation.

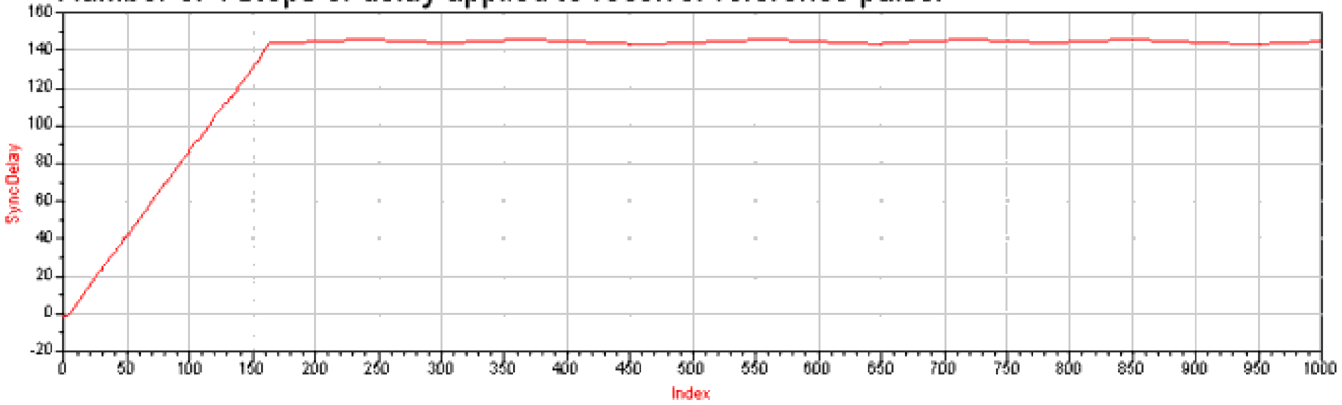
Synchronization Test Bench Design Parameters

Name	Description
ChipInterval	Time between pulses. For PPM this is the nominal time between pulses.
ChipsPerBit	Number of pulses transmitted for each bit. This also determines the number of pulses integrated by the data correlator for each bit.
DitherBits	Number of bits used to determine the dither position of a PPM pulse within each pulse interval. There are $2^{DitherBits}$ possible pulse positions within each PPM pulse interval. (For pulse position modulation.)
TStepsPerPulseWidth	Approximate number of simulation time steps in one pulsewidth interval.
TStepsPerDither	Number of simulation time steps between each possible pulse position with PPM modulation. This value is an integer.
DithersPerPPMBitOffset	Length of offset between a value of 1 and 0 in a PPM modulated code. Length is given in the number of dither interval long intervals in the offset time. This make the offset between 0 and 1 an integer number of possible pulse positions. (For pulse position modulation.)
TStepsPerPPMBitOffset	Length of offset between a value of 1 and 0 in a PPM modulated code. Length is given in number of TStep long intervals in the offset time. This make the offset between 0 and 1 an integer number of possible pulse positions. (For pulse position modulation.)
DitherTime	Time between possible dither positions in a PPM pulse interval. (For pulse position modulation.)
TStep	Time step of the simulation. An integer division of ChipInterval, it is approximately PulseWidth/TStepsPerPulseWidth.
PulseWidth	1/2 amplitude pulsewidth of a Gaussian monopulse output. For the Gaussian derivative pulse shapes, this is the 1/2 amplitude pulsewidth of the Gaussian monopulse from which it is derived.
DoubletSeparation	Time between positive and negative peaks of the waveform when doublet pulse is used. (Used when doublet waveform is selected.)
PulseEnergy_joule	Total energy in Joules of a single pulse output from the PULSE_SHAPE_GENERATOR. For the Gaussian doublet pulse shape it is the energy of each individual polarity pulse in the doublet.
NoiseBandWidthRatio	Determines the Bandwidth of the interfering noise source. The bandwidth is equal to 1/(TStep×NoiseBandWidthRatio).
NoisePower_dBm	Power of interfering noise source in dBm.
SyncIntegTime	Integration time of synchronization correlator.
CodeOffset	Delay applied to the transmit signal is give by CodeOffset×ChipsPerBit×ChipInterval. This determines the length of the time offset that must be scanned by the receiver to obtain synchronization.
RelSyncAmplitude	If the ratio of current correlator output to the absolute average negative correlator output is less than the RelSyncAmplitude value, the coarse synchronization algorithm increases the delay of the correlator reference signal by the value of CoarseTimeStep.
CoarseSyncStep	Size of time adjustment applied to correlator timing when coarse synchronization block determines that correlator is not synchronized.
NumFineSyncAvg	Number of samples averaged for each fine sync iteration.
StopBits	Number of bits that the DataOutput NumericSink will receive during simulation. This determines the overall length of the simulation.

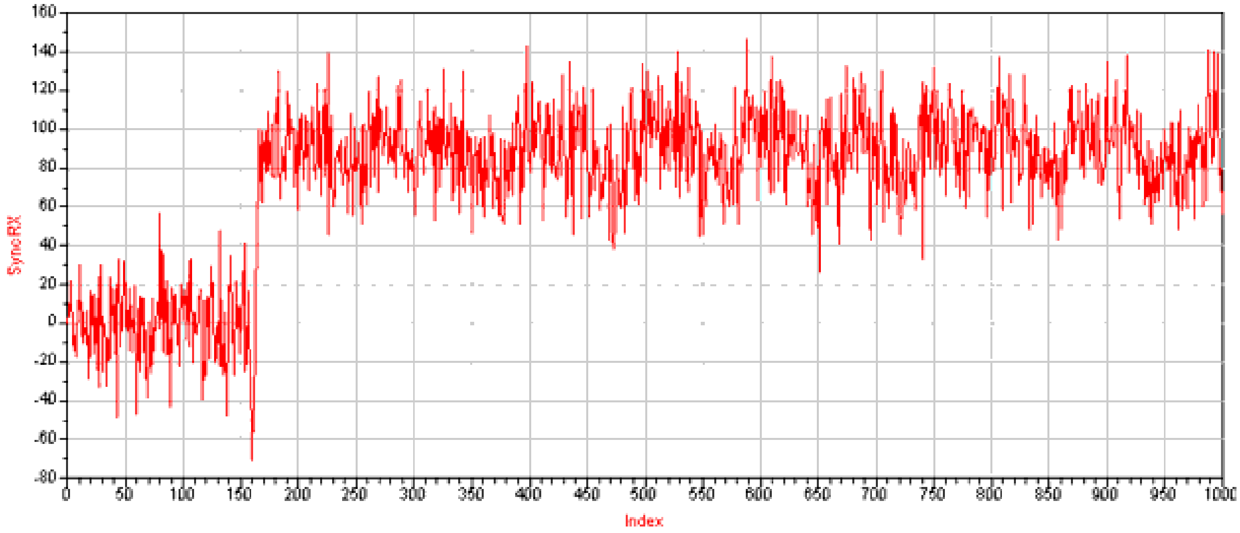
Simulation Results

The Data Display window shows the output value of the synchronization correlator as a function of time; when synchronization is obtained, this value will be greater than when the receiver is unsynchronized. The delay value applied to the receive reference signal as a function of time is also shown; when synchronization is achieved changes in this value should be limited to small oscillation due to the fine synchronization loop.

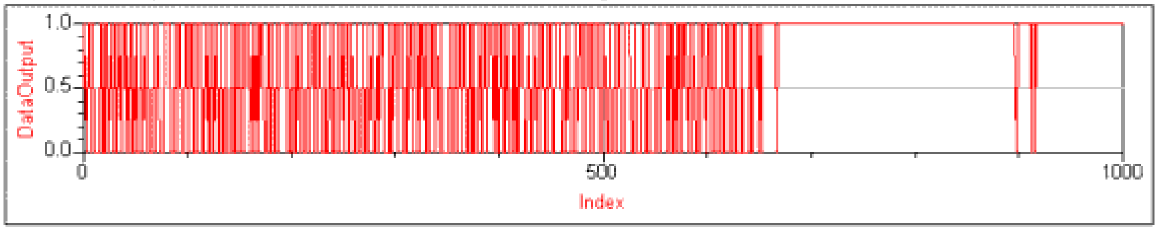
Number of TSteps of delay applied to receiver reference pulser



Synchronization Correlator Output Signal Level



Data Output becomes all 1s when synchronized



TotalDelay is number of TSteps of delay applied to receive signal.
maxSynDelay is the maximum number of TSteps of delay applied to the receiver despreading code in order to achieve synchronization.

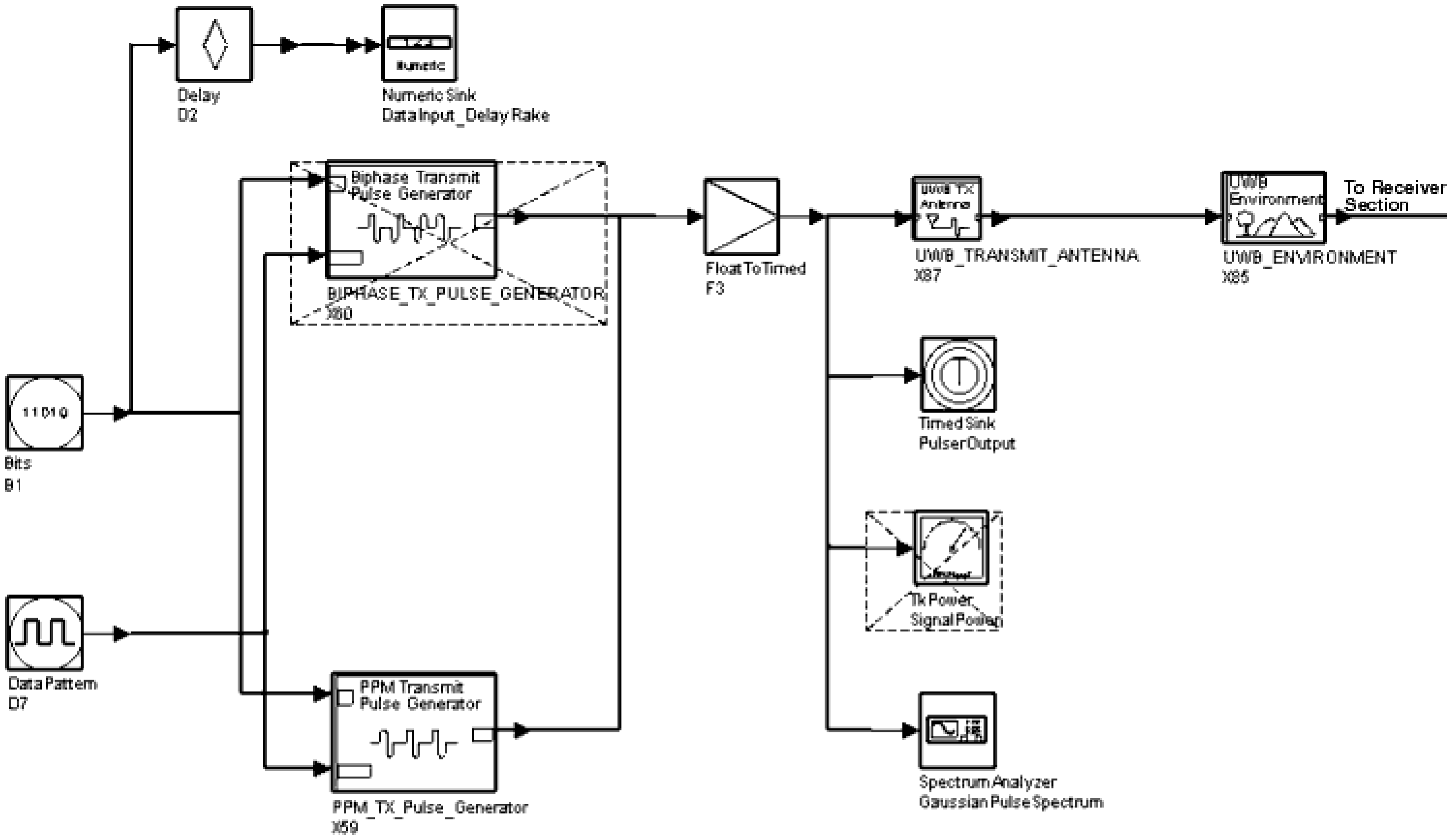
time	TotalDelay	maxSynDelay
0.0000 sec	144.000 / 0.000	146.000

Eqn maxSynDelay=max(SyncDelay)

Eqn TotalDelay=CodeOffset*ChipInterval/TStep

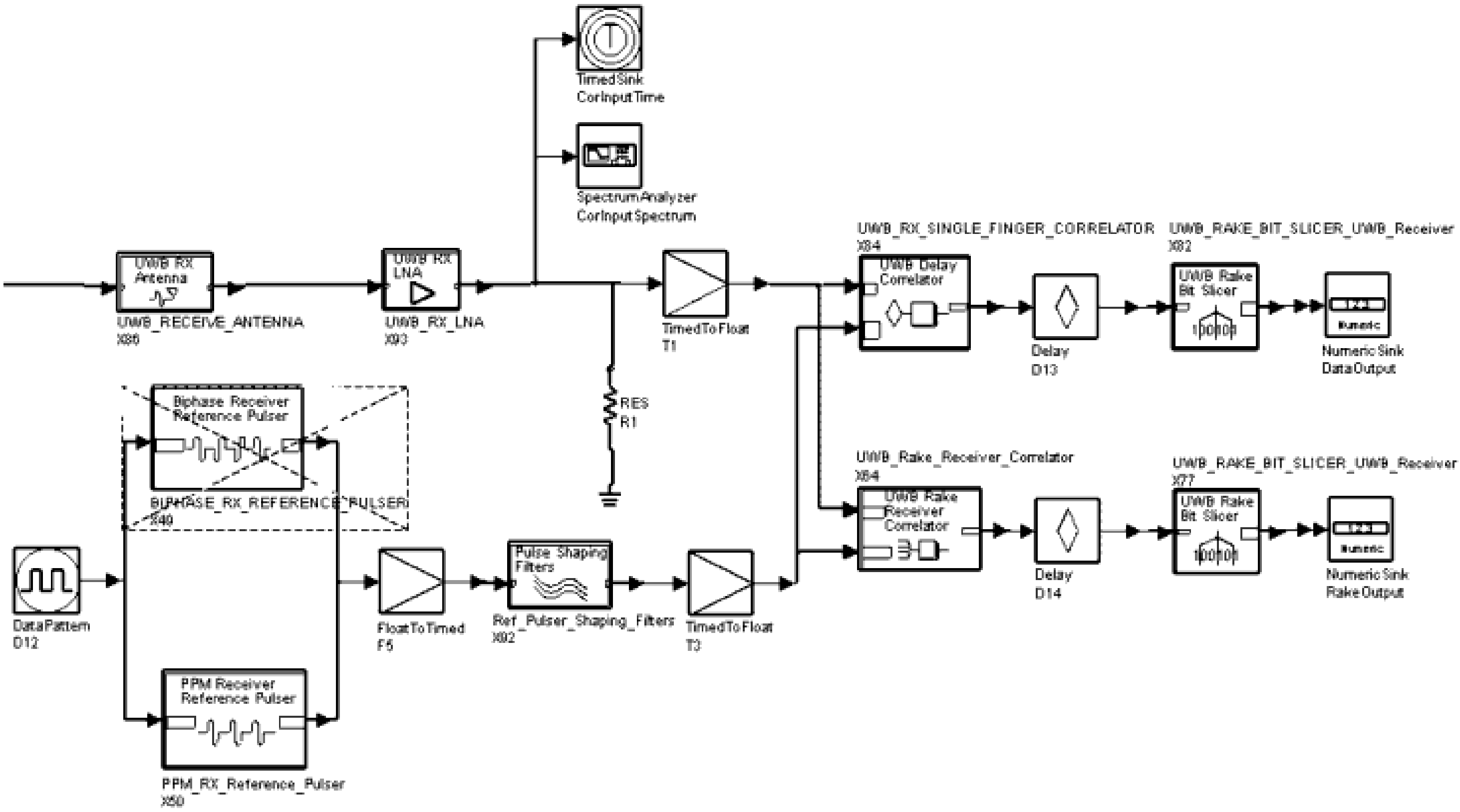
Rake Receiver

The Rake Receiver Test Bench simulates a rake receiver in a multi-path environment.



Rake Receiver Test Bench Schematic, Transmitter Section

(*_UWB_Rake_Receiver*)



Rake Receiver Test Bench Schematic, Receiver Section

(*_UWB_Rake_Receiver*)
To choose pulse position or bi-phase modulation, enable the PPM or bi-phase pulse generator component; deselect the unused component. Pulse shapes, as well as pulse rates, amplitudes, and widths can be selected. For details, refer to *BIPHASE_TX_PULSE_GENERATOR* (dguwb) or *PPM_TX_Pulse_Generator.* (dguwb)

The transmit filter represents the effects of transmitter front-end and antenna; for details, refer to *UWB_TRANSMIT_ANTENNA.* (dguwb)

A pseudorandom bit sequence is used as the spreading code for UWB modulation. For bi-phase modulation, one bit from the spreading code is consumed per transmitted pulse. The spreading code repeat time is equal to the pulse interval multiplied by the pseudorandom bit sequence length. For pulse position modulation, the value of DitherBits determines the number of spreading code bits consumed for each transmitted pulse. The spreading code repeat time for pulse position modulation is the pulse interval multiplied by the spreading code length divided by the number of value of DitherBits. The spreading code repeat time determines the spacing of spectral lines in the modulated transmit spectrum.
UWB_ENVIRONMENT contains the propagation loss and multi-path models. For component details, refer to *UWB_ENVIRONMENT.* (dguwb)

UWB_RECEIVE_ANTENNA consists of an SBlock component that reads a file of S-parameters representing the RF frontend and antenna of the receiver. The receive antenna also contains a loss component to allow scale of the overall loss of the antenna. For component details, refer to *UWB_RECEIVE_ANTENNA.* (dguwb)

UWB_RX_LNA is used to set the receiver low noise amplifier noise figure and bandwidth to simulate receiver RF frontend performance. For component details, refer to *UWB_RX_LNA.* (dguwb)

For bi-phase modulation, the BIPHASE_RX_REFERENCE_PULSER component outputs the same waveform as BIPHASE_TX_PULSE_GENERATOR when data input is all 1s. The output represents a bi-phase modulated UWB waveform. Input data bits are spread using a spreading code. For component details, refer to *BIPHASE_RX_REFERENCE_PULSER.* (dguwb)

For pulse position modulation, the PPM_RX_Reference_Pulser component outputs the same waveform as PPM_TX_Pulse_Generator when the data input is 1 with the inverse of the output when the data input is 0. The output represents a pulse position modulated UWB waveform. The input data bits are spread using a spreading code. For component details, refer to *PPM_RX_Reference_Pulser.* (dguwb)

The reference pulser shaping filters apply the same filtering to the reference pulse as is applied to the transmit signal by the transmitter and receiver. For component details, refer to *Ref_Pulser_Shaping_Filters.* (dguwb)

An SBlock component represents the response of the transmitter circuitry and antenna. A propagation loss model is applied with variable attenuation rates to allow modeling of different terrains and antenna configurations. Multipath reflected signals are applied, creating multiple arrivals for each pulse at varying amplitudes and delays. The rake receiver uses four fingers by default and can be expanded as necessary. Each finger correlates the receiver reference signal with an individual arrival of the multi-path signal. The correlator outputs of each finger are scaled relative to its signal-to-noise ratio. The scaled outputs of all fingers are summed. A single correlator receiver is also included to allow comparison of a single correlator with the multi-finger receiver performance.

To simulate a UWB system that is compliant with the FCC regulations for indoor communications, the transmitter pulse energy can be adjusted to produce a maximum average spectral power density of -41.3 dB/MHz in the 3.1 GHz to 10.6 GHz band. The pulse energy required achieve the proper level can be determined using the Modulated Transmit Spectrum Test Bench design (*_UWB_Modulated_Transmit_Spectrum)* . -41.3 dBm/MHz is the total power radiated from an isotropic antenna.

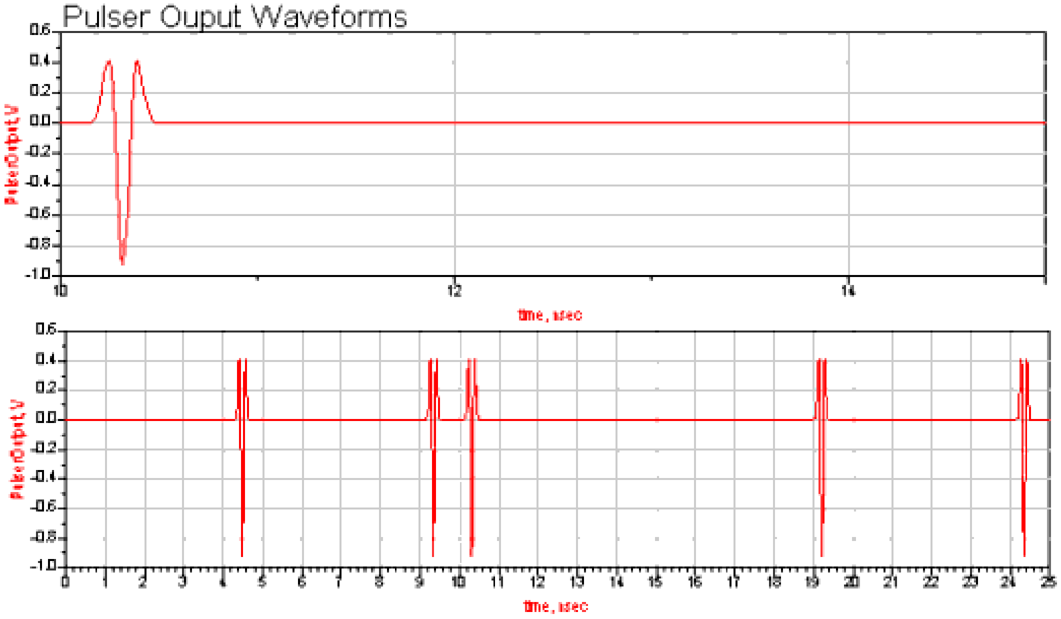
The StopBits variable determines the number of bits to be collected by the DataOutput sink. This controls the run time of the simulation. To speed simulations that sweep the interfering noise power, StopBits can be defined in a VAR equation using the piecewise-linear function so the number of bits simulated is 10 times the estimated BER at that noise power. For high noise power levels, the BER will be higher, and fewer bits are required to accurately determine the BER. At lower noise power levels, simulation of more bits will be required. The user can rely on information from previously completed simulations to configure the piecewise-linear function. If the SpectrumAnalyzer and TimedSink components are to be active during a simulation sweep, the user can also use a piecewise-linear function to optimize the TimeStop variable for the sweep. TimeStop determines the amount of data to be collected by these components.

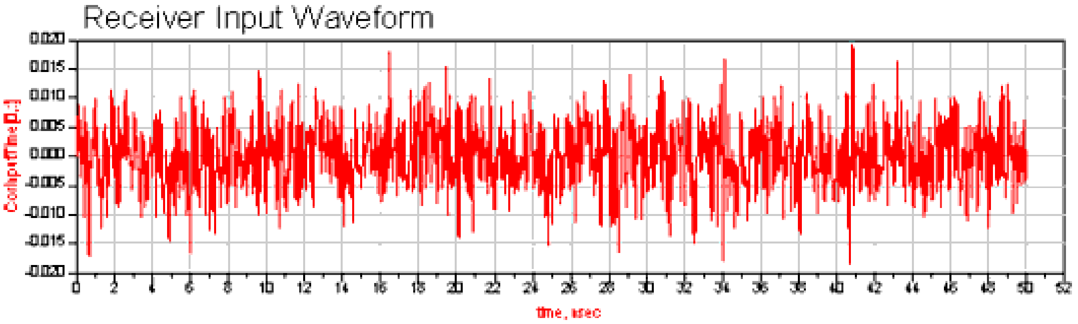
Rake Receiver Test Bench Design Parameters

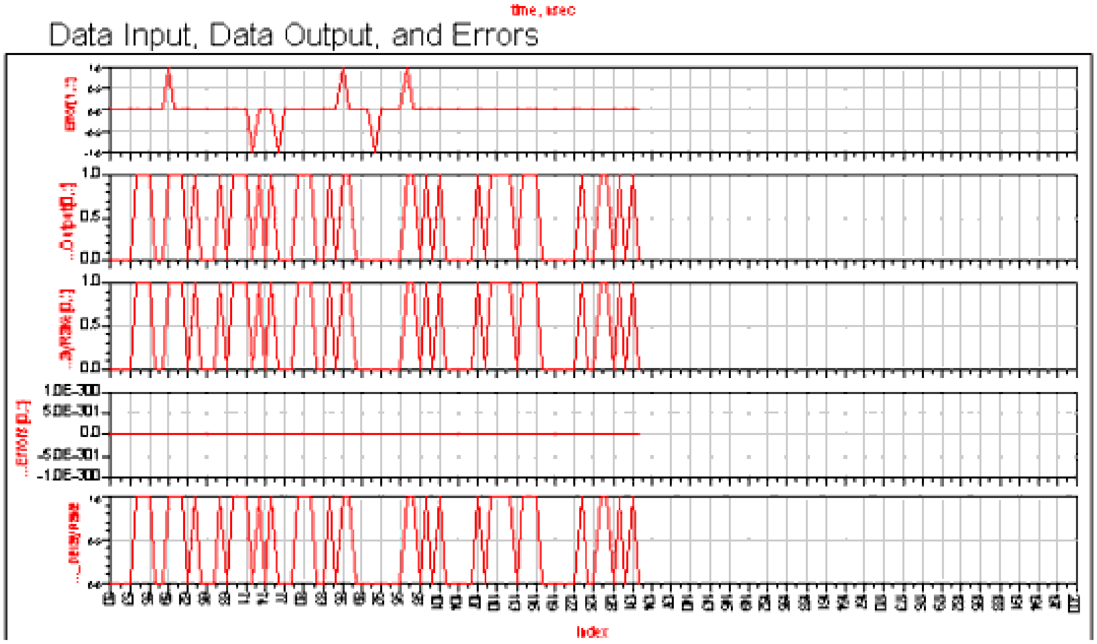
Name	Description
StopBits	Number of bits simulated.
TimeStop	Length of time SpectrumAnalyzer components collect data.
ChipInterval	Time between pulses. For PPM this is the nominal time between pulses.
ChipsPerBit	Number of pulses transmitted for each bit.
DitherBits	Number of bits used to determine the dither position of a PPM pulse within each pulse interval. There are $2^{DitherBits}$ possible pulse positions within each PPM pulse interval. (For pulse position modulation.)
TStepsPerDither	Determines the number of simulation time steps between each possible position.
DithersPerPPMBitOffset	Length of offset between a value of 1 and 0 in a PPM modulated code. Length is given in the number of dither interval long intervals in the offset time. This makes the offset between 0 and 1 an integer number of possible pulse positions. (For pulse position modulation.)
TStepsPerPPMBitOffset	Length of offset between a value of 1 and 0 in a PPM modulated code. The length is given in the number of TStep long intervals in the offset time. This make the offset between 0 and 1 an integer number of possible pulse positions. (For pulse position modulation.)
DitherTime	Time between possible dither positions in a PPM pulse interval. (For pulse position modulation.)
TStep	Time step of the simulation. It is an integer division of the ChipInterval. It is approximately PulseWidth/TStepsPerPulseWidth.
PulseWidth	1/2 amplitude pulsewidth of a Gaussian monopulse output. For the Gaussian derivative pulse shapes, this is the 1/2 amplitude pulsewidth of the Gaussian monopulse from which it is derived.
DoubletSeparation	Time between the positive and negative peaks of the waveform when doublet pulse is used. (Used when doublet waveform is selected.)
NoiseFigure	Noise figure of the receiver frontend.
RXNoiseBandwidth	Noise passband of the receiver frontend.
PulseEnergy_joule	Energy in Joules of single pulse output from the pulse generator.
MaxFingerDelay	Maximum delay applied to a correlator finger in the rake receiver.
FilterDelay	Delay applied to the integrator reset and bit slicer to allow for signal delay through filters.
gamma	Power of distance from the source at which signal amplitude decays. For an isotropic antenna in free space, gamma equals 2.
Range	Distance from the transmitter to the receiver.
RangePower	A base distance is multiplied by $2^{RangePower}$ in simulation sweeps of distance.
DelayTime1, ... , DelayTime6	Time of arrival of each multipath component relative to the first arrival time.
Delay1, ... , Delay6	Number of TSteps of delay applied to each series delay in the multipath component. This is calculated such that round off errors do not accumulate.
Mag0, ... , Mag5	Relative magnitude of each multi-path arrival.

Simulation Results

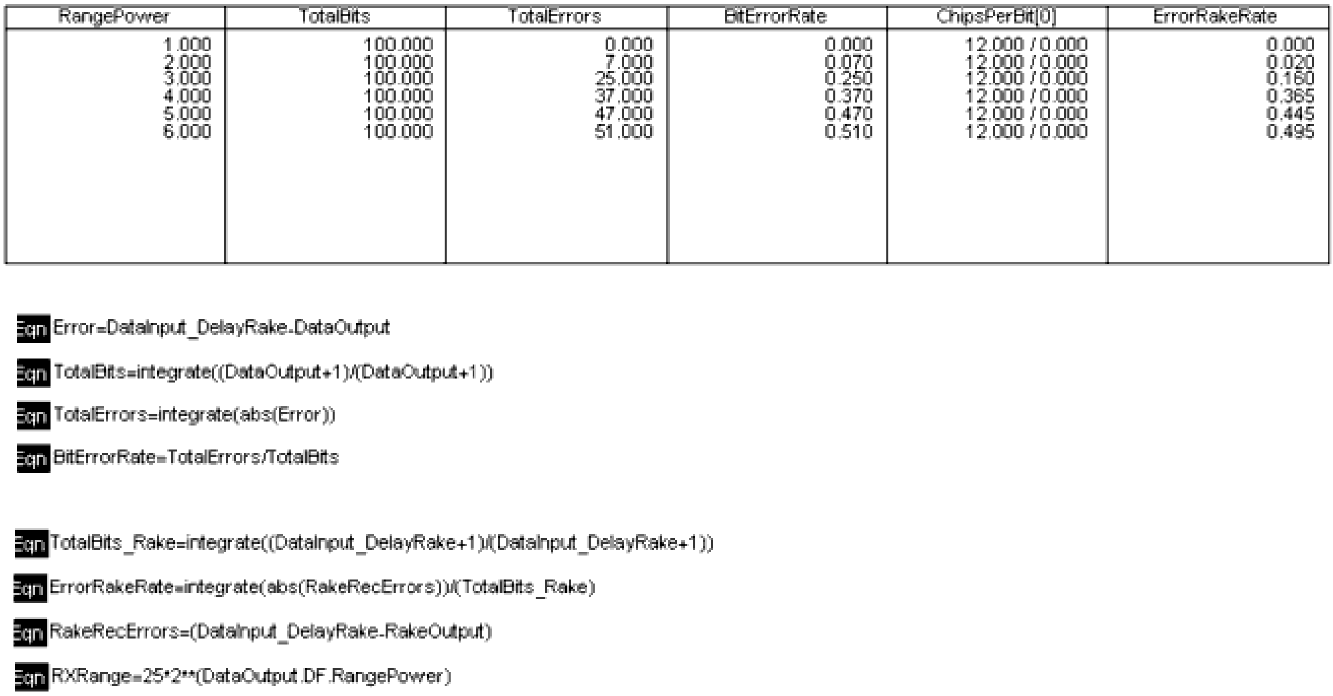
The Data Display window shows the transmit signal and the receiver input signal. Input data bits and the rake and single correlator receiver outputs are shown.







The Eb/No plot shows the bit error rate of both receivers.



RangePower	TotalBits	TotalErrors	BitErrorRate	ChipsPerBit[0]	ErrorRakeRate
1.000	100.000	0.000	0.000	12.000 / 0.000	0.000
2.000	100.000	7.000	0.070	12.000 / 0.000	0.020
3.000	100.000	25.000	0.250	12.000 / 0.000	0.160
4.000	100.000	37.000	0.370	12.000 / 0.000	0.365
5.000	100.000	47.000	0.470	12.000 / 0.000	0.445
6.000	100.000	51.000	0.510	12.000 / 0.000	0.495

```
Eqn Error=DataInput_DelayRake-DataOutput
Eqn TotalBits=integrate((DataOutput+1)/(DataOutput+1))
Eqn TotalErrors=integrate(abs(Error))
Eqn BitErrorRate=TotalErrors/TotalBits

Eqn TotalBits_Rake=integrate((DataInput_DelayRake+1)/(DataInput_DelayRake+1))
Eqn ErrorRakeRate=integrate(abs(RakeRecErrors))/(TotalBits_Rake)
Eqn RakeRecErrors=(DataInput_DelayRake-RakeOutput)
Eqn RXRange=25*2**(DataOutput.DF.RangePower)
```

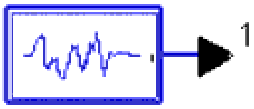
Pulse Mode Test Bench Component Details

Components designed specifically for pulse mode test benches are described in this section.

- *BAND LIMITED NOISE SOURCE UWB Channel* (dguwb)
- *BiPhase Polarity Select UWB Transmitter* (dguwb)
- *BIPHASE RX REFERENCE PULSER* (dguwb)
- *BIPHASE TX PULSE GENERATOR* (dguwb)
- *INTERFERENCE SOURCE 80211a UWB Channel* (dguwb)
- *INTERFERENCE SOURCE 80211B UWB Channel* (dguwb)
- *MultipathDelayBlock UWB Channel* (dguwb)
- *PPM MOD UWB Transmitter* (dguwb)
- *PPM RX Reference Pulser* (dguwb)
- *PPM TX Pulse Generator* (dguwb)
- *PropagationLoss* (dguwb)
- *PULSE SHAPE GENERATOR* (dguwb)
- *PULSE TRIGGER UWB Transmitter* (dguwb)
- *Rake Finger UWB Receiver* (dguwb)
- *Ref Pulser Shaping Filters* (dguwb)
- *Sync Coarse UWB Receiver* (dguwb)
- *Sync Fine Tune UWB Receiver* (dguwb)
- *UWB BIT SLICER UWB Receiver* (dguwb)
- *UWB ENVIRONMENT* (dguwb)
- *UWB Interference Source* (dguwb)
- *UWB RAKE BIT SLICER UWB Receiver* (dguwb)
- *UWB Rake Receiver Correlator* (dguwb)
- *UWB RECEIVE ANTENNA* (dguwb)

- *UWB RX Correlator UWB Receiver* (dguwb)
- *UWB RX Correlator with Integrator Reset UWB Receiver* (dguwb)
- *UWB RX LNA* (dguwb)
- *UWB RX SINGLE FINGER CORRELATOR* (dguwb)
- *UWB TRANSMIT ANTENNA* (dguwb)

BAND_LIMITED_NOISE_SOURCE_UWB_Channel



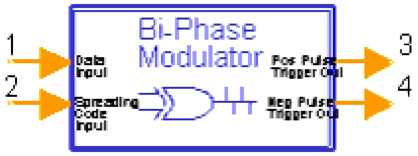
Parameters

Name	Description	Default	Unit	Type	Range
NoiseBandWidthRatio	Integer ratio of the simulation bandwidth to the bandwidth of the noise source. Simulation bandwidth is given by 1/(2×TStep).	7		Integer	>0
NoisePower_dBm	Total rms power from the noise source.	-40		Real	<∞
TStep	Time step of the simulation	10 psec	Time	Real	>0

Inputs

Pin	Name	Description	Signal Type
1	1	Gaussian PDF noise	Timed

BiPhase_Polarity_Select_UWB_Transmitter



This component is used to trigger a positive or negative polarity pulse for bi-phase modulation. The input data bit is exclusive OR-ed with the spreading code. If the result or the exclusive OR is a 1, a pulse on the positive pulse trigger is generated; otherwise, a pulse on the negative pulse trigger is generated.

Parameters

Name	Description	Default	Unit	Type	Range
ChipsPerBit	Number of pulses transmitted for each bit.	1		Integer	>1
ChipInterval	Time between pulses.	1 nsec	Time	Real	>0
TStep	Time step of the simulation.	10 psec	Time	Real	>0

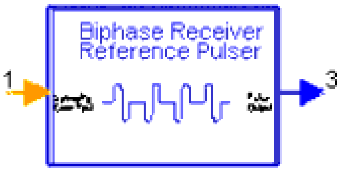
Inputs

Pin	Name	Description	Signal Type
1	Data Input	Data transmitted by UWB system input as binary bits.	Integer
2	Spreading Code Input	Coding sequence used to spread data being transmitted input as binary bits.	Integer

Outputs

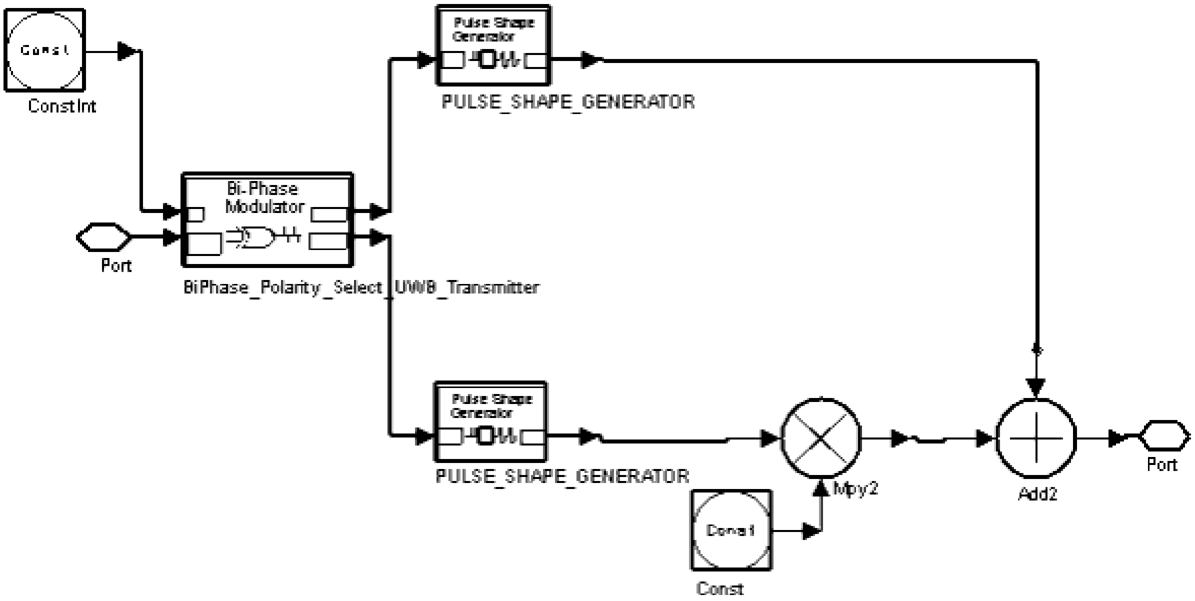
Pin	Name	Description	Signal Type
3	Positive Pulse Trigger	Impulse to trigger generation of a positive polarity pulse.	Float
4	Negative Pulse Trigger	Impulse to trigger generation of a positive polarity pulse.	Float

BIPHASE_RX_REFERENCE_PULSER



This component outputs the same waveform as BIPHASE_TX_PULSE_GENERATOR when the data input is all 1s. The waveform output from this test block represents a bi-phase modulated UWB waveform. The input data bits are spread using a spreading code. The pulse shape output is determined by selecting PULSE_SHAPE_GENERATOR sub-components; for details, refer to *PULSE SHAPE GENERATOR* (dguwb).

Subnetwork



Parameters

Name	Description	Default	Unit	Type	Range
ChipsPerBit	Number of pulses transmitted for each bit.	1		Integer	>1
ChipInterval	Time between pulses.	1 nsec	Time	Real	>0
PulseWidth	Width of output pulse.	250 psec	Time	Real	>0
PulseEnergy_joule	Total energy of the output pulse in Joules.	1e-12		Real	>0
DoubletSeparation	Time between the positive and negative peaks of the waveform.	350 psec	Time	Real	>0
TStep	Time step of the simulation.	10 psec	Time	Real	>0

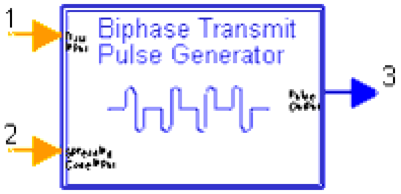
Inputs

Pin	Name	Description	Signal Type
1	Spreading Code Input	Coding sequence used to spread data being transmitted input as binary bits.	Integer

Outputs

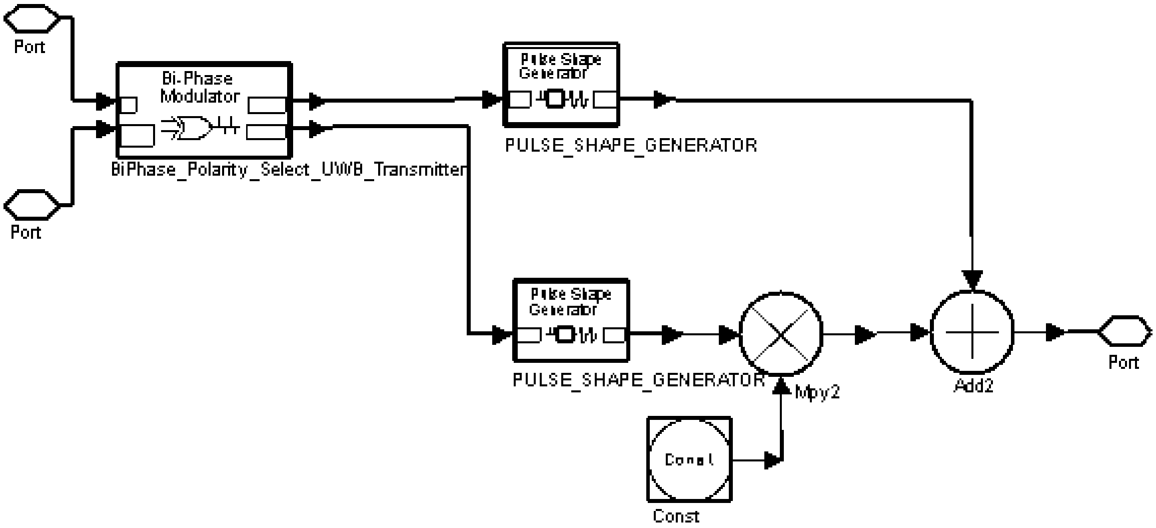
Pin	Name	Description	Signal Type
2	Pulse Output	Outputs pulse with shape determined by which pulse generator subcomponent is selected.	Float

BIPHASE_TX_PULSE_GENERATOR



The BIPHASE_TX_PULSE_GENERATOR output represents a bi-phase modulated UWB waveform. Input data bits are spread using a spreading code. The pulse shape output is determined by selecting sub-components of PULSE_SHAPE_GENERATOR (see *PULSE_SHAPE_GENERATOR* (dguwb)).

Subnetwork



Name	Description	Default	Unit	Type	Range
ChipsPerBit	Number of pulses transmitted for each bit.	1		Integer	>1
ChipInterval	Time between pulses.	1 nsec	Time	Real	>0
PulseWidth	Width of output pulse.	250 psec	Time	Real	>0
PulseEnergy_joule	Total energy of the output pulse in Joules.	1e-12		Real	>0
DoubletSeparation	Time between the positive and negative peaks of the waveform.	350 psec	Time	Real	>0
TStep	Time step of the simulation.	10 psec	Time	Real	>0

Pin	Name	Description	Signal Type
1	Data Input	Data transmitted by UWB system input as binary bits.	Integer
2	Spreading Code Input	Coding sequence used to spread data being transmitted input as binary bits.	Integer

Pin	Name	Description	Signal Type
3	Pulse Output	Outputs pulse with shape determined by which pulse generator subcomponent is selected and polarity determined by data and spreading code.	Float

INTERFERENCE_SOURCE_80211a_UWB_Channel

The waveform for the 802.11a source is read from dataset file *WLAN_80211a_Order11.ds*. The interfering source operates at 100% duty cycle. The baseband signal is upsampled and downsampled using a ratio that adjusts its time step to be equal to TStep. The baseband waveform is a quadrature mixed carrier that is also represented in baseband form; the modulated signal at carrier frequency is represented in baseband form in the simulation.

Name	Description	Default	Unit	Type	Range
wlanPower_dBm	Power level of the 802.11a interference source in dBm	0		Real	<∞
wlanFreq	Center frequency of the 802.11a interference source	2.412e9	Frequency	Real	>0
TStep	Time step of the simulation	10 psec	Time	Real	>0

Pin	Name	Description	Signal Type
1	Output	Output from 802.11a source	Timed

INTERFERENCE_SOURCE_80211B_UWB_Channel

The waveform for the 802.11b source is read from dataset file *WLAN_80211b_8Xoversample.ds*. The interfering source operates at 100% duty cycle. The baseband signal is upsampled and downsampled using a ratio that adjusts its time step to be equal to TStep. The baseband waveform is a quadrature mixed carrier that is also represented in baseband form; the modulated signal at carrier frequency is represented in baseband form in the simulation.

Name	Description	Default	Unit	Type	Range
wlanPower_dBm	Power level of the 802.11b interference source in dBm	0		Real	<∞
wlanFreq	Center frequency of the 802.11b interference source	2.412e9	Frequency	Real	>0
TStep	Time step of the simulation	10 psec	Time	Real	>0

Pin	Name	Description	Signal Type
1	Output	Output from 802.11b source	Timed

MultipathDelayBlock_UWB_Channel

5 Multipath Arrivals
1
2
3

Delay inputs are the incremental delays from one multi-path input to the next. Relative magnitudes of each arrival are applied. Multi-path delay blocks can be connected in series to produce as many arrival times as required. Series connection uses Single Delayed Output pin 2. Delay1 through Delay5 and Mag0 through Mag4 affect the delay paths; when one MultipathDelayBlock_UWB_Channel is used, set Delay6=1 and Mag5=1, and connect as in *UWB_ENVIRONMENT.* (dguwb) For low pulse-rate applications where arrivals of sequential pulses do not overlap, it may be necessary to include arrivals within the integration window of the receiver correlators only.

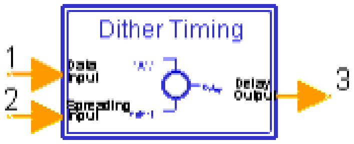
> **Note**
> Multipath terms can be defined using the IEEE P802.15.3 SG3a study group standard model based on a model developed by Saleh and Valenzuela (S-V) which includes a log-normal decay of the clusters along with a log-normal decay of rays within each cluster. See the SG3A "Channel Modeling Sub-committee Report (Final)," IEEE P802.15-02/490r0-SG3a, December 2002.

Name	Description	Default	Type	Range
Delay1, ... , Delay6	Number of TSteps of delay applied to each multipath arrival.	1	Integer	<∞
Mag0, ... , Mag5	Relative magnitude of each multipath arrival. The rake receiver uses these values to set the polarity of each correlator finger.	1	Real	<∞

Pin	Name	Description	Signal Type
1	Input	Transmit waveform	Float

Pin	Name	Description	Signal Type
2	Single Delayed Output	Input waveform delayed by the value of SumDelay5×TStep.	Float
3	Combined Multipath Output	Input waveform summed with SumDelay1×TStep through SumDelay5×TStep with respective magnitudes applied.	Float

PPM_MOD_UWB_Transmitter



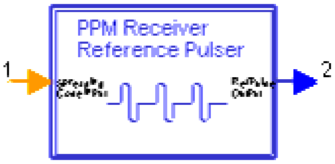
PPM_Mod_UWB_Transmitter determines the timing of a PPM modulated pulse stream. DitherBits determines the number of bits consumed from the spreading input for each data bit input. The time delay is calculated by multiplying TStepsPerDither by the integer value of the spreading code bits consumed. TStepsPerPPMBitOffset is added to the delay value if the data bit is 1. The delay output is an integer indicating the TSteps delay to be applied to the pulse.

Name	Description	Default	Unit	Type	Range
TStepsPerPPMBitOffset	Length of offset between a value of 1 and 0 in a PPM modulated code. Length is given in the number of TStep long intervals in the offset time. This make the offset between 0 and 1 an integer number of possible pulse positions.	1		Integer	>0
ChipsPerBit	Number of pulses transmitted for each bit.	1		Integer	>1
DitherBits	Number of bits used to the dither position of a PPM pulse within each pulse interval. $2^{DitherBits}$ pulse positions are possible within each PPM pulse interval.	5		Integer	>0
TStepsPerDither	Determines the number of simulation time steps between each possible position.	5		Integer	>0
ChipInterval	Time between pulses.	1 nsec	Time	Real	>0
PulseWidth	Width of output pulse.	250 psec	Time	Real	>0
PulseEnergy_joule	Total energy of the output pulse in Joules.	1e-12		Real	>0
DoubletSeparation	Time between the positive and negative peaks of the waveform. Only used for doublet pulses.	350 psec	Time	Real	>0
TStep	Time step of the simulation.	10 psec	Time	Real	>0

Pin	Name	Description	Signal Type
1	Spreading Code Input	Coding sequence used to spread data being transmitted input as binary bits.	Integer

Pin	Name	Description	Signal Type
2	Pulse Output	Outputs pulse with shape determined by which pulse generator subcomponent is selected.	Float

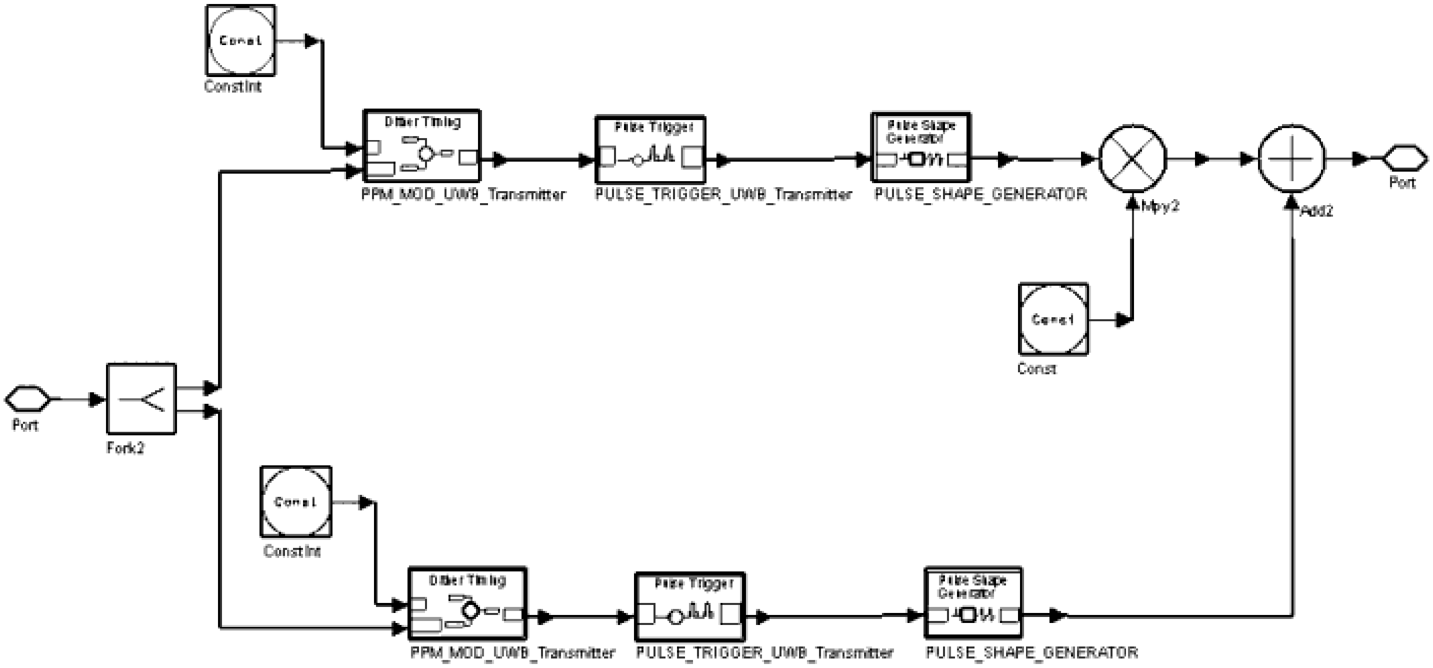
PPM_RX_Reference_Pulser



PPM_RX_Reference_Pulser outputs a waveform that is the sum of the pulse waveform produced by PPM_TX_PULSE_GENERATOR when the data input is 1 with the inverse of the output when the data input is 0.

The output represents a pulse position modulated UWB waveform. The input data bits are spread using a spreading code. The pulse shape output is determined by selecting sub-components of PULSE_SHAPE_GENERATOR; for details, refer to *PULSE_SHAPE_GENERATOR.* (dguwb)

Subnetwork

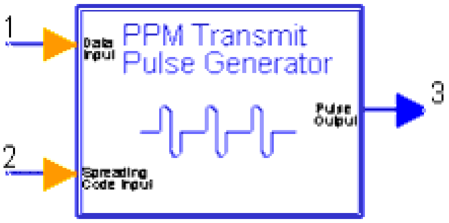


Name	Description	Default	Unit	Type	Range
TStepsPerPPMBitOffset	Length of offset between a value of 1 and 0 in a PPM modulated code. The length is given in the number of TStep long intervals in the offset time. This make the offset between 0 and 1 an integer number of possible pulse positions.	1		Integer	>0
ChipsPerBit	Number of pulses transmitted for each bit.	1		Integer	>1
DitherBits	Number of bits used to the dither position of a PPM pulse within each pulse interval. $2^{DitherBits}$ pulse positions are possible within each PPM pulse interval.	5		Integer	>0
TStepsPerDither	Determines the number of simulation time steps between each possible position.	5		Integer	>0
ChipInterval	Time between pulses.	1 nsec	Time	Real	>0
PulseWidth	Width of output pulse.	250 psec	Time	Real	>0
PulseEnergy_joule	Total energy of the output pulse in Joules.	1e-12		Real	>0
DoubletSeparation	Time between the positive and negative peaks of the waveform. Only used for doublet pulses.	350 psec	Time	Real	>0
TStep	Time step of the simulation.	10 psec	Time	Real	>0

Pin	Name	Description	Signal Type
1	Spreading Code Input	Coding sequence used to spread data being transmitted input as binary bits.	Integer

Pin	Name	Description	Signal Type
2	Pulse Output	Outputs pulse with shape determined by which pulse generator subcomponent is selected.	Float

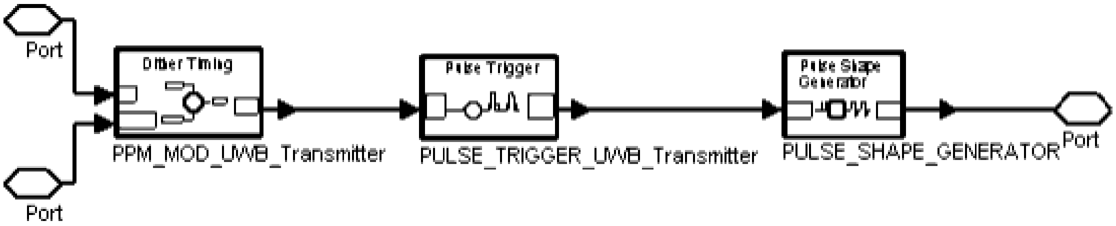
PPM_TX_Pulse_Generator



The PPM_TX_Pulse_Generator output represents a pulse position modulated UWB waveform. Input data bits are spread using a spreading code.

The pulse shape output is determined by selecting sub-components of PULSE_SHAPE_GENERATOR; for details, refer to *PULSE_SHAPE_GENERATOR.* (dguwb)

Subnetwork



Name	Description	Default	Unit	Type	Range
TStepsPerPPMBitOffset	Length of offset between a value of 1 and 0 in a PPM modulated code; Length is given in the number of TStep long intervals in the offset time. This make the offset between 0 and 1 an integer number of possible pulse positions.	1		Integer	>0
ChipsPerBit	Number of pulses transmitted for each bit.	1		Integer	>0
TStepsPerDither	Determines number of simulation time steps between each possible position.	1		Integer	>0
DitherBits	Number of bits used to the dither position of a PPM pulse within each pulse interval. $2^{DitherBits}$ pulse positions are possible within each PPM pulse interval.	5		Integer	>0
ChipInterval	Time between pulses.	1 nsec	Time	Real	>0
PulseWidth	Width of output pulse.	250 psec	Time	Real	>0
PulseEnergy_joule	Total energy of the output pulse in Joules.	1e-12		Real	>0
DoubletSeparation	Time between the positive and negative peaks of the waveform.	350 psec	Time	Real	>0
TStep	Time step of the simulation.	10 psec	Time	Real	>0

Pin	Name	Description	Signal Type
1	Data Input	Data transmitted by UWB system input as binary bits.	Integer
2	Spreading Code Input	Coding sequence used to spread data being transmitted input as binary bits.	Integer

Pin	Name	Description	Signal Type
3	Pulse Output	Outputs pulse with shape determined by which pulse generator subcomponent is selected.	Float

PropagationLoss

Propagation Loss
1
2

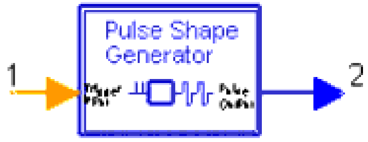
A propagation loss model is applied with variable attenuation rates to allow modeling of different terrains and antenna configurations. The value of gamma determines the rate of attenuation of the signal with distance. For free space, gamma equals 2.

Name	Description	Default	Unit	Type	Range
gamma	Power of distance from the source at which signal amplitude decays. In free space, gamma equals 2.	2		Real	>0
Range	Distance from transmitter to receiver.	3	Distance	Real	>0
do	Reference range for propagation loss. There is 0 dB attenuation at the reference range.	3	Distance	Real	>0
numwalls	Number of walls between transmit and receive antennas.	0		Integer	>or=0
LossWall	Attenuation of signal by each wall.	10	dB	Real	>0
numfloors	Number of floors between transmit and receive antennas.	0		Integer	>or=0
LossFloor	Attenuation of signal by each floor.	10	dB	Real	>0

Pin	Name	Description	Signal Type
1	Input	Transmit waveform.	Float

Pin	Name	Description	Signal Type
2	Output	Attenuated copy of input waveform.	Float

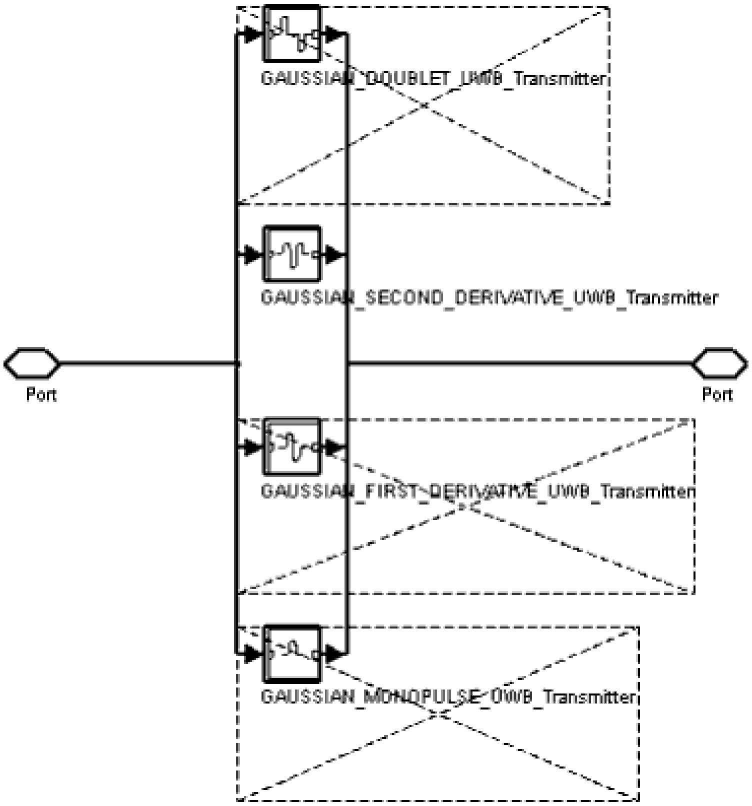
PULSE_SHAPE_GENERATOR



To select a pulse shape, push into the PULSE_SHAPE_GENERATOR component; enable the component for the desired pulse shape; disable the unused components.

In simulations, TStep is approximately PulseWidth/10. One pulse waveform is output each time the input signal amplitude equals 1. The total waveform width is 3.603×PulseWidth. When pulses overlap, the first pulse is truncated.

Subnetwork


GAUSSIAN_DOUBLET_UWB_Transmitter
GAUSSIAN_SECOND_DERIVATIVE_UWB_Transmitter
Port
Port
GAUSSIAN_FIRST_DERIVATIVE_UWB_Transmitter
GAUSSIAN_MONOPULSE_UWB_Transmitter

Pin	Name	Description	Signal Type
1	Trigger Input	An input impulse with magnitude. 1 triggers PULSE_SHAPE_GENERATOR to output pulse.	Float

Pin	Name	Description	Signal Type
2	Pulse Output	Outputs pulse with shape determined by pulse generator subcomponent.	Float

The GAUSSIAN_DOUBLET_UWB_Transmitter
generates a pair of Gaussian monopulse waveforms of opposite polarity.

Name	Description	Default	Unit	Type	Range
PulseWidth	Width of output pulse at 1/2 maximum amplitude of each of Gaussian monopulses that make up the doublet.	250 psec	Time	Real	>0
DoubletSeparation	Time between the positive and negative peaks of the waveform.	250 psec	Time	Real	>0
PulseEnergy_joule	Total energy of the output pulse in Joules. This value will not be calibrated when the individual pulses of the doublet overlap.	1e-12		Real	>0
TStep	Time step of the simulation.	10 psec	Time	Real	>0

Pin	Name	Description	Signal Type
1	Trigger Input	An input impulse with magnitude 1 triggers the output of a Gaussian doublet pulse.	Float

Pin	Name	Description	Signal Type
2	Pulse Output	Outputs Gaussian doublet pulse waveform.	Float

The GAUSSIAN_FIRST_DERIVATIVE_UWB_Transmitter
outputs a pulse with the shape of the first derivative of a Gaussian monopulse when a 1 is input.

Name	Description	Default	Unit	Type	Range
PulseWidth	Width of output pulse at 1/2 maximum amplitude of Gaussian monopulse of which the output waveform is a first derivative.	250 psec	Time	Real	>0
PulseEnergy_joule	Total energy of the output pulse in Joules.	1e-12		Real	>0
TStep	Time step of the simulation.	10 psec	Time	Real	>0

Pin	Name	Description	Signal Type
1	Trigger Input	An input impulse with magnitude 1 triggers the output of a Gaussian first derivative pulse.	Float

Pin	Name	Description	Signal Type
2	Pulse Output	Outputs Gaussian first derivative waveform.	Float

The GAUSSIAN_MONOPULSE_UWB_Transmitter
outputs a Gaussian-shaped monopulse when a 1 is input.

Name	Description	Default	Unit	Type	Range
PulseWidth	Width of output pulse at 1/2 maximum amplitude.	250 psec	Time	Real	>0
PulseEnergy_joule	Total energy of the output pulse in Joules.	1e-12		Real	>0
TStep	Time step of the simulation.	10 psec	Time	Real	>0

Pin	Name	Description	Signal Type
1	Trigger Input	An input impulse with magnitude 1 triggers the output of a Gaussian monopulse.	Float

Pin	Name	Description	Signal Type
2	Pulse Output	Outputs pulse with single Gaussian shaped pulse.	Float

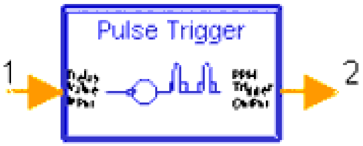
The GAUSSIAN_SECOND_DERIVATIVE_UWB_Transmitter outputs a pulse with the shape of the second derivative of a Gaussian monopulse when a 1 is input.

Name	Description	Default	Unit	Type	Range
PulseWidth	Width of output pulse at 1/2 maximum amplitude of Gaussian monopulse of which the output waveform is a second derivative.	250 psec	Time	Real	>0
PulseEnergy_joule	Total energy of the output pulse in Joules.	1e-12		Real	>0
TStep	Time step of the simulation.	10 psec	Time	Real	>0

Pin	Name	Description	Signal Type
1	Trigger Input	An input impulse with magnitude 1 triggers the output of a Gaussian second derivative pulse.	Float

Pin	Name	Description	Signal Type
2	Pulse Output	Outputs Gaussian second derivative waveform.	Float

PULSE_TRIGGER_UWB_Transmitter



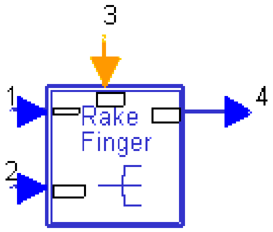
PULSE_TRIGGER_UWB_Transmitter produces an impulse every ChipInterval period. The input delay value is then applied to the impulse before it is output.

Name	Description	Default	Unit	Type	Range
ChipInterval	Time between pulses	1 nsec	Time	Real	>0
TStep	Time step of the simulation	10 psec	Time	Real	>0

Pin	Name	Description	Signal Type
1	Input Delay Value	Number of TSteps of delay to be applied to each pulse.	Integer

Pin	Name	Description	Signal Type
2	PPM Trigger Output	A pulse position modulated impulse used to trigger the pulse shape generator.	Float

Rake_Finger_UWB_Receiver



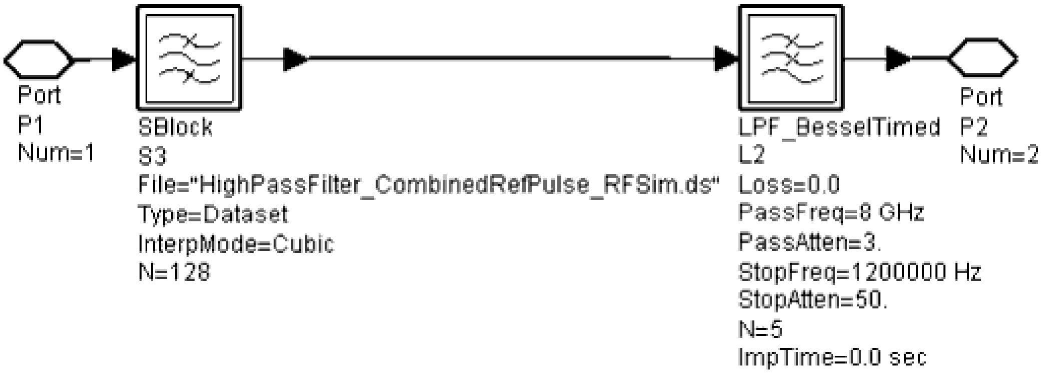
Name	Description	Default	Unit	Type	Range
DelayFinger	Delay time of this finger	0	sec	Float	>=0
MaxFingerDelay	Delay of rake finger with the maximum value		sec	Float	>=0
TStep	Time step of simulation		sec	Float	>=0
Polarity	Polarity of multipath component	1		Integer	-1, 1

Ref_Pulser_Shaping_Filters

Pulse Shaping
Filters
1
2

The reference pulser shaping filters apply the same filtering to the reference pulse as is applied to the transmit signal by the transmitter and receiver.

Subnetwork



Port
P1
Num=1

SBlock
S3
File="HighPassFilter_CombinedRefPulse_RFSim.ds"
Type=Dataset
InterpMode=Cubic
N=128

LPF_BesselTimed
L2
Loss=0.0
PassFreq=8 GHz
PassAtten=3.
StopFreq=1200000 Hz
StopAtten=50.
N=5
ImpTime=0.0 sec

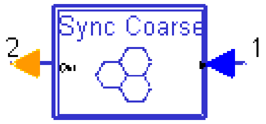
Port
P2
Num=2

Name	Description	Default	Unit	Type	Range
TStep	Time step of the simulation	10 psec	Time	Real	>0

Pin	Name	Description	Signal Type
1	Input	Output of receive reference pulse	Timed

Pin	Name	Description	Signal Type
2	Output	Output of receive reference pulse signal with series combination of transmit and receive filtering applied.	Timed

Sync_Coarse_UWB_Receiver



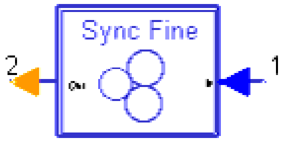
Sync_Coarse_UWB_Receiver compares the correlator integrator output to the magnitude of the average negative value of the correlator integrator output. If the most recent correlator output is less than RelSyncAmplitude times the average negative correlator output value the coarse sync adjustment outputs 1; otherwise, it outputs 0. The Sync_Coarse algorithm stops incrementing the time offsets when the correlator integrator output exceeds the threshold level.

Name	Description	Default	Type	Range
RelSyncAmplitude	If the ratio of current correlator output value to the absolute value of the average negative correlator output value is less than the value of RelSyncAmplitude, the coarse synchronization algorithm increases the delay of the correlator reference signal by the value of CoarseTimeStep.	1.5	Float	>0

Pin	Name	Description	Signal Type
1	In	Correlator output value	Float

Pin	Name	Description	Signal Type
2	Out	If coarse sync adjustment is to be applied, it is 1.; otherwise, it is 0.	Integer

Sync_Fine_Tune_UWB_Receiver



Sync_Fine_Tune_UWB_Receiver compares a time average of the correlator integrator output to the previous value of time averaged correlator output. After collecting each time average sample it either increments or decrements the timing of the correlator by one TStep. If the most recent averaged correlator output value is greater than the previous, the offset delay increment is in the same direction as the previous; if the most recent averaged correlator output value is less than the previous, the offset delay increment is in the opposite direction of the previous.

Name	Description	Default	Type	Range
NumFineSyncAverage	Number of inputs to average	32	Integer	>1

Pin	Name	Description	Signal Type
1	In	Correlator output value	Float

Pin	Name	Description	Signal Type
2	Out	Determines the direction of the fine synchronization adjustment by outputting 1 or -1.	Integer

UWB_BIT_SLICER_UWB_Receiver

1
UWB
Bit Slicer
110
111
010
2

The bit slicer captures the correlator integrator output value immediately before the integrator resets. This value is used to determine whether the output is a 0 or 1 bit.

Name	Description	Default	Unit	Type	Range
ChipsPerBit	Number of pulses transmitted for each bit	1		Integer	>1
ChipInterval	Time between pulses	1 nsec	Time	Real	>0
TStep	Time step of the simulation	10 psec	Time	Real	>0

Pin	Name	Description	Signal Type
1	Input	Signal from output of correlator integrator.	Float

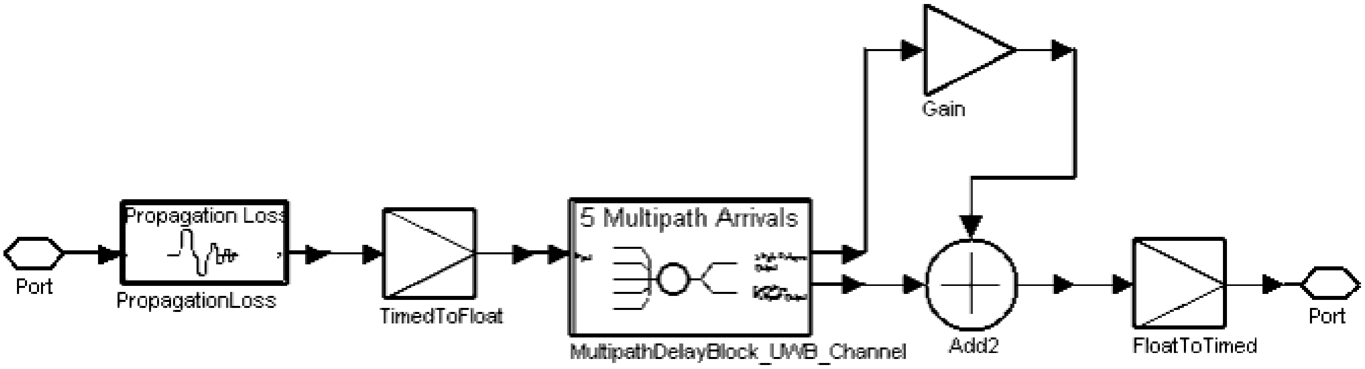
Pin	Name	Description	Signal Type
2	Bits Output	Demodulated data bits	Integer

UWB_ENVIRONMENT

1
UWB
Environment
2

This subnetwork contains the propagation loss and multi-path models.

Subnetwork

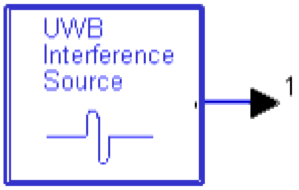


Name	Description	Default	Unit	Type	Range
TStep	Time step of the simulation	10 psec	Time	Real	>0
Delay1,2,3,...	Number of TSteps of delay applied to each series delay in the multipath component. This is calculated such that round off errors do not accumulate.				
Mag0,1,2,...	Relative magnitude of each multi-path arrival.				
gamma	Power of distance from the source at which signal amplitude decays. For an isotropic antenna in free space, gamma equals 2.				

Pin	Name	Description	Signal Type
1	Input	Outputs of transmitter antenna.	Timed

Pin	Name	Description	Signal Type
2	Output	Transmit signal with multipath components and loss relative to the signal level at do applied.	Timed

UWB_Interference_Source

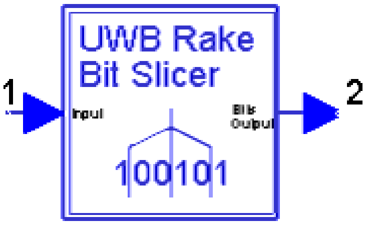


The component provides a second UWB signal source for use as a wideband interfering signal.

Name	Description	Default	Unit	Type	Range
TStepsPerPPMBitOffset	Number of TStep intervals between 1 and a 0 position in a PPM pulse interval	50		Integer	>0
ChipsPerBit	Number of pulses transmitted for each bit	1		Integer	>0
ChipInterval	Time between pulses. For PPM this is the nominal time between pulses.	1 nsec	Time	Real	>0
TStepsPerDither	Determines the number of simulation time steps between each possible position.	5		Integer	>0
DitherBits	Number of bits used to the dither position of a PPM pulse within each pulse interval. $2^{DitherBits}$ pulse positions are possible within each PPM pulse interval.	5		Integer	>0
PulseWidth	1/2 amplitude pulsewidth of a Gaussian monopulse output. For the Gaussian derivative pulse shapes, this is the 1/2 amplitude pulsewidth of the Gaussian monopulse from which it is derived.	100 psec	Time	Real	>0
DoubletSeparation	Time between the positive and negative peaks of the waveform when doublet pulse is used. (Used when doublet waveform is selected.)	400 psec	Time	Real	>0
PulseEnergy_joule	Energy in Joules of single pulse output from the pulse generator.	1e-12		Real	>0

Pin	Name	Description	Signal Type
1	Output	Output a UWB waveform	Time

UWB_RAKE_BIT_SLICER_UWB_Receiver



UWB_RAKE_BIT_SLICER_UWB_Receiver performs a function similar to UWB_BIT_SLICER_UWB_Receiver. Both bit slicers sample the integrator output at the last timestep before the integrator is reset in each bit interval; the polarity of the sampled integrator output is used to determine the value of the bit.

In the Rake receiver, the signal from each receiver finger is delayed to align it with the signal from the Rake finger synrhonized with the greatest delay arrival being received. UWB_RAKE_BIT_SLICER_UWB_Receiver uses the value of MaxFingerDelay to adjust the sampling time by the amount of additional delay applied to the first signal arrival. (In UWB_BIT_SLICER_UWB_Receiver, the integrator sampling time is fixed in the bit interval because variable delays are not present in the test benches that use it.)

Name	Description	Default	Unit	Type	Range
ChipsPerBit	Number of pulses transmitted for each bit	1		Integer	>1
ChipInterval	Time between pulses	1 nsec	Time	Real	>0
MaxFingerDelay	Largest finger delay	1 nsec	Time	Real	>O
TStep	Time step of the simulation	10 psec	Time	Real	>0

Pin	Name	Description	Signal Type
1	Input	Signal from output of correlator integrator.	Float

Pin	Name	Description	Signal Type
2	Bits Output	Demodulated data bits	Integer

UWB_Rake_Receiver_Correlator

1
2
UWB Rake
Receiver
Correlator
3

The correlator multiplies the receive signal with a reference signal and integrates the results over a period of time. The integrator in the correlator integrates the multiplier output signal over the period of ChipInterval×ChipsPerBit. It resets the integrator value to 0 and restarts the integration.

UWB_RX_SINGLE_FINGER_CORRELATOR has a single finger path; UWB_Rake_Receiver_Correlator has 4 finger paths.

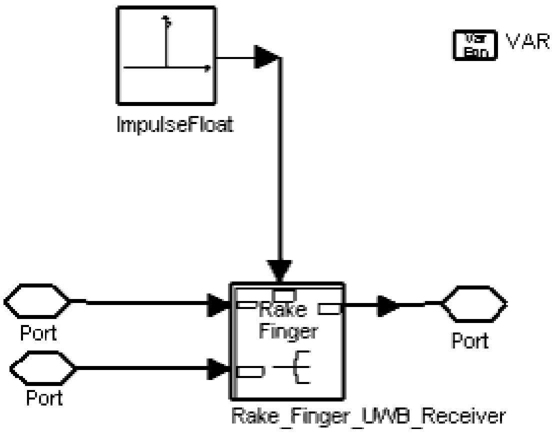
Name	Description	Default	Unit	Type	Range
ChipsPerBit	Number of pulses transmitted for each bit.	1		Integer	>1
ChipInterval	Time between pulses.	1 nsec	Time	Real	>0
TStep	Time step of the simulation	10 psec	Time	Real	>0

Name	Description	Default	Type	Range
Mag0	The relative magnitude of each multipath arrival. The rake receiver uses these values to set the polarity of each correlator finger.	1	Real	<∞
DelayFinger0	The number of TSteps of delay applied to 1st multipath arrival.	1	Integer	<∞
MaxFingerDelay	The maximum value DelayFinger0 through DelayFingerN	1	Integer	<∞

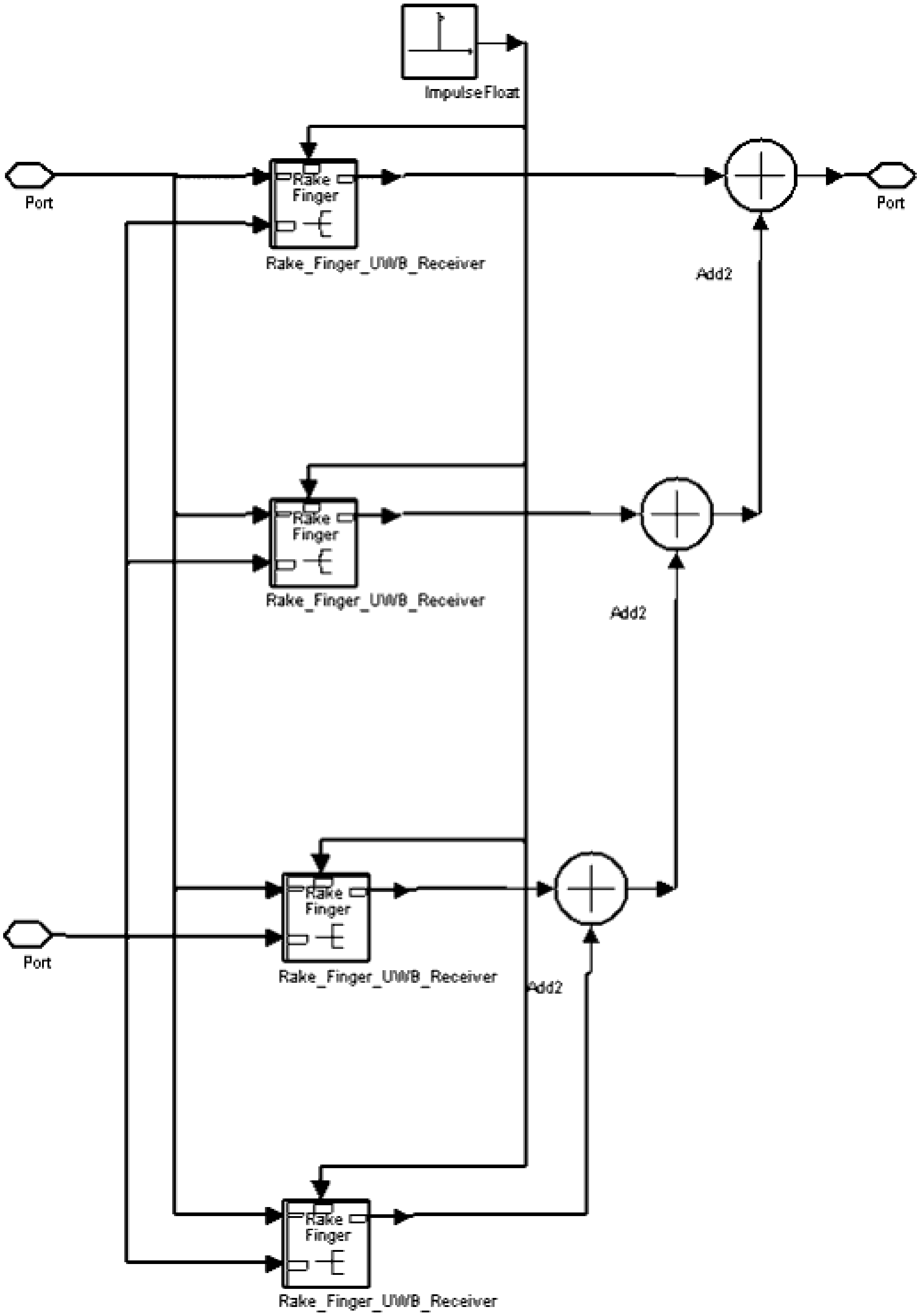
Pin	Name	Description	Signal Type
1	Antenna Input	UWB receiver input waveform	Float
2	Reference Pulse Input	Reference pulse waveform for correlator	Float

Pin	Name	Description	Signal Type
3	Output	Scaled and combined output from all correlator fingers	Float

UWB_RX_SINGLE_FINGER_CORRELATOR Subnetwork



UWB_Rake_Receiver_Correlator Subnetwork



UWB_RECEIVE_ANTENNA

1
UWB RX
Antenna
2

This component consists of an SBlock component that reads a file of S-parameters representing the RF frontend and antenna of the receiver. The receive antenna also contains a loss component to allow scale of the overall loss of the antenna.

S-parameters can be measured using a network analyzer. In the default setup, they are created by conducting RF simulations of a series of filters. *HighPassFilter_RFsim* is used to generate the default S-parameter file.
After reading the S-parameters from the file, the SBlock component produces the equivalent impulse response for use in ADS Ptolemy simulations. For a practical impulse response to be produced, a high-frequency cutoff lowpass filter must be applied when S-parameters for a highpass filter are generated.

> **Note**
> The SBlock N value must be large enough for the filter impulse response to adequately decay. The impulse response will be truncated after a time of N×TStep. Using the lowest acceptable value of N will speed simulation.

A receive antenna can capture only a portion of this energy; the portion of the total radiated power that a receive antenna can capture is determined by the receive antenna aperture, the distance between transmit and receive antennas, and the antenna gains. The receive antenna is modeled using an additional SBlock component and an attenuator. The receive antenna contains a 1/frequency roll-off term to account for the reduction of antenna aperture with frequency. An additional attenuator is applied to adjust the total loss of the antenna at each frequency to the expected value at 3 meters. For example, the antenna aperture at 5.5 GHz is given by the equation

$$A_e = \frac{\lambda^2}{4\Pi}$$

l is the signal wavelength and the expected aperture loss of the antenna at three meters is given by the equation

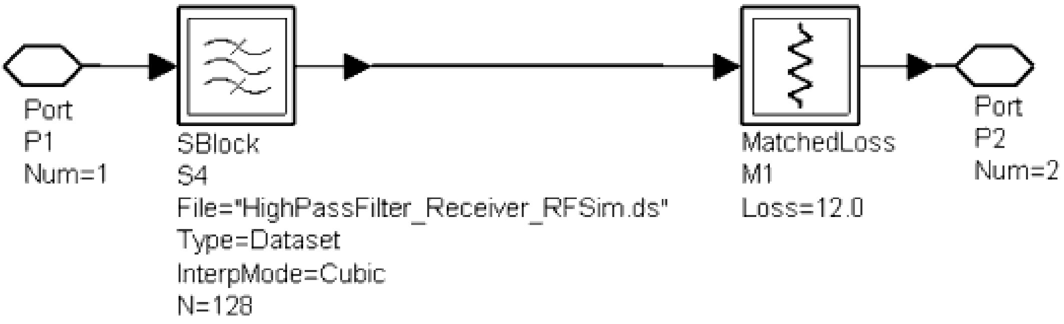
$$ApertureLoss = 10\log\left|\frac{A_e}{4\Pi\langle 3m\rangle^2}\right|$$

If the directional gain of the antenna relative to an isotropic antenna is 0, then at 5.5 GHz the signal level the receive antenna delivers to the receiver is 57 dB below the total transmit power at 3 meters. The loss of the default receiver SBlock component is 45 dB at 5.5 GHz, so 12 dB of loss is applied by the attenuator. To adjust the signal level for distances other than 3 meters, the propagation loss model is applied with do equal to 3 meters.

The signal from the receive antenna goes to an amplifier; the noise figure of this amplifier can be set by the user. A lowpass filter following the amplifier limits the noise bandwidth of the receiver frontend.

For more details regarding generating S-parameter files, refer to *UWB_TRANSMIT_ANTENNA.* (dguwb)

Subnetwork

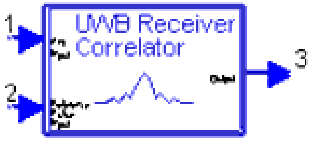


Name	Description	Default	Unit	Type	Range
TStep	Time step of the simulation	10 psec	Time	Real	>0

Pin	Name	Description	Signal Type
1	Input	Input to receive antenna	Timed

Pin	Name	Description	Signal Type
2	Output	Receive antenna input waveform modified by front-end and antenna characteristics. Signal level is adjusted to account for amount of energy capture in antenna aperture at distance do.	Timed

UWB_RX_Correlator_UWB_Receiver



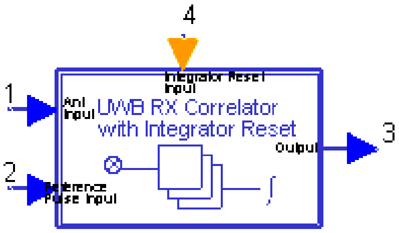
This component provides multiple correlators for receiving arrivals of a multipath signal. Each correlator multiplies the received signal by an appropriately delayed reference signal. The integrator in the correlator integrates the multiplier output signal over the period of ChipInterval×ChipsPerBit. It resets the integrator value to 0 and restarts the integration. The outputs of each correlator are scaled relative to its signal-to-noise ratio, and the outputs of all correlators are summed.

Name	Description	Default	Unit	Type	Range
ChipsPerBit	Number of pulses transmitted for each bit.	1	none	Integer	>1
ChipInterval	Time between pulses.	1 nsec	Time	Real	>0
TStep	Time step of the simulation	10 psec	Time	Real	>0

Pin	Name	Description	Signal Type
1	Antenna Input	UWB receiver input waveform	Float
2	Reference Pulse Input	Reference pulse waveform for correlator	Float

Pin	Name	Description	Signal Type
3	Output	Scaled and combined output from all correlator fingers	Float

UWB_RX_Correlator_with Integrator_Reset_UWB_Receiver



This correlator multiplies the receive signal with a reference signal and integrates the results over a period of time. The integrator in the correlator integrates the multiplier output signal between >1 values at the Reference Pulse Input pin 2; it resets the integrator value to 0 and restarts the integration when a pulse is received on this pin.

Pin	Name	Description	Signal Type
1	Antenna Input	UWB receiver input waveform	Float
2	Reference Pulse Input	Reference pulse waveform for correlator	Float
4	Integrator Reset Input		Integer

Pin	Name	Description	Signal Type
3	Output	Output from correlator integrator	Float

UWB_RX_LNA

UWB RX LNA
1
2

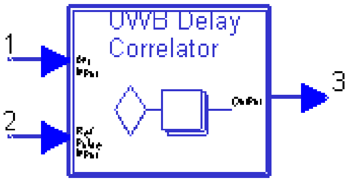
The receiver LNA noise figure and bandwidth can be set to simulate receiver RF frontend performance.

Name	Description	Default	Unit	Type	Range
NoseFigure_dB	Noise figure in dB of the receiver frontend.	4		Real	>0
RXNoiseBandwidth	Noise passband of the receiver frontend.	8 GHz	Frequency	Real	>0

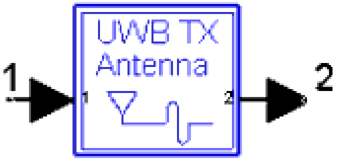
Pin	Name	Description	Signal Type
1	Input	Output of receive antenna	Timed

Pin	Name	Description	Signal Type
2	Output	Output of receive antenna with frontend noise and bandwidth limits applied.	Timed

UWB_RX_SINGLE_FINGER_CORRELATOR



UWB_TRANSMIT_ANTENNA



The transmit filter component represents frequency-dependent effects in the pulser, antenna, and environment.

The *UWB_TRANSMIT_ANTENNA* subnetwork consists of an SBlock component that reads a file of S-parameters representing the RF front-end and antenna of the transmit pulser. The S-parameters can be measured using a network analyzer. In the default setup, they are created by conducting RF simulations of a series of filters.

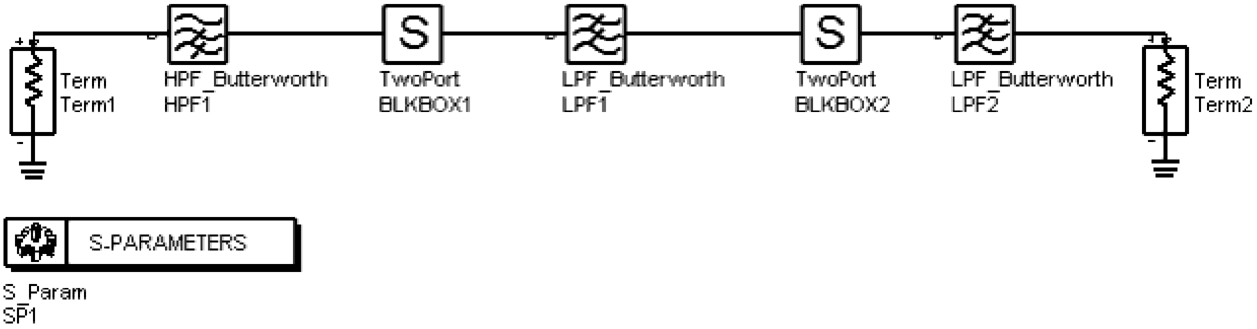
HighPassFilter_RFsim (HighPassFilter_RFSim Schematic) is used to generate the default S-parameter file. After reading S-parameters from the file, the SBlock component produces the equivalent impulse response for use in ADS Ptolemy simulations. To produce a practical impulse response, a high-frequency cutoff lowpass filter must be applied when the S-parameters for a highpass filter are generated.

The SBlock component determines transmit filter characteristics by reading S-parameters from a file. S-parameter files can be generated using templates provided in the *DesignGuide > UWB > Frequency Dependent Models* menu. These designs use the ADS RF simulator (S_Param) to generate S21 for a series combination of lowpass and highpass filters. The default combination of filters provides a representation of the response of a transmit chain and antenna so the transmitter can meet FCC indoor spectral mask regulations. The *Test Impulse Response for use with SBlock* template (*HighPassFilter_SBlockTest,* HighPassFilter_SBlockTest Schematic) provides an ADS Ptolemy simulation for testing the impulse response using filter S-parameters.

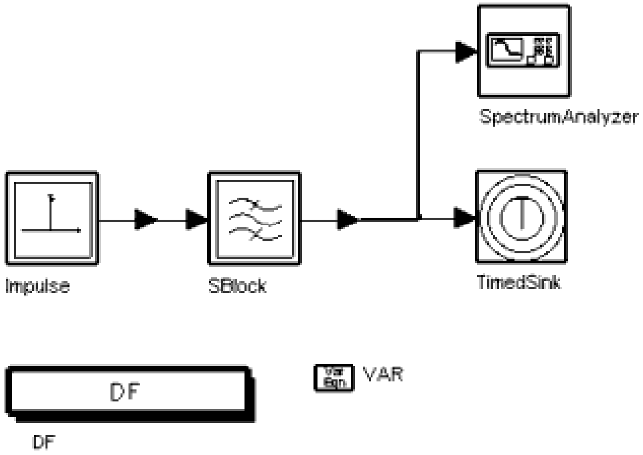
> **Note**
> The SBlock N value must be large enough for the filter impulse response to adequately decay. The impulse response will be truncated after a time of N×TStep. Using the lowest acceptable value of N will speed simulation.

If a highpass filter is used in the RF simulator, a lowpass filter must be included to limit the high-frequency response; otherwise, the SBlock component cannot generate a practically solvable impulse response from the resulting set of S-parameters. S-parameters of the user's circuits can be included in the simulation using the SBlock component.

S-parameters measured using a network analyzer can be used in the input file for the filter. Refer to SBlock component *Help* for input file format information.

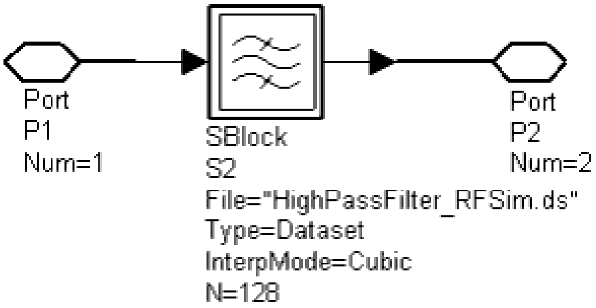


***HighPassFilter_RFSim* Schematic**



***HighPassFilter_SBlockTest* Schematic**

Subnetwork



Name	Description	Default	Unit	Type	Range
TStep	Time step of the simulation	10 psec	Time	Real	>0

Pin	Name	Description	Signal Type
1	Input	Input to transmit front-end.	Timed

Pin	Name	Description	Signal Type
2	Output	Transmit waveform modified by front-end and antenna characteristics	Timed