Agilent Technologies

Advanced Design System 2011.01

Feburary 2011
Simulation Instruments

Acknowledgments

SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

7-zip - 7-Zip Copyright: Copyright (C) 1999-2009 Igor Pavlov. Licenses for files are: 7z.dll: GNU LGPL + unRAR restriction, All other files: GNU LGPL. 7-zip License: This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version. This library is distributed in the hope that it will be useful,but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details. You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA. unRAR copyright: The decompression engine for RAR archives was developed using source code of unRAR program.All copyrights to original unRAR code are owned by Alexander Roshal. unRAR License: The unRAR sources cannot be used to re-create the RAR compression algorithm, which is proprietary. Distribution of modified unRAR sources in separate form or as a part of other software is permitted, provided that it is clearly stated in the documentation and source comments that the code may not be used to develop a RAR (WinRAR) compatible archiver. 7-zip Availability: http://www.7-zip.org/

AMD Version 2.2 - AMD Notice: The AMD code was modified. Used by permission. AMD copyright: AMD Version 2.2, Copyright © 2007 by Timothy A. Davis, Patrick R. Amestoy, and Iain S. Duff. All Rights Reserved. AMD License: Your use or distribution of AMD or any modified version of AMD implies that you agree to this License. This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version. This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details. You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA Permission is hereby granted to use or copy this program under the terms of the GNU LGPL, provided that the Copyright, this License, and the Availability of the original version is retained on all copies.User documentation of any code that uses this code or any modified version of this code must cite the Copyright, this License, the Availability note, and "Used by permission." Permission to modify the code and to distribute modified code is granted, provided the Copyright, this License, and the Availability note are retained, and a notice that the code was modified is included. AMD Availability: http://www.cise.ufl.edu/research/sparse/amd

UMFPACK 5.0.2 - UMFPACK Notice: The UMFPACK code was modified. Used by permission. UMFPACK Copyright: UMFPACK Copyright © 1995-2006 by Timothy A. Davis. All Rights Reserved. UMFPACK License: Your use or distribution of UMFPACK or any modified version of UMFPACK implies that you agree to this License. This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License

as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version. This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details. You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA Permission is hereby granted to use or copy this program under the terms of the GNU LGPL, provided that the Copyright, this License, and the Availability of the original version is retained on all copies. User documentation of any code that uses this code or any modified version of this code must cite the Copyright, this License, the Availability note, and "Used by permission." Permission to modify the code and to distribute modified code is granted, provided the Copyright, this License, and the Availability note are retained, and a notice that the code was modified is included. UMFPACK Availability: http://www.cise.ufl.edu/research/sparse/umfpack UMFPACK (including versions 2.2.1 and earlier, in FORTRAN) is available at http://www.cise.ufl.edu/research/sparse . MA38 is available in the Harwell Subroutine Library. This version of UMFPACK includes a modified form of COLAMD Version 2.0, originally released on Jan. 31, 2000, also available at http://www.cise.ufl.edu/research/sparse . COLAMD V2.0 is also incorporated as a built-in function in MATLAB version 6.1, by The MathWorks, Inc. http://www.mathworks.com . COLAMD V1.0 appears as a column-preordering in SuperLU (SuperLU is available at http://www.netlib.org). UMFPACK v4.0 is a built-in routine in MATLAB 6.5. UMFPACK v4.3 is a built-in routine in MATLAB 7.1.

Qt Version 4.6.3 - Qt Notice: The Qt code was modified. Used by permission. Qt copyright: Qt Version 4.6.3, Copyright (c) 2010 by Nokia Corporation. All Rights Reserved. Qt License: Your use or distribution of Qt or any modified version of Qt implies that you agree to this License. This library is free software; you can redistribute it and/or modify it under the
terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version. This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details. You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA Permission is hereby granted to use or copy this program under the terms of the GNU LGPL, provided that the Copyright, this License, and the Availability of the original version is retained on all copies.User
documentation of any code that uses this code or any modified version of this code must cite the Copyright, this License, the Availability note, and "Used by permission." Permission to modify the code and to distribute modified code is granted, provided the Copyright, this License, and the Availability note are retained, and a notice that the code was modified is included. Qt Availability: http://www.qtsoftware.com/downloads Patches Applied to Qt can be found in the installation at:
$HPEESOF_DIR/prod/licenses/thirdparty/qt/patches. You may also contact Brian Buchanan at Agilent Inc. at brian_buchanan@agilent.com for more information.

The HiSIM_HV source code, and all copyrights, trade secrets or other intellectual property rights in and to the source code, is owned by Hiroshima University and/or STARC.

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 Simulation Instruments

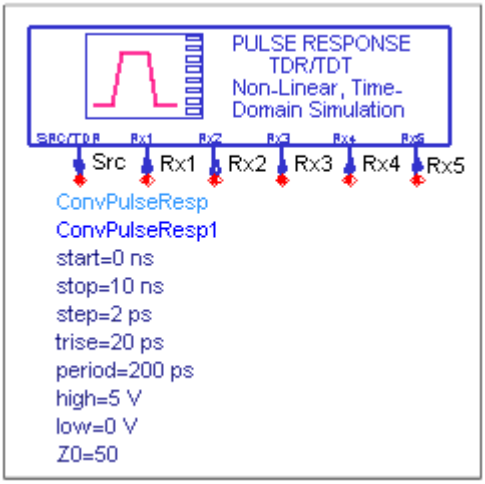
This topic describes the parameters for the simulation instrument components available in the *Simulation - Instrument* component library. These components are used in several of the simulation templates, which are accessed from the Schematic window by selecting **Insert** > **Template**.

Simulation Instrument Components

- *ConvPulseResp (Convolution Pulse Response)* (cktsiminst)
- *ConvStepResp (Convolution Step Response)* (cktsiminst)
- *DC_BJT (Curve Tracer for BJT)* (cktsiminst)
- *DC_FET (Curve Tracer for FET)* (cktsiminst)
- *ImpulseWriter (Impulse Response File Writer)* (cktsiminst)
- *LinearPulseResp (Pulse Response from Frequency Response)* (cktsiminst)
- *LinearStepResp (Linear Response from Frequency Response)* (cktsiminst)
- *SP_BJT (S-Parameters vs. Bias for BJT)* (cktsiminst)
- *SP_Diff (Differential-Mode S-Parameters)* (cktsiminst)
- *SP_FET (S-Parameters vs. Bias for FET)* (cktsiminst)
- *SP_NWA_4Port (4-Port Network Analyzer)* (cktsiminst)
- *SP_NWA_4PortBias (4-Port Network Analyzer with Bias Sources)* (cktsiminst)
- *SP_NWA_4PortBiasLog (4-Port Network Analyzer with Bias, Log Sweep)* (cktsiminst)
- *SP_NWA_4PortLog (4-Port Network Analyzer, Log Sweep)* (cktsiminst)
- *SP_NWA_Log (Network Analyzer for S-Parameters, Log Sweep)* (cktsiminst)
- *SP_NWA (Network Analyzer for S-Parameters)* (cktsiminst)

ConvPulseResp (Convolution Pulse Response)

Symbol



Parameters

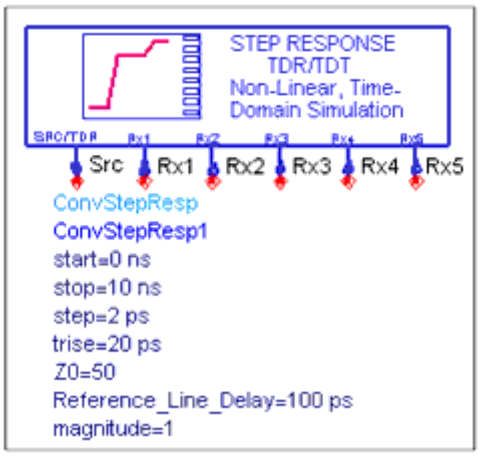
Parameter Name	Description	Units
start	start time for output data	ns
stop	stop time for output data	ns
step	step time for output data	ns
trise	pulse rise time	ps (10 to 90%)
period	pulse period	ps
high	high value of pulse	fV, pv, nv, uv, mv, or V (default)
low	low value of pulse	fV, pv, nv, uv, mv, or V (default)
Z0	impedance of transmit and receive ports	

Notes

1. A template using this item can be accessed by selecting **Insert** > **Template** > **ConvPulseRespT** from the Schematic window.
2. ConvPulseResp emulates an instrument for measuring the reflection and transmission of a network. The test signal is a pulse waveform, whose characteristics you specify. There is one source port, and five receive ports. The source port is also used to measure the reflected signal. The simulation is carried out in the time domain, and if distributed elements are present in the network being simulated, the convolution simulator will be used. The example, *RF_Board/TDRcrosstalk_wrk* shows this component applied.
3. This is a simulation component. No other simulation or control components are needed.

ConvStepResp (Convolution Step Response)

Symbol



Parameters

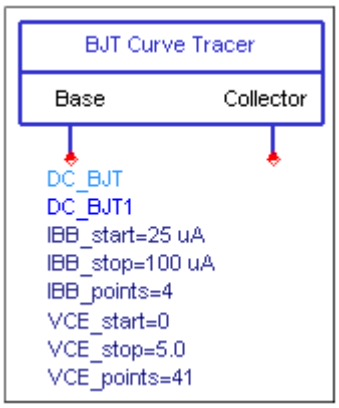
Parameter Name	Description	Units
start	start time for output data	ns
stop	stop time for output data	ns
step	step time for output data	ps
trise	step rise time	ps (10 to 90%)
period	pulse period	ps
Z0	impedance of transmit and receive ports	
Reference_Line_delay	Reference line time delay. This is the time delay of an ideal transmission line internal to the instrument. It just delays the test signal coming out of the *Src* port	
magnitude	step amplitude at transmit port	

Notes

1. A template using this item can be accessed by selecting **Insert** > **Template** > **ConvStepT** from the Schematic window.
2. ConvStepResp emulates an instrument for measuring the reflection and transmission of a network. The test signal is a step waveform, whose characteristics you specify. There is one source port, and five receive ports. The source port is also used to measure the reflected signal. The simulation is carried out in the time domain, and if distributed elements are present in the network being simulated, the convolution simulator will be used. The example, *RF_Board/TDRcrosstalk_wrk* shows this component applied.
3. This is a simulation component. No other simulation or control components are needed.

DC_BJT (Curve Tracer for BJT)

Symbol



Parameters

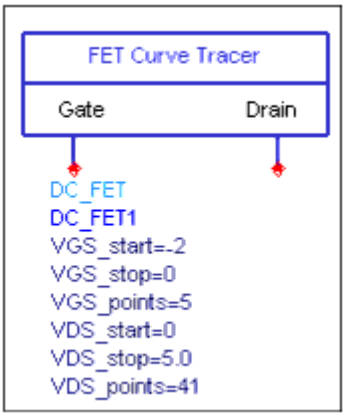
Parameter Name	Description	Units
IBB_start	initial base current	uA
IBB_stop	last base current	uA
IBB_points	number of base current values	
VCE_start	initial collector emitter voltage	
VCE_stop	last collector emitter voltage	
VCE_points	number of collector-emitter values	

Notes

1. A template using this item can be accessed by selecting **Insert** > **Template** > **DC_BJT_T** from the Schematic window.
2. DC_BJT is a DC curve-tracer with a swept voltage source for the collector bias and a swept current source for the base bias.
3. This is a simulation component. No other simulation or control components are needed.

DC_FET (Curve Tracer for FET)

Symbol



Parameters

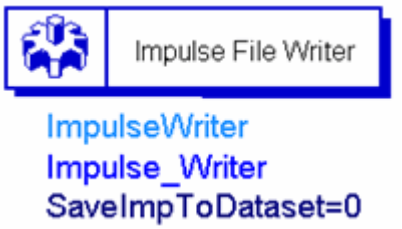
Parameter Name	Description
VGS_start	initial gate voltage
VGS_stop	last gate voltage
VGS_points	number of gate current values
VDS_start	initial drain-source voltage
VDS_stop	last drain-source voltage
VDS_points	number of drain-source values

Notes

1. A template using this item can be accessed by selecting **Insert** > **Template** > **DC_FET_T** from the Schematic window.
2. DC_FET is a DC curve-tracer with two swept voltage sources, one for the gate bias and the other for the drain bias.
3. This is a simulation component. No other simulation or control components are needed.

ImpulseWriter (Impulse Response File Writer)

Symbol



Parameters

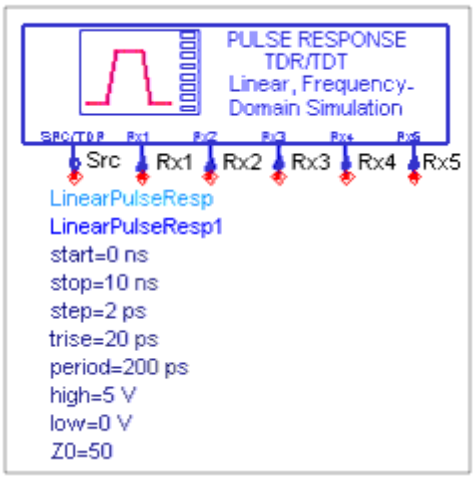
Parameter Name	Description
SaveImpToDataset	save computed impulse responses to dataset (allowed values: 0, 1, no, yes)

Notes

1. ImpulseWriter converts S-parameters imported by SnP block into time domain impulse responses which accurately represent the original spectral. The causality problem associated with band-limited spectrum is circumvented using proprietary technology and the computed impulses are guaranteed to be causal. They are also passive within the original spectrum frequency range. To turn off passivity enforcement, set *EnforcePassivity*=no in the SnP component parameter.
2. The result of impulse responses is exported into an ASCII file created in the workspace data directory. The file name follows the SnP instance name with a suffix of *.imp*. The data format is described in detail in *ADS Impulse File Format* (cktsim).

LinearPulseResp (Pulse Response from Frequency Response)

Symbol



Parameters

Parameters Name	Description	Units
start	start time for output data	ns
stop	stop time for output data	ns
step	step time for output data	ns
trise	pulse rise time	ps (10 to 90%)
period	pulse period	ps
high	high value of pulse	fV, pv, nv, uv, mv, or V (default)
low	low value of pulse	fV, pv, nv, uv, mv, or V (default)
Z0	impedance of transmit and receive ports	

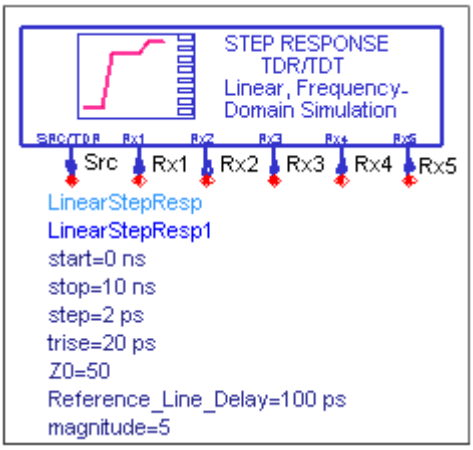
Notes

1. A template using this item can be accessed by selecting **Insert** > **Template** > **LinearPulseRespT** from the Schematic window.
2. LinearPulseResp emulates an instrument for measuring the reflection and transmission of a network. The test signal is a pulse waveform, whose characteristics you specify. There is one source port, and five receive ports. The source port is also used to measure the reflected signal. The simulation is carried out in the frequency domain, and the frequency-domain data is post-processed via the ts() function to get time-domain responses. If any nonlinear elements are included in the network being simulated, they will be modeled as linear elements, linearized around their bias points. The example, *RF_Board/TDRcrosstalk_wrk* shows this component applied.

3. This is a simulation component. No other simulation or control components are needed.

LinearStepResp (Linear Response from Frequency Response)

Symbol



Parameters

Parameter Name	Description	Units
start	start time for output data	ns
stop	stop time for output data	ns
step	step time	ps
trise	step rise time	ps (10 to 90%)
Z0	impedance of transmit and receive ports	
Reference_Line_delay	Reference line time delay	
magnitude	step amplitude at transmit port	

Notes

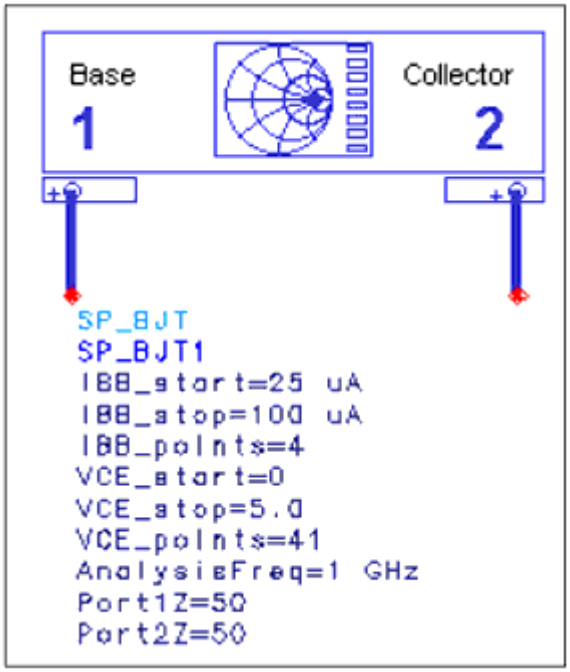
1. A template using this item can be accessed by selecting **Insert** > **Template** > **LinearStepRespT** from the Schematic window.
2. LinearStepResp emulates an instrument for measuring the reflection and transmission of a network. The test signal is a step waveform, whose characteristics you specify. There is one source port, and five receive ports. The source port is also used to measure the reflected signal. The simulation is carried out in the frequency domain, and the frequency-domain data is post-processed via the *ts()* function to get

time-domain responses. If any nonlinear elements are included in the network being simulated, they will be modeled as linear elements, linearized around their bias points. The example, *RF_Board/TDRcrosstalk_wrk* shows this component applied.

3. This is a simulation component. No other simulation or control components are needed.

SP_BJT (S-Parameters vs. Bias for BJT)

Symbol



Parameters

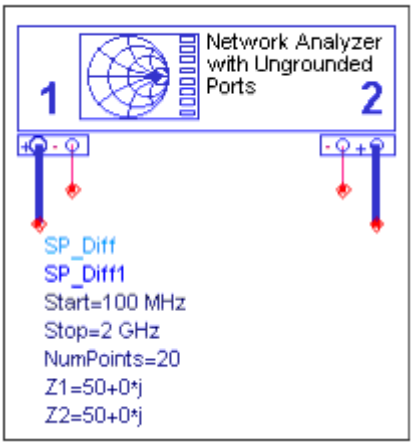
Parameters Name	Description	Units
IBB_start	initial base current	uA
IBB_stop	last base current	uA
IBB_points	number of base current values	
VCE_start	initial collector emitter voltage	
VCE_stop	last collector emitter voltage	
VCE_points	number of collector-emitter values	
AnalysisFreq	Single S-parameter analysis frequency	GHz
Port1Z	port 1 port impedance (complex)	
Port2Z	port 2 port impedance (complex)	

Notes

1. A template using this item can be accessed by selecting **Insert** > **Template** > **SP_BJT_T** from the Schematic window.
2. SP_BJT sets up an S-parameter analysis at one frequency with swept current and swept voltage for the base and collector biases, respectively. This component helps select an operating point for desired gain. Connect it to a bipolar junction transistor, as indicated in the schematic symbol.
3. This is a simulation component. No other simulation or control components are needed.

SP_Diff (Differential-Mode S-Parameters)

Symbol



Parameters

Parameter Name	Description	Unit
Start	start frequency	Hz, KHz, GHz, or MHz (default)
Stop	stop frequency	Hz, KHz, GHz, or MHz (default)
NumPoints	number of points in a linear sweep	
Z1	port 1 port impedance (complex)	
Z2	port 2 port impedance (complex)	

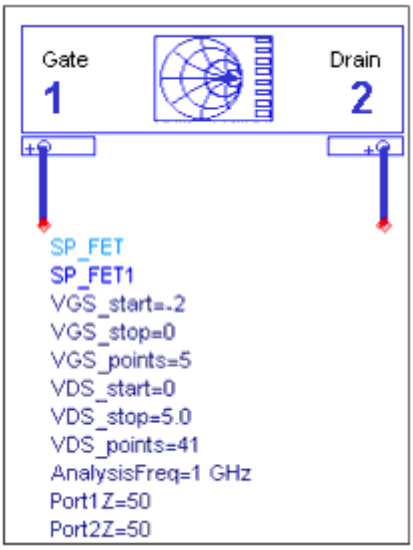
Notes

1. A template using this item can be accessed by selecting **Insert** > **Template** > **SP_DiffT** from the Schematic window.

2. SP_Diff sets up a swept-frequency S-parameter analysis. Ports 1 and 2 of the Network Analyzer are ungrounded, so the S-parameters of differential networks can be simulated without using baluns.
3. This is a simulation component. No other simulation or control components are needed.

SP_FET (S-Parameters vs. Bias for FET)

Symbol



Parameters

Parameter Name	Description
VGS_start	initial gate-source voltage
VGS_stop	last gate-source voltage
VGS_points	number of gate-source current values
VDS_start	initial drain-source voltage
VDS_stop	last drain-source voltage
VDS_points	number of drain-source voltage values
AnalysisFreq	single S-parameter analysis frequency
Port1Z	port 1 port impedance (complex)
Port2Z	port 2 port impedance (complex)

Notes

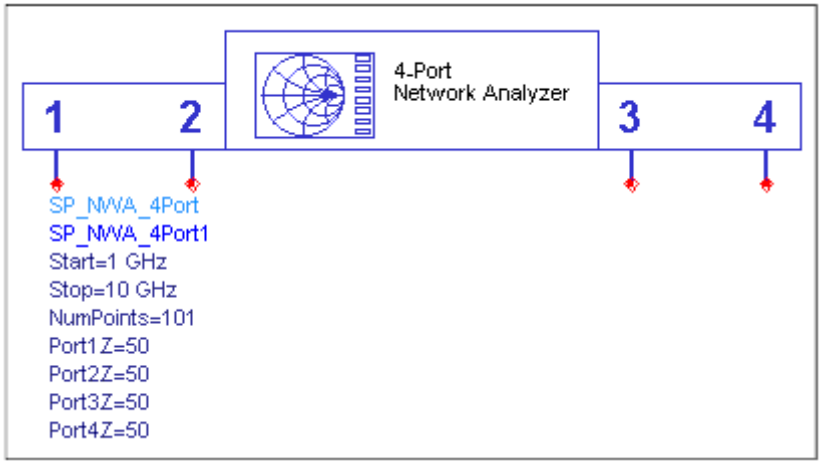
1. A template using this item can be accessed by selecting **Insert** > **Template** >

SP_FET_T from the Schematic window.

2. SP_FET sets up an S-parameter analysis at one frequency with two swept voltage, one each for the for the gate and drain bias. This component helps select an operating point for desired gain. Connect it to a field effect transistor, as indicated in the schematic symbol.
3. This is a simulation component. No other simulation or control components are needed.

SP_NWA_4Port (4-Port Network Analyzer)

Symbol



Parameters

Parameter Name	Description	Units
Start	start frequency	Ghz
Stop	stop frequency	GHz
NumPoints	number of frequency points	
Port1Z	port 1 port impedance (complex)	
Port2Z	port 2 port impedance (complex)	
Port3Z	port 3 port impedance (complex)	
Port4Z	port 4 port impedance (complex)	

Notes

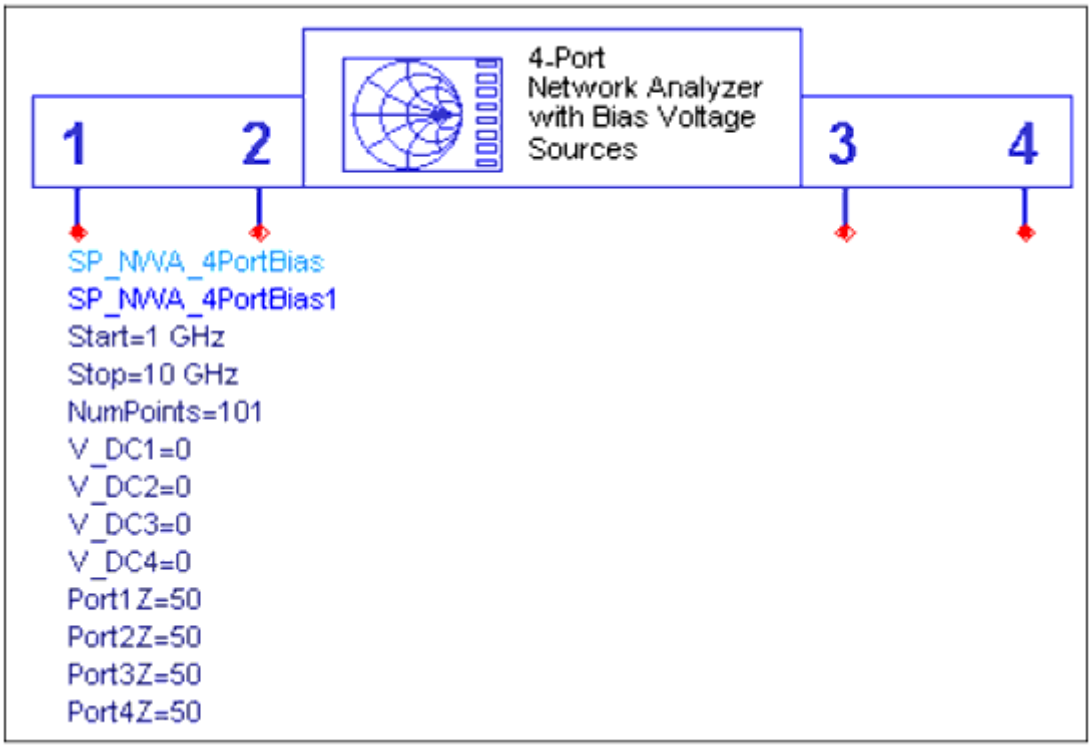
1. A template using this item can be accessed by selecting **Insert** > **Template** > **SP_NWA_4PortT** from the Schematic window.
2. SP_NWA_4Port simulates a four-port S-parameter network analyzer. The extra ports

are useful for testing multi-port devices and for optimizing the 2-port devices side by side. Each port has a separate ideal bias tee to allow a device to be biased directly from this component.

3. This is a simulation component. No other simulation or control components are needed.

SP_NWA_4PortBias (4-Port Network Analyzer with Bias Sources)

Symbol



Parameters

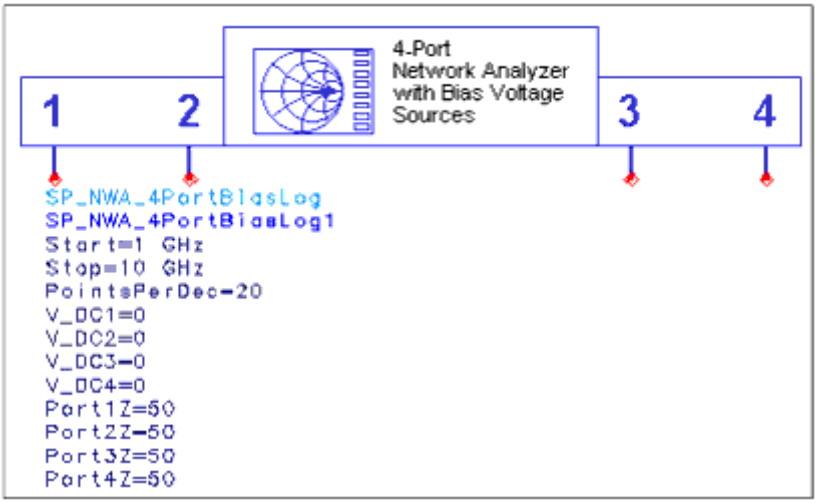
Parameter Name	Description	Units
Start	start frequency	GHz
Stop	stop frequency	GHz
NumPoints	number of frequency points	
V_DC1	port 1 bias voltage	
V_DC2	port 2 bias voltage	
V_DC3	port 3 bias voltage	
V_DC4	port 4 bias voltage	
Port1Z	port 1 port impedance (complex)	
Port2Z	port 2 port impedance (complex)	
Port3Z	port 3 port impedance (complex)	
Port4Z	port 4 port impedance (complex)	

Notes

1. A template using this item can be accessed by selecting **Insert** > **Template** > **SP_NWA_4Port_BiasT** from the Schematic window.
2. SP_NWA_4Port_Bias simulates a four-port S-parameter network analyzer. The extra ports are useful for testing multi-port devices and for optimizing the 2-port devices side by side. Each port has a separate ideal bias tee to allow a device to be biased directly from this component.
3. This component is identical to the SP_NWA_4Port, except that it has ideal bias tees at each port.
4. This is a simulation component. No other simulation or control components are needed.

SP_NWA_4PortBiasLog (4-Port Network Analyzer with Bias, Log Sweep)

Symbol



Parameters

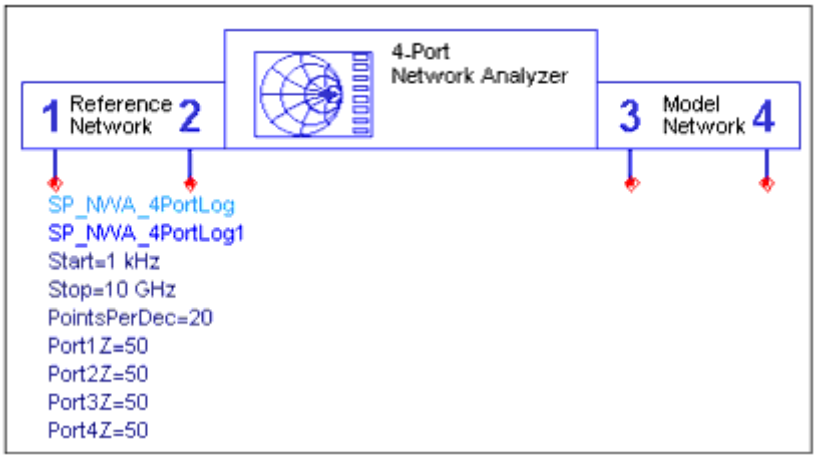
Parameter Name	Description	Units
Start	start frequency	GHz
Stop	stop frequency	GHz
PointsperDec	number of frequency points per decade	
V_DC1	port 1 bias voltage	
V_DC2	port 2 bias voltage	
V_DC3	port 3 bias voltage	
V_DC4	port 4 bias voltage	
Port1Z	port 1 port impedance (complex)	
Port2Z	port 2 port impedance (complex)	
Port3Z	port 3 port impedance (complex)	
Port4Z	port 4 port impedance (complex)	

Notes

1. A template using this item can be accessed by selecting **Insert** > **Template** > **SP_NWA_4Port_BiasLogT** from the Schematic window.
2. SP_NWA_4Port_BiasLog is identical to the SP_NWA_4Port_Bias, except that it has ideal bias tees at each port.
3. This is a simulation component. No other simulation or control components are needed.

SP_NWA_4PortLog (4-Port Network Analyzer, Log Sweep)

Symbol



Parameters

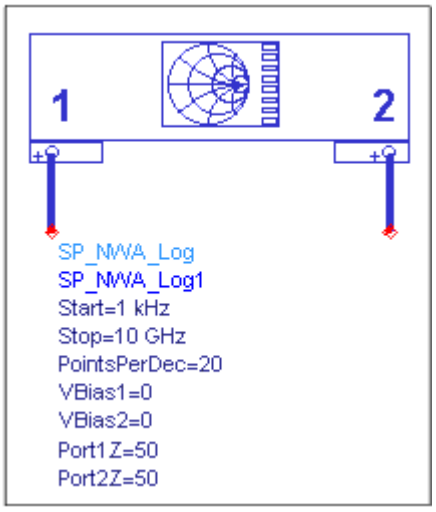
Parameter Name	Description	Units
Start	start frequency	kHz
Stop	stop frequency	GHz
PointsPerDec	frequency points per decade	
Port1Z	port 1 port impedance (complex)	
Port2Z	port 2 port impedance (complex)	
Port3Z	port 3 port impedance (complex)	
Port4Z	port 4 port impedance (complex)	

Notes

1. A template using this item can be accessed by selecting **Insert** > **Template** > **SP_NWA_4PortLogT** from the Schematic window.
2. SP_NWA_4Port_Log is identical to SP_NWA_4Port, except the frequency is swept logarithmically. It simulates a four-port S-parameter network analyzer. The extra ports are useful for testing multi-port devices and for optimizing the 2-port devices side by side. Each port has a separate ideal bias tee to allow a device to be biased directly from this component.
3. This is a simulation component. No other simulation or control components are needed.

SP_NWA_Log (Network Analyzer for S-Parameters, Log Sweep)

Symbol



Parameters

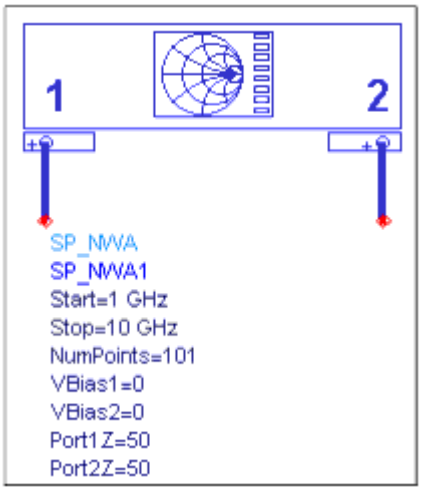
Parameter Name	Description	Units
Start	start frequency	kHz
Stop	stop frequency	GHz
PointsPerDec	frequency points per decade	
Vbias1	port 1 bias voltage	
Vbias2	port 2 bias voltage	
Port1Z	port 1 port impedance (complex)	
Port2Z	port 2 port impedance (complex)	

Notes

1. A template using this item can be accessed by selecting **Insert** > **Template** > **SP_NWA_LogT** from the Schematic window.
2. SP_NWA_LOG is identical to SP_NWA except that frequency is swept logarithmically. It emulates a two-port S-parameter network analyzer. Each port has a separate ideal bias tee to allow a device to be biased directly from the component.
3. This is a simulation component. No other simulation or control components are needed.

SP_NWA (Network Analyzer for S-Parameters)

Symbol



Parameters

Parameter Name	Description	Units
Start	start frequency	GHz
Stop	stop frequency	GHz
NumPoints	number of frequency points	
Vbias1	port 1 bias voltage	
Vbias2	port 1 bias voltage	
Port1Z	port 1 port impedance (complex)	
Port2Z	port 2 port impedance (complex)	

Notes

1. A template using this item can be accessed by selecting **Insert** > **Template** > **SP_NWA_T** from the Schematic window.
2. SP_NWA emulates a two-port S-parameter network analyzer. each port has a separate ideal bias tee to allow a device to be biased directly from this component.
3. This is a simulation component. No other simulation or control components are needed.