Notice

The information contained in this document is subject to change without notice.

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

Warranty

A copy of the specific warranty terms that apply to this software product is available upon request from your Agilent Technologies representative.

Restricted Rights Legend

Use, duplication or disclosure by the U. S. Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 for DoD agencies, and subparagraphs (c) (1) and (c) (2) of the Commercial Computer Software Restricted Rights clause at FAR 52.227-19 for other agencies.

Agilent Technologies
395 Page Mill Road
Palo Alto, CA 94304 U.S.A.


Acknowledgments

Mentor Graphics is a trademark of Mentor Graphics Corporation in the U.S. and other countries.

Microsoft®, Windows®, MS Windows®, Windows NT®, and MS-DOS® 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.

Java™ is a U.S. trademark of Sun Microsystems, Inc.
Contents

1 Controlled Sources
   CCCS (Linear Current-Controlled Current Source) ........................................... 1-2
   CCCS_Z (Current-Controlled Current Source, Z-Domain) .................................. 1-4
   CCVS (Linear Current-Controlled Voltage Source) ........................................... 1-6
   CCVS_Z (Current-Controlled Voltage Source, Z-Domain) .................................. 1-8
   VCCS (Linear Voltage-Controlled Current Source) ........................................... 1-10
   VCCS_Z (Voltage-Controlled Current Source, Z-Domain) .................................. 1-12
   VCVS (Linear Voltage-Controlled Voltage Source) ........................................... 1-14
   VCVS_Z (Voltage-Controlled Voltage Source, Z-Domain) .................................. 1-16

2 Frequency Domain Sources
   Introduction ................................................................................................. 2-1
   I_AC (AC current source) ............................................................................. 2-2
   I_DC (DC current source) ........................................................................... 2-3
   I_1Tone (Current Source, Single Frequency) ................................................. 2-4
   I_nHarm (Current Source, Fundamental Frequency with N-Harmonics) .......... 2-7
   I_nTone (Current Source, N Frequencies and Amplitudes) ............................ 2-10
   I_HB_Dataset (Current Source, HB Dataset Variable) ..................................... 2-13
   I_SpectrumDataset (Current Source, Frequency Spectrum Defined in Dataset) .. 2-14
   OSCwPhNoise (Oscillator with Phase Noise) .................................................. 2-16
   P_AC (AC Power Source) ............................................................................. 2-18
   P_1Tone (Power Source, Single Frequency) ................................................... 2-20
   P_nHarm (Power Source, Fundamental Frequency with N-Harmonics) .......... 2-24
   P_nTone (Power Source, N Frequencies and Power Levels) ............................ 2-28
   P_SpectrumDataset (Power Source, Frequency Spectrum Defined in Dataset) .. 2-31
   V_1Tone (Voltage Source, Single Frequency) ................................................. 2-33
   V_AC (AC Voltage Source) ........................................................................... 2-36
   V_DC (DC Voltage Source) ........................................................................... 2-37
   V_nHarm (Voltage Source, Fundamental Frequency with N-Harmonics) .......... 2-38
   V_nTone (Voltage Source, N Frequencies and Amplitudes) ............................ 2-41
   V_SpectrumDataset (Voltage Source, Frequency Spectrum Defined in Dataset) .. 2-44
   Vf_BitSeq (Fourier Transform of Bit Sequence Waveform) ............................. 2-46
   Vf_Pulse (Voltage Source, Fourier Series Expansion of Period Pulse Wave) .... 2-49
   Vf_Sawtooth (Voltage Source, Fourier Series Expansion of Periodic Sawtooth) .. 2-51
   Vf_Square (Voltage Source, Fourier Series Expansion of Period Square Wave) .. 2-53
   Vf_Triangle (Voltage Source, Fourier Series Expansion of Period Triangle Wave) .. 2-55
   V_HB_Dataset (Voltage Source, HB Dataset Variable) ..................................... 2-57

3 Modulated Sources
   P1RF_3GPP_Uplink (Pwr Src, RF Carrier Modulated by 3GPP Uplink Signal) .... 3-2
PtRF_CDMA_ESG_FWD (Pwr Src, RF Carrier Modulated by ESG Fwd Link CDMA Signal) ................................................................................ 3-4
PtRF_CDMA_ESG_REV (Pwr Src, RF Carrier Modulated by ESG Rev. Link CDMA Signal) ................................................................................ 3-6
PtRF_CDMA_IS95_FWD (Pwr Src, RF Carrier Modulated by IS95 Fwd Link CDMA Signal) ................................................................................. 3-8
PtRF_CDMA_IS95_REV (Pwr Src, RF Carrier Modulated by IS95 Rev. Link CDMA Signal) ................................................................................ 3-10
PtRF_CDMA2K_REV (Pwr Src, RF Carrier Modulated by CDMA2K Reverse Link Signal) .............................................................................. 3-12
PtRF_DECT (Pwr Src, RF Carrier Modulated by DECT Signal) ...................................................................................................................... 3-14
PtRF_EDGE_Uplink (Pwr Src, RF Carrier Modulated by EDGE Uplink Signal) ................................................................................................. 3-15
PtRF_GSM (Pwr Src, RF Carrier Modulated by GSM Signal) .......................................................................................................................... 3-17
PtRF_NADC (Pwr Src, RF Carrier Modulated by NADC Signal) .................................................................................................................... 3-18
PtRF_PHS (Pwr Src, RF Carrier Modulated by PHS Signal) ............................................................................................................................ 3-20
PtRF_Pulse (Pwr Src, RF Pulse Train) ....................................................................................................................................................... 3-21
PtRF_Step (Pwr Src, RF Step) ................................................................................................................................................................. 3-23
VtRF_Pulse (Voltage Source, RF Pulse) .................................................................................................................................................. 3-24
VtRF_Step (Voltage Source, RF Step) ................................................................................................................................................... 3-26

4 Noise Sources
I_Noise (Noise Current Source) ............................................................................... 4-2
I_NoiseBD (Bias-Dependent Noise Current Source) ......................................................................................................................................................... 4-3
NoiseCorr (Noise Source Correlation) ........................................................................ 4-5
Noisy2Port (Linear Noisy 2port Network) ................................................................... 4-6
V_Noise (Noise Voltage Source) .............................................................................. 4-7
V_NoiseBD (Bias-dependent Noise Voltage Source) .................................................................................................................................................. 4-9

5 Time Domain Sources
Introduction............................................................................................................... 5-1
ClockWjitter (Voltage Source: Clock with Jitter).......................................................... 5-2
I_DC (DC current source).......................................................................................... 5-4
ItDataset (Current Source, Time Domain Waveform Defined in Dataset) ................. 5-5
ItExp (Current Source, Exponential Decay) ................................................................ 5-9
ItPulse (Current Source, Pulse with Linear, Cosine or Error Function Edge Shape) ........................................................................................................ 5-11
ItPWL (Current Source, Piecewise Linear) ................................................................ 5-14
ItSFFM (Current Source, Decaying Single-Frequency FM Wave) ................................ 5-15
ItSine (Current Source, Decaying Sine Wave) ................................................................ 5-16
ItStep (Current Source, Step) .................................................................................. 5-18
ItUserDef (Current Source, User-Defined) ................................................................ 5-19
V_DC (DC Voltage Source) ..................................................................................... 5-21
VtBitSeq (Voltage Source, Pseudo Random Pulse Train Defined at Continuous Time by Bit Sequence) ...................................................... 5-22
VtDataset (Voltage Source, Time Domain Waveform Defined in Dataset) .......... 5-24
VtExp (Voltage Source, Exponential Decay) ...................................................... 5-28
VtImpulseDT (Voltage Source, Impulse Train Defined at Discrete Time Steps) .... 5-30
VtLFSR_DT (Voltage Source, Pseudo-Random Pulse Train Defined at
Discrete Time Steps) ..................................................................................... 5-32
VtOneShot (Voltage Source, Retriggerable Pulse Train) ................................... 5-38
VtPulse (Voltage Source, Pulse with Linear, Cosine, or Error Function
Edge Shape) ..................................................................................................... 5-39
VtPulseDT (Voltage Source, Pulse Train Defined at Discrete Time Steps) .......... 5-42
VtPWL (Voltage Source, Piecewise Linear) ......................................................... 5-44
VtRetrig (Voltage Source, Retriggerable, User-Defined Waveform) ................. 5-45
VtSFFM (Voltage Source, Single Frequency FM, SFFM Wave) ......................... 5-47
VtSine (Voltage Source, Decaying Sine Wave) .................................................. 5-48
VtStep (Voltage Source, Step) ........................................................................ 5-50
VtUserDef (Voltage Source, User-Defined) ...................................................... 5-51

Index
Chapter 1: Controlled Sources
**Controlled Sources**

**CCCS (Linear Current-Controlled Current Source)**

**Symbol**

![CCCS Symbol](image)

**Parameters**

- \( G \) = complex current gain; for example, \( \text{polar}(10,45) \), or \( P(j \times \omega)/Q(j \times \omega) \)
- \( R_1 \) = input resistance, in ohms
- \( R_2 \) = output resistance, in ohms
- \( F \) = frequency at which current gain magnitude is down by 3 dB, in hertz
- \( T \) = time delay associated with current gain, in seconds

**Range of Usage**

For ideal current source use the following settings:

<table>
<thead>
<tr>
<th>Setting</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>( F = 0 )</td>
<td>( F = \infty )</td>
</tr>
<tr>
<td>( T = 0 )</td>
<td>( T = 0 )</td>
</tr>
<tr>
<td>( R_1 = 0 )</td>
<td>( R_1 = 0 )</td>
</tr>
<tr>
<td>( R_2 = 0 )</td>
<td>( R_2 = \infty )</td>
</tr>
</tbody>
</table>

**Notes/Equations**

1. This is a purely linear, dependent source model. Nonlinear controlled sources are available in the Nonlinear Devices library.
2. This source is assumed to be noiseless.

3. \[ \beta(f) = G \times \frac{e^{-j(2\pi f T)}}{1 + j(f/F)} \quad \text{(for } F > 0) \]

   \[ \beta(f) = G \times e^{-j(2\pi f T)} \quad \text{(for } F = 0) \]
where

\[ f = \text{simulation frequency in hertz} \]
\[ F = \text{reference frequency in hertz} \]
\[ T = \text{CCCS time delay in seconds} \]
\[ \beta(f) = \text{frequency-dependent current gain} \]

4. For time-domain analysis, the frequency-domain analytical model is used.

5. This source has no default artwork associated with it.
CCCS_Z (Current-Controlled Current Source, Z-Domain)

Symbol

Parameters
- Gain = constant gain term
- Num = numerator coefficients of transfer function
- Den = denominator coefficients of transfer function
- TimeStep = sampling time period

Notes/Equations

1. This model is a current source whose output is linearly proportional to its short circuit input current. It is similar to the CCCS model; instead of specifying the current gain transfer function $A_i$ as a function of frequency, this model allows the transfer function to be defined as a rational polynomial in the Z-Domain. This model can be used in all analysis modes, but becomes especially efficient in the transient and circuit envelope time-domain modes where direct recursive convolution is used instead of inverse FFT convolution. In other modes, the rational polynomial is simply evaluated at

$$z^{-1} = e^{-j \times 2 \pi \times \text{freq} \times \text{TimeStep}}$$

where $\text{freq}$ is the analysis frequency.

The transfer function is

$$A_i(z) = \frac{l_{out}(z)}{l_{in}(z)} = \frac{\text{Gain} \times a_0 + a_1 \times z^{-1} + \ldots + a_{M-1} \times z^{M-1} + a_M \times z^{-M}}{b_0 + b_1 \times z^{-1} + \ldots + b_{N-1} \times z^{N-1} + b_N \times z^{-N}}$$

The $a_i$ coefficients are defined by the Num parameter list, which can be created by either a LIST ARRAY equation or a DATASET ARRAY equation. $a_0$ is first and $a_M$ is last in the list. Similarly, the $b_i$ coefficients are defined by the Den
parameter list. The value if $b_0$ must not be 0. If the Den parameter is not given, it is assumed to equal 1.0.

2. The Gain parameter must be a constant and must not depend on frequency. It, and the polynomial coefficients, should not be complex valued.

3. The TimeStep parameter determines the unit delay time of each $z^{-1}$ block and, in a sampled system, would correspond to the sampling interval. The input to this transfer function is not automatically sampled in this model. For a sampled signal, preface this model with either the sample-hold or sampler model. Note that the frequency response of the Z-domain transfer function is cyclical and repeats every $1/$TimeStep Hertz.

4. In circuit envelope simulation, only the baseband spectral component is filtered by the transfer function.
CCVS (Linear Current-Controlled Voltage Source)

**Symbol**

![Symbol diagram]

**Parameters**

- \( G \) = complex transresistance; for example, polar(10,45) or \( P(j\omega)/Q(j\omega) \)
- \( T \) = time delay associated with transresistance, in seconds
- \( R_1 \) = input resistance, in ohms
- \( R_2 \) = output resistance, in ohms
- \( F \) = frequency at which transresistance magnitude is down by 3dB, in hertz
- \( \text{ImpNoncausalLength} \) = non-causal function impulse response order (value type: integer)
- \( \text{ImpMode} \) = convolution mode (value type: integer)
- \( \text{ImpMaxFreq} \) = maximum frequency to which device is evaluated, in hertz
- \( \text{ImpDeltaFreq} \) = sample spacing in frequency, in hertz
- \( \text{ImpMaxOrder} \) = maximum impulse response order (value type: integer)
- \( \text{ImpWindow} \) = smoothing window (value type: integer)
- \( \text{ImpRelTol} \) = relative impulse response truncation factor
- \( \text{ImpAbsTol} \) = absolute impulse response truncation factor
Range of Usage

For ideal current source use the following settings:

<table>
<thead>
<tr>
<th>Setting</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>F = 0</td>
<td>F = ∞</td>
</tr>
<tr>
<td>T = 0</td>
<td>T = 0</td>
</tr>
<tr>
<td>R1 = 0</td>
<td>R1 = 0</td>
</tr>
<tr>
<td>R2 = 0</td>
<td>R2 = 0</td>
</tr>
</tbody>
</table>

Notes/Equations

1. This is a purely linear, dependent source model. (Nonlinear controlled sources are available in the Eqn Based-Nonlinear library.)

2. This source is assumed to be noiseless.

3. \( f(f) = G \times \frac{e^{-j(2\pi F f)}}{1 + j(\frac{f}{F})} \) (for \( F > 0 \))

\[ (f) = G \times e^{-j(2\pi f T)} \] (for \( F = 0 \))

where

- \( R(f) \) = frequency-dependent transresistance
- \( f \) = simulation frequency in hertz
- \( F \) = reference frequency in hertz
- \( T \) = CCVS time delay in seconds

4. For transient analysis, the transresistance is independent of frequency, and there is no phase shift or time delay associated with the transresistance.

5. For convolution analysis, the frequency-domain analytical model is used.

6. This source has no default artwork associated with it.
CCVS_Z (Current-Controlled Voltage Source, Z-Domain)

Symbol

Parameters

Gain = constant gain term
Num = numerator coefficients of transfer function
Den = denominator coefficients of transfer function
TimeStep = sampling time period

Notes/Equations

1. This model is a voltage source whose output is linearly proportional to its short circuit input current. Similar to the CCVS model, instead of specifying the transfer function \( Z_{21} \) as a function of frequency, this model allows the transfer function to be defined as a rational polynomial in the Z-Domain.

This model can be used in all analysis modes, but becomes especially efficient in the transient and circuit envelope time-domain modes where direct recursive convolution is used instead of inverse FFT convolution. In the other modes, the rational polynomial is simply evaluated at

\[ z^{-1} = e^{-j \times 2\pi \times \text{freq} \times \text{TimeStep}} \]

where freq is the analysis frequency.

The transfer function is

\[
Z_{21}(z) = \frac{V_{\text{out}}(z)}{I_{\text{in}}(z)} = \text{Gain} \times \frac{a_0 + a_1 \times z^{-1} + \ldots + a_{M-1} \times z^{M-1} + a_M \times z^M}{b_0 + b_1 \times z^{-1} + \ldots + b_{N-1} \times z^{N-1} + b_N \times z^N}
\]

The \( a_i \) coefficients are defined by the Num parameter list, which can be created by either a LIST ARRAY equation or a DATASET ARRAY equation. \( a_0 \) is first
and $a_M$ is last in the list. Similarly, the $b_i$ coefficients are defined by the Den parameter list. The value if $b_0$ must not be 0. If the Den parameter is not given, it is assumed to equal 1.0.

2. The Gain parameter must be a constant and not depend on frequency. It, and the polynomial coefficients, should not be complex valued.

3. The TimeStep parameter determines the unit delay time of each $z^{-1}$ block and, in a sampled system, would correspond to the sampling interval. The input to this transfer function is not automatically sampled in this model. For a sampled signal, preface this model with either the sample-hold or sampler model. Note that the frequency response of the Z-domain transfer function is cyclical and repeats every $1/$TimeStep Hertz.

4. In circuit envelope simulation, only the baseband spectral component is filtered by the transfer function.
Controlled Sources

VCCS (Linear Voltage-Controlled Current Source)

Symbol

Parameters

\( G = \) complex transconductance; for example, polar(15,45), or \( P(j\omega)/Q(j\omega) \)

\( R_1 = \) input resistance, in ohms

\( R_2 = \) output resistance, in ohms

\( F = \) frequency at which transconductance magnitude is down by 3dB, in hertz

Range of Usage

<table>
<thead>
<tr>
<th>Setting</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>( F = 0 )</td>
<td>( F = \infty )</td>
</tr>
<tr>
<td>( T = 0 )</td>
<td>( T = 0 )</td>
</tr>
<tr>
<td>( R_1 = 0 )</td>
<td>( R_1 = \infty )</td>
</tr>
<tr>
<td>( R_2 = 0 )</td>
<td>( R_2 = \infty )</td>
</tr>
</tbody>
</table>

Notes/Equations

1. This is a purely linear, dependent source model. Nonlinear controlled sources are available in the Nonlinear Devices library.

2. This source is assumed to be noiseless.

3. \( G(f) = G \times e^{-j(2\pi f T)} \) (for \( F_0 \neq 0 \))

\[ G(f) = G \times e^{-j(2\pi f T)} \quad (\text{for } F = 0) \]

where
\( f = \) simulation frequency in hertz
\( F = \) reference frequency in hertz
\( T = \) VCCS time delay in seconds
\( G(f) = \) frequency-dependent transconductance

4. For time-domain analysis, the frequency-domain analytical model is used.
5. This component has no default artwork associated with it.
Controlled Sources

**VCCS_Z (Voltage-Controlled Current Source, Z-Domain)**

**Symbol**

![Symbol](image)

**Parameters**
- **Gain** = constant gain term
- **Num** = numerator coefficients of transfer function
- **Den** = denominator coefficients of transfer function
- **TimeStep** = sampling time period

**Notes/Equations**

1. This model is a voltage source whose output is linearly proportional to its short circuit input current. Similar to the VCCS model, instead of specifying the transfer function $Z_{21}$ as a function of frequency, this model allows the transfer function to be defined as a rational polynomial in the Z-Domain. This model can be used in all simulations, but becomes especially efficient in the transient and circuit envelope time-domain modes where direct recursive convolution is used instead of inverse FFT convolution. In the other modes, the rational polynomial is simply evaluated at

$$z^{-1} = e^{-j \times 2\pi \times \text{freq} \times \text{TimeStep}}$$

where **freq** is the analysis frequency.

The transfer function is

$$Z_{21}(z) = \frac{V_{out}(z)}{I_{in}(z)} = \text{Gain} \times \frac{a_0 + a_1 z^{-1} + \ldots + a_{M-1} z^{-M-1} + a_M z^{-M}}{b_0 + b_1 z^{-1} + \ldots + b_{N-1} z^{-N-1} + b_N z^{-N}}$$

The $a_i$ coefficients are defined by the Num parameter list, which can be created by either a LIST ARRAY equation or a DATASET ARRAY equation. $a_0$ is first and $a_M$ is last in the list. Similarly, the $b_i$ coefficients are defined by the Den...
parameter list. The value if $b_0$ must not be 0. If the Den parameter is not given, it is assumed to equal 1.0.

2. The Gain parameter must be a constant and not depend on frequency. It, and the polynomial coefficients, should not be complex valued.

3. The TimeStep parameter determines the unit delay time of each $z^{-1}$ block, and, in a sampled system, would correspond to the sampling interval. The input to this transfer function is not automatically sampled in this model. For a sampled signal, preface this model with either the sample-hold or sampler model. Note that the frequency response of the Z-domain transfer function is cyclical and repeats every $1/$TimeStep Hertz.

   The default value for TimeStep is timestep, which is a global variable. If using Circuit Envelope analysis, it is set using the TimeStep parameter. For AC simulation, TimeStep is zero.

4. In circuit envelope analysis, only the baseband spectral component is filtered by the transfer function.
VCVS (Linear Voltage-Controlled Voltage Source)

Symbol

Parameters

- $G =$ complex voltage gain; for example, polar(10,45), or $P(j \omega)/Q(j \omega)$
- $R_1 =$ input resistance, in ohms
- $R_2 =$ output resistance, in ohms
- $F =$ frequency at which voltage gain magnitude is down by 3 dB, in hertz

Range of Usage

<table>
<thead>
<tr>
<th>Setting</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>$F = 0$</td>
<td>$F = \infty$</td>
</tr>
<tr>
<td>$T = 0$</td>
<td>$T = 0$</td>
</tr>
<tr>
<td>$R_1 = 0$</td>
<td>$R_1 = \infty$</td>
</tr>
<tr>
<td>$R_2 = 0$</td>
<td>$R_2 = 0$</td>
</tr>
</tbody>
</table>

Notes/Equations

1. This is a purely linear, dependent source model. Nonlinear controlled sources are available in the Nonlinear Devices library.

2. This component is assumed to be noiseless.

3. Voltage gain =

$$
\mu(f) = G \times \frac{e^{-j(2\pi f T)}}{1 + j \frac{f}{F}} \quad \text{(for } F \neq 0)$$
\[ \mu(f) = G \times e^{-j(2\pi f T)} \quad \text{(for } F = 0) \]

where
- \( f \) = simulation frequency in hertz
- \( F \) = reference frequency in hertz
- \( T \) = VCVS time delay in seconds
- \( \mu(f) \) = frequency-dependent voltage gain

4. For time-domain analysis, the frequency-domain analytical model is used.
5. This component has no default artwork associated with it.
**VCVS_Z (Voltage-Controlled Voltage Source, Z-Domain)**

**Symbol**

```
1 + 2
\[ \cdot \cdot \cdot \]
```

**Parameters**

- **Gain** = constant gain term
- **Num** = numerator coefficients of transfer function
- **Den** = denominator coefficients of transfer function
- **TimeStep** = sampling time period

**Notes/Equations**

1. This model is a voltage source whose output is linearly proportional to its open circuit input voltage. Similar to the VCVS model, instead of specifying the voltage gain transfer function \( A_v \) as a function of frequency, this model allows the transfer function to be defined as a rational polynomial in the Z-Domain. This model is usable in all analysis modes, but becomes especially efficient in the transient and circuit envelope time-domain modes, where direct recursive convolution is used instead of inverse FFT convolution. In the other analysis modes, the rational polynomial is simply evaluated at

\[
\begin{align*}
\hat{z}^{-1} &= e^{-j \times 2 \pi \times \text{freq} \times \text{TimeStep}},
\end{align*}
\]

where freq is the analysis frequency.

The transfer function is

\[
\frac{V_{out}(z)}{V_{in}(z)} = \text{Gain} \times \frac{a_0 + a_1 \times z^{-1} + \ldots + a_{M-1} \times z^{M-1} + a_M \times z^{-M}}{b_0 + b_1 \times z^{-1} + \ldots + b_{N-1} \times z^{N-1} + b_N \times z^{-N}}
\]

The \( a_i \) coefficients are defined by the Num parameter list, which can be created by either a LIST ARRAY equation or a DATASET ARRAY equation.
a₀ is first in the list and aₘ is last. Similarly, the bₗ coefficients are defined by
the Den parameter list. The value if b₀ must not be 0. If the Den parameter is
not given, it is assumed to equal 1.0.

2. The Gain parameter must be a constant and must not depend on frequency. It,
and the polynomial coefficients, should not be complex valued.

3. The TimeStep parameter determines the unit delay time of each z⁻¹ block, and,
in a sampled system, would correspond to the sampling interval. The input to
this transfer function is not automatically sampled in this model.

For a sampled signal, preface this model with either the sample-hold or
sampler model. Note that the frequency response of the Z-domain transfer
function is cyclical and repeats every 1/TimeStep Hertz.

The default value for TimeStep is timestep, which is a global variable. If using
Circuit Envelope analysis, it is set using the TimeStep parameter. For AC
simulation, TimeStep is zero.

4. In circuit envelope analysis, only the baseband spectral component is filtered by
the transfer function.
Controlled Sources
Chapter 2: Frequency Domain Sources

Introduction

A frequency domain source generates a periodic waveform or a superposition of periodic waveforms. Frequency domain sources are often used as stimuli to find the steady-state response of a circuit.

Independent voltage sources, current sources, and power source are provided in Advanced Design System. Power sources have built-in impedances that can also be used as reference impedance for S-parameter simulation.

Frequency domain sources can be used in all simulations. In S-parameter simulation, voltage sources are treated as short circuits, current sources are treated as open circuits, and power sources are treated as impedances.

Amplitudes in frequency domain sources can be set to complex values such as $V = \text{Re} + j \times \text{Im}$, $I = \text{polar (Mag, Angle)}$, $P = \text{polar(dbmtow(dBm), Angle)}$. When these sources are used in baseband transient simulation, only the real part of the signal is used, the imaginary part is dropped.
**Frequency Domain Sources**

**I_AC (AC current source)**

**Symbol**

```
1
\[\text{\downarrow}\]
2
```

**Parameters**

- **Idc** = dc current
- **Iac** = ac current; value used for ac analysis only
- **Freq** = frequency; default is global analysis frequency
- **I_Noise** = noise current magnitude, per sqrt(Hz)

**Notes/Equations**

1. **I_AC** is an ideal ac current source. Positive current flows into the source at pin 1 and out of the source at pin 2.

2. This source is used in all simulations. When not in use, it is treated as an open circuit.

3. **Table 2-1** lists the DC operating point parameters that can be sent to the dataset.

**Table 2-1. DC Operating Point Information**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

4. For general information regarding frequency domain sources, refer to the “Introduction” on page 2-1.
I_DC (DC current source)

Symbol

Parameters

I_{dc} = dc current

I_{ac} = ac current; value used for ac analysis only

Notes/Equations

1. I_DC is an ideal dc current source. Positive current flows into the source at pin 1 and out of the source at pin 2.

2. This source is used in all simulations. When not in use, it is treated as an open circuit.

3. Table 2-2 lists the DC operating point parameters that can be sent to the dataset.

Table 2-2. DC Operating Point Information

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

4. For general information regarding frequency domain sources, refer to the “Introduction” on page 2-1.
Frequency Domain Sources

I_1Tone (Current Source, Single Frequency)

Symbol

Parameters

I = current at center frequency
Freq = center frequency
I_USB = current of upper sideband small signal tone; value used for small-signal mixer simulation
I_LSB = current of lower sideband small-signal tone; value used for small-signal mixer simulation
Idc = dc current
Iac = ac current; value used for ac analysis
FundIndex = frequency index; an alternate way of specifying center frequency, used in MDS
PhaseNoise = list of offset frequency, phase noise pairs
Other = output string to netlist

Range of Usage
Freq > 0

Notes/Equations

1. This current source is defined by its frequency and its current and can be used in all simulations. The phase of the source is specified by a complex value I, such as I=polar(1mA, 45).

For ac simulations, only Iac is used—all other parameters are ignored. When frequency conversion ac analysis is performed, the Freq parameter is used to set the frequency of the source. For harmonic balance and envelope, the Freq parameter identifies the closest analysis frequency. If the frequency is not close
enough, a warning is generated and the source current is set to 0 for that analysis. In envelope analysis, the frequency difference can be up to 0.5/timestep and the given frequency is still properly generated.

This source can also be used in transient analysis to generate an arbitrary waveform at the specified carrier frequency. When this source represents a real only, baseband current (that is, transient or the baseband part of an envelope signal), then only the real part of the signal is generated. Otherwise, the full complex value of I is used to define both the amplitude and phase relationships.

2. For time-domain analyses, transient and envelope, the I current parameter can be an expression of time. When a source with a time varying expression is used in a steady-state analysis such as harmonic balance, then its value at time=0 is used. Care must be exercised if the I current parameter is a function of frequency, since this is not fully supported in all analysis modes.

3. A dc term can be defined on this device.

4. In small signal mixer simulation, the simulator sets the operating point of the circuit using the carrier signal alone as the source. The two sidebands must be set such that they have no effect on the operating point of the circuit.

5. Positive current flows into pin 1 and out of pin 2.

6. Table 2-3 lists the DC operating point parameters that can be sent to the dataset.

Table 2-3. DC Operating Point Information

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

7. The phase noise is specified using the list function:

\[ \text{list}(10\text{Hz}, -20\text{dB}, 100\text{Hz}, -40\text{dB}, 1\text{kHz}, -50\text{dB}) \]

It consists of pairs of offset frequencies in Hertz and phase noise values in \( \text{dBc/Hz} \). When evaluated using offset frequencies that are smaller or larger than those given in the list, the phase noise corresponding to the smallest or largest frequency is used; the data are not extrapolated. No noise is generated if
Frequency Domain Sources

the phase noise is less than -300 dBc/Hz. Any data pair that contains a frequency <= 0 Hz is ignored.

Figure 2-1 shows the phase noise results for the sample data list given above.

---

**Figure 2-1. Phase Noise Results**

Noise is generated by this source in AC, S-parameter and HB analyses. The noise on either side of the carrier is correlated to produce pure phase noise with no amplitude noise. For example, with a carrier frequency of 1 MHz, the noise at 0.99 MHz and 1.01 MHz is correlated and has a value that depends on the phase noise at 10 kHz and the magnitude of the carrier signal.

This source generates broadband noise. Noise will be generated at any offset frequency from the carrier. If the offset frequency is larger than the last one in the phase noise list, that value will be used. The phase noise is not extrapolated and does not stop for large offsets.

Noise is not generated by this source during Circuit Envelope or transient analysis.

8. For general information regarding frequency domain sources, refer to the “Introduction” on page 2-1.
I_nHarm (Current Source, Fundamental Frequency with N-Harmonics)

Symbol

Parameters
Freq = fundamental frequency
I = Nth harmonic amplitude
Idc = dc component
Iac = ac current; value used for ac analysis only
FundIndex = frequency index; an alternate way of specifying fundamental frequency used in MDS
PhaseNoise = list of offset frequency, phase noise pairs
Other = output string to netlist

Range of Usage
Freq > 0

Notes/Equations
1. This current source has a fundamental frequency component and N harmonics of the fundamental frequency, where 1 ≤ N < ∞. The phase of each harmonic is specified by a complex I, such as I=polar(1mA, 45).

For ac simulations, only Iac is used—all other parameters are ignored. When frequency conversion ac analysis is performed, the Freq parameter is used to set the frequency of the source. For harmonic balance and envelope, the Freq parameter identifies the closest analysis frequency. If the frequency is not close enough, a warning is generated and the source current is set to 0 for that analysis. In envelope analysis, the frequency difference can be up to 0.5/timestep and the given frequency is still properly generated.

This source can also be used in transient analysis to generate an arbitrary waveform at the specified carrier frequency. When this source represents a real
Frequency Domain Sources

only, baseband current (that is, transient or the baseband part of an envelope signal), then only the real part of the signal is generated.

Otherwise, the full complex value of I is used to define both the amplitude and phase relationships.

2. For time-domain analyses, transient and envelope, the I current parameter can be an expression of time. When a source with a time varying expression is used in a steady-state analysis such as harmonic balance, then its value at time=0 is used. Care must be exercised if the I current parameter is a function of frequency, since this is not fully supported in all analysis modes.

3. A dc term can be defined on this device.

4. In small signal mixer simulation, the simulator sets the operating point of the circuit using the carrier signal alone as the source. The two sidebands must be set such that they have no effect on the operating point of the circuit.

5. Positive current flows into pin 1 and out of pin 2.

6. Table 2-4 lists the DC operating point parameters that can be sent to the dataset.

7. The phase noise is specified using the list function:

   list(10Hz, -20dB, 100Hz,-40dB, 1kHz,-50dB)

   It consists of pairs of offset frequencies in Hertz and phase noise values in dBC/Hz. When evaluated using offset frequencies that are smaller or larger than those given in the list, the phase noise corresponding to the smallest or largest frequency is used; the data are not extrapolated. No noise is generated if the phase noise is less than -300 dBC/Hz. Any data pair that contains a frequency <= 0 Hz is ignored.

   Figure 2-2 shows the phase noise results for the sample data list given above.

Table 2-4. DC Operating Point Information

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

---

2-8 I_nHarm (Current Source, Fundamental Frequency with N-Harmonics)
Noise is generated by this source in AC, S-parameter and HB analyses. The noise on either side of the carrier is correlated to produce pure phase noise with no amplitude noise. For example, with a carrier frequency of 1 MHz, the noise at 0.99 MHz and 1.01 MHz is correlated and has a value that depends on the phase noise at 10 kHz and the magnitude of the carrier signal.

This source generates broadband noise. Noise will be generated at any offset frequency from the carrier. If the offset frequency is larger than the last one in the phase noise list, that value will be used. The phase noise is not extrapolated and does not stop for large offsets.

Noise is not generated by this source during Circuit Envelope or transient analysis.

8. The phase noise is only specified for the fundamental. Phase noise is generated by the rest of the specified harmonics using the fundamental phase noise specification plus 20*log10(N) dB, where N is the harmonic number.

9. For general information regarding frequency domain sources, refer to the “Introduction” on page 2-1.
Frequency Domain Sources

I_nTone (Current Source, N Frequencies and Amplitudes)

Symbol

Parameters
Freq = Nth frequency tone
I = Nth tone amplitude
Idc = dc component
Iac = ac current; value used for ac analysis only
PhaseNoise = list of offset frequency, phase noise pairs
Other = output string to netlist

Range of Usage
Freq > 0

Notes/Equations
1. This current source can have an arbitrary number \((1 \leq N < \infty)\) of harmonically independent tones, and can be used in all simulations. The phase of each tone is specified by a complex \(I\) value such as \(I=\text{polar}(1\text{mA}, 45)\).

For ac simulations, only \(I_{ac}\) is used—all other parameters are ignored. When frequency conversion ac analysis is performed, the \(Freq\) parameter is used to set the frequency of the source. For harmonic balance and envelope, the \(Freq\) parameter identifies the closest analysis frequency. If the frequency is not close enough, a warning is generated and the source current is set to 0 for that analysis. In envelope analysis, the frequency difference can be up to \(0.5/\text{timestep}\) and the given frequency is still properly generated.

This source can also be used in transient analysis to generate an arbitrary waveform at the specified carrier frequency. When this source represents a real only, baseband current (that is, transient or the baseband part of an envelope signal), then only the real part of the signal is generated. Otherwise, the full complex value of \(I\) is used to define both the amplitude and phase relationships.
2. For time-domain analyses, transient and envelope, the I current parameter can be an expression of time. When a source with a time varying expression is used in a steady-state analysis such as harmonic balance, then its value at time=0 is used. Care must be exercised if the I current parameter is a function of frequency, since this is not fully supported in all analysis modes.

3. A dc term can be defined on this device.

4. In small signal mixer simulation, the simulator sets the operating point of the circuit using the carrier signal alone as the source. The two sidebands must be set such that they have no effect on the operating point of the circuit.

5. Positive current flows into pin 1 and out of pin 2.

6. Table 2-5 lists the DC operating point parameters that can be sent to the dataset.

Table 2-5. DC Operating Point Information

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>I_s</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>V_s</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

7. The phase noise is specified using the list function:

list(10Hz, -20dB, 100Hz,-40dB, 1kHz,-50dB)

It consists of pairs of offset frequencies in Hertz and phase noise values in dBc/Hz. When evaluated using offset frequencies that are smaller or larger than those given in the list, the phase noise corresponding to the smallest or largest frequency is used; the data are not extrapolated. No noise is generated if the phase noise is less than -300 dBc/Hz. Any data pair that contains a frequency <= 0 Hz is ignored.

Figure 2-3 shows the phase noise results for the sample data list given above.
Noise is generated by this source in AC, S-parameter and HB analyses. The noise on either side of the carrier is correlated to produce pure phase noise with no amplitude noise. For example, with a carrier frequency of 1 MHz, the noise at 0.99 MHz and 1.01 MHz is correlated and has a value that depends on the phase noise at 10 kHz and the magnitude of the carrier signal.

This source generates broadband noise. Noise will be generated at any offset frequency from the carrier. If the offset frequency is larger than the last one in the phase noise list, that value will be used. The phase noise is not extrapolated and does not stop for large offsets.

Noise is not generated by this source during Circuit Envelope or transient analysis.

8. Phase noise can be specified for each of the independent tones. If the phase noise is not specified for one of the tones, then that tone generates no phase noise but the other tones do.

9. For general information regarding frequency domain sources, refer to the “Introduction” on page 2-1.
I_HB_Dataset (Current Source, HB Dataset Variable)

Symbol

Parameters

Dataset = Dataset filename
Variable = Dataset variable (string and reference or file-based)
Idc = DC component (default: 0 mA)
Iac = AC current, use polar () for phase (default: polar(1,0) mA)

Notes/Equations

1. Table 2-6 lists the DC operating point parameters that can be sent to the dataset.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

2. For general information regarding frequency domain sources, refer to the “Introduction” on page 2-1.
Frequency Domain Sources

I_SpectrumDataset (Current Source, Frequency Spectrum Defined in Dataset)

Symbol

Parameters
Dataset = dataset name
Expression = dataset variable or expression
Freq = fundamental frequency

Notes/Equations
1. Each dataset-based source has these fields:
   • Dataset, for the name of the dataset.
   • Expression, for an expression or a dataset variable. Values of this expression will be used as the harmonics for this source.
   • Freq, for the fundamental frequency of this source.
2. The frequency of the fundamental is specified on this source. It is not affected by the actual value of the fundamental associated with the harmonic data in a dataset.
3. Expression must evaluate to a one-dimensional array with two or more entries. The entries in this array are interpreted as dc, fundamental, second harmonic, third harmonic, etc. There cannot be gaps in the list of harmonics; if the list of harmonics includes the 6th harmonic, harmonics 0-5 must be present.
4. Because no interpolation or extrapolation is done by the system, the harmonic data must be single-dimensional. While multi-dimensional data is handled for dataset variables, it is not handled for this source.
5. The harmonic values must be numeric and can be given as real (integer) or complex values. If given as real values, they will be converted to complex (zero imaginary part) before being used in a simulation.

6. The designer must ensure the rationality of the expression that is used. If an expression is used that actually evaluates to a value of resistance, the system has no way of detecting this and will treat these resistance values as harmonics in a harmonic balance simulation.

7. The independent data extracted from the design system expression will be ignored; only the dependent data will be used, and it will be treated as harmonics. The independent data is ignored so that the source can be frequency shifted (the fundamental value can be changed to something other than that associated with the harmonics in a dataset).

8. Table 2-7 lists the DC operating point parameters that can be sent to the dataset.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

9. For general information regarding frequency domain sources, refer to the “Introduction” on page 2-1.
Frequency Domain Sources

**OScwPhNoise (Oscillator with Phase Noise)**

**Symbol**

![Symbol](image)

**Parameters**

- **Freq** = frequency
- **P** = output power
- **Rout** = output resistance
- **PhaseNoise** = phase noise data

**Range of Usage**

All phase noise dBc values should be less than $-10$

**Notes/Equations**

1. In Circuit Envelope simulation, the output power **P** of OscwPhNoise represents the total power output from this source over the entire simulation bandwidth. This implies that the phase noise level specified in this source should be small enough with respect to this total power such that the specified phase noise level below the carrier signal can be maintained without violating the conservation of total power. In the event that this specified phase noise is too high, the overall power output from this source will be fixed at **P**, and the phase noise level and the carrier signal power from the source will be adjusted by the program to be different from the levels specified in order to maintain the total power over the bandwidth to be **P**.

2. OSCwPhNoise can be used in harmonic balance and circuit envelope simulations—it is not recommended for transient simulation.

   A harmonic balance simulation example is shown in Figure 2-4 and Figure 2-5.

3. For new designs, the use of the **P_1Tone** source is preferred, unless Circuit Envelope noise is required. The **P_1Tone** source can now specify phase noise, using the same syntax as this element. The **P_1Tone** with phase noise generates the proper broadband phase noise when analyzing circuits and systems with mixers and/or multiple tones.
Figure 2-4. Harmonic Balance Setup

Figure 2-5. Harmonic Balance Noise Simulation Results
Frequency Domain Sources

P_AC (AC Power Source)

Symbol

Parameters

- **Num** = port number
- **Z** = reference impedance, use 1+i*0 for complex
- **Pac** = ac power, use polar(dbmtow(0), 0) for phase
- **Freq** = frequency, in Hz
- **Noise** = enable/disable port thermal noise: yes (default), no
- **Vdc** = open circuit dc voltage
- **Temp** = temperature of port in degrees Celsius. Default equals the circuit ambient temperature

Notes/Equations

1. P_AC is an ac power source used for ac simulation. When not in use it is treated as an impedance.
2. The Noise parameter only affects the amount of noise generated by the port. If Noise=yes, the amount of noise generated is based on this temperature. This parameter is not used on the input and output ports when calculating the noise figure, as the noise figure definition requires the input port temperature to be 290K and the output port is noiseless.
3. Table 2-8 lists the DC operating point parameters that can be sent to the dataset.
4. For general information regarding frequency domain sources, refer to the “Introduction” on page 2-1.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iport</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vport</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>
Frequency Domain Sources

**P_1Tone (Power Source, Single Frequency)**

Symbol

![Symbol Image]

**Parameters**
- **Num** = port number
- **Z** = source impedance
- **P** = power at center frequency
- **Freq** = center frequency
- **P_USB** = power of upper sideband small-signal tone; value used for small-signal mixer simulation
- **P_LSB** = power of lower sideband small-signal tone; value used for small-signal mixer simulation
- **Mod** = modulation function
- **Noise** = enable/disable port thermal noise: yes or no
- **Pac** = ac power; value used for ac simulation
- **FundIndex** = frequency index; an alternate way of specifying center frequency, used in MDS
- **Vdc** = open circuit dc voltage
- **Temp** = element temperature in degrees Celsius. Default equals the circuit ambient temperature
- **PhaseNoise** = list of offset frequency, phase noise pairs
- **Other** = output string to netlist

**Notes/Equations**

---

2-20  P_1Tone (Power Source, Single Frequency)
The phase of the source is specified by a complex value P such as 
\[ P = \text{polar}(\text{dbmtow}(0), 45) \]. The same applies to P_USB and P_LSB. The unit for 
power is W, mW, and so on; dBm must be converted to W by using dbmtow().

This power source is defined by its frequency, power, impedance, and linear 
modulation. It can be used in all circuit simulations.

For ac analysis, only Z and Pac are used—all other parameters are ignored. 
When frequency conversion ac analysis is performed, the Freq parameter is 
used to set the frequency of the source. For harmonic balance and envelope, the 
Freq parameter identifies the closest analysis frequency. If the frequency is not 
close enough, a warning is generated and the source generates no signal for 
that analysis. In the envelope analysis, the frequency difference can be up to 
0.5/timestep and the given frequency is still properly generated.

This source can also be used in transient analysis to generate an arbitrary 
waveform at the specified carrier frequency. The time domain output waveform 
of P_1Tone is a cosine. In order to get a sinusoidal waveform, set 
P = \text{polar}(\text{magnitude}, -90), where magnitude is the power magnitude. The 
transient waveform amplitude is affected by the load termination such that, for 
a matched load this amplitude is scaled by 1/2.

The output impedance of the source is defined by the Z-parameter. The output 
impedance may be complex, but dc, transient, and baseband envelope analyses 
do not support non-real impedances.

The signal level is defined by the power parameter P and the Mod parameter. 
The signal level is set such that the power delivered to a conjugately matched 
load is equal to P, assuming the Mod parameter is equal to 1.0. The Mod 
parameter can be used to apply complex, linearly scaled, modulation to the 
output signal. When this source represents a real-only, baseband signal 
(transient or the baseband part of an envelope signal), only the real part of the 
signal is generated. Otherwise, the full complex value of Mod can be used to 
modify both the amplitude and phase of the signal.

For time-domain analyses, transient and envelope, both the P and Mod 
parameters can be expressions of time. A time varying P provides a 
logarithmically scaled modulation, but is limited to scalar, magnitude-only 
variations. A time varying Mod provides both linearly scaled amplitude 
modulation as well as a linear phase modulation by using a complex expression. 
When a source with a time varying expression is used in a steady-state analysis 
such as harmonic balance, then its value at time=0 is used. Care must be
Frequency Domain Sources

exercised if these parameters are expressed as a function of frequency, because
this is not fully supported in all analysis modes.

6. In small-signal mixer simulation, the simulator sets the operating point of the
circuit using the carrier signal alone as the source. The two sidebands must be
set such that they have no effect on the operating point of the circuit.

7. Set Noise=0 to have no noise generated by this source.

8. The Temp parameter only affects the amount of noise generated by the port. If
Noise=yes, the amount of noise generated is based on this temperature. This
parameter is not used on the input and output ports when calculating noise
figure, as the noise figure definition requires the input port temperature to be
290K and the output port is noiseless.

9. Table 2-9 lists the DC operating point parameters that can be sent to the
dataset.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iport</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vport</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

10. The phase noise is specified using the list function:

    list(10Hz, -20dB, 100Hz,-40dB, 1kHz,-50dB)

    It consists of pairs of offset frequencies in Hertz and phase noise values in
dBC/Hz. When evaluated using offset frequencies that are smaller or larger
than those given in the list, the phase noise corresponding to the smallest or
largest frequency is used; the data are not extrapolated. No noise is generated if
the phase noise is less than -300 dBc/Hz. Any data pair that contains a
frequency <= 0 Hz is ignored.

    Figure 2-6 shows the phase noise results for the sample data list given above.
Noise is generated by this source in AC, S-parameter and HB analyses. The noise on either side of the carrier is correlated to produce pure phase noise with no amplitude noise. For example, with a carrier frequency of 1 MHz, the noise at 0.99 MHz and 1.01 MHz is correlated and has a value that depends on the phase noise at 10 kHz and the magnitude of the carrier signal.

This source generates broadband noise. Noise will be generated at any offset frequency from the carrier. If the offset frequency is larger than the last one in the phase noise list, that value will be used. The phase noise is not extrapolated and does not stop for large offsets.

11. Noise is not generated by this source during Circuit Envelope or transient analysis. The IncludePortNoise parameter on the analysis controller must be set to yes for this source to generate phase noise and the instance parameter Noise must be set to yes. This source also adds white thermal noise based on the reference impedance. If this extra white noise is not desired, use a separate V_1Tone with a noiseless resistor.

12. For general information regarding frequency domain sources, refer to the “Introduction” on page 2-1.
Frequency Domain Sources

P_nHarm (Power Source, Fundamental Frequency with N-Harmonics)

Symbol

Parameters

Num = port number
Z = source impedance
Freq = fundamental frequency
P = Nth harmonic power level
Noise = enable/disable port thermal noise: yes or no
Pac = ac power; value used for ac simulation
FundIndex = frequency index; an alternate way of specifying center frequency, used in MDS
Vdc = open circuit dc voltage
Temp = element temperature in degrees Celsius. Default equals the circuit ambient temperature
PhaseNoise = list of offset frequency, phase noise pairs
Other = output string to netlist

Notes/Equations

1. The phase of the source is specified by a complex value P such as P = polar(dbmtow(0), 45). The unit for power is W, mW, and so on; dBm must be converted to W by using dbmtow().

2. This power source is defined by a fundamental frequency component, and N harmonics of the fundamental frequency. It can be used in all circuit simulations.
For ac analysis, only Z and Pac are used—all other parameters are ignored. When frequency conversion ac analysis is performed, the Freq parameter is used to set the frequency of the source. For harmonic balance and envelope, the Freq parameter identifies the closest analysis frequency. If the frequency is not close enough, a warning is generated and the source generates no signal for that analysis. In the envelope analysis, the frequency difference can be up to 0.5/timestep and the given frequency is still properly generated. This source can also be used in transient analysis to generate an arbitrary waveform at the specified carrier frequency.

3. The output impedance of the source is defined by the Z-parameter. The output impedance may be complex, but dc, transient, and baseband envelope analyses do not support non-real impedances.

4. The signal level is defined by the power parameter P. The signal level is set such that the power delivered to a conjugately matched load is equal to P. When this source represents a real-only, baseband signal (transient or the baseband part of an envelope signal), only the real part of the signal is generated.

5. For time-domain analyses, transient and envelope, the P parameter may be an expression of time. A time varying P provides a logarithmically scaled modulation, but is limited to scalar, magnitude-only variations. When a source with a time varying expression is used in a steady-state analysis such as harmonic balance, then its value at time=0 is used. Care must be exercised if these parameters are expressed as a function of frequency, since this is not fully supported in all analysis modes.

6. Set Noise=0 to have no noise generated by this source.

7. The Temp parameter only affects the amount of noise generated by the port. If Noise=yes, the amount of noise generated is based on this temperature. This parameter is not used on the input and output ports when calculating noise figure, as the noise figure definition requires the input port temperature to be 290K and the output port is noiseless.

8. Table 2-10 lists the DC operating point parameters that can be sent to the dataset.
9. The phase noise is specified using the list function:

```
list(10Hz, -20dB, 100Hz,-40dB, 1kHz,-50dB)
```

It consists of pairs of offset frequencies in Hertz and phase noise values in dBc/Hz. When evaluated using offset frequencies that are smaller or larger than those given in the list, the phase noise corresponding to the smallest or largest frequency is used; the data are not extrapolated. No noise is generated if the phase noise is less than -300 dBc/Hz. Any data pair that contains a frequency <= 0 Hz is ignored.

Figure 2-7 shows the phase noise results for the sample data list given above.

![Figure 2-7. Phase Noise Results](image)

Noise is generated by this source in AC, S-parameter and HB analyses. The noise on either side of the carrier is correlated to produce pure phase noise with no amplitude noise. For example, with a carrier frequency of 1 MHz, the noise at 0.99 MHz and 1.01 MHz is correlated and has a value that depends on the phase noise at 10 kHz and the magnitude of the carrier signal.

This source generates broadband noise. Noise will be generated at any offset frequency from the carrier. If the offset frequency is larger than the last one in

---

**Table 2-10. DC Operating Point Information**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iport</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vport</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>
the phase noise list, that value will be used. The phase noise is not extrapolated and does not stop for large offsets.

Noise is not generated by this source during Circuit Envelope or transient analysis.

10. The IncludePortNoise parameter on the analysis controller must be set to yes for this source to generate phase noise and the instance parameter Noise must be set to yes. This source also adds white thermal noise based on the reference impedance. If this extra white noise is not desired, use a separate V_1Tone with a noiseless resistor.

11. The phase noise is only specified for the fundamental. Phase noise is generated by the rest of the specified harmonics using the fundamental phase noise specification plus 20*log10(N) dB, where N is the harmonic number.

12. For general information regarding frequency domain sources, refer to the “Introduction” on page 2-1.
Frequency Domain Sources

P_NTone (Power Source, N Frequencies and Power Levels)

Symbol

Parameters

Num = port number
Z = source impedance
Freq = Nth frequency tone
P = Nth tone power level
Noise = enable/disable port thermal noise: yes or no
Pac = ac power; value used for ac simulation
Vdc = open circuit dc voltage
Temp = element temperature in degrees Celsius. Default equals the circuit ambient temperature
PhaseNoise = list of offset frequency, phase noise pairs
Other = output string to netlist

Notes/Equations

1. The phase of the source is specified by a complex value \( P \) such as \( P = \text{polar(dbmtow(0), 45)} \). The unit for power is W, mW, and so on; dBm must be converted to W by using dbmtow().

2. This power source can have an arbitrary number \( 1 \leq N < \infty \) of harmonically independent tones. It can be used in all circuit simulations.

For ac analysis, only Z and Pac are used—all other parameters are ignored. When frequency conversion ac analysis is performed, the Freq parameter is used to set the frequency of the source. For harmonic balance and envelope, the Freq parameter identifies the closest analysis frequency. If the frequency is not close enough, a warning is generated and the source generates no signal for that analysis. In the envelope analysis, the frequency difference can be up to 0.5/timestep and the given frequency is still properly generated. This source can
also be used in transient analysis to generate an arbitrary waveform at the specified carrier frequency.

3. The output impedance of the source is defined by the Z-parameter. The output impedance may be complex, but dc, transient, and baseband envelope analyses do not support non-real impedances.

4. The signal level is defined by the power parameter P. The signal level is set such that the power delivered to a conjugately matched load is equal to P. When this source represents a real-only, baseband signal (transient or the baseband part of an envelope signal), only the real part of the signal is generated.

5. For time-domain analyses, transient and envelope, the P parameter may be an expression of time. A time varying P provides a logarithmically scaled modulation, but is limited to scalar, magnitude-only variations. When a source with a time varying expression is used in a steady-state analysis such as harmonic balance, then its value at time=0 is used. Care must be exercised if these parameters are expressed as a function of frequency, since this is not fully supported in all analysis modes.

6. Set Noise=0 to have no noise generated by this source.

7. The Temp parameter only affects the amount of noise generated by the port. If Noise = yes, the amount of noise generated is based on this temperature. This parameter is not used on the input and output ports when calculating noise figure, as the noise figure definition requires the input port temperature to be 290K and the output port is noiseless.

8. Table 2-11 lists the DC operating point parameters that can be sent to the dataset.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iport</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vport</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

9. The phase noise is specified using the list function:

\[
\text{list}(10\text{Hz}, -20\text{dB}, 100\text{Hz}, -40\text{dB}, 1\text{kHz}, -50\text{dB})
\]
Frequency Domain Sources

It consists of pairs of offset frequencies in Hertz and phase noise values in dBc/Hz. When evaluated using offset frequencies that are smaller or larger than those given in the list, the phase noise corresponding to the smallest or largest frequency is used; the data are not extrapolated. No noise is generated if the phase noise is less than -300 dBc/Hz. Any data pair that contains a frequency <= 0 Hz is ignored.

Figure 2-8 shows the phase noise results for the sample data list given above.

![Figure 2-8. Phase Noise Results](image)

Noise is generated by this source in AC, S-parameter and HB analyses. The noise on either side of the carrier is correlated to produce pure phase noise with no amplitude noise. For example, with a carrier frequency of 1 MHz, the noise at 0.99 MHz and 1.01 MHz is correlated and has a value that depends on the phase noise at 10 kHz and the magnitude of the carrier signal.

This source generates broadband noise. Noise will be generated at any offset frequency from the carrier. If the offset frequency is larger than the last one in the phase noise list, that value will be used. The phase noise is not extrapolated and does not stop for large offsets.

10. Noise is not generated by this source during Circuit Envelope or transient analysis. The IncludePortNoise parameter on the analysis controller must be set to yes for this source to generate phase noise and the instance parameter Noise must be set to yes. This source also adds white thermal noise based on the reference impedance. If this extra white noise is not desired, use a separate V_1Tone with a noiseless resistor.

11. Phase noise can be specified for each of the independent tones. If the phase noise is not specified for one of the tones, then that tone generates no phase noise but the other tones do.

12. For general information regarding frequency domain sources, refer to the “Introduction” on page 2-1.
\textbf{Parameters}

\begin{itemize}
  \item \texttt{Num} = port number
  \item \texttt{Z} = source impedance
  \item \texttt{Freq} = fundamental frequency
  \item \texttt{Dataset} = dataset name
  \item \texttt{Expression} = dataset variable or expression
\end{itemize}

\textbf{Notes/Equations}

1. Each dataset-based source has these fields:
   \begin{itemize}
     \item A field \texttt{Dataset}, for the name of the dataset.
     \item A field \texttt{Expression}, for an expression or a dataset variable. The values of this expression will be used as the harmonics for this source.
     \item A field \texttt{Freq} for the fundamental frequency of this source.
   \end{itemize}

2. The frequency of the fundamental is specified on this source. It is not affected by the actual value of the fundamental associated with the harmonic data in a dataset.

3. Expression must evaluate to a one-dimensional array with two or more entries. The entries in this array are interpreted as dc, fundamental, second harmonic, third harmonic, etc. There cannot be gaps in the list of harmonics; if the list of harmonics includes the 6th harmonic, harmonics 0-5 must be present.

4. Because no interpolation or extrapolation is done by the system, the harmonic data must be single-dimensional. While multi-dimensional data is handled for dataset variables, it is not handled for this source.
5. The harmonic values must be numeric and can be given as real (integer) or complex values. If given as real values, they will be converted to complex (zero imaginary part) before being used in a simulation.

6. The designer must ensure the rationality of the expression that is used. If an expression is used that actually evaluates to a value of resistance, the simulator has no way of detecting this, and will treat these resistance values as harmonics in a harmonic balance simulation. The independent data extracted from the design system expression will be ignored; only the dependent data will be used, and it will be treated as harmonics. The independent data is ignored so that the source can be frequency shifted (the fundamental value can be changed to something other than that associated with the harmonics in a dataset).

7. Table 2-12 lists the DC operating point parameters that can be sent to the dataset.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iport</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vport</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

8. For general information regarding frequency domain sources, refer to the “Introduction” on page 2-1.
V_1Tone (Voltage Source, Single Frequency)

Symbol

```
+------------------
<p>| |
|                  |
|                  |
|                  |
|                  |
|                  |
|                  |</p>
<table>
<thead>
<tr>
<th></th>
</tr>
</thead>
</table>
+------------------
```

Parameters

- \( V = \) voltage at center frequency
- \( \text{Freq} = \) center frequency
- \( V_{USB} = \) voltage of upper sideband small-signal tone; value used for small-signal mixer simulation
- \( V_{LSB} = \) voltage of lower sideband small-signal tone; value used for small-signal mixer simulation
- \( V_{dc} = \) dc voltage
- \( V_{ac} = \) ac voltage; value used for ac simulation
- \( \text{SaveCurrent} = \) flag to save branch current
- \( \text{FundIndex} = \) frequency index; an alternate way of specifying center freq, used in MDS
- \( \text{PhaseNoise} = \) list of offset frequency, phase noise pairs
- \( \text{Other} = \) output string to netlist

Range of Usage

\( \text{Freq} > 0 \)

Notes/Equations

1. This single frequency voltage source is defined by its frequency and its voltage and can be used in all circuit simulations. The phase of the source is specified by a complex value \( V \), such as \( V = \text{polar}(1V, 45) \).

   For ac simulation, only \( V_{ac} \) is used and all other parameters are ignored. When frequency conversion ac analysis is performed, the \( \text{Freq} \) parameter is used to set the frequency of the source. For harmonic balance and circuit envelope, the
Frequency Domain Sources

Freq parameter identifies the closest analysis frequency. If the frequency is not close enough, a warning is generated and the source voltage is set to 0 for that analysis. In the envelope analysis, the frequency difference can be up to 0.5/timestep and the given frequency is still properly generated.

This source can also be used in transient analysis to generate an arbitrary waveform at the specified carrier frequency. When this source represents a real-only, baseband voltage (that is, transient or the baseband part of an envelope signal), then only the real part of the signal is generated. Otherwise, the full complex value of V is used to define both the amplitude and phase relationships.

2. For time-domain analyses, transient and envelope, the voltage parameter may be an expression of time. When a source with a time varying expression is used in a steady-state analysis such as harmonic balance, then its value at time=0 is used. Care must be exercised if the voltage parameter is a function of frequency, since this is not fully supported in all analysis modes.

3. A dc term can be defined for this device. Also, as with the other voltage sources, a SaveCurrent parameter is available to disable the storing of the voltage source current to the dataset.

4. Table 2-13 lists the DC operating point parameters that can be sent to the dataset.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

5. In small signal mixer simulation, the simulator sets the operating point of the circuit using the carrier signal alone as the source. The two sidebands must be set such that they have no effect on the operating point of the circuit.

6. In S-parameter analysis, this component is treated as an ideal short circuit.

7. The phase noise is specified using the list function:

   list(10Hz, -20dB, 100Hz, -40dB, 1kHz, -50dB)
It consists of pairs of offset frequencies in Hertz and phase noise values in dBc/Hz. When evaluated using offset frequencies that are smaller or larger than those given in the list, the phase noise corresponding to the smallest or largest frequency is used; the data are not extrapolated. No noise is generated if the phase noise is less than -300 dBc/Hz. Any data pair that contains a frequency <= 0 Hz is ignored.

Figure 2-9 shows the phase noise results for the sample data list given above.

![Figure 2-9. Phase Noise Results](image)

Noise is generated by this source in AC, S-parameter and HB analyses. The noise on either side of the carrier is correlated to produce pure phase noise with no amplitude noise. For example, with a carrier frequency of 1 MHz, the noise at 0.99 MHz and 1.01 MHz is correlated and has a value that depends on the phase noise at 10 kHz and the magnitude of the carrier signal.

This source generates broadband noise. Noise will be generated at any offset frequency from the carrier. If the offset frequency is larger than the last one in the phase noise list, that value will be used. The phase noise is not extrapolated and does not stop for large offsets.

Noise is not generated by this source during Circuit Envelope or transient analysis.

8. For general information regarding frequency domain sources, refer to the “Introduction” on page 2-1.
Frequency Domain Sources

V_AC (AC Voltage Source)

Symbol

Parameters

Vdc = dc voltage, in volts
Vac = ac voltage, in volts; value used for ac analysis only
SaveCurrent = flag to save branch current
Freq = frequency

Notes/Equations

1. For AC simulations with no mixer component, leave Freq equal to freq, where freq is a global variable.

   For frequency-conversion AC analysis, Freq = source frequency.

2. V_AC is only meaningful in AC simulation. When used in other simulations, it is treated as a short circuit.

3. Table 2-14 lists the DC operating point parameters that can be sent to the dataset.

Table 2-14. DC Operating Point Information

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

4. For general information regarding frequency domain sources, refer to the "Introduction" on page 2-1.
**V\_DC (DC Voltage Source)**

Symbol

![Symbol Image]

**Parameters**

- $V_{dc} =$ dc voltage, in volts
- $V_{ac} =$ ac voltage, in volts; value used for ac analysis only
- SaveCurrent =$\neg$ flag to save branch current

**Notes/Equations**

1. $V\_DC$ can be used in all simulations. When not in use, it is treated as a short circuit.
2. Table 2-15 lists the DC operating point parameters that can be sent to the dataset.

**Table 2-15. DC Operating Point Information**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>
Frequency Domain Sources

**V_nHarm (Voltage Source, Fundamental Frequency with N-Harmonics)**

**Symbol**

![Symbol](image)

**Parameters**

- **Freq** = fundamental frequency
- **V** = Nth harmonic amplitude
- **Vdc** = dc voltage
- **Vac** = ac voltage; value used for ac analysis only
- **SaveCurrent** = flag to save branch current
- **FundIndex** = frequency index; an alternate way of specifying fundamental freq, used in MDS
- **PhaseNoise** = list of offset frequency, phase noise pairs
- **Other** = output string to netlist

**Notes/Equations**

1. This voltage source has a fundamental freq. component and N (1 ≤ N < ∞) harmonics of the fundamental freq. The phase of each harmonic is specified by a complex V, such as V=polar(IV, 45). This source is used in all simulations. The phase of the source is specified by a complex value V, such as V=polar(1V, 45).

   For ac simulation, only Vac is used and all other parameters are ignored. When frequency conversion ac analysis is performed, the Freq parameter is used to set the frequency of the source. For harmonic balance and circuit envelope, the Freq parameter identifies the closest analysis frequency. If the frequency is not close enough, a warning is generated and the source voltage is set to 0 for that analysis. In the envelope analysis, the frequency difference can be up to 0.5/timestep and the given frequency is still properly generated.

   This source can also be used in transient analysis to generate an arbitrary waveform at the specified carrier frequency. When this source represents a
real-only, baseband voltage (that is, transient or the baseband part of an envelope signal), then only the real part of the signal is generated. Otherwise, the full complex value of \( V \) is used to define both the amplitude and phase relationships.

2. For time-domain analyses, transient and envelope, the voltage parameter may be an expression of time. When a source with a time varying expression is used in a steady-state analysis such as harmonic balance, then its value at time=0 is used. Care must be exercised if the voltage parameter is a function of frequency, since this is not fully supported in all analysis modes.

3. A dc term can be defined for this device. Also, as with the other voltage sources, a SaveCurrent parameter is available to disable the storing of the voltage source current to the dataset.

4. Table 2-16 lists the dc operating point parameters that can be sent to the dataset.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

5. In S-parameter analysis, this component is treated as an ideal short circuit.

6. The phase noise is specified using the list function:

   \[
   \text{list}(10\text{Hz}, -20\text{dB}, 100\text{Hz}, -40\text{dB}, 1\text{kHz}, -50\text{dB})
   \]

   It consists of pairs of offset frequencies in Hertz and phase noise values in \( \text{dBc/Hz} \). When evaluated using offset frequencies that are smaller or larger than those given in the list, the phase noise corresponding to the smallest or largest frequency is used; the data are not extrapolated. No noise is generated if the phase noise is less than -300 dBc/Hz. Any data pair that contains a frequency <= 0 Hz is ignored.

   Figure 2-10 shows the phase noise results for the sample data list given above.
Noise is generated by this source in AC, S-parameter and HB analyses. The noise on either side of the carrier is correlated to produce pure phase noise with no amplitude noise. For example, with a carrier frequency of 1 MHz, the noise at 0.99 MHz and 1.01 MHz is correlated and has a value that depends on the phase noise at 10 kHz and the magnitude of the carrier signal.

This source generates broadband noise. Noise will be generated at any offset frequency from the carrier. If the offset frequency is larger than the last one in the phase noise list, that value will be used. The phase noise is not extrapolated and does not stop for large offsets.

Noise is not generated by this source during Circuit Envelope or transient analysis.

7. The phase noise is only specified for the fundamental. Phase noise is generated by the rest of the specified harmonics using the fundamental phase noise specification plus $20\log_{10}(N)$ dB, where $N$ is the harmonic number.

8. For general information regarding frequency domain sources, refer to the “Introduction” on page 2-1.
**V_nTone (Voltage Source, N Frequencies and Amplitudes)**

**Symbol**

![Symbol Diagram]

**Parameters**
- **Freq** = Nth frequency tone
- **V** = Nth tone amplitude
- **Vdc** = dc voltage
- **Vac** = ac voltage; value used for ac simulation only
- **SaveCurrent** = flag to save branch current
- **PhaseNoise** = list of offset frequency, phase noise pairs
- **Other** = output string to netlist

**Notes/Equations**

1. This voltage source can have an arbitrary number \(1 \leq N < \infty\) of harmonically independent tones and can be used in all simulations. The phase of each tone is specified by a complex \(V\) value such as \(V = \text{polar}(1V, 45)\).

   For ac simulation, only Vac is used and all other parameters are ignored. When frequency conversion ac analysis is performed, the Freq parameter is used to set the frequency of the source. For harmonic balance and circuit envelope, the Freq parameter identifies the closest analysis frequency. If the frequency is not close enough, a warning is generated and the source voltage is set to 0 for that analysis. In the envelope analysis, the frequency difference can be up to 0.5/timestep and the given frequency is still properly generated. For ac analysis, the Freq parameter is ignored.

2. For time-domain analyses, transient and envelope, the voltage parameter may be an expression of time. When a source with a time varying expression is used in a steady-state analysis such as harmonic balance, then its value at time=0 is used. Care must be exercised if the voltage parameter is a function of frequency, since this is not fully supported in all analysis modes.
3. A dc term can be defined for this device. Also, as with the other voltage sources, a SaveCurrent parameter is available to disable the storing of the voltage source current to the dataset.

4. Table 2-17 lists the dc operating point parameters that can be sent to the dataset.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

5. In S-parameter analysis, this component is treated as an ideal short circuit.

6. The phase noise is specified using the list function:

\[
\text{list}(10\text{Hz}, -20\text{dB}, 100\text{Hz}, -40\text{dB}, 1\text{kHz}, -50\text{dB})
\]

It consists of pairs of offset frequencies in Hertz and phase noise values in dBc/Hz. When evaluated using offset frequencies that are smaller or larger than those given in the list, the phase noise corresponding to the smallest or largest frequency is used; the data are not extrapolated. No noise is generated if the phase noise is less than -300 dBc/Hz. Any data pair that contains a frequency \( \leq 0 \) Hz is ignored.

Figure 2-11 shows the phase noise results for the sample data list given above.

![Phase Noise Results](image)

Figure 2-11. Phase Noise Results

Noise is generated by this source in AC, S-parameter and HB analyses. The noise on either side of the carrier is correlated to produce pure phase noise with no amplitude noise. For example, with a carrier frequency of 1 MHz, the noise...
at 0.99 MHz and 1.01 MHz is correlated and has a value that depends on the phase noise at 10 kHz and the magnitude of the carrier signal.

This source generates broadband noise. Noise will be generated at any offset frequency from the carrier. If the offset frequency is larger than the last one in the phase noise list, that value will be used. The phase noise is not extrapolated and does not stop for large offsets.

Noise is not generated by this source during Circuit Envelope or transient analysis.

7. Phase noise can be specified for each of the independent tones. If the phase noise is not specified for one of the tones, then that tone generates no phase noise but the other tones do.

8. For general information regarding frequency domain sources, refer to the “Introduction” on page 2-1.
V_SpectrumDataset (Voltage Source, Frequency Spectrum Defined in Dataset)

Symbol

Parameters
Dataset = dataset name
Expression = dataset variable or expression
Freq = fundamental frequency
SaveCurrent = flag to save branch current

Notes/Equations
1. Each dataset-based source has these fields:
   - A field Dataset, for the name of the dataset.
   - A field Expression, for an expression or a dataset variable. The values of this expression will be used as the harmonics for this source.
   - A field Freq for the fundamental frequency of this source.

2. The frequency of the fundamental is specified on this source. It is not affected by the actual value of the fundamental associated with the harmonic data in a dataset.

3. Expression must evaluate to a one-dimensional array with two or more entries. The entries in this array are interpreted as dc, fundamental, second harmonic, third harmonic, and so on. There cannot be gaps in the list of harmonics; if the list of harmonics includes the 6th harmonic, harmonics 0-5 must be present.

4. Because no interpolation or extrapolation is done by the system, the harmonic data must be single-dimensional. While multi-dimensional data is handled for dataset variables, it is not handled for this source.
5. The harmonic values must be numeric and can be given as real (integer) or complex values. If given as real values, they will be converted to complex (zero imaginary part) before being used in a simulation.

6. The designer must ensure the rationality of the expression that is used. If an expression is used that actually evaluates to a value of resistance, the system has no way of detecting this, and will treat these resistance values as harmonics in a harmonic balance simulation.

   The independent data extracted from the design system expression will be ignored; only the dependent data will be used, and it will be treated as harmonics. The independent data is ignored so that the source can be frequency shifted (the fundamental value can be changed to something other than that associated with the harmonics in a dataset).

7. Table 2-18 lists the dc operating point parameters that can be sent to the dataset.

```
Table 2-18. DC Operating Point Information

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>
```

8. For general information regarding frequency domain sources, refer to the “Introduction” on page 2-1.
Vf_BitSeq (Fourier Transform of Bit Sequence Waveform)

Symbol

Parameters
Vlow = minimum voltage level, in V, fV, pV, nV, uV, mV, or kV (default: 0V)
Vhigh = maximum voltage level, in V, fV, pV, nV, uV, or mV (default: 5V)
Rate = bit rate, in MHz, Hz, kHz, GHz (default: 500 MHz)
Rise = rise time of pulse, in nsec, fsec, psec, nsec, usec, or msec (default: 1 nsec)
Fall = fall time of pulse, in nsec, fsec, psec, nsec, usec, or msec (default: 1 nsec)
BitSeq = bit sequence (default: 1011011100111100)
Tstart = start time of bit sequence (default: 0 sec)
Tstop = stop time of bit sequence (default: 32 nsec)
Tstep = time domain resolution (default: 0.01 nsec)

Notes/Equations
1. Vf_BitSeq is recommended for use in frequency domain analyses only; for time
domain analyses (such as transient simulations) use the VtBitSeq component
for comparable functionality.

2. The BitSeq parameter enables you to input the waveform of a pulse via an
arbitrary bit pattern such as 1101011100111100 (default). When the end of the
sequence is reached, the sequence is repeated. A specification of 1 sets voltage
to Vhigh, 0 sets it to Vlow.

Note  To edit BitSeq, enter a value enclosed with double quote symbols.

The bit sequence is generated over the time range [Tstart, Tstop] and its time
domain resolution is expressed in Tstep. For good results, Tstop should be
exactly one cycle of the complete sequence. For example, if Rate=500MHz, then
bit period=1/Rate=1/500MHz=2nsec. For a 16-bit sequence, for example BitSeq="110111100111100", define Tstop=32nsec (16bits × 2nsec). (Refer to Figure 2-12 and Figure 2-13.)

Prior to ADS 2003C, an explicit VAR block for specification of Tstart, Tstop and Tstep was required. The use of an explicit VAR block is no longer necessary as these variables are included as component parameters.

3. For Harmonic Balance simulations the recommended controller setting for Freq[1]=1/Tstop. Increasing Order[1] to a large value (as shown in Figure 2-13), ensures that a sufficient number of harmonics are included to give an accurate Fourier series representation of the user-defined pulse waveform.

4. Figure 2-12 and Figure 2-13 show the use of Vf_BitSeq in typical AC and Harmonic Balance analysis scenarios in ADS.

For RFDE users, create the design variables StopT and StepT in the Analog Environment dialog box. Select AC analysis and perform a linear frequency sweep from 0 to 1/(2*StepT), in steps of 1/(StopT). Or, when performing a harmonic balance analysis, set Frequency to 1/StopT and Order to 511 (or set the order to a large enough value to represent the waveform).

Figure 2-12. Vt_BitSeq in Typical AC Simulation Analysis
Frequency Domain Sources

Figure 2-13. Vf_BitSeq in Typical Harmonic Balance Simulation Analysis

5. Table 2-19 lists the dc operating point parameters that can be sent to the dataset.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

6. For general information regarding frequency domain sources, refer to the “Introduction” on page 2-1.
Vf_Pulse (Voltage Source, Fourier Series Expansion of Period Pulse Wave)

Symbol

Parameters

Vpeak = peak voltage amplitude of pulse, in volts
Vdc = dc offset
Freq = fundamental frequency component \((1 / T_0, \text{where } T_0 \text{ is the pulse-period)}\) of pulse-train
Width = pulse-width, in seconds
Rise = rise-time, in seconds
Fall = fall-time, in seconds
Delay = time delay, in seconds
Weight = compensation for Gibb's Phenomenon if rise- or fall-time is 0; activated when Weight=yes; ignored if both Rise and Fall are > 0
Harmonics = number of harmonics
SaveCurrent = flag to save branch current
FundIndex = frequency index; an alternate way of specifying fundamental freq, used in MDS

Range of Usage

Width \(\geq 0\)
Freq > 0
Rise \(\geq 0\)
Fall \(\geq 0\)
Delay \(\geq 0\)
Rise + Fall + Width \(\leq T_0 = 1/\text{Freq}\)
Frequency Domain Sources

Notes/Equations

1. Vf_Pulse is a time-periodic rectangular pulse-train voltage source that can be used in all simulations. However, the Vf_Pulse source is short-circuited for AC simulation. The time-periodic signal is converted to discrete frequency components that are harmonically related and represented using the signal's equivalent Fourier series.

2. The source produces a positive voltage with respect to pin 1.

3. If either rise-time (Rise) or fall-time (Fall) is 0, the discontinuity in the pulse gives rise to Gibb's Phenomenon when the pulse is synthesized from its Fourier components. The ripple effect at the discontinuity can be smoothed by specifying Weight=yes, which scales the Fourier coefficients of the source by Lanczos factors or weights.

4. The number of terms in the Fourier series used to represent this source in the frequency domain is equivalent to the order chosen for the harmonic balance simulation.

5. You can synthesize a similar time-periodic signal by connecting in series a number of sinusoidal steady-state sources at harmonically related frequencies and having amplitudes and phases that are the corresponding Fourier coefficients.

6. Table 2-20 lists the dc operating point parameters that can be sent to the dataset.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

7. For general information regarding frequency domain sources, refer to the “Introduction” on page 2-1.
Vf_Sawtooth (Voltage Source, Fourier Series Expansion of Periodic Sawtooth)

Symbol

Parameters

Vpeak = peak voltage amplitude of wave, in volts
Vdc = dc offset
Freq = frequency
Delay = time delay, in seconds
Weight = compensation for Gibb’s Phenomenon if rise- or fall-time is 0; activated when Weight=yes; ignored if rise and fall are non-0
Harmonics = number of harmonics
SaveCurrent = flag to save branch current
FundIndex = frequency index; an alternate way of specifying fundamental freq, used in MDS

Range of Usage

Delay ≥ 0

Notes/Equations

1. This item is a time-periodic sawtooth voltage source that can be used in all simulations. However, the Vf_Sawtooth source is short circuited for AC simulation. The time-periodic signal is converted to discrete frequency components that are harmonically related and represented using the signal’s equivalent Fourier series.

2. The source produces a positive voltage with respect to pin 1.

3. The discontinuity in the pulse caused by 0 fall-time gives rise to Gibb’s Phenomenon when the pulse is synthesized from Fourier components. The
ripple effect at the discontinuity can be smoothed by specifying Weight=yes, which scales the Fourier coefficients of the source by Lanczos factors or weights.

4. The number of terms in the Fourier series used to represent this source in the frequency domain is equivalent to the order chosen for the harmonic balance simulation.

5. You can synthesize a similar time-periodic signal by connecting in series a number of sinusoidal steady-state sources at harmonically related frequencies and having amplitudes and phases that are the corresponding Fourier coefficients.

6. **Table 2-21** lists the dc operating point parameters that can be sent to the dataset.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

7. For general information regarding frequency domain sources, refer to the “Introduction” on page 2-1.
Vf_Square (Voltage Source, Fourier Series Expansion of Period Square Wave)

Symbol

Parameters

- Vpeak = peak voltage amplitude of pulse, in volts
- Vdc = dc offset
- Freq = frequency
- Rise = rise-time, in seconds
- Fall = fall-time, in seconds
- Delay = time delay, in seconds
- Weight = compensation for Gibb's Phenomenon if rise- or fall-time is 0; activated=yes; ignored if rise and fall are non-0
- Harmonics = number of harmonics
- SaveCurrent = flag to save branch current
- FundIndex = frequency index; an alternate way of specifying fundamental freq, used in MDS

Range of Usage

Delay ≥ 0; Rise ≥ 0; Fall ≥ 0; Rise + Fall < T₀/2

Notes/Equations

1. This time-periodic square-wave voltage source can be used in all simulations. However, the Vf_Square source is short circuited for AC simulation. The time-periodic signal is converted to discrete frequency components that are harmonically related and represented using the signal's equivalent Fourier series.
Frequency Domain Sources

2. The source produces a positive voltage with respect to pin 1.

3. If either rise-time (Rise) or fall-time (Fall) is zero, the discontinuity in the pulse gives rise to Gibb’s Phenomenon when the pulse is synthesized from its Fourier components. The ripple effect at the discontinuity can be smoothed by specifying Weight=yes, which scales the Fourier coefficients of the source by Lanczos factors or weights.

4. The number of terms in the Fourier series used to represent this source in the frequency domain is equivalent to the order chosen for the harmonic balance simulation.

5. A similar time-periodic signal can be synthesized by connecting in series a number of sinusoidal steady-state sources at harmonically related frequencies with amplitudes and phases of the corresponding Fourier coefficients.

6. Table 2-22 lists the dc operating point parameters that can be sent to the dataset.

Table 2-22. DC Operating Point Information

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

7. For general information regarding frequency domain sources, refer to the “Introduction” on page 2-1.
Vf_Triangle (Voltage Source, Fourier Series Expansion of Period Triangle Wave)

Symbol

Parameters

Vpeak = peak voltage amplitude of wave, in volts

Vdc = dc offset

Freq = frequency

Delay = time delay, in seconds

Harmonics = number of harmonics

SaveCurrent = flag to save branch current

FundIndex = frequency index; an alternate way of specifying fundamental freq, used in MDS

Range of Usage

Delay ≥ 0

Notes/Equations

1. This is a time-periodic triangle-wave voltage source that can be used in all simulations. However, the Vf_Triangle source is short circuited for AC simulation. The time-periodic signal is converted to discrete frequency components that are harmonically related and represented using the signal’s equivalent Fourier series.

2. The source produces a positive voltage with respect to pin 1.

3. The number of terms in the Fourier series used to represent this source in the frequency domain is equivalent to the order chosen for the harmonic balance simulation.
Frequency Domain Sources

4. Table 2-23 lists the dc operating point parameters that can be sent to the dataset.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

Table 2-23. DC Operating Point Information

5. For general information regarding frequency domain sources, refer to the "Introduction" on page 2-1.
V_HB_Dataset (Voltage Source, HB Dataset Variable)

Symbol

Parameters
Dataset = dataset filename
Variable = dataset variable
Vdc = DC voltage (default: 0 V)
Vac = AC voltage, use polar() for phase; default: polar (1,0) V
SaveCurrent = flag to save branch current (default: YES)

Notes/Equations
1. This data-based, frequency domain waveform voltage source is defined by a frequency-domain dataset variable. The dataset variable must have frequency as its independent swept axis.
2. The dataset filename and dataset variable must be enclosed in double quotes.
3. When performing a Harmonic Balance simulation with V_HB_Dataset, set the frequency on the HB simulation controller to the same frequency (from the earlier HB simulation controller) that was used to generate the dataset and the variable.
4. This source does not handle datasets that contain results from multiple simulations (such as those with more than one simulation controller). This does not mean that the dataset cannot have more than one variable. The dataset may contain several variables; however, they must be from the same simulation with a single simulation controller.
5. This source makes it possible to use a dataset variable from one Harmonic Balance simulation in another Harmonic Balance simulation.

In the following simple example, the output from a single stage amplifier is used as the input for another single stage amplifier using the V_HB_Dataset source.
From the schematic (amplifier_ckt1.dsn) in Figure 2-14, a dataset named amp_ckt1.ds is generated and includes the nodal voltage variable Vout. The circuit in amp_ckt1.dsn contains an amplifier with a gain of 10 dB.

The schematic in Figure 2-15 (both_amps.dsn) contains the V_HB_Dataset source with the variable Vout from the dataset amp_ckt1.ds as the input to an amplifier with a gain of 5 dB. This is equivalent to having two amplifiers cascaded in series, the first with a gain of 10 dB; the second with a gain of 5 dB. Both circuits are shown, and the results from each are also given. Note that both output waveforms are the same.

Simulation results are shown in Figure 2-16 (the output spectrum and waveform are shown for the swept values of the amplifier gain).

Figure 2-14. Single Stage Amplifier

Figure 2-15. Single Stage Amplifier using V_HB_Dataset Source
6. **Table 2-24** lists the dc operating point parameters that can be sent to the dataset.

```
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>
```

7. For general information regarding frequency domain sources, refer to the “Introduction” on page 2-1.
Frequency Domain Sources
Chapter 3: Modulated Sources
Modulated Sources

PtRF_3GPP_Uplink (Pwr Src, RF Carrier Modulated by 3GPP Uplink Signal)

Symbol

Parameters

Freq = Carrier frequency
Power = Output power at RF output
R = Output impedance of RF output

Notes/Equations

1. This model generates a digitally-modulated RF signal that has the modulation characteristics of a user equipment 3GPP (WCDMA) signal. It also contains framing characteristics. This source is implemented by interpolating data from a stored data file, which is faster than recalculating the source.

2. The stored data file is generated by the 3GPP Design Library. The pulse-shaping filter is a root raised-cosine filter with roll-off $\alpha=0.22$, according to 3GPP specifications.

3. The data file contains 1 frame (10 msec) of 3GPP data (38400 chips at 1/3.84 µsec per chip).

4. It is recommended that simulation timestep be set to (1/3.84/4 µsec), that is, taking four samples per chip. For other timestep values the source interpolates between data samples and results in different or lower fidelity signal spectrum.

5. Recommended controller setups for Envelope simulation are:
   - Envelope item
     Freq[1] = RFfreq
     Order[1] = 1
     StatusLevel=2
     Stop=tstop

3-2  PtRF_3GPP_Uplink (Pwr Src, RF Carrier Modulated by 3GPP Uplink Signal)
Step=tstep
Other=SavetoDataset=no

• VAR item
  chip_rate=3.84 MHz
  RFfreq = 1.95 GHz
  Pavs = 0 dBm
  sam_per_chip = 4
  tstep = 1/(chip_rate \times sam\_per\_chip)
  numChips = 256
  tstop = numChips / chip_rate

• PtRF_3GPP_Uplink item
  Freq = RFfreq
  Power = dbmtow(Pavs)
  (R = 50 Ohm)

6. For an overview of 3GPP (WCDMA) systems, refer to the 3GPPFDD Design Library, Introduction chapter (Manuals > Components > Signal Processing Components > 3GPPFDD).
Modulated Sources

PtRF_CDMA_ESG_FWD (Pwr Src, RF Carrier Modulated by ESG Fwd Link CDMA Signal)

Symbol

Parameters

$F_0$ = carrier frequency  
Power = RF output power  
$Z$ = RF output impedance

Notes/Equations

1. This model generates a digitally-modulated RF signal that has the modulation characteristics of a base station CDMA signal. It does not contain any framing characteristics. This source is implemented by interpolating data from a stored data file, which is faster than recalculating the source.

2. The stored data file is generated by the Agilent ESG series of signal generators. This source has lower adjacent channel power than that of PtRF_CDMA_IS95_FWD.

3. An identical source called IS95FwdLinkSrc that you can modify is located in the examples/Tutorial/ModSources_prj directory.

4. It is recommended that simulation timestep is equal to (0.25/1.2288 MHz), i.e, taking four samples per bit. Using other timestep values causes the source to interpolate between data samples and thus result in a distorted spectrum.

5. Recommended controller setups for Envelope simulation are:

- Envelope item  
  $Freq[1] = RFfreq$  
  $Order[1] = 1$  
  $StatusLevel=2$  
  $Stop=tstop$

---

3-4  PtRF_CDMA_ESG_FWD (Pwr Src, RF Carrier Modulated by ESG Fwd Link CDMA Signal)
Step=tstep
Other=SavetoDataset=no

• VAR item
  bit_rate=1.2288 MHz
  RFfreq = 1.9 GHz
  Pavs = 0 dBm
  sam_per_bit = 4
  tstep = 1/(bit_rate × sam_per_bit)
  numSymbols = 256
  tstop = num Symbols/(bit_rate/2)

• PtRF_CDMA_ESG_FWD item
  FO=RFfreq
  Power = dbmtow(Pavs)
  Z=50 Ohm
Modulated Sources

PtRF_CDMA_ESG_REV (Pwr Src, RF Carrier Modulated by ESG Rev. Link CDMA Signal)

Symbol

Parameters
- F0 = carrier frequency
- Power = RF output power
- Z = RF output impedance

Notes/Equations
1. This model generates a digitally-modulated RF signal that has the modulation characteristics of a handset CDMA signal. It does not contain any framing characteristics. This source is implemented by interpolating data from a stored data file, which is faster than recalculating the source.

2. The stored data file is generated by the Agilent ESG series of signal generators. This source has lower adjacent channel power than that of PtRF_CDMA_IS95_REV.

3. An identical source called IS95RevLinkSrc2 that you can modify is located in the examples/Tutorial/ModSources_prj directory.

4. It is recommended that simulation timestep is equal to (0.25/1.2288 MHz), i.e., taking four samples per bit. Using other timestep values the source interpolates between data samples and results in distorted spectrum.

5. Recommended controller setups for Envelope simulation are:
   - Envelope item
     Freq[1] = RFfreq
     Order[1] = 1
     StatusLevel=2
     Stop=tstop

---

3-6  PtRF_CDMA_ESG_REV (Pwr Src, RF Carrier Modulated by ESG Rev. Link CDMA Signal)
Step=tstep
Other=SavetoDataset=no

• VAR item
  bit_rate=1.2288 MHz
  RFfreq = 1.9 GHz
  Pavs = 0 dBm
  sam_per_bit = 4
  tstep = 1/(bit_rate × sam_per_bit)
  numSymbols = 256
  tstop = num Symbols/(bit_rate/2)

• PtRF_CDMA_ESG_REV item
  FO=RFfreq
  Power = dbmtow(Pavs)
  Z=50 Ohm
Modulated Sources

PtRF_CDMA_IS95_FWD (Pwr Src, RF Carrier Modulated by IS95 Fwd Link CDMA Signal)

Symbol

Parameters
F0 = carrier frequency
Power = RF output power
Z = RF output impedance
LinMod = additional linear modulation
Toffset = time offset into data array

Notes/Equations
1. This model generates a digitally-modulated RF signal that has the modulation characteristics of a base station CDMA signal. It does not contain any framing characteristics.
2. The bandlimiting filter coefficients come from IS-95 specifications. This source has higher adjacent channel power than that of PtRF_CDMA_ESG_FWD.
3. It is recommended that simulation timestep is equal to (0.25/1.2288 MHz), i.e., taking four samples per bit. Using other timestep values makes the source to interpolate between data samples and result in distorted spectrum.
4. Recommended controller setups for Envelope simulation are:
   - Envelope item
     Freq[1] = RFfreq
     Order[1] = 1
     StatusLevel = 2
     Stop = tstop
     Step = tstep
     Other = SavetoDataset = no
• VAR item
  bit_rate=1.2288 MHz
  RFfreq = 1.9 GHz
  Pavs = 0 dBm
  sam_per_bit = 4
  tstep = 1/(bit_rate × sam_per_bit)
  numSymbols = 256
  tstop = num Symbols/(bit_rate/2)

• PtRF_CDMA_IS95_FWD item
  FO=RFfreq
  Power = dbmtow(Pavs)
  Z=50 Ohm
Modulated Sources

PtRF_CDMA_IS95_REV (Pwr Src, RF Carrier Modulated by IS95 Rev. Link CDMA Signal)

Symbol

Parameters
F0 = carrier frequency
Power = RF output power
Z = RF output impedance

Notes/Equations
1. This model generates a digitally-modulated RF signal that has the modulation characteristics of a handset CDMA signal. It does not contain any framing characteristics. This source is implemented by interpolating data from a stored data file, which is faster than recalculating the source.

2. The bandlimiting filter coefficients come from IS-95 specifications. This source has higher adjacent channel power than that of PtRF_CDMA_ESG_REV.

3. An identical source called IS95RevLinkSrc that you can modify is located in the examples/Tutorial/ModSources_prj directory.

4. It is recommended that simulation timestep is equal to (0.25/1.2288 MHz), i.e, taking four samples per bit. Using other timestep values makes the source to interpolate between data samples and result in distorted spectrum.

5. Recommended controller setups for Envelope simulation are:
   - Envelope item
     Freq[1] = RFfreq
     Order[1] = 1
     StatusLevel=2
     Stop=tstop
     Step=tstep
     Other=SavetoDataset=no
• VAR item
  bit_rate=1.2288 MHz
  RFfreq = 1.9 GHz
  Pavs = 0 dBm
  sam_per_bit = 4
  tstep = 1/(bit_rate × sam_per_bit)
  numSymbols = 256
  tstop = num Symbols/(bit_rate/2)

• PtRF_CDMA_IS95_FWD item
  FO=RFfreq
  Power = dbmtow(Pavs)
  Z=50 Ohm
Modulated Sources

PtRF_CDMA2K_REV (Pwr Src, RF Carrier Modulated by CDMA2K Reverse Link Signal)

Symbol

Parameters
Freq = Carrier frequency
Power = Output power at RF output
R = Output impedance of RF output

Notes/Equations
1. This model generates a digitally-modulated RF signal that has the modulation characteristics of a mobile station CDMA2000 signal. It also contains framing characteristics. This source is implemented by interpolating data from a stored data file, which is faster than recalculating the source.

2. The stored data file is generated by the CDMA2000 Design Library. The bandlimiting filter coefficients come from IS-2000 specifications.

3. The data file contains 1 frame (20 msec) of CDMA2000 data (24576 chips at 1/1.2288 µsec per chip).

4. It is recommended that simulation timestep be set to (1/1.2288/4 µsec), that is, taking four samples per chip. For other timestep values the source interpolates between data samples and results in different or lower fidelity signal spectrum.

5. Recommended controller setups for Envelope simulation are:
   - Envelope item
     Freq[1] = RFfreq
     Order[1] = 1
     StatusLevel=2
     Stop=tstop
     Step=tstep
     Other=SavetoDataset=no
• VAR item
  chip_rate=1.2288 MHz
  RFfreq = 825 MHz
  Pavs = 0 dBm
  sam_per_chip = 4
  tstep = 1/(chip_rate \times sam_per_chip)
  numChips = 256
  tstop = numChips / chip_rate

• PtRF_CDMA2K_REV item
  Freq = RFfreq
  Power = dbmtow(Pavs)
  (R = 50 Ohm)

PtRF_DECT (Pwr Src, RF Carrier Modulated by DECT Signal)

Symbol

Parameters

- $F_0 =$ carrier frequency
- Power = RF output power
- $Z =$ RF output impedance

Notes/Equations

1. This model generates a digitally modulated RF signal that has the modulation characteristics of a DECT signal. Bit time is 0.868 µsec. NRZ data is Gaussian-filtered with $BT=0.5$.
2. The third terminal is the digital (-1/1 volt) bit sequence.
PtRF_EDGE_Uplink (Pwr Src, RF Carrier Modulated by EDGE Uplink Signal)

Symbol

Parameters
Freq = Carrier frequency
Power = Output power at RF output
R = Output impedance of RF output

Notes/Equations
1. This model generates a digitally-modulated RF signal that has the modulation characteristics of a mobile station EDGE signal. It also contains framing characteristics. This source is implemented by interpolating data from a stored data file, which is faster than recalculating the source.

2. The stored data file is generated by the EDGE Design Library.

3. The data file contains 1 TDMA frame (120/26 msec) of EDGE data (1250 symbols at 48/13 µsec per symbol). One EDGE frame contains 8 time slots with each time slot containing 156.25 symbols. The EDGE frame generated by this source contains data (normal burst with 8PSK modulation) in the second time slot, all other seven time slots are idle (no signal). This frame represents one active user in the EDGE uplink.

4. It is recommended that simulation timestep be set to (6/1625/8 msec), that is, taking eight samples per symbol. For other timestep values, the source interpolates between data samples and results in a different or lower fidelity signal spectrum.

5. Recommended controller setups for Envelope simulation are:
   • Envelope item
     Freq[1] = RFfres
     Order[1] = 1
Modulated Sources

\[
\begin{align*}
\text{StatusLevel} &= 2 \\
\text{Stop} &= \text{tstop} \\
\text{Step} &= \text{tstep} \\
\text{Other} &= \text{SavetoDataset} = \text{no}
\end{align*}
\]

- **VAR item**
  - \(\text{sym\_rate} = \frac{1625}{6} \text{ kHz}\)
  - \(\text{RFfreq} = 890.2 \text{ MHz}\)
  - \(\text{Pavs} = 0 \text{ dBm}\)
  - \(\text{sam\_per\_sym} = 8\)
  - \(\text{tstep} = \frac{1}{(\text{sym\_rate} \times \text{sam\_per\_sym})}\)
  - \(\text{numSymbols} = 256\)
  - \(\text{tstop} = \frac{\text{numSymbols}}{\text{sym\_rate}}\)

- **PtRF\_EDGE\_Uplink item**
  - \(\text{FO} = \text{RFfreq}\)
  - \(\text{Power} = \text{dbmtow}(\text{Pavs})\)
  - \(\text{R} = 50 \text{ Ohm}\)

6. For an overview of EDGE systems, refer to the EDGE Design Library, Introduction chapter (Manuals > Components > Signal Processing Components > EDGE).
PtRF_GSM (Pwr Src, RF Carrier Modulated by GSM Signal)

Symbol

Parameters

- $F_0 =$ carrier frequency
- Power = RF output power
- $R_{out} =$ RF output resistance
- DataRate = digital modulation data rate
- InitBits = initial state of PRBS data generator

Notes/Equations

1. This model generates a continuously digitally modulated RF signal that has the modulation characteristics of a transmitted GSM signal. It does not contain GSM framing or pulse modulation characteristics. It consists of a pseudorandom data generator (PRBS) feeding a Gaussian filter with a bandwidth time product of 0.3 that then FM modulates a voltage source to generate the RF output waveform. The baseband digital waveform is also output from this source.

2. The user can define the carrier frequency, power and output resistance of the RF output. The data rate can also be set, along with the initial seed value of the PRBS generator. The PRBS generator has 17 stages, with maximal length taps at bits 17 and 3. The baseband digital output is a $-1V$ to $+1V$ digital bit stream with a 1-ohm output impedance.

3. There is a time delay of 2.5-bit periods plus one analysis timestep between the digital output and the modulated RF output. The RF output can be floated with any common mode voltage on its two outputs, whereas the baseband output is always referenced to ground.
Modulated Sources

PtRF_NADC (Pwr Src, RF Carrier Modulated by NADC Signal)

Symbol

Parameters

- \( F_0 \) = carrier frequency
- Power = RF output power
- \( Z \) = RF output impedance
- LinMod = additional linear modulation
- Toffset = time offset into data array

Notes/Equations

1. This model generates a continuously digitally modulated RF signal that has the modulation characteristics of a transmitted NADC signal. It does not contain framing or pulse modulation characteristics. This source is implemented by interpolating data from a stored data file, which is faster than recalculating the source.

2. The user can define the carrier frequency, power and output impedance of the RF output.

3. Additional amplitude or phase modulation can be added by using LinMod to define an additional time varying modulation function. And, Toffset can be set to delay into the pseudorandom sequence to vary the effective starting point of the digital modulation sequence.

   The \(-1V\) to \(+1V\) baseband digital data stream is also available as an output at the third terminal and has a 1-ohm source resistance.

4. This 48.6 Kbps data stream was generated by using a PRBS source to modulate an RF source using the pi/4DQPSK modulator and filtering the signal with a root raised-cosine filter amplifier with a rolloff factor of 0.35. The filter uses an impulse response equal to 40 symbol periods and a Hanning window. The data
sequence is 1024 PRBS symbols in addition to 46 zero-padding symbols. This pattern repeats if necessary, depending on the analysis stop time.

5. The data is stored at 10 samples per symbol. If the analysis timestep is a multiple of this value, then there is no interpolation error. With other timestep values, spurious spectra may appear, but are more than 80dB below the main signal. Cubic interpolation is used on the RF output to minimize this error. Linear interpolation is used on the baseband, digital output to maintain its digital nature.
Modulated Sources

PtRF_PHS (Pwr Src, RF Carrier Modulated by PHS Signal)

Symbol

Parameters

F0 = carrier frequency
Power = RF output power
Z = RF output impedance
LinMod = additional linear modulation
Toffset = time offset into data array

Notes/Equations

1. This model generates a continuously digitally modulated RF signal that has the modulation characteristics of a transmitted PHS signal. It does not contain framing or pulse modulation characteristics. This source is implemented by interpolating data from a stored data file, which is faster than recalculating the source.

2. The user can define the carrier frequency, power and output impedance of the RF output.

3. Additional amplitude or phase modulation can be added using LinMod to define an additional time-varying modulation function. And, Toffset can be set to delay into the pseudorandom sequence, to vary the effective starting point of the digital modulation sequence.

   The −1V to +1V baseband digital data stream is also available as an output at the third terminal and has a 1-ohm source resistance.
PtRF_Pulse (Pwr Src, RF Pulse Train)

Symbol

Parameters

Num = port number (value type: integer)
Z = source impedance
P = carrier power during pulse
Freq = RF carrier frequency
OffRatio = linear amplitude ratio of off to on pulse portions
Delay = delay time before first pulse
Rise = rise time of pulse
Fall = fall time of pulse
Width = width of constant portion of pulse
Period = pulse repetition period
Chirp = linear frequency modulation during pulse
Phase0 = initial phase of pulse carrier
Noise = enable port thermal noise: YES, NO
Pac = ac power
Vdc = open circuit dc voltage

Notes/Equations

1. This RF pulse power source creates a pulse modulated RF carrier with optional frequency chirping. The carrier frequency at the start of the pulse is defined by the Freq parameter. For envelope simulation, Freq identifies the closest analysis frequency. If the frequency is not close enough to an analysis frequency, a warning is issued and the source output is set to zero for that analysis.
Modulated Sources

2. The pulse amplitude characteristics are defined using the Gaussian-shaped erf_pulse function, so the pulse parameters have the same definition as in the VTPulse model with Edge=erf. OffRatio defines the low state of the pulse relative to the high state, defined by P; P may be complex (such as P=polar(dBmtoW(0),45), and time-varying to provide additional amplitude and phase modulation. When this source represents a baseband signal (transient or the baseband part of an envelope signal), then only the real part of the signal is output.

3. The additional frequency chirp is referenced to the frequency value at the Delay time point where the pulse first starts turning on. The chirp rate is calculated by dividing the Chirp parameter by the sum of Width, Rise and Fall time. The Chirp value then represents the amount of frequency shift over the full, extended pulse width. If OffRatio is not 0, this same chirp rate will continue until the next pulse starts, when it is reset to the Freq value.
PtRF_Step (Pwr Src, RF Step)

Symbol

Parameters

Num = port number (value type: integer)

Z = source impedance

P = steady state power

Freq = RF frequency

Delay = time delay before step

Rise = rise time of step

Noise = enable port thermal noise: YES, NO

Pac = ac power

Vdc = open circuit dc voltage

Notes/Equations

1. This RF step power source creates an RF carrier that is turned on after the start of the time-domain simulation. The carrier frequency is defined by Freq. For envelope simulations, Freq identifies the closest analysis frequency. If the frequency is not close enough to an analysis frequency, a warning is issued and the source output is set to zero for that analysis.

2. The carrier is turned on at the time specified by Delay. The turn-on duration is defined by Rise and uses the Gaussian-shaped rise time defined by the erf_pulse() function.

3. The P parameter can be complex and a function of time and will provide amplitude-only modulation with logarithmic scaling. Refer to the VtRF_Step source if linear or other modulation is desired. When this source represents a baseband signal (transient or the baseband part of an envelope signal), then only the real part of the signal is output.
Modulated Sources

**VtRF_Pulse (Voltage Source, RF Pulse)**

**Symbol**

![Symbol](image)

**Parameters**

- **Freq** = RF carrier frequency
- **Vpeak** = voltage envelope of pulse
- **OffRatio** = linear amplitude ratio of off to on pulse portions
- **Delay** = time delay before first pulse
- **Rise** = rise time of pulse
- **Fall** = fall time of pulse
- **Width** = width of constant portion of pulse
- **Period** = pulse repetition period
- **Chirp** = linear frequency modulation during pulse
- **Phase0** = initial phase of pulse carrier
- **Vdc** = dc voltage
- **Vac** = ac voltage
- **SaveCurrent** = save branch current: YES, NO

**Notes/Equations**

1. This RF pulse voltage source creates a pulse modulated RF carrier with optional frequency chirping. The carrier frequency at the start of the pulse is defined by the Freq parameter. For envelope simulation, the Freq parameter identifies the closest analysis frequency. If the frequency is not close enough to an analysis frequency, a warning is issued and the source output is set to zero for that analysis.
2. The pulse amplitude characteristics are defined using the Gaussian-shaped erf_pulse function, so the parameters have the same definition as in the VPulse with Edge=erf. OffRatio defines the low state of the pulse relative to the high state, defined by Vpeak. The Vpeak parameter can be complex and time-varying to provide additional amplitude and phase modulation. When this source represents a baseband signal (transient or the baseband part of an envelope signal), only the real part of the signal is output.

3. The additional frequency chirp is referenced to the frequency value at the Delay time point where the pulse first starts turning on. The chirp rate is calculated by dividing the Chirp parameter by the sum of the Width, Rise and Fall time. The Chirp value then represents the amount of frequency shift over the full, extended pulse width. If the OffRatio is not zero, this same chirp rate will continue until the next pulse starts, when it is reset to the Freq parameter value.

4. This source output in harmonic balance analyses is only the value at time=0. Additional source parameters that are available can be found in the perform/edit component dialog box.
Modulated Sources

VtRF_Step (Voltage Source, RF Step)

Symbol

Parameters
Freq = RF frequency
V = voltage envelope of step
Delay = time delay before step
Rise = rise time of step
Vdc = dc voltage
Vac = ac voltage
SaveCurrent = save branch current: YES, NO

Notes/Equations
1. This RF step voltage source creates an RF carrier that is turned on after the start of the time-domain simulation. The carrier frequency is defined by the Freq parameter. For envelope analyses, the Freq parameter identifies the closest analysis frequency. If the frequency is not close enough to an analysis frequency, a warning is issued and the source voltage is set to 0 for that analysis.

2. The carrier is turned on at the time specified by the Delay parameter. The turn-on duration is defined by the Rise parameter and uses the Gaussian-shaped rise time defined by the (erf_pulse() function).

3. The voltage parameter can be a complex value to define both the amplitude and phase of the carrier. It can also be a time-varying expression to put additional amplitude or phase modulation on the carrier. When this source represents a baseband signal (transient or the baseband part of an envelope signal), then only the real part of the signal is output.
Chapter 4: Noise Sources
Noise Sources

I_Noise (Noise Current Source)

Symbol

Parameters
I_Noise = noise current magnitude, in amperes

Notes/Equations
1. I_Noise is the rms noise current. For simulations other than noise analysis, it will be replaced by an open circuit.
I_NoiseBD (Bias-Dependent Noise Current Source)

Symbol

Parameters
K = multiplicative constant
Ie = dc bias current exponent
A0 = additive constant in the denominator
A1 = multiplication factor for the frequency
Fe = frequency exponent
Elem = ID of an element such as R, FET, BJT
Pin = element pin number or name

Range of Usage
A0 and A1 cannot be simultaneously set to zero.

Notes/Equations
1. For simulations other than noise analysis, I_NoiseBD is treated as an open circuit.
2. The values and the units of K, Ie, A0, A1, and Fe should be such that the strength of the noise source as calculated from the following expression results in amperes²/Hz.
Noise Sources

The noise spectral density of this source is given by

$$\langle \dot{i}^2 \rangle = \frac{K \times I_{dc} \times I_e}{A_0 + A_1 \times f \times F_e}$$

where $I_{dc}$ is the dc bias current in amperes and $f$ is the simulation frequency in hertz. The dc current is that flowing into the Pin of Elem. Depending on the values of $K$, $I_e$, $A_0$, $A_1$, and $F_e$ this source can be used as a flicker, burst, shot or thermal noise source. This can be explained by comparing the noise spectral density with the spectral density of a flicker, burst, shot and thermal noise source, given:

Flicker noise:

$$\langle \dot{i}^2 \rangle = \frac{K_f \times I_{dc} \times A_f}{f \times F_{fe}}$$

Burst noise:

$$\langle \dot{i}^2 \rangle = \frac{K_b \times I_{dc} \times A_b}{1 + \left( \frac{f}{F_b} \right)^2}$$

Shot noise: $\langle \dot{i}^2 \rangle = 2 \times q \times I_{dc}$

Thermal noise: $\langle \dot{i}^2 \rangle = 4 \times k \times T \times g$

Table 4-1 summarizes the values to which the parameters must be set to realize the types of noise sources.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Flicker</th>
<th>Burst</th>
<th>Shot</th>
<th>Thermal</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K$</td>
<td>$K_f$</td>
<td>$K_b$</td>
<td>$2\times q$</td>
<td>$4\times k \times T \times g$</td>
</tr>
<tr>
<td>$I_e$</td>
<td>$A_f$</td>
<td>$A_b$</td>
<td>1.0</td>
<td>0.0</td>
</tr>
<tr>
<td>$A_0$</td>
<td>0.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
</tr>
<tr>
<td>$A_1$</td>
<td>1.0</td>
<td>$(1/F_b)^2$</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>$F_e$</td>
<td>$F_{fe}$</td>
<td>2.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
</tbody>
</table>

3. This component has no default artwork associated with it.
NoiseCorr (Noise Source Correlation)

Symbol

Parameters

CorrCoeff = correlation coefficient
Source1 = source 1 name
Source2 = source 2 name

Notes/Equations

1. This source is used in noise analysis; it is not supported for Envelope or Transient noise analysis—a warning message will be issued when used in these analyses.

2. The sources that are correlated can be current or voltage sources.

The correlation coefficient is defined in this equation:

\[
\text{CorrCoeff} = \frac{\langle n_1, n_2 \rangle}{\sqrt{|n_1|^2 |n_2|^2}}
\]

where

CorrCoeff is the correlation coefficient between the sources

\( n_1 \) and \( n_2 \) are rms values of the noise generated by each source.
Noise Sources

**Noisy2Port (Linear Noisy 2ort Network)**

**Symbol**

![Symbol](image)

**Parameters**

- $NF_{\text{min}}$ = minimum noise figure
- $R_n$ = noise resistance
- $Sopt$ = optimum match for minimum noise figure

**Notes/Equations**

1. This source is used in noise analysis only; for other simulations, voltage source will be replaced by a short circuit and current source will be replaced by an open circuit.

2. If $NF_{\text{min}}$, $Sopt$, and $R_n$ are used to characterize noise, the following relation must be satisfied for a realistic model.

   \[
   \frac{R_n}{Z_0} \geq \frac{T_0(F_{\text{min}} - 1)|1 + Sopt|^2}{T^4} \cdot \frac{(1 - |S_{11}|^2)}{|1 - Sopt S_{11}|^2}
   \]

   A warning message will be issued if $R_n$ does not meet this criterion. If the noise parameters attempt to describe a system that requires negative noise (due to $R_n$ being too small), the negative part of the noise will be set to zero and a warning message will be issued.

3. $Sopt$ is always with respect to a reference impedance of 50 ohms. The reference impedance is not changeable.
**V_Noise (Noise Voltage Source)**

**Symbol**

![Diagram](image)

**Parameters**

- **V_Noise** = noise voltage amplitude, in volts
- **SaveCurrent** = save branch current: YES, NO

**Notes/Equations**

1. This source is the rms noise voltage. For simulations other than noise analysis, it will be replaced by a short circuit.

2. Setting \( V_{\text{Noise}} = 1 \mu\text{V} \) specifies a spectral noise density in units of \( \frac{\text{volts}}{\sqrt{\text{Hz}}} \).

RMS noise voltage calculated by

\[
\sqrt{\text{mean}(V_{\text{source}}^2)}
\]

where \( V_{\text{source}} \) is the random noise voltage at each timestep is

\[
1 \mu\text{V} \sqrt{\frac{1}{\text{Step}}}
\]

Therefore, rms noise voltage is

\[
1 \mu\text{V} \sqrt{\frac{1}{0.1\text{msec}}} = 100 \mu\text{V}
\]

For baseband envelope, noise is distributed in a bandwidth out to 0.5/Step, rms noise voltage of the baseband envelope is

\[
1 \mu\text{V} \sqrt{\frac{0.5}{0.1\text{msec}}} = 70.7 \mu\text{V}
\]
3. Where V_Noise information refers to step, it is referring to the Circuit Envelope timestep only. The transient noise analysis gives the exact answer only when fixed timestep is used and the noise bandwidth is equal to $1/(2 \times \text{timestep})$. The answer is slightly less when other timesteps or the variable timestep algorithm is used. For those cases, a special algorithm is used to keep the noise from changing as the timestep changes.

The noise bandwidth is used to define a sampling timestep:

$$\text{step} = 1/(2 \times \text{NoiseBandwidth})$$

A random value with the proper Gaussian distribution is generated at each time $= i \times \text{step}$, where $i$ is an integer from 1 to infinity. When the simulator wants to get the random value of the source in variable timestep mode, it gets the two random values (fully uncorrelated) on either side of the current time and performs linear interpolation.
**V_NoiseBD (Bias-dependent Noise Voltage Source)**

Symbol

Parameters

- **K** = multiplicative constant
- **Ie** = dc bias current exponent
- **A0** = additive constant in the denominator
- **A1** = multiplication factor for the frequency
- **Fe** = frequency exponent
- **Elem** = ID of an element such as R, FET, BJT
- **Pin** = element pin number or name

**Range of Usage**

A0 and A1 cannot be simultaneously set to zero.

**Notes/Equations**

1. For simulations other than noise analysis, V_NoiseBD is treated as a short circuit.

2. The values and the units of K, Ie, A0, A1, and Fe should be such that the strength of the noise source as calculated from the following expression results in volts²/Hz.

   The noise spectral density of this source is given by:

   \[
   \langle V^2 \rangle = \frac{K \times I_{dc}^{Ie}}{A0 + A1 \times f^{Fe}}
   \]

   where \(I_{dc}\) is the dc bias current in amperes and \(f\) is the simulation frequency in hertz. The dc current is that flowing into the Pin of Elem. Depending on the values of K, Ie, A0, A1, and Fe this source can be used as a flicker, burst, shot or...
Noise Sources

thermal noise source. This can be explained by comparing the noise spectral density with the spectral density of a flicker, burst, shot and thermal noise source.

Flicker noise:

\[ <v^2> = \frac{K_f \times I_{dc}^{Af}}{F^{F_f}} \]

Burst noise:

\[ <v^2> = \frac{K_b \times I_{dc}^{Ab}}{1 + \left(\frac{f}{F_b}\right)^2} \]

Shot noise: \( <v^2> = 2 \times q \times I_{dc} \)

Thermal noise: \( <v^2> = 4 \times k \times T \times g \)

Table 4-2 summarizes the values to which the parameters must be set to realize the types of noise sources.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Flicker</th>
<th>Burst</th>
<th>Shot</th>
<th>Thermal</th>
</tr>
</thead>
<tbody>
<tr>
<td>K</td>
<td>K_f</td>
<td>K_b</td>
<td>2xq</td>
<td>4xkxTxg</td>
</tr>
<tr>
<td>I_e</td>
<td>A_f</td>
<td>A_b</td>
<td>1.0</td>
<td>0.0</td>
</tr>
<tr>
<td>A_0</td>
<td>0.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
</tr>
<tr>
<td>A_1</td>
<td>1.0</td>
<td>(1/F_b)^2</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>F_e</td>
<td>F_fe</td>
<td>2.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
</tbody>
</table>

3. When using V_NoiseBD, the parameter K should be properly scaled such that it yields Thevenin equivalent of the above current sources.

4. This component has no default artwork associated with it.
Chapter 5: Time Domain Sources

Introduction

Independent sources that do not fit in the frequency-domain category are placed in
the time-domain sources category.

Vt prefixes are transient voltage sources; It prefixes are transient current sources; Pt
prefixes are transient power sources.

When time domain sources are used in S-parameter simulation, voltage sources are
treated as short circuits, current sources are treated as open sources, and power
sources are treated as impedances.

Time-domain sources are generally not used for frequency-domain simulation such as
ac and harmonic balance.
Time Domain Sources

ClockWjitter (Voltage Source: Clock with Jitter)

Symbol

Parameters

Low = low-level voltage (default: 0V)
High = high-level voltage (default: 1V)
Rout = output resistance (default: 1 ohm)
Delay = delay time (default: 0 nsec)
Rise = rise time (default: 1 nsec)
Fall = fall time (default: 1 nsec)
Width = pulse width (default: 3 nsec)
Period = pulse period (default: 10 nsec)
Jitter = jitter time (default: 0 nsec)

Range of Usage

Delay \geq 0, \text{ Rise} \geq 0, \text{ Fall} \geq 0
Width > 0
Width + \text{ Rise} + \text{ Fall} \leq \text{ Period}

Notes/Equations

1. This source is a voltage source in series with a resistor Rout. If Rout is very small, it behaves like an ideal voltage source.

2. Jitter is specified in seconds. It models the timing jitter of a clock signal. The period of the pulse varies from nominal with a Gaussian distribution, where $\sigma = \text{Jitter}$. It exhibits a maximum deviation of $\pm 3\sigma$. The pulse width is not affected by the jitter.

3. A transient simulation example is shown in Figure 5-1.
4. For general information regarding time domain sources, refer to the “Introduction” on page 5-1.
Time Domain Sources

I_DC (DC current source)

Symbol

Parameters

$Idc = $ dc current (default: 1 mA)

$Iac = $ ac current; value used for ac analysis only

Notes/Equations

1. $I_DC$ is an ideal dc current source. Positive current flows into the source at pin 1 and out of the source at pin 2.

2. This source is used in all simulations. When not in use, it is treated as an open circuit.

3. Table 5-1 lists the DC operating point parameters that can be sent to the dataset.

4. For general information regarding time domain sources, refer to the “Introduction” on page 5-1.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

Table 5-1. DC Operating Point Information
**ItDataset (Current Source, Time Domain Waveform Defined in Dataset)**

**Symbol**

![Symbol Diagram]

**Parameters**

- **Dataset** = dataset name
- **Expression** = dataset variable or expression
- **Freq** = carrier frequency (default: 0 GHz)
- **Gain** = gain to apply to dataset values; can be complex and time varying (default: 1.0)
- **Tmax** = maximum dataset time to use
- **Toffset** = initial dataset time offset
- **Tscale** = time speed-up scaling factor
- **Idc** = dc offset current (default: 0 mA)
- **Interpolation** = interpolation mode: Linear (default), Cubic Spline, Cubic, Value Lookup, Ceiling Value Lookup, Floor Value Lookup, and Value

**Notes/Equations**

1. This data-based, time-domain waveform current source is defined by a time domain dataset variable. The dataset variable must have time as its independent swept axis. This source can be used in transient or envelope simulation.

2. Set the **Expression** parameter to the dependent variable name of the dataset. If the dataset has time & current for example, this must be set to current.

3. The carrier frequency defined by the **Freq** parameter is independent of the dataset.

   For envelope simulation, the **Freq** parameter identifies the closest analysis frequency. If the frequency is not close enough to an analysis frequency, a warning is issued and the source current is set to zero for that analysis.
Time Domain Sources

4. If $T_{\text{max}}$ is not given, the simulation $T_{\text{stop}}$ must not exceed the time range of the stored variable. The output current at a given time is the interpolated dataset variable value at that time multiplied by the Gain parameter, evaluated at that time value. The dataset interpolation, if needed, is performed using linear or spline interpolation of the real and imaginary values. The Gain parameter can be complex and time varying. When this source represents a baseband signal (transient or the baseband part of an envelope signal), then only the real part of the signal is generated. For non-baseband signals this output current is the complex envelope at the specified carrier frequency. The dataset variable and Gain parameter may be real, even for non-baseband signals, in which case they are simply defining the amplitude modulation of the carrier.

If $T_{\text{max}}$ is given, this source also allows the time axis to be scaled and will re-cycle through the dataset as many times as is necessary. This allows a single waveform that was captured, either by measurements or by simulation, and stored into a dataset to be used in different simulations with different time scales, and be translated to different carrier frequencies and converted into an indefinitely long, periodic waveform.

The $T_{\text{max}}$ parameter is the maximum dataset time value to use from the dataset. If time values greater than this are requested by the simulation, it will cycle back to dataset time=0.

The $T_{\text{offset}}$ parameter is the dataset time value that this source initially starts at when simulation time=0. This allows different instances of this source to effectively create different waveforms by starting at different points in the dataset.

The $T_{\text{scale}}$ parameter is the scaling applied to the simulator time to get the dataset time. $T_{\text{scale}} > 1$ speeds up the waveform, increasing the apparent frequency and bandwidth of the stored waveform.

The relationship between the dataset time, $T_{\text{ds}}$, and the actual simulation time, time, is

$$T_{\text{ds}} = \text{time} \quad T_{\text{max}} = 0$$

$$T_{\text{ds}} = (T_{\text{offset}} + \text{rem}(T_{\text{scale}} \times \text{time}, T_{\text{max}}), \quad T_{\text{max}} \neq 0)$$

with the modulo remainder function

$$\text{rem}(x, y) = \left( x - \text{int}\left(\frac{x}{y}\right) \times y \right)$$
It is possible to use a negative $T_{	ext{scale}}$ factor to time-reverse a waveform, although $T_{	ext{offset}}$ must be set to greater than $T_{	ext{scale}} \times T_{\text{stop}}$, to avoid using a negative number in the $\text{rem}(\ )$ function.

5. The interpolation modes are: Linear, Cubic Spline, Cubic, Value Lookup, Ceiling Value Lookup, Floor Value Lookup, and Value. Consider an arbitrary set of time-current data pairs, namely $I_0$ at $t=0$, $I_1$ at $t=t_1$, $I_2$ at $t=t_2$, ..., $I_n$ at $t=t_n$.

Using the Value Lookup interpolation mode, the interpolated current will be determined as follows:

\[
\begin{align*}
    i_0 & \quad 0 < t < \frac{t_1}{2} \\
    \frac{i_0 + i_1}{2} & \quad t = \frac{t_1}{2} \\
    i_1 & \quad \frac{t_1}{2} < t < \frac{t_1 + t_2}{2} \\
    \frac{i_1 + i_2}{2} & \quad t = \frac{t_1 + t_2}{2} \\
    i_2 & \quad \frac{t_1 + t_2}{2} < t < \frac{t_2 + t_3}{2} \\
    \frac{i_2 + i_3}{2} & \quad t = \frac{t_2 + t_3}{2} \\
    i_n & \quad \frac{t_n - 1 + t_{n-1}}{2} < t < \frac{t_n + t_{n-1}}{2} \\
    \frac{i_{n-1} + i_n}{2} & \quad t = \frac{t_n + t_{n-1}}{2}
\end{align*}
\]

Using the Ceiling Value Lookup interpolation mode, the interpolated current will be determined as follows:

\[
\begin{align*}
    i_0 & \quad t = 0 \\
    i_1 & \quad 0 < t \leq t_1 \\
    \frac{i_1 + i_2}{2} & \quad t_1 < t \leq t_2 \\
    i_n & \quad t_{n-1} < t \leq t_n
\end{align*}
\]

Using the Floor Value Lookup interpolation mode, the interpolated current will be determined as follows:
Time Domain Sources

The Value interpolation mode is to be used when the mode is variable or unknown. The entered parameter for Value interpolation mode should be a string (or integer) from the following set:

- "linear" (0)
- "spline" (1)
- "cubic" (2)
- "index_lookup" (3)
- "value_lookup" (4)
- "ceil_value_lookup" (5)
- "floor_value_lookup" (6)

Refer to the DataAccessComponent documentation for additional information about the interpolation modes available for ItDataset.

6. Table 5-2 lists the DC operating point parameters that can be sent to the dataset.

Table 5-2. DC Operating Point Information

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

7. For general information regarding time domain sources, refer to the “Introduction” on page 5-1.
ItExp (Current Source, Exponential Decay)

Symbol

Parameters
- \( I_{\text{Low}} \) = initial current (default: 0 mA)
- \( I_{\text{High}} \) = pulse current (default: 1 mA)
- \( \text{Delay1} \) = rise delay time (default: 0 nsec)
- \( \text{Tau1} \) = rise time constant (default: 1 nsec)
- \( \text{Delay2} \) = fall delay time (default: 1 nsec)
- \( \text{Tau2} \) = fall time constant (default: 1 nsec)

Range of Usage
- \( \text{Delay1} \geq 0 \)
- \( \text{Tau1} \geq 0 \)
- \( \text{Delay2} \geq 0 \)
- \( \text{Tau2} \geq 0 \)

Notes/Equations
1. If \( \text{Tau1} \) or \( \text{Tau2} \) = 0, it is replaced by \( \text{MaxTimeStep} \) from the transient simulation, or by \( \text{Step} \) from the envelope simulation.

   In SPICE, the equivalent to this source is a current source with the exponential waveform argument \( \text{EXP} \) and its parameters.

2. The current is given by:
   - \( I = I_{\text{Low}} \quad 0 \leq t \leq \text{Delay1} \)
   - \( I = I_{\text{Low}} + (I_{\text{High}} - I_{\text{Low}}) \times \left[ 1 - e^{-\frac{t - \text{Delay1}}{\text{Tau1}}} \right] \quad \text{Delay1} < t \leq \text{Delay2} \)
Time Domain Sources

\[ I = I_{Low} + (I_{High} - I_{Low}) \times \left[ \frac{-(t - Delay_1)}{\tau_1} \right] \]
\[ + (I_{High} - I_{Low}) \times \left[ \frac{-(t - Delay_2)}{\tau_2} \right] \quad Delay_2 < t \]

3. Table 5-3 lists the DC operating point parameters that can be sent to the dataset.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

4. For general information regarding time domain sources, refer to the “Introduction” on page 5-1.
ItPulse (Current Source, Pulse with Linear, Cosine or Error Function Edge Shape)

Symbol

Parameters

I_Low = initial current (default: 0 mA)
I_High = pulse current (default: 1 mA)
Delay = time delay (default: 0 nsec)
Edge = rise and fall edge type: linear (default), cosine, erf
Rise = rise time (default: 1 nsec)
Fall = fall time (default: 1 nsec)
Width = pulse width (default: 3 nsec)
Period = pulse period (default: 10 nsec)

Notes/Equations

1. ItPulse is a time-periodic pulse-train current source for use with transient or envelope simulations; it is treated as an open circuit in all other simulations.

2. If Rise or Fall=0, it is replaced by MaxTimeStep from the transient simulation, or Step from the envelope simulation.

3. If Edge=linear, the rising and falling edge is a linear ramp. In SPICE, the equivalent to this source is a current or voltage source with the pulse waveform argument PULSE and its parameters.

   The intermediate points are determined by linear interpolation. Values greater than those specified are set by the parameter Period.
Time Domain Sources

<table>
<thead>
<tr>
<th>Time</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>low</td>
</tr>
<tr>
<td>Delay</td>
<td>low</td>
</tr>
<tr>
<td>Delay + Rise</td>
<td>high</td>
</tr>
<tr>
<td>Delay + Rise + Width</td>
<td>high</td>
</tr>
<tr>
<td>Delay + Rise + Width + Fall</td>
<td>low</td>
</tr>
<tr>
<td>Period</td>
<td>low</td>
</tr>
</tbody>
</table>

If Edge=erf, instead of the rise and fall portions being linear ramps, this source generates a pulse based on the error function, giving a different shape to the rising and falling edges. By not having abrupt changes in slope, the pulse shape is more realistic and its frequency spectrum decreases more rapidly.

For the error function pulse, the rise and fall time define the total transition period and the maximum slope is greater than \((I_{\text{High}} - I_{\text{Low}})/\text{Rise}\). (See Figure 5-3.)

This source uses \(1-\text{erfc}(x), (-2 < x < 2)\) to generate the transition region and has a peak slope that is approximately 2.25 times the linear rise time. Due to the faster slope, the 3db bandwidth of the output pulse are larger for a given rise time.

The shape of the waveform is shown in Figure 5-4; the intermediate points during rise and fall time are determined by interpolation.
If Edge=cosine, this source generates cosine-shaped rising and falling edges. By not having abrupt changes in slope, the pulse shape is more realistic and its frequency spectrum decreases more rapidly.

For the cosine pulse, the rise and fall time define the total transition period and the maximum slope is greater than \((I_{\text{High}}-I_{\text{Low}})/\text{Rise}\). (See Figure 5-3.)

4. Table 5-4 lists the DC operating point parameters that can be sent to the dataset.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

5. For general information regarding time domain sources, refer to the “Introduction” on page 5-1.
Time Domain Sources

ItPWL (Current Source, Piecewise Linear)

Symbol

Parameters

\[ I_{T\text{ran}} = \text{pwl (time, time-current pairs) or pwlr (time, Ncycles, time-current pairs)} \]

Notes/Equations

1. The piecewise linear current versus time data are specified with a \( \text{pwl()} \) function. The syntax for \( \text{pwl(time, T_i, I_i, ...)} \) specifies that at time \( = T_i \) the current is \( I_i \). The value of the source at intermediate values of time is determined by using linear interpolation on the input values.

   In SPICE, the equivalent to this source is a current source with the piecewise linear waveform argument PWL and its parameters.

2. If the piecewise linear waveform needs to be repeated for several cycles, a \( \text{pwlr()} \) function can be used. The syntax for \( \text{pwlr(time, Ncycles, T_i, I_i, ...)} \) where Ncycles is the number of cycles to be repeated.

3. Table 5-5 lists the DC operating point parameters that can be sent to the dataset.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

4. For general information regarding time domain sources, refer to the “Introduction” on page 5-1.
ItSFFM (Current Source, Decaying Single-Frequency FM Wave)

Symbol

<table>
<thead>
<tr>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>( I_{dc} ) = initial current offset, in amperes (default: 0 mA)</td>
</tr>
<tr>
<td>Amplitude = sinusoidal wave amplitude, in amperes (default: 1 mA)</td>
</tr>
<tr>
<td>CarrierFreq = carrier frequency, in Hertz (default: 1 GHz)</td>
</tr>
<tr>
<td>ModIndex = modulation index (default: 0.5)</td>
</tr>
<tr>
<td>SignalFreq = signal frequency, in Hertz (default: 1 MHz)</td>
</tr>
</tbody>
</table>

Notes/Equations

1. In SPICE, the equivalent to this source is a current source with the single-frequency FM source waveform argument SFFM and its parameters.

2. The shape of the waveform is described in the following equation.

   \[
   I_{ac} = I_{dc} + \text{Amplitude} \times \sin(2\pi \text{CarrierFreq} \times \text{time}) + \text{ModIndex} \times \sin(2\pi \text{SignalFreq} \times \text{time}) \\
   = A \times \sin(2\pi f_c t + \alpha \times \sin 2\pi f_s t)
   \]

3. Table 5-6 lists the DC operating point parameters that can be sent to the dataset.

Table 5-6. DC Operating Point Information

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>
ItSine (Current Source, Decaying Sine Wave)

Symbol

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Idc</td>
<td>Initial current offset</td>
<td>0 mA</td>
</tr>
<tr>
<td>Amplitude</td>
<td>Sinusoidal wave amplitude</td>
<td>1 mA</td>
</tr>
<tr>
<td>Freq</td>
<td>Sinusoidal wave frequency</td>
<td>1 GHz</td>
</tr>
<tr>
<td>Delay</td>
<td>Time delay</td>
<td>0 nsec</td>
</tr>
<tr>
<td>Damping</td>
<td>Damping factor</td>
<td>0 1/sec</td>
</tr>
<tr>
<td>Phase</td>
<td>Initial phase</td>
<td>0</td>
</tr>
</tbody>
</table>

Range of Usage

Freq > 0
Delay ≥ 0

Notes/Equations

1. ItSine defines an ac sinusoidal current source, at a specified frequency and phase, including its turn-on characteristics for use with transient analysis. In SPICE, the equivalent to this source is a current source with the sinusoidal waveform argument SIN and its parameters.

2. ItSine has a value of \[Idc + \text{Amplitude} \times \sin(\text{phase})\] from \(t=0\), until \(t=\text{Delay}\). It then becomes an exponentially damped sine wave described by

\[
I = \text{Idc} + \text{Amplitude} \times \sin \left[ 2\pi \left( Freq(t - \text{Delay}) + \frac{\text{Phase}}{360} \right) \right] 
\times e^{-(t - \text{Delay}) \times \text{Damping}}
\]

where \(t\) is time.
3. Table 5-7 lists the DC operating point parameters that can be sent to the dataset.

Table 5-7. DC Operating Point Information

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

4. For general information regarding time domain sources, refer to the “Introduction” on page 5-1.
Time Domain Sources

**ItStep (Current Source, Step)**

**Symbol**

![Symbols](image)

**Parameters**
- \( I_{\text{Low}} \) = initial current (default: 0 mA)
- \( I_{\text{High}} \) = pulse current (default: 1 mA)
- Delay = time delay (default: 0 nsec)
- Rise = rise time (default: 1 nsec)

**Notes/Equations**

1. In SPICE, the equivalent to this source is a current or voltage source with the step waveform argument STEP and its parameters.

2. Table 5-8 lists the DC operating point parameters that can be sent to the dataset.

**Table 5-8. DC Operating Point Information**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

3. For general information regarding time domain sources, refer to the “Introduction” on page 5-1.
ItUserDef (Current Source, User-Defined)

Symbol

Parameters

I\_Tran = transient current (default: damped\_sin(time) )
Idc = dc current
Iac = ac current (default 1 mA)

Notes/Equations

1. Typically, I\_Tran is assigned an equation. This equation can be defined as a function of time by using the program reserved variable time in it. As the value of time is swept in transient or envelope simulation, the amplitude of the current source will take on the value of the equation.

2. Note that a variable or equation is unitless. However, the value of I\_Tran as given by the result of a variable or equation will be assumed to be in amperes. The value of time will be the current simulation time in seconds.

3. There are several built-in functions that implement the standard SPICE sources, such as pwl and pulse. For a transient analysis, the ItUserDef source current is the sum of the value specified in the Idc and I\_Tran parameters.

Example

\[ i(t) = \text{pwl}(t, 0, 0, 1, 10\text{ns}, 1, 15\text{ns}, 0) + \text{damped\_sin}(t) \]

4. The Iac parameter is used in AC simulations and does not affect transient simulation. An example for specifying magnitude and phase would be Iac=polar(2,45), where 2 is the magnitude and 45 is the phase. For more parameter options (such as frequency) on an AC source, use the I\_AC component on the Sources-Freq Domain palette.

5. Table 5-9 lists the DC operating point parameters that can be sent to the dataset.
Time Domain Sources

Table 5-9. DC Operating Point Information

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

6. For general information regarding time domain sources, refer to the “Introduction” on page 5-1.
V_DC (DC Voltage Source)

Symbol

Parameters

Vdc = dc voltage, in volts (default: 1.0V)
Vac = ac voltage, in volts; value used for ac analysis only
SaveCurrent = save branch current (default: yes)

Notes/Equations

1. V_DC can be used in all simulations. When not in use, it is treated as a short circuit.

2. Table 5-10 lists the dc operating point parameters that can be sent to the dataset.

Table 5-10. DC Operating Point Information

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

3. For general information regarding time domain sources, refer to the “Introduction” on page 5-1.
Time Domain Sources

VtBitSeq (Voltage Source, Pseudo Random Pulse Train Defined at Continuous Time by Bit Sequence)

Symbol

\[ + \]

\[ - \]

Parameters

- Vlow = minimum voltage level (default: 0V)
- Vhigh = maximum voltage level (default: 5V)
- Rate = bit rate (default: 50 MHz)
- Rise = rise time of pulse (default: 1 nsec)
- Fall = fall time of pulse (default: 1 nsec)
- BitSeq = bit sequence (default: 101010)
- SaveCurrent = save branch current: YES (default), NO

Notes/Equations

1. BitSeq allows you to vary the waveform of a pulse: an arbitrary bit pattern such as 101010 (default), or considerably longer and more varied, such as 11100001111101. When the end of the sequence is reached, the sequence is repeated. A specification of 1 sets voltage to Vhigh, 0 sets it to Vlow.

   Note To edit BitSeq, enter a value enclosed with double quote symbols.

2. VtBitSeq is used for transient simulations; Vf_BitSeq is recommended for frequency simulations.

3. Table 5-11 lists the dc operating point parameters that can be sent to the dataset.
Table 5-11. DC Operating Point Information

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

4. For general information regarding time domain sources, refer to the “Introduction” on page 5-1.
Time Domain Sources

VtDataset (Voltage Source, Time Domain Waveform Defined in Dataset)

Symbol

Parameters

Dataset = dataset name
Expression = dataset variable or expression
Freq = carrier frequency (default: 0 GHz)
Gain = apply to dataset values; can be complex and time varying (default: 1.0)
Tmax = maximum dataset time to use
Toffset = initial dataset time offset
Tscale = time speed-up scaling factor
Vdc = dc offset voltage (default: 0 V)
Interpolation = interpolation mode: Linear (default), Cubic Spline, Cubic, Value Lookup, Ceiling Value Lookup, Floor Value Lookup, and Value
SaveCurrent = save branch current: yes (default), no

Notes/Equations

1. This data-based, time-domain waveform voltage source is defined by a time domain dataset variable. The dataset variable must have time as its independent swept axis. This source can be used in transient or envelope simulation.

2. Set the Expression parameter to the dependent variable name of the dataset. If the dataset has time & voltage for example, this must be set to voltage.

3. The carrier frequency defined by the Freq parameter is independent of the dataset.
For envelope simulation, the Freq parameter identifies the closest analysis frequency. If the frequency is not close enough to an analysis frequency, a warning is issued and the source voltage is set to zero for that analysis.

4. If Tmax is not given, the simulation Tstop must not exceed the time range of the stored variable. The output voltage at a given time is the interpolated dataset variable value at that time multiplied by the Gain parameter, evaluated at that time value. The dataset interpolation, if needed, is performed using linear or spline interpolation of the real and imaginary values. The Gain parameter can be complex and time varying. When this source represents a baseband signal (transient or the baseband part of an envelope signal), then only the real part of the signal is generated. For non-baseband signals this output voltage is the complex envelope at the specified carrier frequency. The dataset variable and Gain parameter may be real, even for non-baseband signals, in which case they are simply defining carrier amplitude modulation.

If Tmax is given, this source also allows the time axis to be scaled and will re-cycle through the dataset as many times as is necessary. This allows a single waveform that was captured, either by measurements or by simulation, and stored into a dataset to be used in different simulations with different time scales, and be translated to different carrier frequencies and converted into an indefinitely long, periodic waveform.

The Tmax parameter is the maximum dataset time value to use from the dataset. If time values greater than this are requested by the simulation, it will cycle back to dataset time=0.

The Toffset parameter is the dataset time value that this source initially starts at when simulation time=0. This allows different instances of this source to effectively create different waveforms by starting at different points in the dataset.

The Tscale parameter is the scaling applied to the simulator time to get the dataset time. A number greater than 1 speeds up the waveform, increasing the apparent frequency and bandwidth of the stored waveform.

The relationship between the dataset time, Tds, and the actual simulation time, time, is

\[
T_{ds} = \text{time} \quad \text{Tmax} = 0
\]

\[
T_{ds} = (\text{Toffset} + \text{rem}(T_{scale} \times \text{time}, \text{Tmax}), \quad \text{Tmax} \neq 0)
\]

with the modulo remainder function
Time Domain Sources

\[ \text{rem}(x, y) = \left( x - \text{int} \left( \frac{x}{y} \right) \right) \times y \]

It is possible to use a negative \( T \text{scale} \) factor to time-reverse a waveform, although \( T \text{offset} \) must be set to greater than \( T \text{scale} \times T \text{stop} \), to avoid using a negative number in the \( \text{rem}( \ ) \) function.

5. The interpolation modes are Linear, Cubic Spline, Cubic, Value Lookup, Ceiling Value Lookup, Floor Value Lookup, and Value. Consider an arbitrary set of time-voltage data pairs: \( V_0 \) at \( t=0 \), \( V_1 \) at \( t=t_1 \), \( V_2 \) at \( t=t_2 \), ..., \( V_n \) at \( t=t_n \).

Using the Value Lookup interpolation mode, the interpolated voltage will be determined as follows:

\[
\begin{align*}
V_0 & \quad 0 < t < \frac{t_1}{2} \\
\frac{V_0 + V_1}{2} & \quad t = \frac{t_1}{2} \\
V_1 & \quad \frac{t_1}{2} < t < \frac{t_1 + t_2}{2} \\
\frac{V_1 + V_2}{2} & \quad t = \frac{t_1 + t_2}{2} \\
V_2 & \quad \frac{t_1 + t_2}{2} < t < \frac{t_2 + t_3}{2} \\
\frac{V_2 + V_3}{2} & \quad t = \frac{t_2 + t_3}{2} \\
& \quad \vdots \\
\frac{V_{n-1} + V_n}{2} & \quad t = \frac{t_n + t_{n-1}}{2} \\
V_n & \quad t > t_n
\end{align*}
\]

Using the Ceiling Value Lookup interpolation mode, the interpolated voltage will be determined as follows:

\[
\begin{align*}
V_0 & \quad t = 0 \\
V_1 & \quad 0 < t \leq t_1 \\
V_2 & \quad t_1 < t \leq t_2 \\
& \quad \vdots \\
V_n & \quad t_{n-1} < t \leq t_n
\end{align*}
\]
Using the Floor Value Lookup interpolation mode, the interpolated voltage will be determined as follows:

\[
\begin{align*}
\text{Floor Value Lookup:} & \\
0 \leq t < t_1 & : v_0 \\
t_1 \leq t < t_2 & : v_1 \\
t_2 \leq t < t_3 & : v_2 \\
t_{n-1} \leq t < t_n & : v_n
\end{align*}
\]

The Value interpolation mode is to be used when the mode is variable or unknown. The entered parameter for Value interpolation mode should be a string (or integer) from the following set:

- "linear" (0)
- "spline" (1)
- "cubic" (2)
- "index_lookup" (3)
- "value_lookup" (4)
- "ceil_value_lookup" (5)
- "floor_value_lookup" (6)

Refer to the DataAccessComponent documentation for additional information about the interpolation modes available for VtDataset.

6. Table 5-12 lists the dc operating point parameters that can be sent to the dataset.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

7. For general information regarding time domain sources, refer to the "Introduction" on page 5-1.
Time Domain Sources

**VtExp (Voltage Source, Exponential Decay)**

**Symbol**

![Symbol Image]

**Parameters**

- $V_{\text{low}} =$ initial voltage (default: 0V)
- $V_{\text{high}} =$ peak voltage (default: 1V)
- $\text{Delay}_1 =$ rise time delay (default: 0 nsec)
- $\text{Tau}_1 =$ rise time constant (default: 1 nsec)
- $\text{Delay}_2 =$ fall time delay (default: 1 nsec)
- $\text{Tau}_2 =$ fall time constant (default: 1 nsec)
- $\text{SaveCurrent} =$ save branch current: yes (default), no

**Range of Usage**

- $\text{Delay}_1 \geq 0$
- $\text{Delay}_2 \geq 0$
- $\text{Tau}_1 \geq 0$
- $\text{ Tau}_2 \geq 0$

**Notes/Equations**

1. In SPICE, the equivalent to this source is a voltage source with the exponential waveform argument EXP and its parameters. If $\text{Tau}_1$ or $\text{Tau}_2 = 0$, it is replaced by MaxTimeStep from the transient simulation or Step from the envelope simulation.

2. The source output voltage, $V$, is given by the following:

$$t_1 = \frac{t - \text{Delay}_1}{\text{Tau}_1} \quad t_2 = \frac{t - \text{Delay}_2}{\text{Tau}_2}$$

Case 1: $\text{Delay}_1 < \text{Delay}_2$

---

5-28 VtExp (Voltage Source, Exponential Decay)
Case 2: Delay2 < Delay1

\[
V = \begin{cases} 
V_{\text{low}} & 0 \leq t \leq \text{Delay}_1 \\
V_{\text{low}} + (V_{\text{high}} - V_{\text{low}}) \times (1 - \exp(-t_1)) & \text{Delay}_1 \leq t \leq \text{Delay}_2 \\
V_{\text{low}} + (V_{\text{high}} - V_{\text{low}}) \times (1 - \exp(-t_1)) + (V_{\text{low}} - V_{\text{high}}) \times (1 - \exp(-t_2)) & \text{Delay}_2 < t 
\end{cases}
\]

3. Table 5-13 lists the dc operating point parameters that can be sent to the dataset.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

4. For general information regarding time domain sources, refer to the “Introduction” on page 5-1.
Time Domain Sources

**VtImpulseDT (Voltage Source, Impulse Train Defined at Discrete Time Steps)**

Symbol

<table>
<thead>
<tr>
<th>Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Symbol Image]</td>
</tr>
</tbody>
</table>

**Parameters**

- **Vlow** = minimum voltage level (default: 0V)
- **Vhigh** = maximum voltage level (default: 1V)
- **Period** = time between repetitive impulses (default: 100 nsec)
- **Delay** = time delay before first impulse (default: 0 nsec)
- **Rout** = output resistance (default: 1 ohm)

**Notes/Equations**

1. This source is used in envelope and transient simulations.
2. Both the delay and the period are rounded to the nearest integer multiple of the analysis time step. The impulse source is in the high state for only one time sample each period, with an open circuit voltage equal to Vhigh and an output impedance set by Rout.
3. It is possible to set Vlow to a voltage more positive than Vhigh in order to generate a negative-going impulse train, as shown in Figure 5-5.
4. Table 5-14 lists the dc operating point parameters that can be sent to the dataset.

Table 5-14. DC Operating Point Information

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

5. For general information regarding time domain sources, refer to the “Introduction” on page 5-1.
Time Domain Sources

VtLFSR_DT (Voltage Source, Pseudo-Random Pulse Train Defined at Discrete Time Steps)

Symbol

Parameters

Vlow = minimum voltage level (default: 0V)
Vhigh = maximum voltage level (default: 1V)
Rate = bit rate (default: 24.3 kHz)
Delay = initial time delay to first transition (default: 0 nsec)
Taps = bits used to generate feedback (default: bin("10000000000000100")
Seed = initial value loaded into the shift register (default: bin("10101010101010101")
Rout = output resistance (default: 1 ohm)

Notes/Equations

1. This is a discrete-time source for use in envelope and transient simulations. The pulse width must be an integer number of simulation time steps.
2. This component can be used to generate PN sequences with user-defined recurrence relations.
3. The linear feedback shift register component can be used to generate PN sequences with user-defined recurrence relations. The input to the LFSR is a binary sequence. Figure 5-6 illustrates an LFSR model.

Data is shifted to the right in the shift register. The length of the shift register is r. The numbers a(1), a(2), ..., a(r) are the binary feedback coefficients specified by Taps.

The shift register length r is defined by the largest value in Taps. For example, a Taps of 7 3 2 1 results in a shift register length of 7; the maximum value allowed in Taps is 31, which results in a maximum shift register length of 31.
The initial contents of the shift are specified by the value of Seed. The maximum meaningful value for Seed is \((2^{*}r)\) for a specific Taps. The maximum Seed value allowed is \((2^{*}31)\).

The following equations describe the operation of LFSR.

\[
D(n) = \left[ \sum_{k=1}^{r} a(k)D(n-k) \right] \mod 2 \quad \text{for } n \geq 1
\]

where

\[
D(0) = Seed_2(0)
\]
\[
D(-1) = Seed_2(1)
\]
\[
\vdots
\]
\[
D(1-r) = Seed_2(r-1)
\]

and

\[
Seed = \sum_{k \geq 0} Seed_2(k)2^k
\]

where

\[
Seed_2(k) \in \{0,1\} \quad \text{for } 0 \leq k < r.
\]

Example: Let Seed=2, and r=7
Time Domain Sources

Then

\[
\begin{align*}
\text{Seed}_2 (0) &= 0 \\
\text{Seed}_2 (1) &= 1 \\
\text{Seed}_2 (2) &= 0 \\
\vdots \\
\text{Seed}_2 (6) &= 0
\end{align*}
\]

Therefore,

\[
\begin{align*}
D(0) &= \text{Seed}_2 (0) = 0 \\
D(-1) &= \text{Seed}_2 (1) = 1 \\
D(-2) &= \text{Seed}_2 (2) = 0 \\
\vdots \\
D(-6) &= \text{Seed}_2 (6) = 0
\end{align*}
\]

4. The binary feedback coefficients are specified by Taps, which is a list of feedback coefficients. The coefficients are specified by listing the locations where the feedback coefficients equal 1. For example, the recurrence relation

\[
D(n) = (D(n-7) + D(n-3) + D(n-2) + D(n-1)) \mod 2
\]

is specified by the list [7, 3, 2, 1].

Table 5-15 is an extensive list of feedback coefficients for linear feedback shift registers showing one or more alternate feedback connections for a given number of stages.

<table>
<thead>
<tr>
<th>Number of Stages</th>
<th>Code Length</th>
<th>Maximal Taps</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>3</td>
<td>[2, 1]</td>
</tr>
<tr>
<td>3</td>
<td>7</td>
<td>[3, 1]</td>
</tr>
<tr>
<td>4</td>
<td>15</td>
<td>[4, 1]</td>
</tr>
<tr>
<td>5</td>
<td>31</td>
<td>[5, 2] [5, 4, 3, 2] [5, 4, 2, 1]</td>
</tr>
<tr>
<td>6</td>
<td>63</td>
<td>[6, 1] [6, 5, 2, 1] [6, 5, 3, 2]</td>
</tr>
</tbody>
</table>

5-34  VtLFSR_DT (Voltage Source, Pseudo-Random Pulse Train Defined at Discrete Time Steps)
Table 5-15. Feedback Connections for Linear m-Sequences (continued)*

<table>
<thead>
<tr>
<th>Number of Stages</th>
<th>Code Length</th>
<th>Maximal Taps</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>127</td>
<td>[7, 4, 3, 2, 1, 7, 6, 4, 2, 7, 6, 3, 1, 7, 6, 5, 2, 7, 6, 5, 4, 2, 1, 7, 5, 4, 3, 2, 1]</td>
</tr>
<tr>
<td>8</td>
<td>255</td>
<td>[8, 6, 5, 3, 8, 6, 5, 2, 8, 5, 3, 1, 8, 6, 5, 1, 8, 7, 6, 1, 8, 7, 6, 5, 2, 8, 6, 4, 3, 2, 1]</td>
</tr>
<tr>
<td>9</td>
<td>511</td>
<td>[9, 4, 9, 8, 4, 3, 9, 8, 5, 4, 9, 8, 4, 1, 9, 5, 3, 2, 9, 8, 6, 5, 9, 8, 7, 2, 9, 6, 5, 4, 2, 9, 7, 6, 4, 3, 1, 9, 8, 7, 6, 5, 3]</td>
</tr>
<tr>
<td>10</td>
<td>1023</td>
<td>[10, 3, 10, 8, 3, 2, 10, 4, 3, 1, 10, 8, 5, 1, 10, 8, 5, 4, 10, 9, 4, 1, 10, 8, 4, 3, 10, 5, 3, 2, 10, 5, 2, 1, 10, 9, 4, 2]</td>
</tr>
<tr>
<td>11</td>
<td>2047</td>
<td>[11, 1, 11, 8, 5, 2, 11, 7, 3, 2, 11, 5, 3, 5, 11, 10, 3, 2, 11, 6, 5, 1, 11, 5, 3, 1, 11, 9, 4, 1, 11, 8, 6, 2, 11, 9, 8, 3]</td>
</tr>
<tr>
<td>12</td>
<td>4095</td>
<td>[12, 6, 4, 1, 12, 9, 3, 2, 12, 11, 10, 5, 2, 1, 12, 11, 6, 4, 2, 1, 12, 11, 9, 7, 6, 5, 12, 11, 9, 5, 3, 1, 12, 11, 9, 8, 7, 4, 12, 11, 9, 7, 6, 5]</td>
</tr>
<tr>
<td>13</td>
<td>8191</td>
<td>[13, 4, 3, 1, 13, 10, 9, 7, 5, 4, 13, 11, 8, 7, 4, 1, 13, 12, 8, 7, 6, 5, 13, 9, 8, 7, 5, 1, 13, 12, 6, 5, 4, 3, 13, 12, 11, 9, 5, 3, 13, 12, 11, 5, 2, 1, 13, 12, 9, 8, 4, 2, 13, 8, 7, 4, 3, 2]</td>
</tr>
<tr>
<td>14</td>
<td>16,383</td>
<td>[14, 12, 2, 1, 14, 13, 4, 2, 14, 13, 11, 9, 14, 10, 6, 1, 14, 11, 6, 1, 14, 12, 11, 1, 14, 16, 4, 2, 14, 11, 9, 6, 5, 2, 14, 13, 6, 5, 3, 1, 14, 13, 12, 8, 4, 1, 14, 8, 7, 6, 4, 2, 14, 10, 6, 5, 4, 1, 14, 13, 12, 7, 6, 3, 14, 13, 11, 10, 8, 3]</td>
</tr>
<tr>
<td>15</td>
<td>32,767</td>
<td>[15, 13, 10, 9, 15, 13, 10, 1, 15, 14, 9, 2, 15, 1, 15, 9, 4, 1, 15, 12, 3, 1, 15, 10, 5, 4, 15, 10, 5, 4, 3, 2, 15, 11, 7, 6, 2, 1, 15, 17, 6, 3, 2, 1, 15, 10, 9, 8, 5, 3, 15, 12, 5, 4, 3, 2, 15, 10, 9, 7, 5, 3, 15, 13, 12, 10, 15, 13, 10, 2, 15, 12, 9, 1, 15, 14, 12, 2, 15, 13, 9, 6, 15, 7, 4, 1, 15, 4, 15, 13, 17, 4]</td>
</tr>
</tbody>
</table>
Table 5-15. Feedback Connections for Linear $m$-Sequences (continued)*

<table>
<thead>
<tr>
<th>Number of Stages</th>
<th>Code Length</th>
<th>Maximal Taps</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>65,535</td>
<td>[16, 12, 3, 1] [16, 12, 9, 6] [16, 9, 4, 3] [16, 12, 7, 2] [16, 10, 7, 6] [16, 15, 7, 2] [16, 9, 5, 2] [16, 13, 9, 6] [16, 15, 4, 2] [16, 15, 9, 4]</td>
</tr>
<tr>
<td>17</td>
<td>131,071</td>
<td>[17, 3] [17, 3, 2] [17, 7, 4, 3] [17, 16, 3, 1] [17, 12, 6, 3, 2, 1] [17, 8, 7, 6, 4, 3] [17, 11, 8, 6, 4, 2] [17, 9, 8, 6, 4, 1] [17, 16, 14, 10, 3, 2] [17, 12, 11, 8, 5, 2]</td>
</tr>
<tr>
<td>18</td>
<td>262,143</td>
<td>[18, 7] [18, 10, 7, 5] [18, 13, 11, 9, 8, 7, 6, 3] [18, 17, 16, 15, 10, 9, 8, 7] [18, 15, 12, 11, 9, 8, 7, 6]</td>
</tr>
<tr>
<td>19</td>
<td>524,287</td>
<td>[19, 5, 2, 1] [19, 13, 8, 5, 4, 3] [19, 12, 10, 9, 7, 3] [19, 17, 15, 14, 13, 12, 6, 1] [19, 17, 15, 14, 13, 9, 8, 4, 2, 1] [19, 16, 13, 11, 19, 9, 4, 1] [19, 9, 8, 7, 6, 3] [19, 16, 15, 13, 12, 9, 5, 4, 2, 1] [19, 18, 15, 14, 11, 10, 8, 5, 3, 2] [19, 18, 17, 16, 12, 7, 6, 5, 3, 1]</td>
</tr>
<tr>
<td>20</td>
<td>1, 048,575</td>
<td>[20, 3] [20, 9, 5, 3] [20, 19, 4, 3] [20, 11, 8, 6, 3, 2] [20, 17, 14, 10, 7, 4, 3, 2]</td>
</tr>
<tr>
<td>21</td>
<td>2,097,151</td>
<td>[21, 2] [21, 14, 7, 2] [21, 13, 5, 2] [21, 8, 7, 4, 3, 2] [21, 10, 6, 4, 3, 2] [21, 15, 10, 9, 5, 4, 3, 2] [21, 14, 12, 7, 6, 4, 3, 2] [21, 20, 19, 18, 5, 4, 3, 2]</td>
</tr>
<tr>
<td>22</td>
<td>4,194,303</td>
<td>[22, 1] [22, 9, 5, 1] [22, 20, 18, 16, 6, 4, 2, 1] [22, 19, 16, 13, 10, 7, 4, 1] [22, 17, 9, 7, 2, 1] [22, 17, 13, 12, 8, 7, 2, 1] [22, 14, 13, 12, 7, 3, 2, 1]</td>
</tr>
<tr>
<td>23</td>
<td>8,388,607</td>
<td>[23, 5] [23, 17, 11, 5] [23, 5, 4, 1] [23, 12, 5, 4] [23, 21, 7, 5] [23, 16, 13, 6, 5, 3] [23, 11, 10, 7, 6, 5] [23, 15, 10, 9, 7, 5, 4, 3] [23, 17, 11, 9, 8, 5, 4, 1] [23, 18, 16, 13, 11, 8, 5, 2]</td>
</tr>
<tr>
<td>24</td>
<td>16,777,215</td>
<td>[24, 7, 2] [24, 4, 3, 1] [24, 22, 20, 18, 16, 14, 11, 9, 8, 7, 5, 4] [24, 21, 19, 18, 17, 16, 15, 14, 13, 10, 9, 5, 4, 1]</td>
</tr>
<tr>
<td>25</td>
<td>33,554, 431</td>
<td>[25, 3] [25, 3, 2, 1] [25, 20, 5, 3] [25, 12, 5, 4] [25, 17, 10, 3, 2, 1] [25, 23, 21, 19, 9, 7, 5, 3] [25, 18, 12, 11, 6, 5, 4] [25, 20, 16, 11, 5, 3, 2, 1] [25, 12, 11, 8, 7, 6, 4, 3]</td>
</tr>
<tr>
<td>26</td>
<td>67,108,863</td>
<td>[26, 6, 2, 1] [26, 22, 21, 16, 12, 11, 10, 8, 5, 4, 3, 1]</td>
</tr>
<tr>
<td>27</td>
<td>134,217,727</td>
<td>[27, 5, 2, 1] [27, 18, 11, 10, 9, 5, 4, 3]</td>
</tr>
</tbody>
</table>
Table 5-15. Feedback Connections for Linear m-Sequences (continued)*

<table>
<thead>
<tr>
<th>Number of Stages</th>
<th>Code Length</th>
<th>Maximal Taps</th>
</tr>
</thead>
<tbody>
<tr>
<td>28</td>
<td>268,435,455</td>
<td>[28, 3, 28, 13, 11, 9, 5, 3] [28, 22, 11, 10, 4, 3] [28, 24, 20, 16, 12, 8, 4, 3, 2, 1]</td>
</tr>
<tr>
<td>29</td>
<td>536,870,911</td>
<td>[29, 2, 29, 20, 11, 2] [29, 13, 7, 2] [29, 21, 5, 2] [29, 26, 5, 2] [29, 19, 16, 6, 3, 2] [29, 18, 14, 6, 3, 2]</td>
</tr>
<tr>
<td>30</td>
<td>1,073,741,823</td>
<td>[30, 23, 2, 1] [30, 6, 4, 1] [30, 24, 20, 16, 14, 13, 11, 7, 2, 1]</td>
</tr>
<tr>
<td>31</td>
<td>2,147,483,646</td>
<td>[31, 29, 21, 17] [31, 28, 19, 15] [31, 3] [31, 3, 2, 1] [31, 13, 8, 3] [31, 21, 12, 3, 2, 1] [31, 20, 18, 7, 5, 3] [31, 30, 29, 25] [31, 28, 24, 10]</td>
</tr>
<tr>
<td>32</td>
<td>4,294,967,295</td>
<td>[32, 22, 2, 1] [32, 7, 5, 3, 2, 1] [32, 28, 19, 18, 16, 14, 11, 10, 9, 6, 5, 1]</td>
</tr>
<tr>
<td>33</td>
<td>8,589,934,591</td>
<td>[33, 13] [33, 22, 13, 11] [33, 26, 14, 10] [33, 6, 4, 1] [33, 22, 16, 13, 11, 8]</td>
</tr>
<tr>
<td>61</td>
<td>2,305,843,009,213, 693, 951</td>
<td>[61, 5, 2, 1]</td>
</tr>
<tr>
<td>89</td>
<td>618,970,019,642,690,137,449,562,112</td>
<td>[89, 6, 5, 3]</td>
</tr>
</tbody>
</table>


5. Table 5-16 lists the dc operating point parameters that can be sent to the dataset.

Table 5-16. DC Operating Point Information

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

6. For general information regarding time domain sources, refer to the “Introduction” on page 5-1.
Time Domain Sources

VtOneShot (Voltage Source, Retriggerable Pulse Train)

Symbol

Parameters
Delay = time delay from trigger to pulse start (default: timestep)

Width = pulse width (default: 5*timestep)

Vhigh = pulse voltage (default: 5V)

Notes/Equations

1. This source is implemented in FDD for use with transient and envelope simulations. The retriggerable one-shot is a predefined application of the retriggerable source VtRetrig. It outputs a pulse of amplitude Vhigh and specified width and delay after every trigger event. Due to the trigger delay of 1 to 2 time steps, the actual pulse width will be shorter than specified by this same amount, if the one-shot delay is specified to be less than 1 time step.

2. The trigger input is an infinite impedance, differential input. A trigger event occurs whenever the baseband voltage difference across the two inputs passes through 0.5V with a positive slope. The output impedance is fixed at 50 ohms.

3. Table 5-17 lists the dc operating point parameters that can be sent to the dataset.

Table 5-17. DC Operating Point Information

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

4. For general information regarding time domain sources, refer to the “Introduction” on page 5-1.
VtPulse (Voltage Source, Pulse with Linear, Cosine, or Error Function Edge Shape)

Symbol

Parameters

- Vlow = initial voltage (default: 0V)
- Vhigh = pulse voltage (default: 1V)
- Delay = delay time (default: 0 nsec)
- Edge = rising and falling edge type (default: linear)
- Rise = rise time (default: 1 nsec)
- Fall = fall time (default: 1 nsec)
- Width = pulse width (default: 3 nsec)
- Period = pulse period (default: 10 nsec)
- SaveCurrent = save branch current: yes (default), no

Range of Usage

- Delay ≥ 0; Rise ≥ 0; Fall ≥ 0
- Width > 0
- Width + Rise + Fall ≤ Period

Notes/Equations

1. This item is a time-periodic rectangular pulse-train voltage source for use with transient and envelope simulation. It is treated as a short circuit in all other simulations.

2. If Rise or Fall = 0, it is replaced by MaxTimeStep from the transient simulation or Step from the envelope simulation.
Time Domain Sources

3. If Edge=linear, the rising and falling edge is a linear ramp. In SPICE, the equivalent to this source is a current or voltage source with the pulse waveform argument PULSE and its parameters.

<table>
<thead>
<tr>
<th>Time</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>low</td>
</tr>
<tr>
<td>Delay</td>
<td>low</td>
</tr>
<tr>
<td>Delay + Rise</td>
<td>high</td>
</tr>
<tr>
<td>Delay + Rise + Width</td>
<td>high</td>
</tr>
<tr>
<td>Delay + Rise + Width + Fall</td>
<td>low</td>
</tr>
<tr>
<td>Period</td>
<td>low</td>
</tr>
</tbody>
</table>

If Edge=erf, instead of the rise and fall portions being linear ramps, this source generates a pulse based on the error function, giving a different shape to the rising and falling edges. By not having abrupt changes in slope, the pulse shape is more realistic and its frequency spectrum decreases more rapidly.

For the error function pulse, the rise and fall time define the total transition period and the maximum slope is greater than \((I_{\text{High}} - I_{\text{Low}})/\text{Rise}\). (See Figure 5-7.)

![Figure 5-7. ItPulse Waveforms with Different Edges](image)

This source uses \(1-\text{erfc}(x), (-2 < x < 2)\) to generate the transition region and has a peak slope that is approximately 2.25 times the linear rise time. Due to the faster slope, the 3db bandwidth of the output pulse is larger for a given rise time.
The shape of the waveform is shown in Figure 5-8; the intermediate points during rise and fall time are determined by interpolation.

![Figure 5-8. Waveform shape](image)

If Edge=cosine, this source generates cosine-shaped rising and falling edges. By not having abrupt changes in slope, the pulse shape is more realistic and its frequency spectrum decreases more rapidly.

For the cosine pulse, the rise and fall time define the total transition period and the maximum slope is greater than \((I_{\text{High}}-I_{\text{Low}})/\text{Rise}\). (See Figure 5-7.)

4. Table 5-18 lists the dc operating point parameters that can be sent to the dataset.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

5. For general information regarding time domain sources, refer to the “Introduction” on page 5-1.
Time Domain Sources

VtPulseDT (Voltage Source, Pulse Train Defined at Discrete Time Steps)

Symbol

Parameters

Vlow = initial voltage (default: 0V)
Vhigh = pulse voltage (default: 1V)
Delay = time delay (default: 0 nsec)
Width = pulse width (default: 3 nsec)
Period = pulse period (default: 10 nsec)
Rout = output resistance (default: 1 ohm)

Notes/Equations

1. This source is used in envelope and transient simulations.
2. Period, Width, and Delay are rounded to the nearest integer multiple of the analysis time step. The pulse width must be an integer number of simulation time steps. The pulse source is in the high state for a time interval equal to Width, during which it has an open circuit output voltage equal to Vhigh. The output impedance is set by Rout.
3. As with the impulse source, Vlow can be set to a voltage more positive than Vhigh in order to generate a negative-going pulse train.
4. The use of a discrete time pulse source, as opposed to a standard pulse source, guarantees that there is no timing jitter in the pulse edges due to the waveform being sampled asynchronously by a fixed time interval simulation. By setting the period, width or delay equal to multiples of the time step variable, the source can be set up to track the analysis time step control, if desired.
5. Table 5-19 lists the dc operating point parameters that can be sent to the dataset.
6. For general information regarding time domain sources, refer to the “Introduction” on page 5-1.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>
Time Domain Sources

VtPWL (Voltage Source, Piecewise Linear)

Symbol

Parameters

\[ V_{\text{Tran}} = \text{time-voltage pairs: } \text{pwl}(\text{time, time-voltage pairs}) \text{ or } \text{pwlr}(\text{time, Ncycles, time-voltage pairs}) \]

\[ \text{SaveCurrent} = \text{save branch current: yes (default), no} \]

Notes/Equations

1. The piecewise linear voltage versus time data are specified with a \text{pwl}() function. The syntax for \text{pwl} is \text{pwl}(\text{time, } T_i, V_i, ...) . Each pair of values \((T_i, V_i)\) specifies that at \text{time}=T_i, the voltage is \text{V}_i . The value of the source at intermediate values of time is determined by using linear interpolation on the input values.

2. In SPICE, the equivalent to this source is a voltage source with the piecewise linear waveform argument PWL and its parameters.

3. If the piecewise linear waveform is to be repeated for several cycles, a \text{pwlr}() function can be used. The syntax for \text{pwlr}() is \text{pwlr}(\text{time, Ncycles, } T_i, V_i, ...) where \text{Ncycles} is the number of cycles to be repeated.

4. Table 5-20 lists the dc operating point parameters that can be sent to the dataset.

Table 5-20. DC Operating Point Information

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>\text{Is}</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>\text{Power}</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>\text{Vs}</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

5. For general information regarding time domain sources, refer to the “Introduction” on page 5-1.
VtRetrig (Voltage Source, Retriggerable, User-Defined Waveform)

Symbol

Parameters

V = user-defined waveform equation (default: 1us-(time_tt(1)))
Rout = output resistance (default: 1 ohm)
Thresh = trigger threshold on rising edge (default: 0.5V)

Notes/Equations

1. This source is used in envelope and transient simulations.
2. This retriggerable source allows you to use an equation to describe a baseband waveform segment; the waveform segment is then output every time an input trigger occurs. The waveform is typically described in terms of the time since the last trigger event, which is (time - _tt(1)). For example, the default equation simply generates a value equal to 1 until 1 msec after the trigger event.

Figure 5-9 shows a more complicated example, where a function wav( ) is defined to create a truncated sinc( ) waveform, which is then generated following every trigger event. This wav( ) example function also uses the FDD function _tn( ) (the trigger count number) to linearly increase the sinc( ) waveform bandwidth with each new trigger. Any of the time-domain equation capabilities of the simulator can be used to define this waveform, including reading data from a dataset or using a random time variable. The output voltage of the source, prior to any triggers, is 0.0. The output impedance of the source is set by Rout.

3. The trigger input is an infinite impedance, differential input. The trigger event is determined whenever the baseband voltage difference across the two inputs passes through the trigger threshold voltage with a positive slope. Due to the delay in the trigger detection, the minimum value of (time - _tt(1)) will be between 1 and 2 time steps. There is fixed delay of one time step in addition to
Time Domain Sources

the time between the interpolated trigger event and the next simulated time point.

4. Because the trigger input and the output voltage are baseband only signals, this model works equally well in either transient or circuit envelope simulations.

![Figure 5-9. Truncated sinc( ) Waveform](image)

5. Table 5-21 lists the dc operating point parameters that can be sent to the dataset.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

6. For general information regarding time domain sources, refer to the “Introduction” on page 5-1.
VtSFFM (Voltage Source, Single Frequency FM, SFFM Wave)

Symbol

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vdc</td>
<td>Initial voltage offset (default: 0V)</td>
</tr>
<tr>
<td>Amplitude</td>
<td>Amplitude of signal (default: 1V)</td>
</tr>
<tr>
<td>CarrierFreq</td>
<td>Carrier frequency (default: 1 GHz)</td>
</tr>
<tr>
<td>ModIndex</td>
<td>Modulation index (default: 0.5)</td>
</tr>
<tr>
<td>SignalFreq</td>
<td>Signal frequency (default: 1 MHz)</td>
</tr>
<tr>
<td>SaveCurrent</td>
<td>Save branch current: yes (default), no</td>
</tr>
</tbody>
</table>

Notes/Equations

1. In SPICE, the equivalent to this source is a voltage source with the single-frequency FM source waveform argument SFFM and its parameters.

2. The shape of the waveform is described in the following equation.

\[ V_{out} = V_{dc} + \text{Amplitude} \times \sin(2\pi \times \text{CarrierFreq} \times \text{time} + \text{ModIndex} \times \sin(2\pi \times \text{SignalFreq} \times \text{time})) \]

3. Table 5-22 lists the dc operating point parameters that can be sent to the dataset.

Table 5-22. DC Operating Point Information

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

4. For general information regarding time domain sources, refer to the “Introduction” on page 5-1.
Time Domain Sources

**VtSine (Voltage Source, Decaying Sine Wave)**

**Symbol**

![Symbol](image)

**Parameters**

- **Vdc** = initial voltage offset (default: 0V)
- **Amplitude** = amplitude of sinusoidal wave (default: 1V)
- **Freq** = frequency of sinusoidal wave (default: 1 GHz)
- **Delay** = time delay (default: 0 nsec)
- **Damping** = damping factor, 1/sec (default: 0 1/sec)
- **Phase** = initial phase value (default: 0)
- **SaveCurrent** = save branch current: yes, no (default: no)

**Range of Usage**

- **Freq** > 0
- **Delay** ≥ 0

**Notes/Equations**

1. In SPICE, the equivalent to this source is a voltage source with the sinusoidal waveform argument sin and its parameters. VtSine defines an ac sinusoidal voltage source, at a specified frequency and phase, including its turn-on characteristics for use with transient analysis.
2. VtSine has a value of \([Vdc + Amplitude \times \sin(\text{phase})]\) from \(t=0\) until \(t=\text{delay}\). Voltage then becomes an exponentially damped sine wave described by

\[
V = Vdc + Amplitude \times \sin\left(2\pi \left(\frac{\text{Freq}(t - \text{Delay}) + \frac{\text{Phase}}{360}}{\text{Delay}}\right)\right) \\
\times e^{-t - \text{Delay} \times \text{Damping}}
\]

where \(t\) is time.

3. This source can also be used in Harmonic Balance and Circuit Envelope simulations.

4. **Table 5-23** lists the dc operating point parameters that can be sent to the dataset.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

5. For general information regarding time domain sources, refer to the “Introduction” on page 5-1.
Time Domain Sources

VtStep (Voltage Source, Step)

Symbol

Parameters
Vlow = initial voltage (default: 0V)
Vhigh = pulse voltage (default: 1V)
Delay = delay time (default: 0 nsec)
Rise = rise time (default: 1 nsec)
SaveCurrent = save branch current: yes (default), no

Notes/Equations
1. In SPICE, the equivalent to this source is a voltage source with the step waveform argument STEP and its parameters.
2. Table 5-24 lists the dc operating point parameters that can be sent to the dataset.

Table 5-24. DC Operating Point Information

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>
VtUserDef (Voltage Source, User-Defined)

Symbol

Parameters

V_Tran = transient voltage (default: damped_sin(time) )
Vdc = dc voltage
Vac = ac voltage (default: 1V)
SaveCurrent = save branch current: yes (default), no

Notes/Equations

1. Typically, V_Tran is assigned an equation. This equation can be defined as a function of time by using the program reserved variable time in it. As the value of time is swept in transient or envelope simulation, the amplitude of the voltage source will take on the value of the equation.

2. A variable or equation is unitless. However, the value of V_Tran as given by the result of a variable or equation will be assumed to be in volts. The value of time will be the current simulation time in seconds.

3. There are several built-in functions that implement the standard SPICE sources, such as pwl and pulse. For a transient analysis, the VtUserDef source voltage is the sum of the value specified in the Vdc and V_Tran parameters.

   Example:
   
   \[
   \text{vt} = \text{pwl (time, 0ns, 0, 1ns, 1, 2ns, -2) X damped_sin (time)}
   \]

4. The Vac parameter is used in AC simulations and does not affect transient simulation. An example for specifying magnitude and phase would be Vac=polar(2,45), where 2 is the magnitude and 45 is the phase. For more parameter options (such as frequency) on an AC source, use the V_AC component on the Sources-Freq Domain palette.
Time Domain Sources

5. Table 5-25 lists the dc operating point parameters that can be sent to the dataset.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Current</td>
<td>A</td>
</tr>
<tr>
<td>Power</td>
<td>DC power dissipated</td>
<td>W</td>
</tr>
<tr>
<td>Vs</td>
<td>Voltage</td>
<td>V</td>
</tr>
</tbody>
</table>

6. For general information regarding time domain sources, refer to the “Introduction” on page 5-1.
Index

C
CCCS, 1-2
CCCS_Z, 1-4
CCVS, 1-6
CCVS_Z, 1-8
ClockWjitter, 5-2

I
I_1Tone, 2-4
I_AC, 2-2
I_DC, 2-3, 5-4
I_HB_Dataset, 2-13
I_nHarm, 2-7
I_Noise, 4-2
I_NoiseBD, 4-3
I_nTone, 2-10
I_SpectrumDataset, 2-14
ItDataset, 5-5
ItExp, 5-9
ItPulse, 5-11
ItPWL, 5-14
ItSFFM, 5-15
ItSine, 5-16
ItStep, 5-18
ItUserDef, 5-19

N
NoiseCorr, 4-5
Noisy2Port, 4-6

O
OSCwPhNoise, 2-16

P
P_1Tone, 2-20
P_AC, 2-18
P_nHarm, 2-24
P_nTone, 2-28
P_SpectrumDataset, 2-31
PtRF_3GPP_Uplink, 3-2
PtRF_CDMA_ESG_FWD, 3-4
PtRF_CDMA_ESG_REV, 3-6
PtRF_CDMA_IS95_FWD, 3-8
PtRF_CDMA_IS95_REV, 3-10
PtRF_CDMA2K_REV, 3-12
PtRF_DECT, 3-14
PtRF_EDGE_Uplink, 3-15
PtRF_GSM, 3-17
PtRF_NADC, 3-18
PtRF_PHS, 3-20
PtRF_Pulse, 3-21
PtRF_Step, 3-23
V
V_1Tone, 2-33
V_AC, 2-36
V_DC, 2-37, 5-21
V_HB_Dataset, 2-57
V_nHarm, 2-38
V_Noise, 4-7
V_NoiseBD, 4-9
V_nTone, 2-41
V_SpectrumDataset, 2-44
VCCS, 1-10
VCCS_Z, 1-12
VCVS, 1-14
VCVS_Z, 1-16
Vf_BitSeq, 2-46
Vf_Pulse, 2-49
Vf_Sawtooth, 2-51
Vf_Square, 2-53
Vf_Triangle, 2-55
VfBitSeq, 5-22
VfDataset, 5-24
VfExp, 5-28
VfImpulseDT, 5-30
VfLFSR_DT, 5-32
VfOneShot, 5-38
VfPulse, 5-39
VfPulseDT, 5-42
VfPWL, 5-44
VfRetrig, 5-45
VfRF_Pulse, 3-24
VfRF_Step, 3-26
VfSFFM, 5-47
VfSine, 5-48
VfStep, 5-50
VfUserDef, 5-51