Simscape™ Electronics™
Reference

MATLAB®&SIMULINK®

MathWorks®

How to Contact MathWorks

Latest news: www.mathworks.com

Sales and services: www.mathworks.com/sales_and_services

User community: www.mathworks.com/matlabcentral

Technical support: www.mathworks.com/support/contact_us

Phone: 508-647-7000

The MathWorks, Inc.
3 Apple Hill Drive
Natick, MA 01760-2098

Simscape ™ Electronics ™ Reference

Contents

Blocks — Alphabetical List

Accelerometer

Behavioral model of MEMS accelerometer
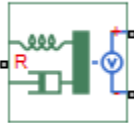
Library

Sensors

Description

The Accelerometer block implements a behavioral model of a MicroElectroMechanical Systems (MEMS) accelerometer. For the default output type `Voltage level`, the accelerometer provides an output voltage that is proportional to the acceleration rate presented at the mechanical translational physical port R. The output voltage is limited according to the values that you provide for maximum and minimum output voltage.

The block also has an alternative output type, `PWM duty cycle`. With this choice, the output of the block is a PWM signal with a duty cycle that is proportional to the measured acceleration. You can limit the variation in duty cycle to a specified range.

Optionally, you can model sensor dynamics by setting the **Dynamics** parameter to `Model sensor bandwidth`. Including dynamics adds a first-order lag between the angular rate presented at port R and the corresponding voltage applied to the electrical + and - ports.

If running your simulation with a fixed-step solver, or generating code for hardware-in-the-loop testing, MathWorks recommends that you set the **Dynamics** parameter to `No dynamics — Suitable for HIL`, because this avoids the need for a small simulation time step if the sensor bandwidth is high.

Variables

Use the **Variables** section of the block interface to set the priority and initial target values for the block variables prior to simulation. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

The **Measured acceleration** variable target specifies the initial output for the sensor.

Parameters

Output type

Select one of the following options to define the block output type:

- `Voltage level` — The amplitude of the output voltage is proportional to the measured acceleration. This is the default option.

- `PWM duty cycle` — The duty cycle (on time divided by the pulse total time) is proportional to the measured acceleration.

Sensitivity

The change in output voltage level per unit change in acceleration when the output is not being limited. This parameter is only visible when you select `Voltage level` for the **Output type** parameter. The default value is `1000` mV/gee.

Output voltage for zero acceleration

The output voltage from the sensor when the acceleration is zero. This parameter is only visible when you select `Voltage level` for the **Output type** parameter. The default value is `2.5` V.

Maximum output voltage

The maximum output voltage from the sensor, which determines the sensor maximum measured positive acceleration. This parameter is only visible when you select `Voltage level` for the **Output type** parameter. The default value is `4` V.

Minimum output voltage

The minimum output voltage from the sensor, which determines the sensor maximum measured negative acceleration. This parameter is only visible when you select `Voltage level` for the **Output type** parameter. The default value is `1` V.

Duty cycle sensitivity (percent per unit acceleration)

The change in duty cycle per unit acceleration. Duty cycle is expressed as a percentage of the PWM period. This parameter is only visible when you select `PWM duty cycle` for the **Output type** parameter. The default value is `10` percent/gee.

Duty cycle for zero acceleration (percent)

The duty cycle output by the sensor when the acceleration is zero. This parameter is only visible when you select `PWM duty cycle` for the **Output type** parameter. The default value is `50`%.

Maximum duty cycle (percent)

The maximum duty cycle output by the sensor. Increasing acceleration levels beyond this point will not register an increase in duty cycle. This parameter is only visible when you select `PWM duty cycle` for the **Output type** parameter. The default value is `75`%.

Minimum duty cycle (percent)

The minimum duty cycle output by the sensor. Decreasing acceleration levels beyond this point will not register a decrease in duty cycle. This parameter is only visible when you select `PWM duty cycle` for the **Output type** parameter. The default value is `25`%.

PWM frequency

The frequency of the output pulse train. This parameter is only visible when you select `PWM duty cycle` for the **Output type** parameter. The default value is `1` kHz.

Output voltage amplitude

The amplitude of the output pulse train when high. This parameter is only visible when you select `PWM duty cycle` for the **Output type** parameter. The default value is `5` V.

Dynamics

Select one of the following options for modeling sensor dynamics:

- `No dynamics — Suitable for HIL` — Do not model sensor dynamics. Use this option when running your simulation fixed step or generating code for hardware-in-the-loop testing, because this avoids the need for a small simulation time step if the sensor bandwidth is high. This is the default option.

- `Model sensor bandwidth` — Model sensor dynamics with a first-order lag approximation, based on the **Bandwidth** parameter value. You can control the initial condition for the lag by specifying the **Measured acceleration** variable target.

Bandwidth

Specifies the 3dB bandwidth for the measured acceleration assuming a first-order time constant. This parameter is only visible when you select `Model sensor bandwidth` for the **Dynamics** parameter. The default value is 3 kHz.

Ports

The block has the following ports:

R

Mechanical translational port

+

Positive electrical port

-

Negative electrical port

Band-Limited Op-Amp

Model band-limited operational amplifier

Library

Integrated Circuits

Description

The Band-Limited Op-Amp block models a band-limited operational amplifier. If the voltages at the positive and negative ports are *Vp* and *Vm*, respectively, the output voltage is:

$$V_{out} = \frac{A\left(V_p - V_m\right)}{\frac{s}{2\pi f} + 1} - I_{out} * R_{out}$$

where:

- *A* is the gain.
- R_{out} is the output resistance.
- I_{out} is the output current.
- *s* is the Laplace operator.
- *f* is the 3-dB bandwidth.

The input current is:

$$\frac{V_p - V_m}{R_{in}}$$

where R_{in} is the input resistance.

The block does not use the initial condition you specify using the **Initial output voltage, V0** parameter if you select the **Start simulation from steady state** check box in the Simscape Solver Configuration block.

Parameters

Gain, A

> The open-loop gain of the operational amplifier. The default value is `1000`.

Input resistance, Rin

> The resistance at the input of the operational amplifier that the block uses to calculate the input current. The default value is `1e+06` Ω.

Output resistance, Rout

> The resistance at the output of the operational amplifier that the block uses to calculate the drop in output voltage due to the output current. The default value is `100` Ω.

Minimum output, Vmin

> The lower limit on the operational amplifier no-load output voltage. The default value is `-15` V.

Maximum output, Vmax

> The upper limit on the operational amplifier no-load output voltage. The default value is `15` V.

Maximum slew rate, Vdot

> The maximum positive or negative rate of change of output voltage magnitude. The default value is `1000` V/s.

Bandwidth, f

> The open-loop bandwidth, that is, the frequency at which the gain drops by 3 dB compared to the low-frequency gain, *A*. The default value is `1e+05` Hz.

Initial output voltage, V0

> The output voltage at the start of the simulation when the output current is zero. The default value is `0` V.

Note This parameter value does not account for the voltage drop across the output resistor.

Ports

The block has the following ports:

+

    Positive electrical voltage

-

    Negative electrical voltage

OUT

    Output voltage

See Also

Finite-Gain Op-Amp | Fully Differential Op-Amp | Op-Amp

Battery

Simple battery model
Library: Simscape / Electronics / Sources

Description

The Battery block represents a simple battery model. The block has four modeling variants, accessible by right-clicking the block in your block diagram and then selecting the appropriate option from the context menu, under **Simscape** > **Block choices**:

- **Uninstrumented | No thermal port** — Basic model that does not output battery charge level or simulate thermal effects. This modeling variant is the default.

- **Uninstrumented | Show thermal port** — Model with exposed thermal port. This model does not measure internal charge level of the battery.

- **Instrumented | No thermal port** — Model with exposed charge output port. This model does not simulate thermal effects.

- **Instrumented | Show thermal port** — Model that lets you measure internal charge level of the battery and simulate thermal effects. Both the thermal port and the charge output port are exposed.

The instrumented variants have an extra physical signal port that outputs the internal state of charge. Use this functionality to change load behavior as a function of state of charge, without the complexity of building a charge state estimator.

The thermal port variants expose a thermal port, which represents the battery thermal mass. When you select this option, provide additional parameters to define battery behavior at a second temperature. For more information, see "Modeling Thermal Effects" on page 1-12.

The battery equivalent circuit is made up of the fundamental battery model, the self-discharge resistance R_{SD}, the charge dynamics model, and the series resistance R_0.
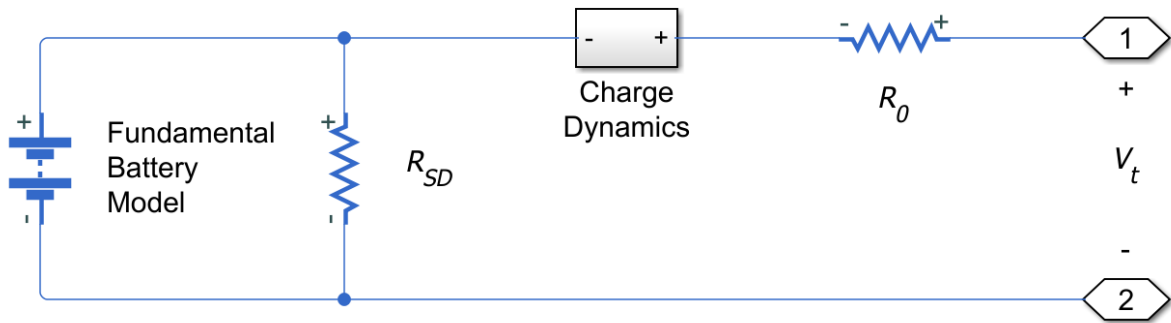
Battery Model

If you select `Infinite` for the **Battery charge capacity** parameter, the block models the battery as a series resistor and a constant voltage source. If you select `Finite` for the **Battery charge capacity** parameter, the block models the battery as a series resistor and a charge-dependent voltage source. In the finite case, the voltage is a function of charge and has the following relationship:

$$V = V_0 \left(\frac{\text{SOC}}{1 - \beta(1 - \text{SOC})} \right)$$

where:

- `SOC` (state-of-charge) is the ratio of current charge to rated battery capacity.
- V_0 is the voltage when the battery is fully charged at no load, as defined by the **Nominal voltage** parameter.
- β is a constant that is calculated so that the battery voltage is *V1* when the charge is *AH1*. Specify the voltage *V1* and ampere-hour rating *AH1* using the **Voltage V1 < Vnom when charge is AH1** and **Charge AH1 when no-load volts are V1** parameters, respectively.

The equation defines an approximate relationship between voltage and remaining charge. This approximation replicates the increasing rate of voltage drop at low charge values, and ensures that the battery voltage becomes zero when the charge level is zero. The advantage of this model is that it requires few parameters, which are readily available on most datasheets.

Modeling Battery Fade

For battery models with finite battery charge capacity, you can model battery performance deterioration depending on the number of discharge cycles. This deterioration is referred to as battery fade. To enable battery fade, set the **Battery fade** parameter to `Enabled`. This setting exposes additional parameters in the **Fade** section.

The block implements battery fade by scaling certain battery parameter values that you specify in the **Main** section, depending on the number of completed discharge cycles. The block uses multipliers λ_{AH}, λ_{R0}, and λ_{V1} on the **Ampere-hour rating**, **Internal resistance**, and **Voltage V1 < Vnom when charge is AH1** parameter values, respectively. These multipliers, in turn, depend on the number of discharge cycles:

$$\lambda_{AH} = 1 - k_1 N^{0.5}$$

$$\lambda_{R0} = 1 + k_2 N^{0.5}$$

$$\lambda_{V1} = 1 - k_3 N$$

$$N = N_0 + \frac{1}{AH} \int_0^t \frac{i(t) \cdot H\left(i(t)\right)}{\lambda_{AH}(t)} dt$$

where:

- λ_{AH} is the multiplier for battery nominal capacity.
- λ_{R0} is the multiplier for battery series resistance.
- λ_{V1} is the multiplier for voltage $V1$.
- N is the number of discharge cycles completed.
- N_0 is the number of full discharge cycles completed before the start of the simulation.
- AH is the rated battery capacity in ampere-hours.
- $i(t)$ is the instantaneous battery output current.
- $H(i(t))$ is the Heaviside function of the instantaneous battery output current. This function returns 0 if the argument is negative, and 1 if the argument is positive.

The block calculates the coefficients k_1, k_2, and k_3 by substituting the parameter values you provide in the **Fade** section into these battery equations. For example, the default set of block parameters corresponds to the following coefficient values:

- k_1 = 1e-2
- k_2 = 1e-3
- k_3 = 1e-3

You can also define a starting point for a simulation based on the previous charge-discharge history by using the high-priority variable **Discharge cycles**. For more information, see "Variables" on page 1-14.

Modeling Thermal Effects

For thermal variants of the block, you provide additional parameters to define battery behavior at a second temperature. The extended equations for the voltage when the thermal port is exposed are:

$$V = V_{0T} \left(\frac{\text{SOC}}{1 - \beta(1 - \text{SOC})} \right)$$

$$V_{0T} = V_0 \left(1 + \lambda_V \left(T - T_1 \right) \right)$$

where:

- T is the battery temperature.
- T_1 is the nominal measurement temperature.
- λ_V is the parameter temperature dependence coefficient for V_0.
- β is calculated in the same way as "Battery Model" on page 1-10, using the temperature-modified nominal voltage V_{0T}.

The internal series resistance, self-discharge resistance, and any charge-dynamic resistances are also functions of temperature:

$$R_T = R \left(1 + \lambda_R \left(T - T_1 \right) \right)$$

where λ_R is the parameter temperature dependence coefficient.

All the temperature dependence coefficients are determined from the corresponding values you provide at the nominal and second measurement temperatures. If you include charge dynamics in the model, the time constants vary with temperature in the same way.

The battery temperature is determined from a summation of all the ohmic losses included in the model:

$$M_{th}\dot{T} = \sum_i V_{T,i}^{\ 2} / R_{T,i}$$

where:

- M_{th} is the battery thermal mass.
- i corresponds to the ith ohmic loss contributor. Depending on how you have configured the block, the losses include:
  - Series resistance
  - Self-discharge resistance
  - First charge dynamics segment
  - Second charge dynamics segment
- $V_{T,i}$ is the voltage drop across resistor i.
- $R_{T,i}$ is resistor i.

Modeling Charge Dynamics

You can model battery charge dynamics using the **Charge dynamics** parameter:

- `No dynamics` — The equivalent circuit contains no parallel RC sections. There is no delay between terminal voltage and internal charging voltage of the battery.
- `Single time-constant dynamics` — The equivalent circuit contains one parallel RC section. Specify the time constant using the `First time constant` parameter.
- `Double time-constant dynamics` — The equivalent circuit contains two parallel RC sections. Specify the time constants using the `First time constant` and `Second time constant` parameters.

This figure shows the equivalent circuit for the block configured with double time-constant dynamics.
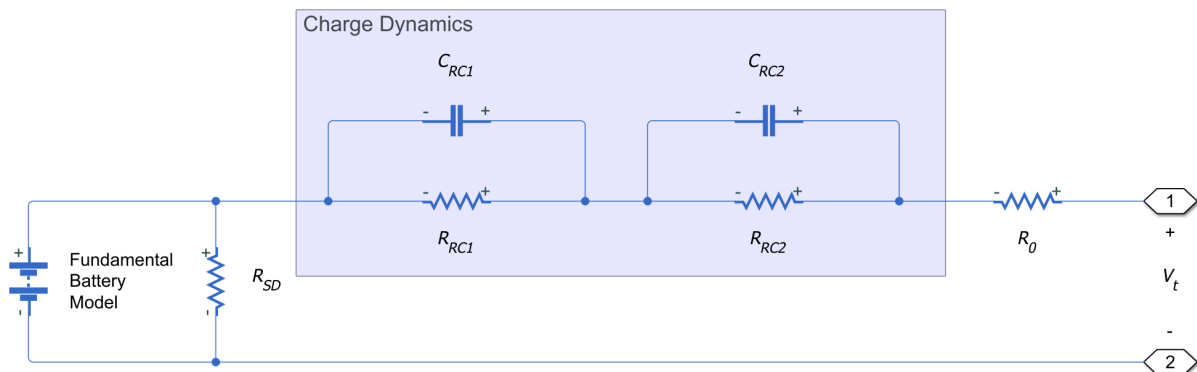
In the diagram:

- R_{RC1} and R_{RC2} are the parallel RC resistances. Specify these values with the **First polarization resistance** and **Second polarization resistance** parameters, respectively.

- C_{RC1} and C_{RC2} are the parallel RC capacitances. The time constant τ for each parallel section relates the R and C values using the relationship $C = \tau / R$. Specify τ for each section using the **First time constant** and **Second time constant** parameters, respectively.

- R_0 is the series resistance. Specify this value with the **Internal resistance** parameter.

Variables

Use the **Variables** section of the block interface to set the priority and initial target values for the block variables prior to simulation. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

Unlike block parameters, variables do not have conditional visibility. The **Variables** section lists all the existing block variables. If a variable is not used in the set of equations corresponding to the selected block configuration, the values specified for this variable are ignored.

When you model battery fade, the **Discharge cycles** variable lets you specify the number of charge-discharge cycles completed prior to the start of simulation. If you disable battery fade modeling, this variable is not used by the block.

Basic Assumptions and Limitations

- The self-discharge resistance is assumed not to depend strongly on the number of discharge cycles.

- For the thermal variant of the battery, you provide fade data only for the reference temperature operation. The block applies the same derived λ_{AH}, λ_{R0}, and λ_{V1} multipliers to parameter values corresponding to the second temperature.

- When using the thermal block variants, use caution when operating at temperatures outside of the temperature range bounded by the **Measurement temperature** and **Second measurement temperature** values. The block uses linear interpolation for the derived equation coefficients, and simulation results can become nonphysical outside of the specified range. The block checks that the internal series resistance, self-discharge resistance, and nominal voltage always remain positive. If there is a violation, the block issues error messages.

Ports

Output

q — Battery charge level, C
physical signal

Physical signal port that outputs the internal charge, in the units of coulomb (C). Use this output port to change load behavior as a function of charge, without the complexity of building a charge state estimator.

Dependencies

Enabled for the instrumented variants of the block: **Instrumented | No thermal port** and **Instrumented | Show thermal port**.

Conserving

+ — Positive terminal
electrical

Electrical conserving port associated with the battery positive terminal.

- — Negative terminal
electrical

Electrical conserving port associated with the battery negative terminal.

H — Battery thermal mass
thermal

Thermal conserving port that represents the battery thermal mass. When you expose this port, provide additional parameters to define battery behavior at a second temperature. For more information, see "Modeling Thermal Effects" on page 1-12.

Dependencies

Enabled for the thermal variants of the block: **Uninstrumented | Show thermal port** and **Instrumented | Show thermal port**.

Parameters

Main

Nominal voltage — Output voltage when battery is fully charged
12 V (default) | positive number

The no-load voltage across the battery when it is fully charged.

Internal resistance — Battery internal resistance
2 Ω (default) | positive number

Internal connection resistance of the battery.

Battery charge capacity — Select battery model
Infinite (default) | Finite

Select one of the options for modeling the charge capacity of the battery:

- Infinite — The battery voltage is independent of charge drawn from the battery.
- Finite — The battery voltage decreases as charge decreases.

Ampere-hour rating — Nominal battery capacity when fully charged
50 hr*A (default) | positive number

The maximum (nominal) battery charge in ampere-hours. To specify a target value for the initial battery charge at the start of simulation, use the high-priority **Charge** variable. For more information, see "Variables" on page 1-14.

Dependencies

Enabled when the **Battery charge capacity** parameter is set to `Finite`.

Voltage V1 < Vnom when charge is AH1 — Output voltage at charge level AH1
11.5 V (default) | positive number

The fundamental battery output voltage when the charge level is AH1, as specified by the **Charge AH1 when no-load volts are V1** parameter.

Dependencies

Enabled when the **Battery charge capacity** parameter is set to `Finite`.

Charge AH1 when no-load volts are V1 — Charge level when the no-load output voltage is V1
25 hr*A (default) | positive number

The battery charge level corresponding to the no-load output voltage specified by the **Voltage V1 < Vnom when charge is AH1** parameter.

Dependencies

Enabled when the **Battery charge capacity** parameter is set to `Finite`.

Self-discharge — Select whether to model the self-discharge resistance of the battery
`Disabled` (default) | `Enabled`

Select whether to model the self-discharge resistance of the battery. The block models this effect as a resistor across the terminals of the fundamental battery model.

As temperature increases, self-discharge resistance decreases, causing self-discharge to increase. If the decrease in resistance is too fast, thermal runaway of the battery and numerical instability can occur. You can resolve this by doing any of the following:

• Decrease the thermal resistance

• Decrease the gradient of the self-discharge resistance with respect to temperature

- Increase the self-discharge resistance

Dependencies

Enabled when the **Battery charge capacity** parameter is set to `Finite`.

Self-discharge resistance — Resistance that represents battery self-discharge
2000 Ω (default) | positive number

The resistance across the fundamental battery model that represents battery self-discharge.

Dependencies

Enabled when the **Self-discharge** parameter is set to `Enabled`.

Measurement temperature — Temperature at which the block parameters are measured
298.15 K (default) | positive number

Temperature T_1, at which the block parameters in the **Main** section are measured. For more information, see "Modeling Thermal Effects" on page 1-12.

Dependencies

Enabled for blocks with exposed thermal port.

Dynamics

Charge dynamics — Battery charge dynamics model
No dynamics (default) | Single time-constant dynamics | Double time-constant dynamics

Select how to model battery charge dynamics. This parameter determines the number of parallel RC sections in the equivalent circuit:

- `No dynamics` — The equivalent circuit contains no parallel RC sections. There is no delay between terminal voltage and internal charging voltage of the battery.
- `Single time-constant dynamics` — The equivalent circuit contains one parallel RC section. Specify the time constant using the **First time constant** parameter.

- `Double time-constant dynamics` — The equivalent circuit contains two parallel RC sections. Specify the time constants using the **First time constant** and **Second time constant** parameters.

First polarization resistance — First RC resistance
`0.005` Ω (default) | positive number

The resistance of the first parallel RC section. This parameter primarily affects the ohmic losses of the RC section.

Dependencies

To enable this parameter, set **Charge dynamics** to `Single time-constant dynamics` or `Double time-constant dynamics`.

First time constant — First RC time constant
`30` s (default) | positive number

The time constant of the first parallel RC section. This value is equal to *RC* and affects the dynamics of the RC section.

Dependencies

To enable this parameter, set **Charge dynamics** to `Single time-constant dynamics` or `Double time-constant dynamics`.

Second polarization resistance — Second RC resistance
`0.005` Ω (default) | positive number

The resistance of the second parallel RC section. This parameter primarily affects the ohmic losses of the RC section.

Dependencies

To enable this parameter, set **Charge dynamics** to `Double time-constant dynamics`.

Second time constant — Second RC time constant
`30` s (default) | positive number

The time constant of the second parallel RC section. This value is equal to *RC* and affects the dynamics of the RC section.

Dependencies

To enable this parameter, set **Charge dynamics** to `Double time-constant dynamics`.

Fade

`Battery fade` — Select whether to model battery performance deterioration with aging
`Disabled` (default) | `Enabled`

Select whether to include battery fade modeling:

- `Disabled` — The battery performance is not age dependent.

- `Enabled` — The battery performance changes depending on the number of completed charge-discharge cycles. Selecting this option exposes additional parameters in this section, which define the battery performance after a certain number of discharge cycles. The block uses these parameter values to calculate the scaling coefficients k_1, k_2, and k_3. For more information, see "Modeling Battery Fade" on page 1-11.

Dependencies

Enabled when the **Battery charge capacity** parameter in the **Main** section is set to `Finite`. If **Battery charge capacity** is `Infinite`, the **Fade** section is empty.

`Number of discharge cycles, N` — Number of cycles that defines a second set of data points
100 (default) | positive number

The number of charge-discharge cycles after which the other parameters in this section are measured. This second set of data points defines the scaling coefficients k_1, k_2, and k_3, used in modeling battery fade.

Dependencies

Enabled when the **Battery fade** parameter is set to `Enabled`.

`Ampere-hour rating after N discharge cycles` — Maximum battery capacity after N discharge cycles
45 hr*A (default) | positive number

The maximum battery charge, in ampere-hours, after the number of discharge cycles specified by the **Number of discharge cycles, N** parameter.

Dependencies

Enabled when the **Battery fade** parameter is set to `Enabled`.

Internal resistance after N discharge cycles — Battery internal resistance after N discharge cycles
2.02 Ω (default) | positive number

The battery internal resistance after the number of discharge cycles specified by the **Number of discharge cycles, N** parameter.

Dependencies

Enabled when the **Battery fade** parameter is set to `Enabled`.

Voltage V1 at charge AH1 after N discharge cycles — Output voltage at charge level AH1 after N discharge cycles
10.35 V (default) | positive number

The fundamental battery model output voltage, at charge level AH1, after the number of discharge cycles specified by the **Number of discharge cycles, N** parameter.

Dependencies

Enabled when the **Battery fade** parameter is set to `Enabled`.

Temperature Dependence

This section appears only for blocks with exposed thermal port. For more information, see "Modeling Thermal Effects" on page 1-12.

Nominal voltage at second measurement temperature — Output voltage when battery is fully charged
12 V (default) | positive number

The no-load voltage across the battery at the second measurement temperature when it is fully charged.

Internal resistance at second measurement temperature — Battery internal resistance
2.2 Ω (default) | positive number

Internal connection resistance of the battery at the second measurement temperature.

Voltage V1 at second measurement temperature — Output voltage at charge level AH1
11.4 V (default) | positive number

The fundamental battery model output voltage at the second measurement temperature and at charge level AH1, as specified by the **Charge AH1 when no-load volts are V1** parameter.

Dependencies

Enabled when the **Battery charge capacity** parameter in the **Main** section is set to `Finite`.

Self-discharge resistance at second measurement temperature — Resistance that represents battery self-discharge
2200 Ω (default) | positive number

The resistance across the fundamental battery model at the second measurement temperature. This resistance represents the self-discharge.

Dependencies

Enabled when the **Self-discharge resistance** parameter in the **Main** section is set to `Enabled`.

First polarization resistance at second measurement temperature — First RC resistance at second measurement temperature
`0.005` Ω (default) | positive number

The resistance of the first parallel RC section at the second measurement temperature.

Dependencies

To enable this parameter, set **Charge dynamics** to `Single time-constant dynamics` or `Double time-constant dynamics`.

First time constant at second measurement temperature — First RC time constant at second measurement temperature
30 s (default) | positive number

The time constant of the first parallel RC section at the second measurement temperature.

Dependencies

To enable this parameter, set **Charge dynamics** to `Single time-constant dynamics` or `Double time-constant dynamics`.

Second polarization resistance at second measurement temperature — Second RC resistance at second measurement temperature
0.005 Ω (default) | positive number

The resistance of the second parallel RC section at the second measurement temperature.

Dependencies

To enable this parameter, set **Charge dynamics** to `Double time-constant dynamics`.

Second time constant at second measurement temperature — Second RC time constant at second measurement temperature
30 s (default) | positive number

The time constant of the second parallel RC section at the second measurement temperature.

Dependencies

To enable this parameter, set **Charge dynamics** to `Double time-constant dynamics`.

Second measurement temperature — Temperature at which the block parameters in this section are measured
273.15 K (default) | positive number

Temperature T_2, at which the block parameters in the **Temperature Dependence** section are measured. For more information, see "Modeling Thermal Effects" on page 1-12.

To specify the initial temperature at the start of simulation, use the high-priority **Temperature** variable. For more information, see "Variables" on page 1-14.

Thermal Port

This section appears only for blocks with exposed thermal port. For more information, see "Modeling Thermal Effects" on page 1-12.

`Thermal mass` — **Thermal mass associated with the thermal port**
30000 J/K (default) | positive number

Thermal mass associated with the thermal port H. It represents the energy required to raise the temperature of the thermal port by one degree.

References

[1] Ramadass, P., B. Haran, R. E. White, and B. N. Popov. "Mathematical modeling of the capacity fade of Li-ion cells." *Journal of Power Sources*. 123 (2003), pp. 230â€"240.

[2] Ning, G., B. Haran, and B. N. Popov. "Capacity fade study of lithium-ion batteries cycled at high discharge rates." *Journal of Power Sources*. 117 (2003), pp. 160â€"169.

See Also
Controlled Voltage Source | DC Voltage Source

Introduced in R2008b

Battery (Table-Based)

Tabulated battery model
Library: Simscape / Electronics / Sources

Description

The Battery (Table-Based) block represents a high-fidelity battery model. The block calculates no-load voltage as a function of charge level and temperature using lookup tables and includes several modeling options:

- Self-discharge
- Battery fade
- Charge dynamics

The plot shows a battery whose performance varies with temperature and state of charge changes, as typically found on a datasheet.
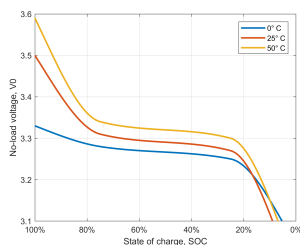
Use this block to parameterize batteries with complex no-load voltage behavior from datasheets or experimental results. For a simpler representation of a battery, see the Battery block.

The Battery (Table-Based) block has four modeling variants, accessible by right-clicking the block in your block diagram and then selecting the appropriate option from the context menu, under **Simscape > Block choices**:

- **Uninstrumented | No thermal port** — Basic model that does not output battery charge level and simulates at a fixed temperature. This modeling variant is the default.
- **Uninstrumented | Show thermal port** — Model with exposed thermal port. This model does not output internal charge level of the battery.
- **Instrumented | No thermal port** — Model with exposed charge output port. This model uses a fixed temperature throughout the simulation.
- **Instrumented | Show thermal port** — Model that lets you output internal charge level of the battery. Both the thermal port and the charge output port are exposed.

The instrumented variants have an extra physical signal port that outputs the internal state of charge. Use this functionality to change load behavior as a function of state of charge, without the complexity of building a charge state estimator.

The thermal port variants expose a thermal port, which represents the battery thermal mass.

The battery equivalent circuit is made up of the fundamental battery model, the self-discharge resistance R_{SD}, the charge dynamics model, and the series resistance R_0.
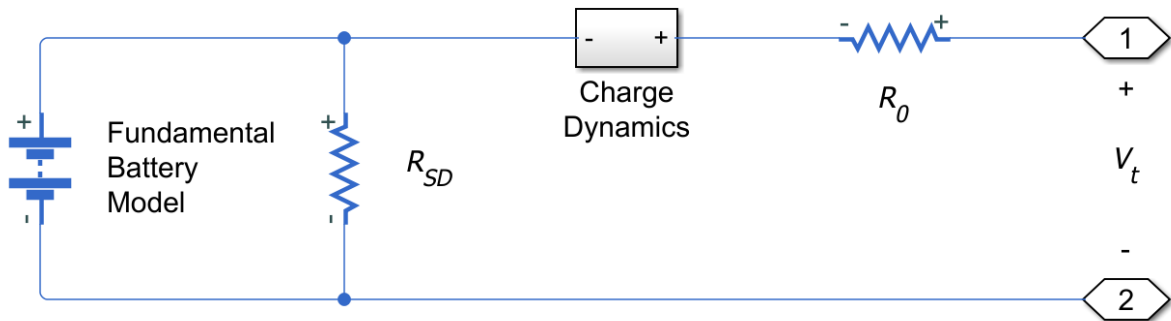
Battery Model

The block calculates the no-load voltage, or the voltage across the fundamental battery model by interpolation:

$$v_0 = v_0(\text{SOC}, T)$$

Where:

- v_0 is the no-load voltage of the battery. Specify the grid of lookup values using the **No-load voltage, V0(SOC,T)** parameter.

- SOC is the ratio of current charge to rated battery capacity. Specify the SOC breakpoints using the **Vector of state-of-charge values, SOC** parameter.

- T is the battery temperature. Specify the T breakpoints using the **Vector of temperatures, T** parameter.

The block also models the series resistance R_0 as a function of state of charge and temperature. Specify the grid of lookup values for the series resistance using the **Terminal resistance, R0(SOC,T)** parameter.

Modeling Self-Discharge

When the battery terminals are open-circuit, it is still possible for internal currents to discharge the battery. This behavior is called self-discharge. To enable this effect, set the **Self-discharge** parameter to Enabled.

The block models these internal currents with a temperature-dependent resistance $R_{SD}(T)$ across the terminals of the fundamental battery model. You can specify the lookup values for this resistance using the **Self-discharge resistance, Rleak(T)** parameter.

Modeling Charge Dynamics

Batteries exhibit greater charging/discharging currents immediately after a change in load than they do after an extended period. This time-varying property is a result of battery charge dynamics and is modeled using parallel RC sections in the equivalent circuit.

You can model battery charge dynamics using the **Charge dynamics** parameter:

- No dynamics — The equivalent circuit contains no parallel RC sections. There is no delay between terminal voltage and internal charging voltage of the battery.

- Single time-constant dynamics — The equivalent circuit contains one parallel RC section. Specify the time constant using the **First time constant, tau1(SOC,T)** parameter.

- Double time-constant dynamics — The equivalent circuit contains two parallel RC sections. Specify the time constants using the **First time constant, tau1(SOC,T)** and **Second time constant, tau2(SOC,T)** parameters.

This diagram shows the equivalent circuit for the block configured with double time-constant dynamics.
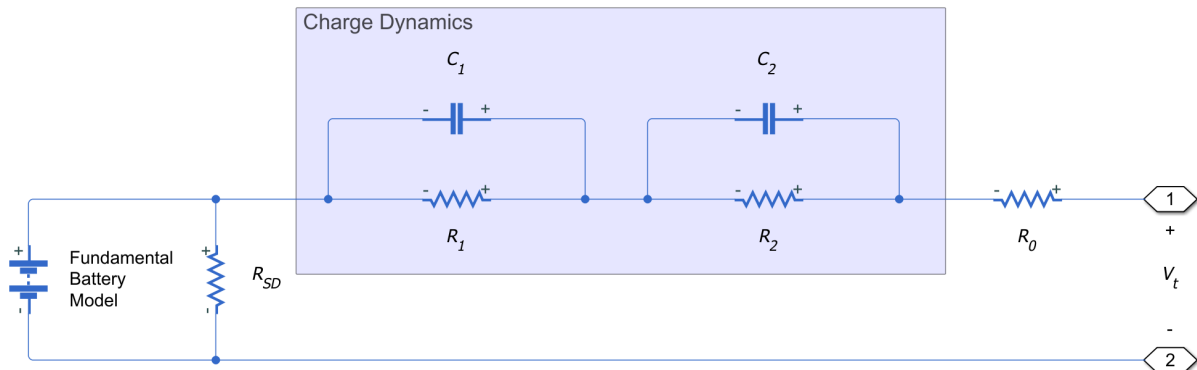
In the diagram:

- R_1 and R_2 are the parallel RC resistances. Specify these values with the **First polarization resistance, R1(SOC,T)** and **Second polarization resistance, R2(SOC,T)** parameters, respectively.
- C_1 and C_2 are the parallel RC capacitances. The time constant τ for each parallel section relates the R and C values using the relationship $C = \tau / R$. Specify τ for each section using the **First time constant, tau1(SOC,T)** and **Second time constant, tau2(SOC,T)** parameters, respectively.
- R_0 is the series resistance. Specify this value with the **Internal resistance, R0(SOC,T)** parameter.

Modeling Battery Fade

Battery fade is the deterioration of battery performance over repeated charge and discharge cycles. The no-load voltage across the fundamental battery model fades proportionally with the number of discharge cycles n:

$$v_{0,\text{fade}} = v_0 \left(1 + \frac{\delta_{v_0}}{100} \frac{n}{N} \right)$$

Where:

- N is the reference number of discharge cycles over which you specify percent change of various battery parameters. Set this value using the **Number of discharge cycles, N** parameter.

- δ_{v_0} is the percent change in no-load voltage after N discharge cycles. Specify δ_{v_0} using the **Change in no-load voltage after N discharge cycles (%)** parameter.

The nominal charge, from which state of charge is calculated, fades with the square root of number of discharge cycles:

$$q_{nom,\text{fade}} = q_{nom} \left(1 + \frac{\delta_{AH}}{100} \sqrt{\frac{n}{N}} \right)$$

Where:

- q_{nom} is the ampere-hour rating of the battery. Specify this value using the **Ampere-hour rating, AH(T)** parameter.

- δ_{AH} is the percent change in ampere-hour rating of the battery after N discharge cycles.

All resistances in the battery model also fade with the square root of the number of discharge cycles:

$$R_{i,\text{fade}} = R_i \left(1 + \frac{\delta_{R_i}}{100} \sqrt{\frac{n}{N}} \right)$$

Where:

- R_i is the ith resistance
- δ_{R_i} is the percent change in this resistance over N cycles

Depending on how you have configured the block, the resistances might include:

- The series resistance — Specify the percent change over N cycles using the **Change in terminal resistance after N discharge cycles (%)** parameter.

- The self-discharge resistance — Specify the percent change over N cycles using the **Change in self-discharge resistance after N discharge cycles (%)** parameter.

- The first charge dynamics resistance — Specify the percent change over *N* cycles using the **Change in first polarization resistance after N discharge cycles (%):** parameter.

- The second charge dynamics resistance — Specify the percent change over *N* cycles using the **Change in second polarization resistance after N discharge cycles (%)** parameter.

Modeling Thermal Effects

For thermal variants of the block, the battery temperature is determined from a summation of all the ohmic losses included in the model:

$$M_{th}\dot{T} = \sum_i V_{T,i}{}^2 / R_{T,i}$$

Where:

- M_{th} is the battery thermal mass.

- *i* corresponds to the *i*th ohmic loss contributor. Depending on how you have configured the block, the losses might include:

  - The series resistance

  - The self-discharge resistance

  - The first charge dynamics segment

  - The second charge dynamics segment

- $V_{T,i}$ is the voltage drop across resistor *i*.

- $R_{T,i}$ is resistor *i*.

Ports

Output

SOC — Battery charge level, C
physical signal

Physical signal port that outputs the internal state of charge. Use this output port to change load behavior as a function of state of charge, without the complexity of building a charge state estimator.

Dependencies

Enabled for the instrumented variants of the block: **Instrumented | No thermal port** and **Instrumented | Show thermal port**.

Conserving

+ — Positive terminal
electrical

Electrical conserving port associated with the battery positive terminal.

- — Negative terminal
electrical

Electrical conserving port associated with the battery negative terminal.

H — Battery thermal mass
thermal

Thermal conserving port that represents the battery thermal mass.

Dependencies

Enabled for the thermal variants of the block: **Uninstrumented | Show thermal port** and **Instrumented | Show thermal port**.

Parameters

Main

Vector of state-of-charge values, SOC — SOC breakpoints
[0, .25, .75, 1] (default) | vector

Vector of state-of-charge breakpoints defining the points at which you specify lookup data. This vector must be strictly ascending.

Vector of temperatures, T — *T* breakpoints
[273.15, 298.15, 323.15] K (default) | vector of positive numbers

Vector of temperature breakpoints defining the points at which you specify lookup data. This vector must be positive and strictly ascending.

No-load voltage, V0(SOC,T) — *V0* lookup table
[3.2, 3.1, 3.14; 3.25, 3.27, 3.3; 3.28, 3.31, 3.34; 3.33, 3.5, 3.59] V (default) | matrix of nonnegative numbers

Lookup data for no-load voltages across the fundamental battery model at the specified SOC and temperature breakpoints.

Terminal resistance, R0(SOC,T) — R0 lookup table
[.03, .015, .002; .04, .017, .008; .039, .012, .006; .027, .013, .021] Ohm (default) | matrix of nonnegative numbers

Lookup data for series resistance of the battery at the specified SOC and temperature breakpoints.

Ampere-hour rating, AH(T) — AH breakpoints
[2.9, 4.1, 4.2] hr*A (default) | vector of nonnegative numbers

Lookup data for the ampere-hour rating of the battery at the specified temperature breakpoints. The block calculates the state of charge by dividing the accumulated charge by this value. The block calculates accumulated charge by integrating the battery current.

Self-discharge — Select whether to model the self-discharge resistance of the battery
Disabled (default) | Enabled

Select whether to model the self-discharge resistance of the battery. The block models this effect as a resistor across the terminals of the fundamental battery model.

As temperature increases, self-discharge resistance decreases, causing self-discharge to increase. If the decrease in resistance is too fast, thermal runaway of the battery and numerical instability can occur. You can resolve this instability by making any of these changes:

• Decrease the thermal resistance

• Decrease the gradient of the self-discharge resistance with respect to temperature

- Increase the self-discharge resistance

Self-discharge resistance, Rleak(T) — Resistance that represents battery self-discharge
[8000, 7000, 6000] Ohm (default) | vector of positive numbers

Lookup data for self-discharge resistance of the battery at the specified temperature breakpoints. This resistance acts across the terminals of the fundamental battery model.

Dependencies

Enabled when the **Self-discharge** parameter is set to Enabled.

Dynamics

Charge dynamics — Battery charge dynamics model
No dynamics (default) | Single time-constant dynamics | Double time-constant dynamics

Select how to model battery charge dynamics. This parameter determines the number of parallel RC sections in the equivalent circuit:

- No dynamics — The equivalent circuit contains no parallel RC sections. There is no delay between terminal voltage and internal charging voltage of the battery.
- Single time-constant dynamics — The equivalent circuit contains one parallel RC section. Specify the time constant using the **First time constant, tau1(SOC,T)** parameter.
- Double time-constant dynamics — The equivalent circuit contains two parallel RC sections. Specify the time constants using the **First time constant, tau1(SOC,T)** and **Second time constant, tau2(SOC,T)** parameters.

First polarization resistance, R1(SOC,T) — First RC resistance
[.089, .076, .01; .042, .022, .099; .019, .007, .002; .051, .043, .029] Ohm (default) | matrix of positive numbers

Lookup data for the first parallel RC resistance at the specified SOC and temperature breakpoints. This parameter primarily affects the ohmic losses of the RC section.

Dependencies

To enable this parameter, set **Charge dynamics** to Single time-constant dynamics or Double time-constant dynamics.

First time constant, tau1(SOC,T) — First RC time constant
[44, 148, 235; 93, 110, 1000; 19, 27, 133; .5, 22, 3] s (default) | matrix of positive numbers

Lookup data for the first parallel RC time constant at the specified SOC and temperature breakpoints.

Dependencies

To enable this parameter, set **Charge dynamics** to Single time-constant dynamics or Double time-constant dynamics.

Second polarization resistance, R2(SOC,T) — Second RC resistance
[.014, .382, .407; .028, .006, .007; .014, .007, .006; .333, .956, .912] Ohm (default) | matrix of positive numbers

Lookup data for the second parallel RC resistance at the specified SOC and temperature breakpoints. This parameter primarily affects the ohmic losses of the RC section.

Dependencies

To enable this parameter, set **Charge dynamics** to Double time-constant dynamics.

Second time constant, tau2(SOC,T) — Second RC time constant
[1, 44, 5644; 11, 24, 506; 2, 14, 330; 3310, 13419, 30216] s (default) | matrix of positive numbers

Lookup data for the second parallel RC time constant at the specified SOC and temperature breakpoints.

Dependencies

To enable this parameter, set **Charge dynamics** to Double time-constant dynamics.

Fade

Number of discharge cycles, N — Reference number of cycles for percent change calculations
100 (default) | number from 1 to Inf

The number of charge-discharge cycles over which the specified percent changes occur.

Change in no-load voltage after N discharge cycles (%) — **Percent change in no-load voltage after *N* cycles**
0 (default) | scalar

Percent change in the no-load voltage after the battery undergoes *N* discharge cycles.

Change in terminal resistance after N discharge cycles (%) — **Percent change in series resistance after *N* cycles**
0 (default) | scalar

Percent change in the series resistance after the battery undergoes *N* discharge cycles.

Change in ampere-hour rating after N discharge cycles (%) — **Percent change in ampere-hour rating after *N* cycles**
0 (default) | scalar

Percent change in the ampere-hour rating after the battery undergoes *N* discharge cycles.

Change in self-discharge resistance after N discharge cycles (%) — **Percent change in self-discharge resistance after *N* cycles**
0 (default) | scalar

Percent change in the self-discharge resistance after the battery undergoes *N* discharge cycles.

Dependencies

To enable this parameter, set **Self-discharge** to Enabled.

Change in first polarization resistance after N discharge cycles (%) — **Percent change in first RC resistance after *N* cycles**
0 (default) | scalar

Percent change in the first RC resistance after the battery undergoes *N* discharge cycles.

Dependencies

To enable this parameter, set **Charge dynamics** to Single time-constant dynamics or Double time-constant dynamics.

Change in second polarization resistance after N discharge cycles (%) — **Percent change in first RC resistance after *N* cycles**
0 (default) | scalar

Percent change in the second RC resistance after the battery undergoes *N* discharge cycles.

Dependencies

To enable this parameter, set **Charge dynamics** to `Double time-constant dynamics`.

Thermal

`Simulation temperature` — **Battery temperature**
`298.15` K (default) | positive number

Battery temperature used in lookup tables throughout simulation.

Dependencies

This section appears only for blocks without an exposed thermal port. For more information, see "Modeling Thermal Effects" on page 1-12.

`Thermal mass` — **Thermal mass associated with the thermal port**
`30000` J/K (default)

Thermal mass associated with the thermal port H. It represents the energy required to raise the temperature of the thermal port by one degree.

Dependencies

Enabled for blocks with an exposed thermal port. For more information, see "Modeling Thermal Effects" on page 1-12.

See Also
Battery

Introduced in R2018a

Capacitor

Capacitor model including nonlinearity, tolerance, operational limits, and fault behavior

Library

Passive Devices

Description

The Capacitor block lets you model linear and nonlinear (table-based) capacitors, including polar capacitors. Optionally, you can also model the following effects:

*   "Tolerances" on page 1-38
*   "Operating Limits" on page 1-39
*   "Faults" on page 1-39

You can turn these modeling options on and off independently of each other. When all the additional options are turned off, the component behavior is identical to the Simscape Foundation library Capacitor block.

In its simplest form, the Capacitor block models a linear capacitor, described with the following equation:

$$I = C \frac{dV}{dt}$$

where:

* I is the current.
* C is the capacitance.

- V is the voltage.

- t is the time.

To model a nonlinear or polar capacitor, set the **Enable table-based capacitance** parameter to `Yes - use table-based capacitance` and provide a lookup table of capacitance-voltage values:

- For polar capacitors, where this lookup table is asymmetric with respect to the applied terminal voltage, set the **Symmetric C-V table** parameter to `No - use C-V data as-is`.

- For other types of nonlinear capacitor, ensure symmetry of the capacitance with regards to the applied terminal voltage by setting the **Symmetric C-V table** parameter to `Yes - use voltage magnitude when computing C`.

Tolerances

You can apply tolerances to the nominal value you provide for the **Capacitance** parameter. Datasheets typically provide a tolerance percentage for a given capacitor type. The table shows how the block applies tolerances and calculates capacitance based on the selected **Tolerance application** option.

Option	Capacitance Value
`None — use nominal value`	C
`Random tolerance`	Uniform distribution: $C \cdot (1 - tol + 2 \cdot tol \cdot$ `rand`) Gaussian distribution: $C \cdot (1 + tol \cdot$ `randn` $/ nSigma)$
`Apply maximum tolerance value`	$C \cdot (1 + tol)$
`Apply minimum tolerance value`	$C \cdot (1 - tol)$

In the table,

- C is the **Capacitance** parameter value, nominal capacitance.

- tol is the fractional tolerance, **Capacitance tolerance (%)** /100.

- $nSigma$ is the value you provide for the **Number of standard deviations for quoted tolerance** parameter.

- `rand` and `randn` are standard MATLAB® functions for generating uniform and normal distribution random numbers.

Operating Limits

You can specify operating limits in terms of maximum working voltage and the maximum (instantaneous) power dissipation in the series resistance and in the parallel conductance of the capacitor.

For polar capacitors, you can define the working voltage range in such a way that the block provides a warning, or an error, if the polarity of the applied voltage becomes incorrect.

When an operating limit is exceeded, the block can either generate a warning or stop the simulation with an error. For more information, see "Operating Limits" on page 1-42.

Faults

Instantaneous changes in capacitor parameters are unphysical. Therefore, when the Capacitor block enters the faulted state, the capacitance, resistance, and conductance transition to their faulted values over a period of time, according to the following formula:

$$CurrentValue = FaultedValue - (FaultedValue - UnfaultedValue) \cdot \text{sech}(\Delta t \, / \, \tau)$$

where:

- Δt is the time since the onset of the fault condition.
- τ is the user-defined time constant associated with the fault transition.

The block can trigger the start of fault transition:

- At a specific time
- When terminal voltage is outside the permissible voltage range for longer than a specific time interval

You can enable or disable these trigger mechanisms separately, or use them together if more than one trigger mechanism is required in a simulation. When more than one mechanism is enabled, the first mechanism to trigger the fault transition takes precedence. In other words, component fails no more than once per simulation.

You can also choose whether to issue an assertion when a fault occurs, by using the **Reporting when a fault occurs** parameter. The assertion can take the form of a warning or an error. By default, the block does not issue an assertion.

Variables

Use the **Variables** section of the block interface to set the priority and initial target values for the block variables prior to simulation. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

The **Capacitor voltage** variable lets you specify a high-priority target for the initial capacitor voltage at the start of simulation.

Parameters

- "Main" on page 1-40
- "Operating Limits" on page 1-42
- "Faults" on page 1-43

Main

Enable table-based capacitance

> Select the type of capacitor:
>
> - `No - use constant capacitance` — Model a linear capacitor, with nominal capacitance defined by the **Capacitance** parameter value. This is the default.
> - `Yes - use table-based capacitance` — Model a nonlinear capacitor, where the nominal capacitance value changes based on the value of applied terminal voltage.

Capacitance

> The nominal capacitance value for linear capacitor. This parameter is visible only if you select `No - use constant capacitance` for the **Enable table-based capacitance** parameter. Capacitance value must be greater than zero. The default value is 1 μF.

Capacitance values

> The vector of capacitance values, for table lookup based on the corresponding voltage value. This parameter is visible only if you select `Yes - use table-based`

`capacitance` for the **Enable table-based capacitance** parameter. Capacitance values must be greater than 0. The vector length must be the same as the voltage vector length. The default value is `[1e-5 1e-6]` F.

Corresponding voltage values

The input vector of voltage values for table-based capacitance calculation. This parameter is visible only if you select `Yes - use table-based capacitance` for the **Enable table-based capacitance** parameter. The vector length must be greater than or equal to 2, and the values must be strictly monotonic, either increasing or decreasing. The default value is `[0 10]` V.

Symmetric C-V table

This parameter is visible only if you select `Yes - use table-based capacitance` for the **Enable table-based capacitance** parameter. Specify how to use the table data:

- `Yes - use voltage magnitude when computing C` — Use this option to ensure symmetry of the capacitance with regards to the applied terminal voltage. This is the default.

- `No - use C-V data as-is` — Use this option to model polar capacitors. For example, with default parameter values for table-based capacitance, applied voltage of –10 V would produce nominal capacitance of 1e-6 F. However, if you select `No - use C-V data as-is` for the **Symmetric C-V table** parameter, the resulting capacitance value is 1e-5 F, because the block uses the nearest input value for extrapolation.

Capacitance tolerance (%)

The capacitor tolerance as defined on the manufacturer datasheet. For table-based capacitors, this tolerance is applied to the entire table at once. The default value is 5%.

Tolerance application

Select how to apply tolerance during simulation:

- `None — use nominal value` — The block does not apply tolerance, uses the nominal capacitance value. This is the default.

- `Random tolerance` — The block applies random offset to the capacitance value, within the tolerance value limit. You can choose Uniform or Gaussian distribution for calculating the random number by using the **Tolerance distribution** parameter.

- `Apply maximum tolerance value` — The capacitance is increased by the specified tolerance percent value.
- `Apply minimum tolerance value` — The capacitance is decreased by the specified tolerance percent value.

Tolerance distribution

This parameter is visible only if you select `Random tolerance` for the **Tolerance application** parameter. Select the distribution type:

- `Uniform` — Uniform distribution. This is the default.
- `Gaussian` — Gaussian distribution.

Number of standard deviations for quoted tolerance

Number of standard deviations for calculating the Gaussian random number. This parameter is visible only if you select `Gaussian` for the **Tolerance distribution** parameter. The default value is 4.

Series resistance

Simulation of some circuits may require the presence of the small series resistance. Equivalent series resistance (ESR) is sometimes specified on manufacturer datasheets. If not, you can define this resistance via the dissipation factor (DF), which is also shown on many datasheets. The relationship is $DF = 2\pi \cdot f \cdot C \cdot ESR$, where f is signal frequency. The default value is 1 μΩ.

Parallel conductance

Parallel leakage path associated with the capacitor. For capacitors connected in series, the presence of a small parallel conductance can help with convergence. The default value is 0 1/Ω.

Operating Limits

Enable operating limits

Select `Yes` to enable reporting when the operational limits are exceeded. The associated parameters become visible on the **Operating Limits** tab to let you select the reporting method and specify the operating limits in terms of power and working voltage. The default value is `No`.

Reporting when operating limits exceeded

Select what happens when an operating limit is exceeded:

- `Warn` — The block issues a warning. This is the default.
- `Error` — Simulation stops with an error.

Working voltage range

Range of voltage values allowed for normal block operation, specified as a vector of size 2. The default value is `[-25 25]` V.

Power rating

Maximum instantaneous power dissipation in the resistance and conductance elements associated with the capacitor. The default value is `1` W.

Faults

Enable faults

Select `Yes` to enable faults modeling. The associated parameters become visible on the **Faults** tab to let you select the reporting method and specify the trigger mechanism (temporal or behavioral). You can enable these trigger mechanisms separately or use them together. The default value is `No`.

Reporting when a fault occurs

Choose whether to issue an assertion when a fault occurs:

- `None` — The block does not issue an assertion. This is the default.
- `Warn` — The block issues a warning.
- `Error` — Simulation stops with an error.

Faulted capacitance as % of unfaulted

Relative change in the capacitance when the block is in the faulted state, as compared to the unfaulted state. For table-based capacitances, the relative change is applied to all elements of the vector. The default value is `100`, which means that the faulted capacitance is equal to the unfaulted capacitance.

Faulted series resistance

Equivalent series resistance of the capacitor when the block is in the faulted state. The default value is `1` mΩ.

Faulted parallel conductance

Parallel leakage conductance of the capacitor when the block is in the faulted state. The default value is `0` 1/Ω.

Fault transition time constant

Time constant associated with the transition to the faulted state, as described in "Faults" on page 1-39. The default value is 1 ms.

Enable temporal fault trigger

Select Yes to enable time-based fault triggering. The default value is No.

Simulation time for a fault event

Set the simulation time at which you want the block to start entering the fault state. This parameter is visible only if the **Enable temporal fault trigger** parameter is set to Yes. The default value is 1 s.

Enable behavioral fault trigger

Select Yes to enable behavioral fault triggering. The default value is No.

Permissible voltage range

Specify the minimum and maximum permissible voltage. If the voltage value is outside this range for longer than the **Time to fail when exceeding voltage range** parameter value, then the block starts entering the fault state. This parameter is visible only if the **Enable behavioral fault trigger** parameter is set to Yes. The default value is [-100 100] V.

Time to fail when exceeding voltage range

Set the maximum length of time that the voltage can be outside the permissible voltage range without triggering the fault. This parameter is visible only if the **Enable behavioral fault trigger** parameter is set to Yes. The default value is 1 s.

Ports

The block has the following ports:

+

Positive electrical port

-

Negative electrical port

See Also

Fault | Inductor | Resistor

CMOS AND

Model CMOS AND gate behaviorally

Library

Logic

Description

The CMOS AND block represents a CMOS AND logic gate behaviorally:

- The block output logic level is HIGH if the logic levels of both of the gate inputs are 1.
- The block output logic level is LOW otherwise.

The block determines the logic levels of the gate inputs as follows:

- If the gate voltage is greater than the threshold voltage, the block interprets the input as logic 1.
- Otherwise, the block interprets the input as logic 0.

The threshold voltage is the voltage value at midpoint between the **High level input voltage** parameter value and the **Low level input voltage** parameter value.

Note To improve simulation speed, the block does not model all the internal individual MOSFET devices that make up the gate. See "Basic Assumptions and Limitations" on page 1-46 for details.

The block models the gate as follows:

- The gate inputs have infinite resistance and finite or zero capacitance.

- The gate output offers a selection of two models: `Linear` and `Quadratic`. For more information, see "Selecting the Output Model for Logic Blocks". Use the **Output current-voltage relationship** parameter to specify the output model.

- You can specify propagation delay for both output models. For `Linear` output, the block sets the value of the gate output capacitor such that the resistor-capacitor time constant equals the **Propagation delay** parameter value. For `Quadratic` output, the gate input demand is lagged to approximate the **Propagation delay** parameter value.

The block output voltage depends on the output model selected:

- For `Linear` model, output high is the **High level output voltage** parameter value, and output low is the **Low level output voltage** parameter value.

- For `Quadratic` model, the output voltage for High and Low states is a function of the output current, as explained in "Quadratic Model Output and Parameters". For zero load current, output high is Vcc (the **Supply voltage** parameter value), and output low is zero volts.

Basic Assumptions and Limitations

The block does not model the internal individual MOSFET devices that make up the gate (except for the final MOSFET pair if you select the `Quadratic` option for the **Output current-voltage relationship** parameter). This limitation has the following implications:

- The block does not accurately model the gate's response to input noise and inputs that are around the logic threshold voltage.

- The block does not accurately model dynamic response.

Circuits that involve a feedback path around a set of logic gates may require a nonzero propagation delay to be set on one or more gates.

Parameters

- "Inputs Tab" on page 1-47
- "Outputs Tab" on page 1-47
- "Initial Conditions Tab" on page 1-49

Inputs Tab

Low level input voltage

Voltage value below which the block interprets the input voltage as logic `LOW`. The default value is 2 V.

High level input voltage

Voltage value above which the block interprets the input voltage as logic `HIGH`. The default value is 3 V.

Average input capacitance

Fixed capacitance that approximates the input capacitance for a MOSFET gate. The MOSFET capacitance depends on the applied voltage. When you drive this block with another gate, the **Average input capacitance** produces a rise time similar to that of the MOSFET. You can usually find this capacitance value on a manufacturer datasheet. The default value is 5 pF. Setting this value to zero may result in faster simulation times.

Outputs Tab

Output current-voltage relationship

Select the output model, `Linear` or `Quadratic`. The default value is `Linear`.

Low level output voltage

Voltage value at the output when the output logic level is `LOW`. The default value is `0` V. This parameter is available when you select the `Linear` option for the **Output current-voltage relationship** parameter.

High level output voltage

Voltage value at the output when the output logic level is `HIGH`. The default value is `5` V. This parameter is available when you select the `Linear` option for the **Output current-voltage relationship** parameter.

Output resistance

Value of the series output resistor that is used to model the drop in output voltage resulting from the output current. The default value is 25 Ω. You can derive this value from a datasheet by dividing the high-level output voltage by the maximum low-level output current. This parameter is available when you select the `Linear` option for the **Output current-voltage relationship** parameter.

Supply voltage

Supply voltage value applied to the gate in your circuit. The default value is 5 V. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Measurement voltage

The gate supply voltage for which mask data output resistances and currents are defined. The default value is 5 V. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic HIGH output resistance at zero current and at I_OH

A row vector [*R_OH1 R_OH2*] of two resistance values. The first value *R_OH1* is the gradient of the output voltage-current relationship when the gate is logic HIGH and there is no output current. The second value *R_OH2* is the gradient of the output voltage-current relationship when the gate is logic HIGH and the output current is *I_OH*. The default value is [25 250] Ω. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic HIGH output current I_OH when shorted to ground

The resulting current when the gate is in the logic HIGH state, but the load forces the output voltage to zero. The default value is 63 mA. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic LOW output resistance at zero current and at I_OL

A row vector [*R_OL1 R_OL2*] of two resistance values. The first value *R_OL1* is the gradient of the output voltage-current relationship when the gate is logic LOW and there is no output current. The second value *R_OL2* is the gradient of the output voltage-current relationship when the gate is logic LOW and the output current is *I_OL*. The default value is [30 800] Ω. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic LOW output current I_OL when shorted to Vcc

The resulting current when the gate is in the logic LOW state, but the load forces the output voltage to the supply voltage Vcc. The default value is -45 mA. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Propagation delay

Time it takes for the output to swing from `LOW` to `HIGH` or `HIGH` to `LOW` after the input logic levels change. The default value is 25 ns.

Protection diode on resistance

The gradient of the voltage-current relationship for the protection diodes when forward biased. The default value is 5 Ω. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Protection diode forward voltage

The voltage above which the protection diode is turned on. The default value is `0.6` V. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Initial Conditions Tab

Output initial state

Specify whether the initial output state of the block is `High` or `Low`. This parameter is used for both linear and quadratic output states, provided that the **Propagation delay** parameter is greater than zero and the Solver Configuration block does not have the **Start simulation from steady state** option selected. The default value is `Low`.

Ports

The block has the following ports:

A

Electrical input port

B

Electrical input port

J

Electrical output port

CMOS Buffer

Model CMOS Buffer gate behaviorally

Library

Logic

Description

The CMOS Buffer block represents a CMOS Buffer logic gate behaviorally:

- The block output logic level is HIGH if the logic level of the gate input is 1.
- The block output logic level is LOW otherwise.

The block determines the logic levels of the gate inputs as follows:

- If the gate voltage is greater than the threshold voltage, the block interprets the input as logic 1.
- Otherwise, the block interprets the input as logic 0.

The threshold voltage is the voltage value at midpoint between the **High level input voltage** parameter value and the **Low level input voltage** parameter value.

Note To improve simulation speed, the block does not model all the internal individual MOSFET devices that make up the gate. See "Basic Assumptions and Limitations" on page 1-51 for details.

The block models the gate as follows:

- The gate inputs have infinite resistance and finite or zero capacitance.

- The gate output offers a selection of two models: `Linear` and `Quadratic`. For more information, see "Selecting the Output Model for Logic Blocks". Use the **Output current-voltage relationship** parameter to specify the output model.

- You can specify propagation delay for both output models. For `Linear` output, the block sets the value of the gate output capacitor such that the resistor-capacitor time constant equals the **Propagation delay** parameter value. For `Quadratic` output, the gate input demand is lagged to approximate the **Propagation delay** parameter value.

The block output voltage depends on the output model selected:

- For `Linear` model, output high is the **High level output voltage** parameter value, and output low is the **Low level output voltage** parameter value.

- For `Quadratic` model, the output voltage for High and Low states is a function of the output current, as explained in "Quadratic Model Output and Parameters". For zero load current, output high is Vcc (the **Supply voltage** parameter value), and output low is zero volts.

Basic Assumptions and Limitations

The block does not model the internal individual MOSFET devices that make up the gate (except for the final MOSFET pair if you select the `Quadratic` option for the **Output current-voltage relationship** parameter). This limitation has the following implications:

- The block does not accurately model the gate's response to input noise and inputs that are around the logic threshold voltage.

- The block does not accurately model dynamic response.

Circuits that involve a feedback path around a set of logic gates may require a nonzero propagation delay to be set on one or more gates.

Parameters

- "Inputs Tab" on page 1-52
- "Outputs Tab" on page 1-52
- "Initial Conditions Tab" on page 1-54

Inputs Tab

Low level input voltage

Voltage value below which the block interprets the input voltage as logic `LOW`. The default value is 2 V.

High level input voltage

Voltage value above which the block interprets the input voltage as logic `HIGH`. The default value is 3 V.

Average input capacitance

Fixed capacitance that approximates the input capacitance for a MOSFET gate. The MOSFET capacitance depends on the applied voltage. When you drive this block with another gate, the **Average input capacitance** produces a rise time similar to that of the MOSFET. You can usually find this capacitance value on a manufacturer datasheet. The default value is 5 pF. Setting this value to zero may result in faster simulation times.

Outputs Tab

Output current-voltage relationship

Select the output model, `Linear` or `Quadratic`. The default value is `Linear`.

Low level output voltage

Voltage value at the output when the output logic level is `LOW`. The default value is `0` V. This parameter is available when you select the `Linear` option for the **Output current-voltage relationship** parameter.

High level output voltage

Voltage value at the output when the output logic level is `HIGH`. The default value is `5` V. This parameter is available when you select the `Linear` option for the **Output current-voltage relationship** parameter.

Output resistance

Value of the series output resistor that is used to model the drop in output voltage resulting from the output current. The default value is 25 Ω. You can derive this value from a datasheet by dividing the high-level output voltage by the maximum low-level output current. This parameter is available when you select the `Linear` option for the **Output current-voltage relationship** parameter.

Supply voltage

Supply voltage value applied to the gate in your circuit. The default value is 5 V. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Measurement voltage

The gate supply voltage for which mask data output resistances and currents are defined. The default value is 5 V. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic HIGH output resistance at zero current and at I_OH

A row vector [*R_OH1 R_OH2*] of two resistance values. The first value *R_OH1* is the gradient of the output voltage-current relationship when the gate is logic HIGH and there is no output current. The second value *R_OH2* is the gradient of the output voltage-current relationship when the gate is logic HIGH and the output current is *I_OH*. The default value is [25 250] Ω. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic HIGH output current I_OH when shorted to ground

The resulting current when the gate is in the logic HIGH state, but the load forces the output voltage to zero. The default value is 63 mA. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic LOW output resistance at zero current and at I_OL

A row vector [*R_OL1 R_OL2*] of two resistance values. The first value *R_OL1* is the gradient of the output voltage-current relationship when the gate is logic LOW and there is no output current. The second value *R_OL2* is the gradient of the output voltage-current relationship when the gate is logic LOW and the output current is *I_OL*. The default value is [30 800] Ω. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic LOW output current I_OL when shorted to Vcc

The resulting current when the gate is in the logic LOW state, but the load forces the output voltage to the supply voltage Vcc. The default value is -45 mA. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Propagation delay

Time it takes for the output to swing from LOW to HIGH or HIGH to LOW after the input logic levels change. The default value is 25 ns.

Protection diode on resistance

The gradient of the voltage-current relationship for the protection diodes when forward biased. The default value is 5 Ω. This parameter is available when you select the Quadratic option for the **Output current-voltage relationship** parameter.

Protection diode forward voltage

The voltage above which the protection diode is turned on. The default value is 0.6 V. This parameter is available when you select the Quadratic option for the **Output current-voltage relationship** parameter.

Initial Conditions Tab

Output initial state

Specify whether the initial output state of the block is High or Low. This parameter is used for both linear and quadratic output states, provided that the **Propagation delay** parameter is greater than zero and the Solver Configuration block does not have the **Start simulation from steady state** option selected. The default value is Low.

Ports

The block has the following ports:

A

Electrical input port

J

Electrical output port

CMOS NAND

Model CMOS NAND gate behaviorally

Library

Logic

Description

The CMOS NAND block represents a CMOS NAND logic gate behaviorally:

- The block output logic level is HIGH if the logic levels of both of the gate inputs are 0.
- The block output logic level is LOW otherwise.

The block determines the logic levels of the gate inputs as follows:

- If the gate voltage is greater than the threshold voltage, the block interprets the input as logic 1.
- Otherwise, the block interprets the input as logic 0.

The threshold voltage is the voltage value at midpoint between the **High level input voltage** parameter value and the **Low level input voltage** parameter value.

Note To improve simulation speed, the block does not model all the internal individual MOSFET devices that make up the gate. See "Basic Assumptions and Limitations" on page 1-56 for details.

The block models the gate as follows:

- The gate inputs have infinite resistance and finite or zero capacitance.

- The gate output offers a selection of two models: `Linear` and `Quadratic`. For more information, see "Selecting the Output Model for Logic Blocks". Use the **Output current-voltage relationship** parameter to specify the output model.

- You can specify propagation delay for both output models. For `Linear` output, the block sets the value of the gate output capacitor such that the resistor-capacitor time constant equals the **Propagation delay** parameter value. For `Quadratic` output, the gate input demand is lagged to approximate the **Propagation delay** parameter value.

The block output voltage depends on the output model selected:

- For `Linear` model, output high is the **High level output voltage** parameter value, and output low is the **Low level output voltage** parameter value.

- For `Quadratic` model, the output voltage for High and Low states is a function of the output current, as explained in "Quadratic Model Output and Parameters". For zero load current, output high is Vcc (the **Supply voltage** parameter value), and output low is zero volts.

Basic Assumptions and Limitations

The block does not model the internal individual MOSFET devices that make up the gate (except for the final MOSFET pair if you select the `Quadratic` option for the **Output current-voltage relationship** parameter). This limitation has the following implications:

- The block does not accurately model the gate's response to input noise and inputs that are around the logic threshold voltage.

- The block does not accurately model dynamic response.

Circuits that involve a feedback path around a set of logic gates may require a nonzero propagation delay to be set on one or more gates.

Parameters

- "Inputs Tab" on page 1-57
- "Outputs Tab" on page 1-57
- "Initial Conditions Tab" on page 1-59

Inputs Tab

Low level input voltage

Voltage value below which the block interprets the input voltage as logic `LOW`. The default value is 2 V.

High level input voltage

Voltage value above which the block interprets the input voltage as logic `HIGH`. The default value is 3 V.

Average input capacitance

Fixed capacitance that approximates the input capacitance for a MOSFET gate. The MOSFET capacitance depends on the applied voltage. When you drive this block with another gate, the **Average input capacitance** produces a rise time similar to that of the MOSFET. You can usually find this capacitance value on a manufacturer datasheet. The default value is 5 pF. Setting this value to zero may result in faster simulation times.

Outputs Tab

Output current-voltage relationship

Select the output model, `Linear` or `Quadratic`. The default value is `Linear`.

Low level output voltage

Voltage value at the output when the output logic level is `LOW`. The default value is `0` V. This parameter is available when you select the `Linear` option for the **Output current-voltage relationship** parameter.

High level output voltage

Voltage value at the output when the output logic level is `HIGH`. The default value is `5` V. This parameter is available when you select the `Linear` option for the **Output current-voltage relationship** parameter.

Output resistance

Value of the series output resistor that is used to model the drop in output voltage resulting from the output current. The default value is 25 Ω. You can derive this value from a datasheet by dividing the high-level output voltage by the maximum low-level output current. This parameter is available when you select the `Linear` option for the **Output current-voltage relationship** parameter.

Supply voltage

Supply voltage value applied to the gate in your circuit. The default value is 5 V. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Measurement voltage

The gate supply voltage for which mask data output resistances and currents are defined. The default value is 5 V. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic HIGH output resistance at zero current and at I_OH

A row vector [*R_OH1 R_OH2*] of two resistance values. The first value *R_OH1* is the gradient of the output voltage-current relationship when the gate is logic HIGH and there is no output current. The second value *R_OH2* is the gradient of the output voltage-current relationship when the gate is logic HIGH and the output current is *I_OH*. The default value is [25 250] Ω. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic HIGH output current I_OH when shorted to ground

The resulting current when the gate is in the logic HIGH state, but the load forces the output voltage to zero. The default value is 63 mA. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic LOW output resistance at zero current and at I_OL

A row vector [*R_OL1 R_OL2*] of two resistance values. The first value *R_OL1* is the gradient of the output voltage-current relationship when the gate is logic LOW and there is no output current. The second value *R_OL2* is the gradient of the output voltage-current relationship when the gate is logic LOW and the output current is *I_OL*. The default value is [30 800] Ω. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic LOW output current I_OL when shorted to Vcc

The resulting current when the gate is in the logic LOW state, but the load forces the output voltage to the supply voltage Vcc. The default value is -45 mA. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Propagation delay

Time it takes for the output to swing from `LOW` to `HIGH` or `HIGH` to `LOW` after the input logic levels change. The default value is 25 ns.

Protection diode on resistance

The gradient of the voltage-current relationship for the protection diodes when forward biased. The default value is 5 Ω. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Protection diode forward voltage

The voltage above which the protection diode is turned on. The default value is `0.6` V. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Initial Conditions Tab

Output initial state

Specify whether the initial output state of the block is `High` or `Low`. This parameter is used for both linear and quadratic output states, provided that the **Propagation delay** parameter is greater than zero and the Solver Configuration block does not have the **Start simulation from steady state** option selected. The default value is `Low`.

Ports

The block has the following ports:

A

Electrical input port

B

Electrical input port

J

Electrical output port

CMOS NOR

Model CMOS NOR gate behaviorally

Library

Logic

Description

The CMOS NOR block represents a CMOS NOR logic gate behaviorally:

- The block output logic level is `LOW` if the logic levels of any of the gate inputs are 1.
- The block output logic level is `HIGH` otherwise.

The block determines the logic levels of the gate inputs as follows:

- If the gate voltage is greater than the threshold voltage, the block interprets the input as logic 1.
- Otherwise, the block interprets the input as logic 0.

The threshold voltage is the voltage value at midpoint between the **High level input voltage** parameter value and the **Low level input voltage** parameter value.

Note To improve simulation speed, the block does not model all the internal individual MOSFET devices that make up the gate. See "Basic Assumptions and Limitations" on page 1-61 for details.

The block models the gate as follows:

- The gate inputs have infinite resistance and finite or zero capacitance.

- The gate output offers a selection of two models: `Linear` and `Quadratic`. For more information, see "Selecting the Output Model for Logic Blocks". Use the **Output current-voltage relationship** parameter to specify the output model.

- You can specify propagation delay for both output models. For `Linear` output, the block sets the value of the gate output capacitor such that the resistor-capacitor time constant equals the **Propagation delay** parameter value. For `Quadratic` output, the gate input demand is lagged to approximate the **Propagation delay** parameter value.

The block output voltage depends on the output model selected:

- For `Linear` model, output high is the **High level output voltage** parameter value, and output low is the **Low level output voltage** parameter value.

- For `Quadratic` model, the output voltage for High and Low states is a function of the output current, as explained in "Quadratic Model Output and Parameters". For zero load current, output high is Vcc (the **Supply voltage** parameter value), and output low is zero volts.

Basic Assumptions and Limitations

The block does not model the internal individual MOSFET devices that make up the gate (except for the final MOSFET pair if you select the `Quadratic` option for the **Output current-voltage relationship** parameter). This limitation has the following implications:

- The block does not accurately model the gate's response to input noise and inputs that are around the logic threshold voltage.

- The block does not accurately model dynamic response.

Circuits that involve a feedback path around a set of logic gates may require a nonzero propagation delay to be set on one or more gates.

Parameters

- "Inputs Tab" on page 1-62
- "Outputs Tab" on page 1-62
- "Initial Conditions Tab" on page 1-64

Inputs Tab

Low level input voltage

Voltage value below which the block interprets the input voltage as logic `LOW`. The default value is 2 V.

High level input voltage

Voltage value above which the block interprets the input voltage as logic `HIGH`. The default value is 3 V.

Average input capacitance

Fixed capacitance that approximates the input capacitance for a MOSFET gate. The MOSFET capacitance depends on the applied voltage. When you drive this block with another gate, the **Average input capacitance** produces a rise time similar to that of the MOSFET. You can usually find this capacitance value on a manufacturer datasheet. The default value is 5 pF. Setting this value to zero may result in faster simulation times.

Outputs Tab

Output current-voltage relationship

Select the output model, `Linear` or `Quadratic`. The default value is `Linear`.

Low level output voltage

Voltage value at the output when the output logic level is `LOW`. The default value is `0` V. This parameter is available when you select the `Linear` option for the **Output current-voltage relationship** parameter.

High level output voltage

Voltage value at the output when the output logic level is `HIGH`. The default value is `5` V. This parameter is available when you select the `Linear` option for the **Output current-voltage relationship** parameter.

Output resistance

Value of the series output resistor that is used to model the drop in output voltage resulting from the output current. The default value is 25 Ω. You can derive this value from a datasheet by dividing the high-level output voltage by the maximum low-level output current. This parameter is available when you select the `Linear` option for the **Output current-voltage relationship** parameter.

Supply voltage

Supply voltage value applied to the gate in your circuit. The default value is 5 V. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Measurement voltage

The gate supply voltage for which mask data output resistances and currents are defined. The default value is 5 V. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic HIGH output resistance at zero current and at I_OH

A row vector [*R_OH1 R_OH2*] of two resistance values. The first value *R_OH1* is the gradient of the output voltage-current relationship when the gate is logic HIGH and there is no output current. The second value *R_OH2* is the gradient of the output voltage-current relationship when the gate is logic HIGH and the output current is *I_OH*. The default value is [25 250] Ω. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic HIGH output current I_OH when shorted to ground

The resulting current when the gate is in the logic HIGH state, but the load forces the output voltage to zero. The default value is 63 mA. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic LOW output resistance at zero current and at I_OL

A row vector [*R_OL1 R_OL2*] of two resistance values. The first value *R_OL1* is the gradient of the output voltage-current relationship when the gate is logic LOW and there is no output current. The second value *R_OL2* is the gradient of the output voltage-current relationship when the gate is logic LOW and the output current is *I_OL*. The default value is [30 800] Ω. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic LOW output current I_OL when shorted to Vcc

The resulting current when the gate is in the logic LOW state, but the load forces the output voltage to the supply voltage Vcc. The default value is -45 mA. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Propagation delay

Time it takes for the output to swing from `LOW` to `HIGH` or `HIGH` to `LOW` after the input logic levels change. The default value is 25 ns.

Protection diode on resistance

The gradient of the voltage-current relationship for the protection diodes when forward biased. The default value is 5 Ω. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Protection diode forward voltage

The voltage above which the protection diode is turned on. The default value is `0.6` V. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Initial Conditions Tab

Output initial state

Specify whether the initial output state of the block is `High` or `Low`. This parameter is used for both linear and quadratic output states, provided that the **Propagation delay** parameter is greater than zero and the Solver Configuration block does not have the **Start simulation from steady state** option selected. The default value is `Low`.

Ports

The block has the following ports:

A

Electrical input port

B

Electrical input port

J

Electrical output port

CMOS NOT

Model CMOS NOT gate behaviorally

Library

Logic

Description

The CMOS NOT block represents a CMOS NOT logic gate behaviorally:

- The block output logic level is HIGH if the logic level of the gate input is 0.
- The block output logic level is LOW otherwise.

The block determines the logic levels of the gate inputs as follows:

- If the gate voltage is greater than the threshold voltage, the block interprets the input as logic 1.
- Otherwise, the block interprets the input as logic 0.

The threshold voltage is the voltage value at midpoint between the **High level input voltage** parameter value and the **Low level input voltage** parameter value.

Note To improve simulation speed, the block does not model all the internal individual MOSFET devices that make up the gate. See "Basic Assumptions and Limitations" on page 1-66 for details.

The block models the gate as follows:

- The gate inputs have infinite resistance and finite or zero capacitance.

- The gate output offers a selection of two models: `Linear` and `Quadratic`. For more information, see "Selecting the Output Model for Logic Blocks". Use the **Output current-voltage relationship** parameter to specify the output model.

- You can specify propagation delay for both output models. For `Linear` output, the block sets the value of the gate output capacitor such that the resistor-capacitor time constant equals the **Propagation delay** parameter value. For `Quadratic` output, the gate input demand is lagged to approximate the **Propagation delay** parameter value.

The block output voltage depends on the output model selected:

- For `Linear` model, output high is the **High level output voltage** parameter value, and output low is the **Low level output voltage** parameter value.

- For `Quadratic` model, the output voltage for High and Low states is a function of the output current, as explained in "Quadratic Model Output and Parameters". For zero load current, output high is Vcc (the **Supply voltage** parameter value), and output low is zero volts.

Basic Assumptions and Limitations

The block does not model the internal individual MOSFET devices that make up the gate (except for the final MOSFET pair if you select the `Quadratic` option for the **Output current-voltage relationship** parameter). This limitation has the following implications:

- The block does not accurately model the gate's response to input noise and inputs that are around the logic threshold voltage.

- The block does not accurately model dynamic response.

Circuits that involve a feedback path around a set of logic gates may require a nonzero propagation delay to be set on one or more gates.

Parameters

- "Inputs Tab" on page 1-67
- "Outputs Tab" on page 1-67
- "Initial Conditions Tab" on page 1-69

Inputs Tab

Low level input voltage

Voltage value below which the block interprets the input voltage as logic `LOW`. The default value is 2 V.

High level input voltage

Voltage value above which the block interprets the input voltage as logic `HIGH`. The default value is 3 V.

Average input capacitance

Fixed capacitance that approximates the input capacitance for a MOSFET gate. The MOSFET capacitance depends on the applied voltage. When you drive this block with another gate, the **Average input capacitance** produces a rise time similar to that of the MOSFET. You can usually find this capacitance value on a manufacturer datasheet. The default value is 5 pF. Setting this value to zero may result in faster simulation times.

Outputs Tab

Output current-voltage relationship

Select the output model, `Linear` or `Quadratic`. The default value is `Linear`.

Low level output voltage

Voltage value at the output when the output logic level is `LOW`. The default value is 0 V. This parameter is available when you select the `Linear` option for the **Output current-voltage relationship** parameter.

High level output voltage

Voltage value at the output when the output logic level is `HIGH`. The default value is 5 V. This parameter is available when you select the `Linear` option for the **Output current-voltage relationship** parameter.

Output resistance

Value of the series output resistor that is used to model the drop in output voltage resulting from the output current. The default value is 25 Ω. You can derive this value from a datasheet by dividing the high-level output voltage by the maximum low-level output current. This parameter is available when you select the `Linear` option for the **Output current-voltage relationship** parameter.

Supply voltage

Supply voltage value applied to the gate in your circuit. The default value is 5 V. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Measurement voltage

The gate supply voltage for which mask data output resistances and currents are defined. The default value is 5 V. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic HIGH output resistance at zero current and at I_OH

A row vector [*R_OH1 R_OH2*] of two resistance values. The first value *R_OH1* is the gradient of the output voltage-current relationship when the gate is logic HIGH and there is no output current. The second value *R_OH2* is the gradient of the output voltage-current relationship when the gate is logic HIGH and the output current is *I_OH*. The default value is [25 250] Ω. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic HIGH output current I_OH when shorted to ground

The resulting current when the gate is in the logic HIGH state, but the load forces the output voltage to zero. The default value is 63 mA. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic LOW output resistance at zero current and at I_OL

A row vector [*R_OL1 R_OL2*] of two resistance values. The first value *R_OL1* is the gradient of the output voltage-current relationship when the gate is logic LOW and there is no output current. The second value *R_OL2* is the gradient of the output voltage-current relationship when the gate is logic LOW and the output current is *I_OL*. The default value is [30 800] Ω. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic LOW output current I_OL when shorted to Vcc

The resulting current when the gate is in the logic LOW state, but the load forces the output voltage to the supply voltage Vcc. The default value is -45 mA. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Propagation delay

Time it takes for the output to swing from LOW to HIGH or HIGH to LOW after the input logic levels change. The default value is 25 ns.

Protection diode on resistance

The gradient of the voltage-current relationship for the protection diodes when forward biased. The default value is 5 Ω. This parameter is available when you select the Quadratic option for the **Output current-voltage relationship** parameter.

Protection diode forward voltage

The voltage above which the protection diode is turned on. The default value is 0.6 V. This parameter is available when you select the Quadratic option for the **Output current-voltage relationship** parameter.

Initial Conditions Tab

Output initial state

Specify whether the initial output state of the block is High or Low. This parameter is used for both linear and quadratic output states, provided that the **Propagation delay** parameter is greater than zero and the Solver Configuration block does not have the **Start simulation from steady state** option selected. The default value is Low.

Ports

The block has the following ports:

A

Electrical input port

J

Electrical output port

CMOS OR

Model CMOS OR gate behaviorally

Library

Logic

Description

The CMOS OR block represents a CMOS OR logic gate behaviorally:

- The block output logic level is HIGH if the logic levels of any of the gate inputs are 1.
- The block output logic level is LOW otherwise.

The block determines the logic levels of the gate inputs as follows:

- If the gate voltage is greater than the threshold voltage, the block interprets the input as logic 1.
- Otherwise, the block interprets the input as logic 0.

The threshold voltage is the voltage value at midpoint between the **High level input voltage** parameter value and the **Low level input voltage** parameter value.

Note To improve simulation speed, the block does not model all the internal individual MOSFET devices that make up the gate. See "Basic Assumptions and Limitations" on page 1-71 for details.

The block models the gate as follows:

- The gate inputs have infinite resistance and finite or zero capacitance.

- The gate output offers a selection of two models: `Linear` and `Quadratic`. For more information, see "Selecting the Output Model for Logic Blocks". Use the **Output current-voltage relationship** parameter to specify the output model.

- You can specify propagation delay for both output models. For `Linear` output, the block sets the value of the gate output capacitor such that the resistor-capacitor time constant equals the **Propagation delay** parameter value. For `Quadratic` output, the gate input demand is lagged to approximate the **Propagation delay** parameter value.

The block output voltage depends on the output model selected:

- For `Linear` model, output high is the **High level output voltage** parameter value, and output low is the **Low level output voltage** parameter value.

- For `Quadratic` model, the output voltage for High and Low states is a function of the output current, as explained in "Quadratic Model Output and Parameters". For zero load current, output high is Vcc (the **Supply voltage** parameter value), and output low is zero volts.

Basic Assumptions and Limitations

The block does not model the internal individual MOSFET devices that make up the gate (except for the final MOSFET pair if you select the `Quadratic` option for the **Output current-voltage relationship** parameter). This limitation has the following implications:

- The block does not accurately model the gate's response to input noise and inputs that are around the logic threshold voltage.

- The block does not accurately model dynamic response.

Circuits that involve a feedback path around a set of logic gates may require a nonzero propagation delay to be set on one or more gates.

Parameters

- "Inputs Tab" on page 1-72
- "Outputs Tab" on page 1-72
- "Initial Conditions Tab" on page 1-74

Inputs Tab

Low level input voltage

Voltage value below which the block interprets the input voltage as logic `LOW`. The default value is 2 V.

High level input voltage

Voltage value above which the block interprets the input voltage as logic `HIGH`. The default value is 3 V.

Average input capacitance

Fixed capacitance that approximates the input capacitance for a MOSFET gate. The MOSFET capacitance depends on the applied voltage. When you drive this block with another gate, the **Average input capacitance** produces a rise time similar to that of the MOSFET. You can usually find this capacitance value on a manufacturer datasheet. The default value is 5 pF. Setting this value to zero may result in faster simulation times.

Outputs Tab

Output current-voltage relationship

Select the output model, `Linear` or `Quadratic`. The default value is `Linear`.

Low level output voltage

Voltage value at the output when the output logic level is `LOW`. The default value is `0` V. This parameter is available when you select the `Linear` option for the **Output current-voltage relationship** parameter.

High level output voltage

Voltage value at the output when the output logic level is `HIGH`. The default value is `5` V. This parameter is available when you select the `Linear` option for the **Output current-voltage relationship** parameter.

Output resistance

Value of the series output resistor that is used to model the drop in output voltage resulting from the output current. The default value is 25 Ω. You can derive this value from a datasheet by dividing the high-level output voltage by the maximum low-level output current. This parameter is available when you select the `Linear` option for the **Output current-voltage relationship** parameter.

Supply voltage

Supply voltage value applied to the gate in your circuit. The default value is 5 V. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Measurement voltage

The gate supply voltage for which mask data output resistances and currents are defined. The default value is 5 V. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic HIGH output resistance at zero current and at I_OH

A row vector [*R_OH1 R_OH2*] of two resistance values. The first value *R_OH1* is the gradient of the output voltage-current relationship when the gate is logic HIGH and there is no output current. The second value *R_OH2* is the gradient of the output voltage-current relationship when the gate is logic HIGH and the output current is *I_OH*. The default value is [25 250] Ω. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic HIGH output current I_OH when shorted to ground

The resulting current when the gate is in the logic HIGH state, but the load forces the output voltage to zero. The default value is 63 mA. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic LOW output resistance at zero current and at I_OL

A row vector [*R_OL1 R_OL2*] of two resistance values. The first value *R_OL1* is the gradient of the output voltage-current relationship when the gate is logic LOW and there is no output current. The second value *R_OL2* is the gradient of the output voltage-current relationship when the gate is logic LOW and the output current is *I_OL*. The default value is [30 800] Ω. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic LOW output current I_OL when shorted to Vcc

The resulting current when the gate is in the logic LOW state, but the load forces the output voltage to the supply voltage Vcc. The default value is -45 mA. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Propagation delay

Time it takes for the output to swing from `LOW` to `HIGH` or `HIGH` to `LOW` after the input logic levels change. The default value is 25 ns.

Protection diode on resistance

The gradient of the voltage-current relationship for the protection diodes when forward biased. The default value is 5 Ω. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Protection diode forward voltage

The voltage above which the protection diode is turned on. The default value is `0.6` V. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Initial Conditions Tab

Output initial state

Specify whether the initial output state of the block is `High` or `Low`. This parameter is used for both linear and quadratic output states, provided that the **Propagation delay** parameter is greater than zero and the Solver Configuration block does not have the **Start simulation from steady state** option selected. The default value is `Low`.

Ports

The block has the following ports:

A

Electrical input port

B

Electrical input port

J

Electrical output port

CMOS XOR

Model CMOS XOR gate behaviorally

Library

Logic

Description

The CMOS XOR block represents a CMOS XOR logic gate behaviorally:

- The block output logic level is HIGH if the logic level of exactly one of the gate inputs is 1.
- The block output logic level is LOW otherwise.

The block determines the logic levels of the gate inputs as follows:

- If the gate voltage is greater than the threshold voltage, the block interprets the input as logic 1.
- Otherwise, the block interprets the input as logic 0.

The threshold voltage is the voltage value at midpoint between the **High level input voltage** parameter value and the **Low level input voltage** parameter value.

Note To improve simulation speed, the block does not model all the internal individual MOSFET devices that make up the gate. See "Basic Assumptions and Limitations" on page 1-76 for details.

The block models the gate as follows:

- The gate inputs have infinite resistance and finite or zero capacitance.

- The gate output offers a selection of two models: `Linear` and `Quadratic`. For more information, see "Selecting the Output Model for Logic Blocks". Use the **Output current-voltage relationship** parameter to specify the output model.

- You can specify propagation delay for both output models. For `Linear` output, the block sets the value of the gate output capacitor such that the resistor-capacitor time constant equals the **Propagation delay** parameter value. For `Quadratic` output, the gate input demand is lagged to approximate the **Propagation delay** parameter value.

The block output voltage depends on the output model selected:

- For `Linear` model, output high is the **High level output voltage** parameter value, and output low is the **Low level output voltage** parameter value.

- For `Quadratic` model, the output voltage for High and Low states is a function of the output current, as explained in "Quadratic Model Output and Parameters". For zero load current, output high is Vcc (the **Supply voltage** parameter value), and output low is zero volts.

Basic Assumptions and Limitations

The block does not model the internal individual MOSFET devices that make up the gate (except for the final MOSFET pair if you select the `Quadratic` option for the **Output current-voltage relationship** parameter). This limitation has the following implications:

- The block does not accurately model the gate's response to input noise and inputs that are around the logic threshold voltage.

- The block does not accurately model dynamic response.

Circuits that involve a feedback path around a set of logic gates may require a nonzero propagation delay to be set on one or more gates.

Parameters

Inputs Tab

Low level input voltage

Voltage value below which the block interprets the input voltage as logic `LOW`. The default value is 2 V.

High level input voltage

Voltage value above which the block interprets the input voltage as logic `HIGH`. The default value is 3 V.

Average input capacitance

Fixed capacitance that approximates the input capacitance for a MOSFET gate. The MOSFET capacitance depends on the applied voltage. When you drive this block with another gate, the **Average input capacitance** produces a rise time similar to that of the MOSFET. You can usually find this capacitance value on a manufacturer datasheet. The default value is 5 pF. Setting this value to zero may result in faster simulation times.

Outputs Tab

Output current-voltage relationship

Select the output model, `Linear` or `Quadratic`. The default value is `Linear`.

Low level output voltage

Voltage value at the output when the output logic level is `LOW`. The default value is `0` V. This parameter is available when you select the `Linear` option for the **Output current-voltage relationship** parameter.

High level output voltage

Voltage value at the output when the output logic level is `HIGH`. The default value is 5 V. This parameter is available when you select the `Linear` option for the **Output current-voltage relationship** parameter.

Output resistance

Value of the series output resistor that is used to model the drop in output voltage resulting from the output current. The default value is 25 Ω. You can derive this value from a datasheet by dividing the high-level output voltage by the maximum low-level output current. This parameter is available when you select the `Linear` option for the **Output current-voltage relationship** parameter.

Supply voltage

Supply voltage value applied to the gate in your circuit. The default value is 5 V. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Measurement voltage

The gate supply voltage for which mask data output resistances and currents are defined. The default value is 5 V. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic HIGH output resistance at zero current and at I_OH

A row vector [*R_OH1 R_OH2*] of two resistance values. The first value *R_OH1* is the gradient of the output voltage-current relationship when the gate is logic HIGH and there is no output current. The second value *R_OH2* is the gradient of the output voltage-current relationship when the gate is logic HIGH and the output current is *I_OH*. The default value is [25 250] Ω. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic HIGH output current I_OH when shorted to ground

The resulting current when the gate is in the logic HIGH state, but the load forces the output voltage to zero. The default value is 63 mA. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic LOW output resistance at zero current and at I_OL

A row vector [*R_OL1 R_OL2*] of two resistance values. The first value *R_OL1* is the gradient of the output voltage-current relationship when the gate is logic LOW and there is no output current. The second value *R_OL2* is the gradient of the output voltage-current relationship when the gate is logic LOW and the output current is *I_OL*. The default value is [30 800] Ω. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic LOW output current I_OL when shorted to Vcc

The resulting current when the gate is in the logic LOW state, but the load forces the output voltage to the supply voltage Vcc. The default value is -45 mA. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Propagation delay

> Time it takes for the output to swing from `LOW` to `HIGH` or `HIGH` to `LOW` after the input logic levels change. The default value is 25 ns.

Protection diode on resistance

> The gradient of the voltage-current relationship for the protection diodes when forward biased. The default value is 5 Ω. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Protection diode forward voltage

> The voltage above which the protection diode is turned on. The default value is `0.6` V. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Initial Conditions Tab

Output initial state

> Specify whether the initial output state of the block is `High` or `Low`. This parameter is used for both linear and quadratic output states, provided that the **Propagation delay** parameter is greater than zero and the Solver Configuration block does not have the **Start simulation from steady state** option selected. The default value is `Low`.

Ports

The block has the following ports:

A

> Electrical input port

B

> Electrical input port

J

> Electrical output port

Comparator

Model a comparator behaviorally

Library

Integrated Circuits

Description

The Comparator block is an abstracted behavioral model of a comparator integrated circuit. It does not model an internal transistor-level implementation. Therefore, the block runs quickly during simulation but retains the correct I/O behavior. The block models differential inputs electrically as having infinite resistance and a finite or zero capacitance.

The block models the gate output as a voltage source driving a series resistor and a capacitor that connects to ground. The output pin connects to the resistor-capacitor connection node. If the difference in the inputs is greater than the input threshold

voltage, then the output is equal to the **High level output voltage** (V_{OL}). Otherwise, the

output is equal to the **Low level output voltage** (V_{OH}).
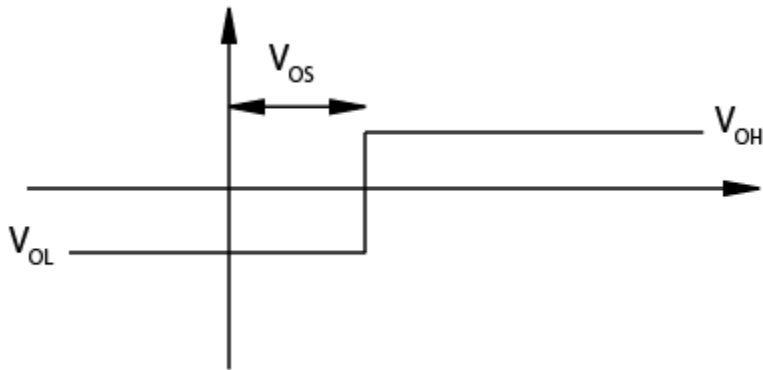
The output model is shown in the following illustration.
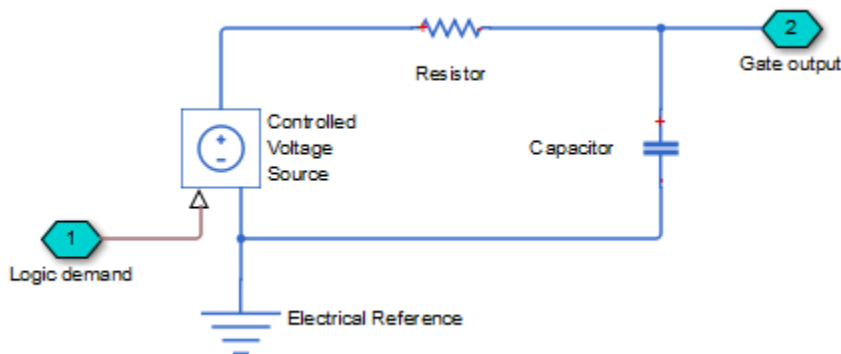
Basic Assumptions and Limitations

Modeling of the output as a controlled voltage source is representative of a totem-pole or push-pull output stage. To model a device with an open-collector:

1   Connect the output pin to the base of an NPN Bipolar Transistor or PNP Bipolar Transistor block.

2   Set the **Output resistance** parameter to a suitable value.

Parameters

- "Inputs Tab" on page 1-82
- "Outputs Tab" on page 1-82

Inputs Tab

Input offset voltage

The voltage which the difference in the input voltages must be greater than so that the comparator gives a logic output 1. The default value is 5 mV.

Average input capacitance

You can usually find this capacitance value on a manufacturer datasheet. The default value is 0 pF. Setting this value to zero can result in faster simulation times.

Outputs Tab

Low level output voltage

The steady-state output voltage, V_{OL}, when the voltage difference across the inputs is less than or equal to the threshold voltage, and the output current is zero. The default value is 0 V.

High level output voltage

The steady-state output voltage, V_{OH}, when the voltage difference across the inputs is greater than the threshold voltage, and the output current is zero. The default value is 5 V.

Output resistance

This parameter is the ratio of output voltage drop to output current. Set this parameter to $(V_{OH} - V_{OH1}) / I_{OH1}$, where V_{OH1} is the reduced output high voltage when the output current is I_{OH1}. The default value is 50 Ω.

Propagation delay

Set this value based on the high-to-low and low-to-high propagation delays. The default value is 0 s.

Ports

This block has the following ports:

+

    Positive electrical input port

-

    Negative electrical input port

OUT

    Electrical output port

See Also

CMOS Buffer

Controlled PWM Voltage

Pulse-width modulated voltage source model
Library: Simscape / Electronics / Actuators & Drivers / Drivers

Description

The Controlled PWM Voltage block represents a pulse-width modulated (PWM) voltage source. The block has two modeling variants, accessible by right-clicking the block in your block diagram and then selecting the appropriate option from the context menu, under **Simscape > Block choices**:

- **Electrical input ports** — The block calculates the duty cycle based on the reference voltage across its **ref+** and **ref-** ports. This modeling variant is the default.

- **Physical signal input** — Specify the duty cycle value directly by using an input physical signal port.

For the **Electrical input ports** variant of the block, the demanded duty cycle is

$$100 * \frac{V_{ref} - V_{\min}}{V_{\max} - V_{\min}} \, percent$$

where:

- V_{ref} is reference voltage across the **ref+** and **ref-** ports.
- $V_{\min}$ is the minimum reference voltage.
- $V_{\max}$ is the maximum reference voltage.

The value of the **Output voltage amplitude** parameter determines amplitude of the output voltage.

At time zero, the pulse is initialized as high, unless the **Pulse delay time** parameter is greater than zero, or the demanded duty cycle is zero.

You can use parameters **Pulse delay time** and **Pulse width offset** to add a small turn-on delay and a small turn-off advance. This can be useful when fine-tuning switching times so as to minimize switching losses.

In PWM mode, the block has two options for the type of switching event when moving between output high and output low states:

- `Asynchronous — Best for variable-step solvers` — Asynchronous events are better suited to variable step solvers, because they require fewer simulation steps for the same level of accuracy. In asynchronous mode the PWM switching events generate zero crossings, and therefore switching times are always determined accurately, regardless of the simulation maximum step size.

- `Discrete—time — Best for fixed-step solvers` — Discrete-time events are better suited to fixed-step operation, because then the switching events are always synchronized with the simulation step. Using an asynchronous implementation with fixed-step solvers may sometimes result in events being up to one simulation step late. For more information, see "Simulating with Fixed Time Step — Local and Global Fixed-Step Solvers" (Simscape).

If you use a fixed-step or local solver and the discrete-time switching event type, the following restrictions apply to the **Sample time** parameter value:

- The sample time must be a multiple of the simulation step size.
- The sample time must be small compared to the PWM period, to ensure sufficient resolution.

Basic Assumptions and Limitations

The model is based on the following assumptions:

- The REF output of this block is floating, it is not tied to the Electrical Reference. One consequence of this is that if you connect the PWM and REF electrical ports directly to the PWM and REF electrical ports of an H-bridge or a gate driver, you must attach an Electrical Reference block to the REF connection line.

- Do not connect the Controlled PWM block directly to a semiconductor gate, because this omits the gate driver output impedance that determines switching dynamics. Use a Gate Driver or a Half-Bridge Driver block to set the gate-source or the gate-emitter voltage.

- Do not use the Controlled PWM block to drive a motor block directly. A PWM motor driver goes open circuit in between pulses. Use the H-Bridge block to drive a motor block.

- When driving a motor via the H-Bridge block, set the **Simulation mode** parameter to `Averaged` to speed up simulations. You must also set the **Simulation mode** parameter of the H-Bridge block to `Averaged` mode. This applies the average of the demanded PWM voltage to the motor. The `Averaged` mode assumes that the impedance of the motor inductive term is small at the **PWM frequency**. To verify this assumption, run the simulation using the `PWM` mode and compare the results to those obtained from using the `Averaged` mode.

- If you are linearizing your model, set the **Simulation mode** parameter to `Averaged` and ensure that you have specified the operating point of the block correctly. You can only linearize the block for inputs corresponding to a duty cycle greater than zero and less than 100 percent.

- When you use this block in `PWM` mode with the **Use local solver option** selected in the Solver Configuration block, set the **Switching event type** parameter to `Discrete —time — Best for fixed-step solvers`. Using the `Asynchronous — Best for variable-step solvers` option in this situation may produce inaccuracies, because simulation with the local solver implies fixed step, and the PWM events will not always coincide precisely with the simulation steps. This results in PWM events sometimes occurring one simulation step late.

Ports

Input

u — Control signal, unitless
physical signal

Input physical signal that specifies the duty cycle.

Dependencies

Enabled for the **Physical signal input** variant of the block.

Conserving

`ref+` — Positive terminal
electrical

Positive electrical reference voltage.

Dependencies

Enabled for the **Electrical input ports** variant of the block.

`ref-` — Negative terminal
electrical

Negative electrical reference voltage

Dependencies

Enabled for the **Electrical input ports** variant of the block.

PWM — Pulse-width modulated signal
electrical

Electrical conserving port associated with the pulse-width modulated signal.

REF — Floating zero volt reference
electrical

Electrical conserving port associated with the floating zero volt reference.

Parameters

PWM

`PWM frequency` — PWM output signal frequency
1000 Hz (default)

Frequency of the PWM output signal.

`Pulse delay time` — Turn-on delay
0 s (default)

The pulse train does not start until the simulation time is equal to the **Pulse delay time**. You can specify a small value for **Pulse delay time** to fine-tune switching times and ensure that an off-going device is fully off before the on-going device starts to turn on. You can also use larger delay times, for example, if you need the pulse train to start only after a number of cycles. The value you provide must be greater than or equal to zero.

Dependencies

Enabled when the **Simulation mode** parameter is set to PWM.

Pulse width offset — Lengthens or shortens the pulse
0 s (default)

The demanded pulse width as defined by the product of the demanded duty cycle and one over the pulse frequency can be offset by the value you provide for **Pulse width offset**. A positive value acts to lengthen the pulse by a fixed amount. A negative value acts to shorten the pulse. You can use this parameter, along with the **Pulse delay time**, to fine-tune switching times so as to minimize switching losses in some circuits.

Dependencies

Enabled when the **Simulation mode** parameter is set to PWM.

Minimum pulse width — Minimum length of pulse
0 s (default)

The minimum pulse length, based on the internal clock or defined programmatically, to protect the device being driven. The value you provide must be greater than or equal to zero.

Dependencies

Enabled when the **Simulation mode** parameter is set to PWM.

Simulation mode — Select type of output voltage
PWM (default) | Averaged

Select the type of output voltage:

- PWM — Produces a pulse-width modulated signal.

- Averaged — The output is a constant whose value is equal to the average value of the PWM signal.

Switching event type — Select type of switching event
Asynchronous — Best for variable-step solvers (default) | Discrete-time — Best for fixed-step solvers

Select the switching event type when moving between output high and output low states:

- Asynchronous — Best for variable-step solvers — This option is more efficient for desktop simulation with variable-step solvers, because it requires fewer simulation steps for the same level of accuracy.
- Discrete-time — Best for fixed-step solvers — Use with fixed-step solvers, including the local solver. For more information, see "Simulating with Fixed Time Step — Local and Global Fixed-Step Solvers" (Simscape).

Dependencies

Enabled when the **Simulation mode** parameter is set to PWM.

Sample time — Discrete sampling time
1e-6 s (default)

The time between updates of the block output state. The sample time must be a multiple of the simulation step size. In order for the PWM control to have sufficient resolution, set the sample time to less than one hundredth of the PWM period. (The PWM period is one over the PWM frequency.)

Dependencies

Enabled when the **Switching event type** parameter is set to Discrete-time — Best for fixed-step solvers.

Input Scaling

Input voltage for 0% duty cycle — Vmin
0 V (default)

Value of the input voltage at which the PWM signal has a 0% duty cycle.

Dependencies

Enabled for the **Electrical input ports** variant of the block.

Input voltage for 100% duty cycle — Vmax
5 V (default)

Value of the input voltage at which the PWM signal has a 100% duty cycle.

Dependencies

Enabled for the **Electrical input ports** variant of the block.

`Input value for 0% duty cycle` — Minimum signal value
0 (default)

Value of the input signal at which the PWM signal has a 0% duty cycle.

Dependencies

Enabled for the **Physical signal input** variant of the block.

`Input value for 100% duty cycle` — Maximum signal value
1 (default)

Value of the input signal at which the PWM signal has a 100% duty cycle

Dependencies

Enabled for the **Physical signal input** variant of the block.

Output Voltage

`Output voltage amplitude` — Signal amplitude for high output
5 V (default)

Amplitude of the PWM signal when the output is high.

See Also
Gate Driver | H-Bridge | Half-Bridge Driver

Introduced in R2008b

Crystal

Model stable resonator

Library

Passive Devices

Description

The Crystal block represents the electrical characteristics of a crystal. The following figure shows the equivalent circuit model of the Crystal block.
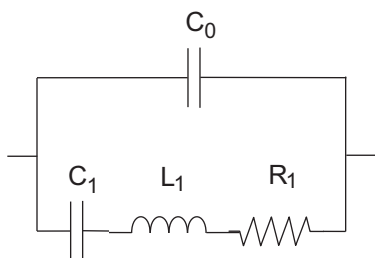
You specify the equivalent circuit parameters for this model when you set the **Parameterization** parameter to `Equivalent circuit parameters`.

- The capacitor $C0$ corresponds to the capacitance you specify in the **Shunt capacitance, C0** parameter.
- The capacitor $C1$ corresponds to the capacitance you specify in the **Motional capacitance, C1** parameter.
- The inductor $L1$ corresponds to the inductance you specify in the **Motional inductance, L1** parameter.

- The resistor *R1* corresponds to the resistance you specify in the **Equivalent series resistance, R1** parameter.

Most datasheets specify crystal frequency rather than inductance, so the block optionally accepts frequency data.

- When you set the **Parameterization** parameter to `Series resonance data`, the block uses the following relationship to calculate *L1* from the series resonant frequency:

$$f_s = \frac{1}{2\pi\sqrt{L_1 C_1}}$$

Where f_s is the **Series resonance, fs** parameter value.

- When you set the **Parameterization** parameter to `Parallel resonance data`, the block uses the following relationship to calculate *L1* from the parallel resonant frequency:

$$f_a = \frac{1}{2\pi\sqrt{L_1 C_1 (C_0 + C_L)/(C_1 + C_0 + C_L)}}$$

Where:

- f_a is the **Parallel resonance, fa** parameter value.
- C_L is the **Load capacitance, CL** parameter value.

Some datasheets specify quality factor rather than equivalent series resistance, so the block optionally accepts quality factor data. When you set the **R1 parameterization** parameter to `Quality factor Q`, the block uses the following relationship to calculate *R1* from the quality factor:

$$Q = \frac{2\pi f L_1}{R_1}$$

Where *Q* is the **Quality factor, Q** parameter value.

Note The **R1 parameterization** parameter is only visible when you select `Series resonance data` or `Parallel resonance data` for the **Parameterization** parameter.

Basic Assumptions and Limitations

The Crystal block models only the fundamental crystal vibration mode.

Parameters

Parameterization

Select one of the following methods for block parameterization:

- `Series resonance data` — Provide series resonant frequency and capacitance data for the crystal. This method is the default.
- `Parallel resonance data` — Provide parallel resonant frequency and capacitance data for the crystal.
- `Equivalent circuit parameters` — Provide electrical parameters for an equivalent circuit model of the crystal.

Series resonance, fs

Crystal series resonant frequency. This parameter is only visible when you select `Series resonance data` for the **Parameterization** parameter. The default value is 32.764 kHz.

Parallel resonance, fa

Crystal parallel resonant frequency that corresponds to operating with a parallel load capacitance specified by the **Load capacitance, CL** parameter. This parameter is only visible when you select `Parallel resonance data` for the **Parameterization** parameter. The default value is 32.768 kHz.

Motional inductance, L1

Inductance that represents the mechanical mass of the crystal. This parameter is only visible when you select `Equivalent circuit parameters` for the **Parameterization** parameter. The default value is 6.742e+03 H.

R1 parameterization

Select one of the following methods for series resistance parameterization:

- `Equivalent series resistance R1` — Provide the resistance value directly. This is the default method.
- `Quality factor Q` — Provide the quality factor that the block uses to calculate the resistance value.

This parameter is only visible when you select `Series resonance data` or `Parallel resonance data` for the **Parameterization** parameter.

Quality factor, Q

Crystal quality factor. This parameter is only visible when you make one of the following selections:

- `Series resonance data` for the **Parameterization** parameter and `Quality factor Q` for the **R1 parameterization** parameter
- `Parallel resonance data` for the **Parameterization** parameter and `Quality factor Q` for the **R1 parameterization** parameter

The default value is `9e+04`.

Equivalent series resistance, R1

Motional damping term. This parameter is only visible when you make one of the following selections:

- `Series resonance data` for the **Parameterization** parameter and `Equivalent series resistance R1` for the **R1 parameterization** parameter
- `Parallel resonance data` for the **Parameterization** parameter and `Equivalent series resistance R1` for the **R1 parameterization** parameter
- `Equivalent circuit parameters` for the **Parameterization** parameter

The default value is 15 kΩ.

Motional capacitance, C1

Capacitance that represents crystal mechanical stiffness under load. The default value is `0.0035` pF.

Load capacitance, CL

Load capacitance that corresponds to the **Parallel resonance, fa** parameter value. This parameter is only visible when you select `Parallel resonance data` for the **Parameterization** parameter. The default value is `12.5` pF.

Shunt capacitance, C0

Electrical capacitance between the two crystal electrical connections. The parameter value must be greater than zero. The default value is `1.6` pF.

Initial voltage

The output voltage at the start of the simulation when the output current is zero. The default value is `0` V.

Ports

The block has the following ports:

+

Positive electrical port

-

Negative electrical port

Current-Controlled Switch

Model current-controlled switch with hysteresis
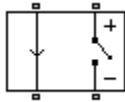
Library

SPICE-Compatible Components/Passive Devices

Description

The Current-Controlled Switch block represents the electrical characteristics of a switch whose state is controlled by the current through the input ports (the controlling current):

- When the controlling current is greater than the sum of the **Threshold current, IT** and **Hysteresis current, IH** parameter values, the switch is closed and has a resistance equal to the **On resistance, RON** parameter value.

- When the controlling current is less than the **Threshold current, IT** parameter value minus the **Hysteresis current, IH** parameter value, the switch is open and has a resistance equal to the **Off resistance, ROFF** parameter value.

- When the controlling current is greater than or less than the **Threshold current, IT** parameter value by an amount less than or equal to the **Hysteresis current, IH** parameter value, the current is in the crossover region and the state of the switch remains unchanged.

Basic Assumptions and Limitations

The block output resistance model is discontinuous during switching. The discontinuity might cause numerical issues. Try the following actions to resolve the issues:

- Set the **On resistance, RON** and **Off resistance, ROFF** parameter values to keep the ratio *RON/ROFF* as small as possible, and less than 1e+12.

- Increase the **Hysteresis current, IH** parameter value to reduce switch chatter.
- Decrease the **Max step size** parameter value (in the Configuration Parameters block dialog box).

Note This increases the simulation time.

Parameters

Threshold current, IT

The current above which the block interprets the controlling current as HIGH. The default value is 0 A.

Note The controlling current must differ from the threshold current by at least the **Hysteresis current, IH** parameter value to change the state of the switch.

Hysteresis current, IH

The amount by which the controlling current must exceed or fall below the **Threshold current, IT** parameter value to change the state of the switch. The default value is 0 A.

On resistance, RON

The resistance of the switch when it is closed. The default value is 1 Ω.

Off resistance, ROFF

The resistance of the switch when it is open. The default value is 1e+12 Ω.

Initial switch state

Select one of the following options for the state of the switch at the start of the simulation:

- On — The switch is initially closed and its resistance value is equal to the **On resistance, RON** parameter value. This is the default option.
- Off — The switch is initially open and its resistance value is equal to the **Off resistance, ROFF** parameter value.

Ports

The block has the following ports:

+

Positive electrical input and output ports.

-

Negative electrical input and output ports.

Current Limiter

Behavioral model of current limiter

Library

Semiconductor Devices

Description

The Current Limiter block provides a behavioral model of a current limiter. Use it to represent current limiting as found in power supplies and motor drives, and also to represent components that are used to limit inrush current.

The current limiting acts for both positive and negative currents. For applications where limiting is required in only one direction, you can augment the Current Limiter block with a series diode (blocks any reverse current) or parallel diode (no limiting in the reverse direction).

The block implements current limiting by using a hyperbolic tangent function:

$$i = i_{LIM} \tanh\left(\frac{4v}{v_{LIM}}\right) + g_{LIM}v$$

where:

- i is the current through the component.
- v is the voltage drop across the component.
- i_{LIM} is the current limit.
- v_{LIM} is the approximate voltage drop across the component when the current limit becomes active.

- g_{LIM} is the rate of change of current with voltage drop when on the current limit (limit-state conductance).

When $v = v_{\text{LIM}}$, then

$$i = i_{LIM} \tanh(4) + g_{LIM} v = 0.9993 i_{LIM} + g_{LIM} v$$

Therefore the current is approximately equal to the limit. Choose the value for g_{LIM} such that $g_{\text{LIM}} \cdot v$ is small compared to i_{LIM} for the maximum expected voltage drop. This term is included in the block equation to improve numerical properties during simulation.

When choosing the value of v_{LIM}, consider that making it too small will require tight solver tolerances and small step sizes. In practice, current limiters can be implemented using a MOSFET and series source resistor, the gate-source voltage being driven by the series resistor. This implementation does not produce a sharp limit, similar to the tanh curve used in this block. You can use a datasheet plot of current against voltage to pick a suitable value for v_{LIM}.

Thermal Port

The block has an optional thermal port, hidden by default. To expose the thermal port, right-click the block in your model, and then from the context menu select **Simscape > Block choices > Show thermal port**. This action displays the thermal port **H** on the block icon, and adds the **Thermal Port** tab to the block dialog box.

The thermal port model contains a thermal mass. The power dissipated by the current limiter, plus the heat flow into the thermal port, drives the thermal mass differential equation:

$$m \frac{dT}{dt} = P_{loss} + Q_H$$

where:

- m is the thermal mass.
- T is the thermal port temperature.
- P_{loss} is the electrical loss, $v \cdot i$.
- Q_{H} is the heat flow from the external network into the thermal port.

Parameters

- "Parameters Tab" on page 1-101
- "Thermal Port Tab" on page 1-101

Parameters Tab

Current limit

The maximum current magnitude. The default value is 1 A.

Voltage drop when current starts to limit

When the voltage drop is equal to this value, then the current is limited at 0.9993 times the current limit value. The default value is 0.1 V.

Limit-state conductance

When the current is limited, this parameter defines the rate of change of current with voltage drop if the current is driven harder onto the limit. The default value is 1e-3 1/Ω.

Thermal Port Tab

This tab appears only for blocks with exposed thermal ports. For more information, see "Thermal Port" on page 1-100.

Thermal mass

The heat energy required to raise the temperature by one degree. The default value is 100 J/K.

Initial temperature

The temperature at the start of simulation. The default value is 25 C.

Ports

The block has the following ports:

+

Electrical conserving port associated with the current limiter positive terminal

-

Electrical conserving port associated with the current limiter negative terminal

Introduced in R2015a

Current Source

Simulate current source with DC, AC, and noise components

Library

Sources

Description

The Current Source block implements a current source with DC, AC, and noise components. The current flowing through the source from the – terminal to the + terminal is given by:

$$i = i_{DC} + i_{AC} \sin(2\pi f t + \phi) + i_N$$

where:

- i_{DC} is the steady-state DC current component.
- i_{AC} is the amplitude of the AC current component.
- f is the frequency of the AC component.
- ϕ is the phase offset of the AC component.
- i_N is the noise current.

You can configure your source as DC-only, AC-only, or a combination of both. By default, both AC and DC components are set to 0. Define the AC/DC current by specifying nonzero parameter values after placing the block in your model.

The noise component is also optional. If you set the **Noise mode** parameter to `Enabled`, then the added noise current is given by:

$$i_N = \sqrt{P_i/2}\,\frac{N(0,1)}{\sqrt{h}}$$

where:

- P_i is the single-sided noise power spectral density for a 1 ohm load, in A^2/Hz.
- N is a Gaussian random number with zero mean and standard deviation of one.
- h is the sampling interval.

By default, the **Noise mode** parameter is set to `Disabled`, and the current source generates no thermal noise.

Noise Options

The block generates Gaussian noise by using the Random Number source in the Simscape Foundation library. You can control the random number seed by setting the **Repeatability** parameter:

- `Not repeatable` — Every time you simulate your model, the block resets the random seed using the MATLAB random number generator:

  `seed = randi(2^32-1);`

- `Repeatable` — The block automatically generates a seed value and stores it inside the block, to always start the simulation with the same random number. This auto-generated seed value is set when you add a Current Source block from the block library to the model. When you make a new copy of the Current Source block from an existing one in a model, a new seed value is generated. The block sets the value using the MATLAB random number generator command shown above.

- `Specify seed` — If you select this option, the additional **Seed** parameter lets you directly specify the random number seed value.

Basic Assumptions and Limitations

Simulating with noise enabled slows down simulation. Choose the sample time (h) so that noise is generated only at frequencies of interest, and not higher.

Parameters

- "DC & AC Components Tab" on page 1-105
- "Noise Tab" on page 1-105

DC & AC Components Tab

DC current

The DC component of the output current. The default value is 0 A. Enter a nonzero value to add a DC component to the current source.

AC current peak amplitude

Amplitude of the AC component of the output current. The default value is 0 A. Enter a nonzero value to add an AC component to the current source.

AC current phase shift

Phase offset of the AC component of the output current. The default value is 0 degrees.

AC current frequency

Frequency of the AC component of the output current. The default value is 60 Hz.

Noise Tab

Noise mode

Select the noise option:

- Disabled — No noise is produced by the current source. This is the default.
- Enabled — The current source generates thermal noise, and the associated parameters become visible on the **Noise** tab.

Power spectral density

The single-sided spectrum noise power. Strictly-speaking, this is a density function for the square of the current, commonly thought of as a power into a 1 ohm load, and therefore units are A^2/Hz. To avoid this unit ambiguity, some datasheets quote noise current as a noise density with units of A/√Hz. In this case, you should enter the square of the noise density quoted in the datasheet as the parameter value. The default value is 0 A^2/Hz.

Sample time

Defines the rate at which the noise source is sampled. Choose it to reflect the frequencies of interest in your model. Making the sample time too small will unnecessarily slow down your simulation. The default value is `1e-3` s.

Repeatability

Select the noise control option:

- `Not repeatable` — The random sequence used for noise generation is not repeatable. This is the default.
- `Repeatable` — The random sequence used for noise generation is repeatable, with a system-generated seed.
- `Specify seed` — The random sequence used for noise generation is repeatable, and you control the seed by using the **Seed** parameter.

Auto-generated seed used for repeatable option

Random number seed stored inside the block to make the random sequence repeatable. This parameter is visible only if you select `Repeatable` for the **Repeatability** parameter. The parameter value is automatically generated using the MATLAB random number generator command. You can modify this parameter value, but it gets overwritten by a new random value if you copy the block to another block in the model. Therefore, if you want to control the seed of the random sequence, use the `Specify seed` option for the **Repeatability** parameter and specify the desired seed value using the **Seed** parameter.

Seed

Random number seed used by the noise random number generator. This parameter is visible only if you select `Specify seed` for the **Repeatability** parameter. The default value is `0`.

Ports

The block has the following ports:

+

Positive electrical port

-

Negative electrical port

See Also

Resistor | Voltage Source

DC Current Source

Model constant current source

Library

SPICE-Compatible Components/Sources

Description

The DC Current Source block represents a constant current source whose output current value is independent of the voltage across its terminals.

The block uses a small conductance internally to prevent numerical simulation issues. The conductance connects the + and - ports of the device and has a conductance *GMIN*:

- By default, *GMIN* matches the **Minimum conductance GMIN** parameter of the SPICE Environment Parameters block, whose default value is 1e–12.
- To change *GMIN*, add a SPICE Environment Parameters block to your model and set the **Minimum conductance GMIN** parameter to the desired value.

Parameters

Constant value, DC

The value of the DC output current. The default value is 0 A.

Ports

The block has the following ports:

+

    Positive electrical voltage.

-

    Negative electrical voltage.

See Also

DC Voltage Source

DC Motor

DC motor model with electrical and torque characteristics
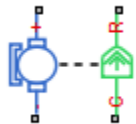
Library

Rotational Actuators

Description

The DC Motor block represents the electrical and torque characteristics of a DC motor using the following equivalent circuit model:
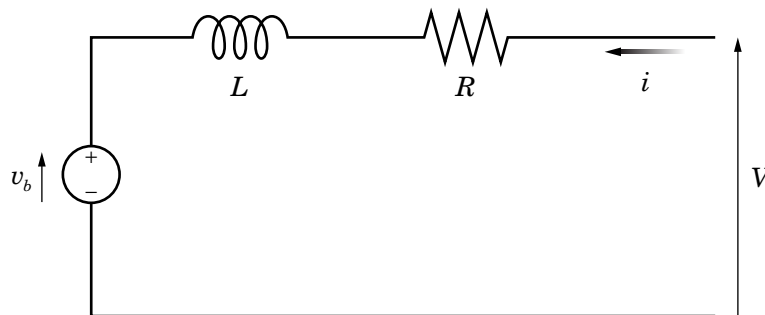
You specify the equivalent circuit parameters for this model when you set the **Model parameterization** parameter to By equivalent circuit parameters. The resistor R corresponds to the resistance you specify in the **Armature resistance** parameter. The inductor L corresponds to the inductance you specify in the **Armature inductance** parameter. The permanent magnets in the motor induce the following back emf v_b in the armature:

$$v_b = k_v \omega$$

where k_v is the **Back-emf constant** and ω is the angular velocity. The motor produces the following torque, which is proportional to the motor current i:

$$T_E = k_t i$$

where k_t is the **Torque constant**. The DC Motor block assumes that there are no electromagnetic losses. This means that mechanical power is equal to the electrical power dissipated by the back emf in the armature. Equating these two terms gives:

$$T_E \omega = v_b i$$
$$k_t i \omega = k_v \omega i$$
$$k_v = k_t$$

As a result, you specify either k_v or k_t in the block dialog box.

The torque-speed characteristic for the DC Motor block is related to the parameters in the preceding figure. When you set the **Model parameterization** parameter to By stall torque & no-load speed or By rated power, rated speed & no-load speed, the block solves for the equivalent circuit parameters as follows:

1   For the steady-state torque-speed relationship, L has no effect.

2   Sum the voltages around the loop and rearrange for i:

$$i = \frac{V - v_b}{R} = \frac{V - k_v \omega}{R}$$

3   Substitute this value of i into the equation for torque:

$$T_E = \frac{k_t}{R}(V - k_v \omega)$$

When you set the **Model parameterization** parameter to By stall torque & no-load speed, the block uses the preceding equation to determine values for R and k_t (and equivalently k_v).

When you set the **Model parameterization** parameter to `By rated power, rated speed & no-load speed`, the block uses the rated speed and power to calculate the rated torque. The block uses the rated torque and no-load speed values in the preceding equation to determine values for R and k_t.

The block models motor inertia J and damping λ for all values of the **Model parameterization** parameter. The resulting torque across the block is:

$$T = \frac{k_t}{R}\left(V - k_v\omega\right) - J\dot{\omega} - \lambda\omega$$

It is not always possible to measure rotor damping, and rotor damping is not always provided on a manufacturer datasheet. An alternative is to use the no-load current to infer a value for rotor damping.

For no-load, the electrically-generated mechanical torque must equal the rotor damping torque:

$$k_t i_{noload} = \lambda\omega_{noload}$$

where i_{noload} is the no-load current. If you select `By no-load current` for the **Rotor damping parameterization** parameter, then this equation is used in addition to the torque-speed equation to determine values for λ and the other equation coefficients.

The value for rotor damping, whether specified directly or in terms of no-load current, is taken into account when determining equivalent circuit parameters for **Model parameterization** options `By stall torque and no-load speed` and `By rated power, rated speed and no-load speed`.

When a positive current flows from the electrical + to - ports, a positive torque acts from the mechanical C to R ports.

Thermal Port

The block has an optional thermal port, hidden by default. To expose the thermal port, right-click the block in your model, and then from the context menu select **Simscape > Block choices > Show thermal port**. This action displays the thermal port H on the block icon, and adds the **Temperature Dependence** and **Thermal Port** tabs to the block dialog box.

Use the thermal port to simulate the effects of copper resistance losses that convert electrical power to heat. For more information on using thermal ports and on the **Temperature Dependence** and **Thermal Port** tab parameters, see "Simulating Thermal Effects in Rotational and Translational Actuators".

Parameters

- "Electrical Torque Tab" on page 1-113
- "Mechanical Tab" on page 1-115

Electrical Torque Tab

Model parameterization

   Select one of the following methods for block parameterization:

   - `By equivalent circuit parameters` — Provide electrical parameters for an equivalent circuit model of the motor. This is the default method.
   - `By stall torque & no-load speed` — Provide torque and speed parameters that the block converts to an equivalent circuit model of the motor.
   - `By rated power, rated speed & no-load speed` — Provide power and speed parameters that the block converts to an equivalent circuit model of the motor.

Armature resistance

   Resistance of the conducting portion of the motor. This parameter is only visible when you select `By equivalent circuit parameters` for the **Model parameterization** parameter. The default value is $3.9\ \Omega$.

Armature inductance

   Inductance of the conducting portion of the motor. If you do not have information about this inductance, set the value of this parameter to a small, nonzero number. The default value is `1.2e-05` H.

Define back-emf or torque constant

   Indicate whether you will specify the motor's back-emf constant or torque constant. When you specify them in SI units, these constants have the same value, so you only specify one or the other in the block dialog box. This parameter is only visible when you select `By equivalent circuit parameters` for the **Model parameterization** parameter. The default value is `Specify back-emf constant`.

Back-emf constant

The ratio of the voltage generated by the motor to the speed at which the motor is spinning. The default value is `7.2e-05` V/rpm. This parameter is only visible when you select `Specify back-emf constant` for the **Define back-emf or torque constant** parameter.

Torque constant

The ratio of the torque generated by the motor to the current delivered to it. This parameter is only visible when you select `Specify torque constant` for the **Define back-emf or torque constant** parameter. The default value is `6.876e-04` N*m/A.

Stall torque

The amount of torque generated by the motor when the speed is approximately zero. This parameter is only visible when you select `By stall torque & no-load speed` for the **Model parameterization** parameter. The default value is `2.4e-04` N*m.

No-load speed

Speed of the motor when not driving a load. This parameter is only visible when you select `By stall torque & no-load speed` or `By rated power, rated speed & no-load speed` for the **Model parameterization** parameter. The default value is `1.91e+04` rpm.

Rated speed (at rated load)

Motor speed at the rated mechanical power level. This parameter is only visible when you select `By rated power, rated speed & no-load speed` for the **Model parameterization** parameter. The default value is `1.5e+04` rpm.

Rated load (mechanical power)

The mechanical power the motor is designed to deliver at the rated speed. This parameter is only visible when you select `By rated power, rated speed & no-load speed` for the **Model parameterization** parameter. The default value is `0.08` W.

Rated DC supply voltage

The voltage at which the motor is rated to operate. This parameter is only visible when you select `By stall torque & no-load speed` or `By rated power, rated speed & no-load speed` for the **Model parameterization** parameter. The default value is `1.5` V.

Rotor damping parameterization

Select one of the following methods to specify rotor damping:

- By damping value — Specify a value for rotor damping directly, by using the **Rotor damping** parameter on the **Mechanical** tab. This is the default.
- By no-load current — The block calculates rotor damping based on the values that you specify for the **No-load current** and **DC supply voltage when measuring no-load current** parameters. If you select this option, the **Rotor damping** parameter is not available on the **Mechanical** tab.

No-load current

Specify the no-load current value, to be used for calculating the rotor damping. This parameter is only visible when you select By no-load current for the **Rotor damping parameterization** parameter. The default value is 0 A.

DC supply voltage when measuring no-load current

Specify the DC supply voltage corresponding to the no-load current value, to be used for calculating the rotor damping. This parameter is only visible when you select By no-load current for the **Rotor damping parameterization** parameter. The default value is 1.5 V.

Mechanical Tab

Rotor inertia

Resistance of the rotor to change in motor motion. The default value is 0.01 g*cm^2. The value can be zero.

Rotor damping

Energy dissipated by the rotor. This parameter is only visible when you select By damping value for the **Rotor damping parameterization** parameter on the **Electrical** tab. The default value is 1e-08 N*m/(rad/s). The value can be zero.

Initial rotor speed

Speed of the rotor at the start of the simulation. The default value is 0 rpm.

Ports

The block has the following ports:

+

Positive electrical input

-

    Negative electrical input

C

    Mechanical rotational conserving port

R

    Mechanical rotational conserving port

Examples

See the following DC motor examples:

- Brushless DC Motor
- Linear Electrical Actuator (Motor Model)
- Linear Electrical Actuator with Control

References

[1] Bolton, W. Mechatronics: Electronic Control Systems in Mechanical and Electrical Engineering, 3rd edition Pearson Education, 2004.

See Also

Induction Motor | Servomotor | Shunt Motor | Universal Motor

DC Voltage Source

Model constant voltage source

Library

SPICE-Compatible Components/Sources

Description

The DC Voltage Source block represents a constant voltage source whose output voltage value is independent of the current through the source.

Parameters

Constant value, DC

The value of the DC output voltage. The default value is 0 V.

Ports

The block has the following ports:

+

Positive electrical voltage.

-

Negative electrical voltage.

1-117

See Also

DC Current Source

DC-DC Converter

Behavioral model of power converter
Library: Simscape / Simscape Electronics / Sources
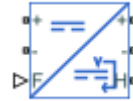
Description

The DC-DC Converter block represents a behavioral model of a power converter. This power converter regulates voltage on the load side. To balance input power, output power, and losses, the required amount of power is drawn from the supply side. Alternatively, the converter can support regenerative power flow from load to supply.

This circuit illustrates the converter's behavior.
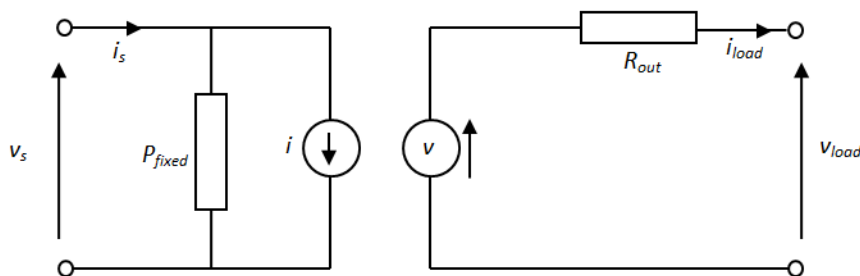
The P_{fixed} component draws a constant power and corresponds to converter losses that are independent of load current. The power drawn is set by the **Fixed converter losses independent of loading** parameter value. The resistor R_{out} corresponds to losses that increase with load current, and is determined from the value you specify for the **Percentage efficiency at rated output power** parameter.

The voltage source is defined by the following equation:

$$v = v_{ref} - i_{load} D + i_{load} R_{out}$$

Where:

- v_{ref} is the load side voltage set point, as defined by the value you specify for the **Output voltage reference demand** parameter.
- D is the value you specify for the **Output voltage droop with output current** parameter. Having a separate value for droop makes control of how output voltage varies with load independent of load-dependent losses. Instead of specifying D directly, you can specify the **Percent voltage droop at rated load**.

The current source value i is calculated so that the power flowing in to the converter equals the sum of the power flowing out plus the converter losses.

To specify the converter behavior when the voltage presented by the load is higher than the converter output voltage reference demand, use the **Power direction** parameter:

- `Unidirectional power flow from supply to regulated side` — Current is blocked by the off-state diode, and the current source current i is zero. Set the conductance of this diode using the **Diode off-state conductance** parameter.
- `Bidirectional power flow` — Power is transmitted to the supply side, and i becomes negative.

Optionally, the block can include voltage regulation dynamics. If you select `Specify voltage regulation time constant` for the **Dynamics** parameter, then a first-order lag is added to the equation defining the voltage source value. With the dynamics enabled, a load step change results in a transient change in output voltage, the time constant being defined by the **Voltage regulation time constant** parameter.

Simulating Faults

You can use the physical signal input port F to simulate both DC supply failure and converter failure. This type of event cannot be simulated by simply disconnecting the DC supply, for example by opening a switch, because the average value model will attempt to increase supply-side current to unrealistic values as supply-side voltage drops.

You control the behavior in response to the physical signal fault input F by the parameters on the **Faults** tab of the block dialog box. With the default parameter settings:

- **Fault condition** is `Output open circuit if F >= Fault threshold`
- **Fault threshold** is `0.5`

You can leave the input F unconnected and the converter will work normally.

If a signal is connected to port F, then the block operates according to the parameter settings on the **Faults** tab. For example, if **Fault condition** is `Output open circuit if F >= Fault threshold`, then when the signal at port F rises above the **Fault threshold** value, the converter stops operating. Zero current is taken from the supply side, and zero current is supplied to the load side.

Modeling thermal effects

The block has an optional thermal port, hidden by default. To expose the thermal port, right-click the block in your model, and then from the context menu select **Simscape > Block choices > Show thermal port**. This action displays the thermal port **H** on the block icon, and adds the **Thermal Port** tab to the block dialog box.

The block transfers heat generated from electrical losses through a Controlled Heat Flow Rate Source to a Thermal Mass block. The electrical properties of the block do not change with temperature. Specify the thermal properties for this block using the parameters **Thermal mass** and **Initial temperature**.

Assumptions

- The two electrical networks connected to the supply-side and regulated-side terminals must each have their own Electrical Reference block.

- The supply-side equation defines a power constraint on the product of the voltage (v_s) and the current (i_s). For simulation, the solver must be able to uniquely determine v_s. To ensure that the solution is unique, the block implements two assertions:

  - $v_s > 0$ — This assertion ensures that the sign of v_s is uniquely defined
  - $i_s < i_{max}$ — This assertion deals with the case when the voltage supply to the block has a series resistance

  When there is a series resistance, there are two possible steady-state solutions for i_s that satisfy the power constraint, the one with the smaller magnitude being the desired one. You should set the value for the **Maximum expected supply-side current** parameter (i_{max}) to be larger than the expected maximum current. This will ensure that when the model is initialized the initial current does not start at the undesired solution.

Ports

Conserving

1+ — Input positive terminal
electrical

Electrical conserving port associated with the positive terminal of the input side.

1- — Input negative terminal
electrical

Electrical conserving port associated with the negative terminal of the input side.

2+ — Output positive terminal
electrical

Electrical conserving port associated with the positive terminal of the output side.

2- — Output negative terminal
electrical

Electrical conserving port associated with the negative terminal of the output side.

F — Fault signal
physical

Physical signal input port that provides the external fault trigger signal. You can leave this port unconnected if the parameters on the **Faults** tab of the block dialog box are set to their default values.

H — Thermal mass
thermal

Thermal conserving port that represents the thermal mass. When you expose this port, provide additional parameters to define battery behavior at a second temperature. For more information, see the "Thermal" on page 1-126 parameters.

Dependencies

To expose this port, right-click the block and select **Simscape** > **Block choices** > **Show thermal port**.

Parameters

Main

Output voltage reference demand — Voltage set point
`10 V` (default)

The set point for the voltage regulator, and the output voltage value when there is no output current.

Rated output power — Rated power
`10 W` (default)

Output power for which the percentage efficiency value is given. This parameter is also used to calculate droop, D, if droop is specified as a percentage.

Droop parameterization — Droop model
`By voltage droop with output current` (default) | `By percent voltage droop at rated load`

Select one of the following methods for droop parameterization:

- `By voltage droop with output current` — Specify the absolute value of droop, D. This is the default option.
- `By percent voltage droop at rated load` — Specify droop, D, as a percentage at rated load.

Output voltage droop with output current — Voltage droop at 1 A
`0.1 V/A` (default)

The number of volts that the output voltage will drop from the set point for an output current of 1 A.

Dependencies

This parameter is visible only if you select `By voltage droop with output current` for the **Droop parameterization** parameter.

Percent voltage droop at rated load — Voltage droop at rated load
`2` (default)

The percentage by which voltage drops compared to the nominal output voltage when supplying the rated load.

Dependencies

This parameter is visible only if you select `By percent voltage droop at rated load` for the **Droop parameterization** parameter.

Power direction — Power flow direction
`Unidirectional power flow from supply to regulated side` (default) |
`Bidirectional power flow`

Select one of the following methods for the direction of power conversion:

- `Unidirectional power flow from supply to regulated side` — Most small power regulators are unidirectional. This is the default option.
- `Bidirectional power flow` — Larger power converters can be bidirectional, for example, converters used in electric vehicles to allow regenerative braking.

Diode off-state conductance — Unidirectional diode
`1e-8 1/Ohm` (default)

Ideal diode incorporated on the output side to prevent current from being forced into the converter in the unidirectional configuration.

Maximum expected supply-side current — Maximum supply current
`2 A` (default)

Set this value to a value greater than the maximum expected supply-side current in your model. Using twice the expected maximum current is generally sufficient. For more information, see "Assumptions" on page 1-121.

Losses

Percentage efficiency at rated output power — Rated efficiency
`80` (default)

Efficiency as defined by 100 times the output load power divided by the input supply power.

Fixed converter losses independent of loading — Constant losses
`1 W` (default)

Power drawn by the P_{fixed} component in the equivalent circuit diagram, which corresponds to converter losses that are independent of load current.

Dynamics

Dynamics — Dynamics model
No dynamics (default) | Specify voltage regulation time constant

Specify whether to include voltage regulation dynamics:

- No dynamics — Do not consider the voltage regulation dynamics.
- Specify voltage regulation time constant — Add a first-order lag to the equation defining the voltage source value. With the dynamics enabled, a load step change results in a transient change in output voltage.

Voltage regulation time constant — Dynamics time constant
0.02 s (default)

Time constant associated with voltage transients when the load current is stepped.

Dependencies

This parameter is only visible when you select Specify voltage regulation time constant for the **Dynamics** parameter.

Initial output voltage demand — Initial demand voltage
10 V (default)

Value of v_{ref} at time zero. Normally, v_{ref} is defined by the **Output voltage reference demand** parameter. However, if you want to initialize the model with no transients when delivering a steady-state load current, you can set the initial v_{ref} value by using this parameter, and increase it accordingly to take account of output resistance and droop.

Dependencies

This parameter is only visible when you select Specify voltage regulation time constant for the **Dynamics** parameter.

Faults

Fault condition — Fault model
`Output open circuit if F >= Fault threshold` (default) | `Output open circuit if F <= Fault threshold`

Selects whether the converter is disabled by a signal that is high or low:

- `Output open circuit if F >= Fault threshold` — Converter is disabled if the signal at port F rises above the threshold value. This is the default option.
- `Output open circuit if F <= Fault threshold` — Converter is disabled if the signal at port F falls below the threshold value.

Fault threshold — Fault limit
`0.5` (default)

Threshold value used to detect a fault.

Thermal

Thermal mass — Thermal mass associated with the thermal port
`100 J/K` (default)

Thermal mass associated with thermal port **H**. It represents the energy required to raise the temperature of the thermal port by one degree.

Initial temperature — Initial temperature at the thermal port
`298.15 K` (default)

Initial temperature associated with thermal port **H**.

See Also
Current Source | Voltage Source

Introduced in R2012b

Diffusion Resistor

Resistor model including velocity saturation, tolerance, operational limits, fault behavior, and noise
Library: Simscape / Electronics / Passive Devices

Description

The Diffusion Resistor block represents a resistor with velocity saturation, while letting you model the following effects:

- "Tolerances" on page 1-129
- "Operating Limits" on page 1-130
- "Faults" on page 1-130
- "Thermal Noise" on page 1-130
- "Thermal Port" on page 1-131

You can turn these modeling options on and off independently of each other.

In its simplest form, the resistance of the Diffusion Resistor block is:

$$R = R_0 \left(1 - p_2 - p_3 + p_2 \sqrt{1 + \left(\theta_2 v_{pn} \right)^2} + p_3 \sqrt[3]{1 + \left| \theta_3 v_{pn} \right|^3} \right)$$

where:

- R_0 is zero-bias resistance.
- p_2 and p_3 are the quadratic and linear voltage coefficients, respectively.
- θ_2 and θ_3 are inverse voltages for quadratic and linear voltage activation, respectively.
- v_{pn} is applied voltage across the resistor.

At low bias,

$$R \approx R_0 \left(1 + \frac{p_2 \theta_2^2 v_{pn}^2}{2} \right)$$

and therefore p_2 and θ_2 determine the low-bias quadratic behavior of the resistor.

At high bias,

$$R \approx R_0 \left(1 - p_2 - p_3 + |v_{pn}| \left(p_2 \theta_2 + p_3 \theta_3 \right) \right)$$

and therefore p_3 and θ_3 impact only the high-bias linear behavior of the resistor.

You can use the voltage-dependence of the resistance to model velocity saturation in a diffused resistor. For sufficiently high voltage,

$$i_{sat} = \frac{1}{R_0 \left(p_2 \theta_2 + p_3 \theta_3 \right)}$$

where i_{sat} is saturation current.

Simplified Parameterization

The simplified parameterization model assumes that the quadratic and linear coefficients are the same. This is one of the recommended assumptions for the r2_cmc model, as a reasonable initial guess when performing parameter extraction (see the r2_cmc documentation at https://projects.si2.org/cmc_index.php). With this assumption, it is possible to define two new parameters, **Critical voltage** and **Corner voltage**, which provide a simpler means for parameterizing models:

$$p_2 = p_3 = \frac{v_{co}}{2 v_{crit}}$$

$$\theta_2 = \theta_3 = \frac{1}{2 v_{co}}$$

where:

- v_{crit} is critical voltage.

- v_{co} is corner voltage.

At high voltage,

$$\frac{dR}{dv_{pn}} \approx \frac{R_0}{v_{crit}}$$

and therefore, critical voltage is the reciprocal of the slope of the increase of R/R_0 with voltage.

With this parameterization, the saturation current is

$$i_{sat} = \frac{v_{crit}}{R_0}$$

Tolerances

You can apply tolerances to the nominal value you provide for the **Resistance** parameter. Datasheets typically provide a tolerance percentage for a given resistor type. The table shows how the block applies tolerances and calculates resistance based on the selected **Tolerance application** option.

Option	Resistance Value
`None — use nominal value`	R_0
`Random tolerance`	Uniform distribution: $R_0 \cdot (1 - tol + 2 \cdot tol \cdot$ `rand`) Gaussian distribution: $R_0 \cdot (1 + tol \cdot$ `randn` $/ nSigma)$
`Apply maximum tolerance value`	$R_0 \cdot (1 + tol)$
`Apply minimum tolerance value`	$R_0 \cdot (1 - tol)$

In the table,

- R_0 is the **Resistance** parameter value, nominal zero-bias resistance.
- tol is fractional tolerance, **Tolerance (%)** /100.
- $nSigma$ is the value you provide for the **Number of standard deviations for quoted tolerance** parameter.

- `rand` and `randn` are standard MATLAB functions for generating uniform and normal distribution random numbers.

Operating Limits

You can specify operating limits in terms of power and maximum working voltage. For the thermal variant of the block (see "Thermal Port" on page 1-131), you can also specify operating limits in terms of temperature.

When an operating limit is exceeded, the block can either generate a warning or stop the simulation with an error. For more information, see the "Operating Limits" on page 1-136 parameters section.

Faults

The Diffusion Resistor block allows you to model an electrical fault as an instantaneous change in resistance. The block can trigger fault events:

- At a specific time
- When a current limit is exceeded for longer than a specific time interval

You can enable or disable these trigger mechanisms separately, or use them together if more than one trigger mechanism is required in a simulation. When more than one mechanism is enabled, the first mechanism to trigger the fault takes precedence. In other words, component fails no more than once per simulation.

When the resistor fails, its resistance is changed to the value you specify for the **Faulted zero-voltage resistance** parameter. You can also choose whether to issue an assertion when a fault occurs, by using the **Reporting when a fault occurs** parameter. The assertion can take the form of a warning or an error. By default, the block does not issue an assertion.

Thermal Noise

The Diffusion Resistor block can generate thermal noise current. If you set the **Noise mode** parameter to `Enabled`, then the block includes a noise current source connected in parallel to the diffusion resistor.

If the sampling time is h, then the thermal noise is given by:

$$i_N = \sqrt{2kT/R} \frac{N(0,1)}{\sqrt{h}}$$

where:

- k is the Boltzmann constant, 1.3806504e-23 J/K.
- T is temperature.
- R is resistance.
- N is a Gaussian random number with zero mean and standard deviation of one.
- $2kT/R$ is the double-sided thermal noise power distribution (the single-sided equivalent is $4kT/R$).

The block generates Gaussian noise by using the Random Number source in the Simscape Foundation library. You can control the random number seed by setting the **Repeatability** parameter:

- `Not repeatable` — Every time you simulate your model, the block resets the random seed using the MATLAB random number generator:

  `seed = randi(2^32-1);`
- `Repeatable` — The block automatically generates a seed value and stores it inside the block, to always start the simulation with the same random number. This auto-generated seed value is set when you add a Diffusion Resistor block from the block library to the model. When you make a new copy of the Diffusion Resistor block from an existing one in a model, a new seed value is generated. The block sets the value using the MATLAB random number generator command shown above.
- `Specify seed` — If you select this option, the additional **Seed** parameter lets you directly specify the random number seed value.

Thermal Port

The block has an optional thermal port, hidden by default. To expose the thermal port, right-click the block in your model, and then from the context menu select **Simscape > Block choices > Show thermal port**. This action displays the thermal port H on the block icon, and adds the **Thermal** tab and the **Variables** tab to the block dialog box.

Use the **Thermal** tab to specify how the resistance value changes with temperature and to set the thermal mass. Use the **Variables** tab to set the initial temperature target.

For the thermal variant, the defining equation for the resistance is augmented with additional temperature scaling:

$$R = R_0 \left(1 + T_{C1}^{eff} \Delta T + T_{C2}^{eff} \left(\Delta T \right)^2 \right) \left(1 - p_2 - p_3 + p_2 \sqrt{1 + \left(\theta_2 v_{pn} \right)^2} + p_3 \sqrt[3]{1 + \left| \theta_3 v_{pn} \right|^3} \right)$$

where T_{C1}^{eff} and T_{C2}^{eff} are the linear and quadratic temperature scaling coefficients, respectively.

$$\Delta T = T_{sim} - T_{meas}$$

where:

- T_{sim} is simulation temperature.
- T_{meas} is measurement temperature.

With the thermal port exposed, the generated noise uses the temperature at the thermal port when determining the instantaneous noise value. Exposing the thermal port also extends the options on the **Operating Limits** tab as follows:

- The **Power rating** parameter becomes temperature dependent. You define a temperature up to which the full power rating is available, plus a higher temperature for which the power rating is reduced to zero. It is assumed that the power rating decreases linearly with temperature between these two values.

- An additional parameter, **Operating temperature range, [Tmin Tmax]**, lets you define the valid temperature range for block operation.

Variables

Use the **Variables** section of the block interface to set the priority and initial target values for the block variables prior to simulation. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

This section appears only for the blocks with exposed thermal port. The **Temperature** variable lets you specify a high-priority target for the temperature at the start of simulation.

Basic Assumptions and Limitations

Simulating with noise enabled slows down simulation. Choose the sample time (h) so that noise is generated only at frequencies of interest, and not higher.

Ports

Conserving

+ — Positive terminal
electrical

Electrical conserving port associated with the resistor positive terminal.

- — Negative terminal
electrical

Electrical conserving port associated with the resistor negative terminal.

H — Resistor thermal mass
thermal

Thermal conserving port that represents the resistor thermal mass.

Dependencies

Enabled for the thermal variant of the block. For more information, see "Thermal Port" on page 1-131.

Parameters

Main

Resistance — Nominal zero-bias resistance
1 Ω (default)

The zero-bias resistance, used as the nominal resistance value. Resistance value must be greater than zero. For the thermal variant of the block, this is the zero-bias resistance at a

temperature equal to the **Measurement temperature** parameter in the "Thermal" on page 1-142 section.

`Tolerance (%) — Resistor tolerance, in percent`
5 (default)

The resistor tolerance as defined on the manufacturer datasheet.

`Tolerance application — Select how to apply tolerance during simulation`
`None — use nominal value` (default) | `Random tolerance` | `Apply maximum tolerance value` | `Apply minimum tolerance value`

Select how to apply tolerance during simulation:

- `None — use nominal value` — The block does not apply tolerance, uses the nominal resistance value. This is the default.
- `Random tolerance` — The block applies random offset to the resistance value, within the tolerance value limit. You can choose Uniform or Gaussian distribution for calculating the random number by using the **Tolerance distribution** parameter.
- `Apply maximum tolerance value` — The resistance is increased by the specified tolerance percent value.
- `Apply minimum tolerance value` — The resistance is decreased by the specified tolerance percent value.

`Tolerance distribution — Select the distribution type`
`Uniform` (default) | `Gaussian`

Select the distribution type for random tolerance:

- `Uniform` — Uniform distribution
- `Gaussian` — Gaussian distribution

Dependencies

Enabled when the **Tolerance application** parameter is set to `Random tolerance`.

`Number of standard deviations for quoted tolerance — Used for calculating the Gaussian random number`
4 (default)

Number of standard deviations for calculating the Gaussian random number.

Dependencies

Enabled when the **Tolerance distribution** parameter is set to `Gaussian`.

Parameterization — Select parameterization method
`Simplified` (default) | `Advanced`

Select how to apply tolerance during simulation:

- `Simplified` — Assume that the quadratic and linear coefficients are the same, and define block behavior using the **Critical voltage** and **Corner voltage** parameters.
- `Advanced` — Explicitly specify values for the quadratic and linear voltage coefficients and for the inverse voltages for quadratic and linear voltage activation.

Critical voltage — Critical voltage for saturation
4 V (default)

Critical voltage for the saturation mechanism. You can determine this parameter value by taking the reciprocal of the slope of the increase of R/R_0 with voltage.

Dependencies

Enabled when the **Parameterization** parameter is set to `Simplified`.

Corner voltage — Voltage at which the resistance increase starts to occur
2 V (default)

Corner voltage, at which the resistance increase starts to occur. The **Corner voltage** must be less than the **Critical voltage**.

Dependencies

Enabled when the **Parameterization** parameter is set to `Simplified`.

Quadratic voltage coefficient — Coefficient p_2
0.25 (default)

Coefficient p_2 from the defining equation.

Dependencies

Enabled when the **Parameterization** parameter is set to `Advanced`.

Inverse voltage for quadratic voltage activation — Coefficient θ_2
0.25 1/V (default)

Coefficient θ_2 from the defining equation.

Dependencies

Enabled when the **Parameterization** parameter is set to `Advanced`.

Linear voltage coefficient — Coefficient p_3
0.25 (default)

Coefficient p_3 from the defining equation.

Dependencies

Enabled when the **Parameterization** parameter is set to `Advanced`.

Inverse voltage for linear voltage activation — Coefficient θ_3
0.25 1/V (default)

Coefficient θ_3 from the defining equation.

Dependencies

Enabled when the **Parameterization** parameter is set to `Advanced`.

Operating Limits

Enable operating limits — Select Yes to enable reporting when the operational limits are exceeded
No (default) | Yes

Select `Yes` to enable reporting when the operational limits are exceeded. The associated parameters in the **Operating Limits** section become visible to let you select the reporting method and specify the operating limits in terms of power and maximum working voltage. Parameters that specify operating limits in terms of temperature are visible only for blocks with exposed thermal port (see "Thermal Port" on page 1-131). The default value is `No`.

Reporting when operating limits exceeded — Select the reporting method
Warn (default) | Error

Select what happens when an operating limit is exceeded:

- `Warn` — The block issues a warning.
- `Error` — Simulation stops with an error.

Dependencies

Enabled when the **Enable operating limits** parameter is set to `Yes`.

Maximum working voltage — Maximum voltage allowed for normal block operation
100 V (default)

Maximum voltage magnitude allowed for normal block operation.

Dependencies

Enabled when the **Enable operating limits** parameter is set to `Yes`.

Power rating — Maximum power allowed for normal block operation
1 W (default)

Maximum power allowed for normal block operation.

If you expose the thermal port of the block, this parameter becomes temperature dependent. The value you specify for the **Power rating** parameter applies up to the temperature specified by the **Temperature below which full power rating is available** parameter value. Then the power rating decreases linearly with temperature, until it becomes 0 at temperature specified by the **Temperature above which power rating is reduced to zero** parameter value.

Dependencies

Enabled when the **Enable operating limits** parameter is set to `Yes`.

Temperature below which full power rating is available — Maximum temperature where full power rating still applies
70 °C (default)

Maximum temperature where full power rating, specified by the **Power rating** parameter value, still applies.

Dependencies

Enabled for the thermal variant of the block. For more information, see "Thermal Port" on page 1-614.

1-137

`Temperature above which power rating is reduced to zero` — Temperature where power rating becomes 0
155 °C (default)

Temperature where power rating becomes 0. Above this temperature, the simulation always issues an assertion regardless of dissipated power. This parameter value must be higher than **Temperature below which full power rating is available**.

Dependencies

Enabled for the thermal variant of the block. For more information, see "Thermal Port" on page 1-614.

`Operating temperature range, [Tmin Tmax]` — Minimum and maximum temperature values allowed for normal block operation
[-50 150] °C (default)

A row vector of length 2 specifying minimum and maximum temperature values allowed for normal block operation. The first element is the lowest allowable operating temperature, and the second element is the largest allowable operating temperature.

Dependencies

Enabled for the thermal variant of the block. For more information, see "Thermal Port" on page 1-614.

Faults

`Enable faults` — Select Yes to enable faults modeling
No (default) | Yes

Select `Yes` to enable faults modeling. The associated parameters in the **Faults** section become visible to let you select the reporting method and specify the trigger mechanism (temporal or behavioral). You can enable these trigger mechanisms separately or use them together.

`Reporting when a fault occurs` — Choose whether to issue an assertion when a fault occurs
None (default) | Warn | Error

Choose whether to issue an assertion when a fault occurs:

- None — The block does not issue an assertion.
- Warn — The block issues a warning.
- Error — Simulation stops with an error.

Dependencies

Enabled when the **Enable faults** parameter is set to Yes.

Faulted zero-voltage resistance — Resistance when block is in faulted state
inf Ω (default)

Zero-voltage resistance between the + and – ports when the block is in the faulted state.

Dependencies

Enabled when the **Enable faults** parameter is set to Yes.

Enable temporal fault trigger — Select Yes to enable time-based fault triggering
No (default) | Yes

Select Yes to enable time-based fault triggering. You can enable the temporal and behavioral trigger mechanisms separately or use them together.

Dependencies

Enabled when the **Enable faults** parameter is set to Yes.

Simulation time for fault event — Time before entering faulted state
1 s (default)

Set the simulation time at which you want the block to enter the faulted state.

Dependencies

Enabled when the **Enable temporal fault trigger** parameter is set to Yes.

Enable behavioral fault trigger — Select Yes to enable behavioral fault triggering
No (default) | Yes

Select Yes to enable behavioral fault triggering. You can enable the temporal and behavioral trigger mechanisms separately or use them together.

Dependencies

Enabled when the **Enable faults** parameter is set to `Yes`.

`Maximum permissible current — Current threshold to fault transition`
1 A (default)

Specify the maximum permissible current value. If the current exceeds this value for longer than the **Time to fail when exceeding maximum permissible current** parameter value, then the block enters the faulted state.

Dependencies

Enabled when the **Enable behavioral fault trigger** parameter is set to `Yes`.

`Time to fail when exceeding maximum permissible current — Maximum length of time the current exceeds the threshold`
1 s (default)

Set the maximum length of time that the current can exceed the maximum permissible value without triggering the fault.

Dependencies

Enabled when the **Enable behavioral fault trigger** parameter is set to `Yes`.

Noise

`Noise mode — Select whether to model thermal noise current`
`Disabled` (default) | `Enabled`

Select whether to model thermal noise current:

- `Disabled` — No noise is produced by the resistor.
- `Enabled` — Resistor generates thermal noise current, and the associated parameters become visible in the **Noise** section.

`Sample time — Rate at which the noise source is sampled`
1e-3 s (default)

Defines the rate at which the noise source is sampled. Choose it to reflect the frequencies of interest in your model. Making the sample time too small will unnecessarily slow down your simulation.

Dependencies

Enabled when the **Noise mode** parameter is set to `Enabled`.

Repeatability — Select the noise control option
`Not repeatable` (default) | `Repeatable` | `Specify seed`

Select the noise control option:

- `Not repeatable` — The random sequence used for noise generation is not repeatable.
- `Repeatable` — The random sequence used for noise generation is repeatable, with a system-generated seed.
- `Specify seed` — The random sequence used for noise generation is repeatable, and you control the seed by using the **Seed** parameter.

Dependencies

Enabled when the **Noise mode** parameter is set to `Enabled`.

Auto-generated seed used for repeatable option — Auto-generated random number seed
random real number

Random number seed stored inside the block to make the random sequence repeatable. The parameter value is automatically generated using the MATLAB random number generator command. You can modify this parameter value, but it gets overwritten by a new random value if you copy the block to another block in the model. Therefore, if you want to control the seed of the random sequence, use the `Specify seed` option for the **Repeatability** parameter and specify the desired seed value using the **Seed** parameter.

Dependencies

Enabled when the **Repeatability** parameter is set to `Repeatable`.

Seed — Random number seed
0 (default)

Seed used by the noise random number generator.

Dependencies

Enabled when the **Repeatability** parameter is set to `Specify seed`.

Device simulation temperature — Temperature of resistor at the start of the simulation
25 °C (default)

The temperature of the resistor at the start of the simulation.

Dependencies

Enabled when the **Noise mode** parameter is set to `Enabled`.

For blocks with an exposed thermal port, this parameter is disabled. Instead, use the **Variables** tab to set the initial temperature target. For more information, see "Variables" on page 1-132.

Thermal

This section appears only for blocks with exposed thermal port. For more information, see "Thermal Port" on page 1-131.

Resistance linear temperature coefficient — Specifies how the resistance value changes with temperature
0 1/K (default)

The coefficient T_{C1}^{eff} in the equation that describes resistance as a function of temperature. See "Thermal Port" on page 1-131 for details.

Resistance quadratic temperature coefficient — Specifies how the resistance value changes with temperature
0 1/K^2 (default)

The coefficient T_{C2}^{eff} in the equation that describes resistance as a function of temperature. See "Thermal Port" on page 1-131 for details.

Measurement temperature — Temperature corresponding to nominal resistance
25 °C (default)

The temperature T_0, for which the nominal resistance R is specified.

Thermal mass — Thermal mass associated with port H
100 J/K (default)

Thermal mass associated with the thermal port H. It represents the energy required to raise the temperature of the thermal port by one degree.

See Also

Fault | Resistor

Introduced in R2017b

Diode

Diode model; piecewise linear, piecewise linear zener, or exponential diode

Library

Semiconductor Devices

Description

The Diode block represents one of the following types of diodes:

- "Piecewise Linear" on page 1-144
- "Piecewise Linear Zener" on page 1-144
- "Exponential" on page 1-145

Piecewise Linear

The piecewise linear diode model is the same model found in the Simscape Diode block, with the addition of a fixed junction capacitance. If the diode forward voltage exceeds the value specified in the **Forward voltage** parameter, the diode behaves as a linear resistor with the resistance specified in the **On resistance** parameter. Otherwise, the diode behaves as a linear resistor with the small conductance specified in the **Off conductance** parameter. Zero voltage across the diode results in zero current flowing.

Piecewise Linear Zener

The piecewise linear zener diode model behaves like the piecewise linear diode model for bias voltages above $-Vz$, where Vz is the **Reverse breakdown voltage Vz** parameter value. For voltages less than $-Vz$, the diode behaves as a linear resistor with the low Zener resistance specified in the **Zener resistance Rz** parameter. This diode model also includes a fixed junction capacitance.

Exponential

The exponential diode model provides the following relationship between the diode current *I* and the diode voltage *V*:

$$I = IS \cdot \left(e^{\frac{qV}{NkT_{m1}}} - 1 \right) \qquad V > -BV$$

$$I = -IS \cdot \left(e^{\frac{-q(V+Vz)}{kT_{m1}}} - e^{\frac{qV}{NkT_{m1}}} \right) \qquad V \le -BV$$

where:

- *q* is the elementary charge on an electron (1.602176e–19 Coulombs).
- *k* is the Boltzmann constant (1.3806503e–23 J/K).
- *BV* is the **Reverse breakdown voltage BV** parameter value.
- *N* is the emission coefficient.
- *IS* is the saturation current.
- T_{m1} is the temperature at which the diode parameters are specified, as defined by the **Measurement temperature** parameter value.

When $(qV / NkT_{m1}) > 80$, the block replaces $e^{\frac{qV}{NkT_{m1}}}$ with $(qV / NkT_{m1} - 79)e^{80}$, which matches the gradient of the diode current at $(qV / NkT_{m1}) = 80$ and extrapolates linearly.

When $(qV / NkT_{m1}) < -79$, the block replaces $e^{\frac{qV}{NkT_{m1}}}$ with $(qV / NkT_{m1} + 80)e^{-79}$, which also matches the gradient and extrapolates linearly. Typical electrical circuits do not reach these extreme values. The block provides this linear extrapolation to help convergence when solving for the constraints during simulation.

When you select `Use parameters IS and N` for the **Parameterization** parameter, you specify the diode in terms of the **Saturation current IS** and **Emission coefficient N**

parameters. When you select `Use I-V curve data points` for the **Parameterization** parameter, you specify two voltage and current measurement points on the diode I-V curve and the block derives the *IS* and *N* values. The block then calculates *IS* and *N* as follows:

- $$N = ((V_1 - V_2)/V_t)/(\log(I_1) - \log(I_2))$$

- $$IS = \left(I_1 / (\exp(V_1 /(NV_t)) - 1) + I_2 / (\exp(V_2 /(NV_t)) - 1) \right)/2$$

where:

- $V_t = kT_{m1}/q$.
- V_1 and V_2 are the values in the **Voltages [V1 V2]** vector.
- I_1 and I_2 are the values in the **Currents [I1 I2]** vector.

When you select `Use an I-V data point and IS` for the **Parameterization** parameter, then the block calculates *N* as follows:

$$N = V_1 / \left(V_t \log\left(\frac{I_1}{IS} + 1 \right) \right)$$

When you select `Use an I-V data point and N` for the **Parameterization** parameter, then the block calculates *IS* as follows:

$$IS = I_1 / \left(\exp\left(V_1 /\left(NV_t \right) - 1 \right) \right)$$

The exponential diode model provides the option to include a junction capacitance:

- When you select `Include fixed or zero junction capacitance` for the **Junction capacitance** parameter, the capacitance is fixed.
- When you select `Use parameters CJO, VJ, M & FC` for the **Junction capacitance** parameter, the block uses the coefficients *CJO*, *VJ*, *M*, and *FC* to calculate a junction capacitance that depends on the junction voltage.
- When you select `Use C-V curve data points` for the **Junction capacitance** parameter, the block uses three capacitance values on the C-V capacitance curve to estimate *CJO*, *VJ*, and *M* and uses these values with the specified value of *FC* to calculate a junction capacitance that depends on the junction voltage. The block calculates *CJO*, *VJ*, and *M* as follows:

- $$CJ0 = C_1((V_{R2} - V_{R1}) / (V_{R2} - V_{R1}(C_2 / C_1)^{-1/M}))^M$$

- $$VJ = -(-V_{R2}(C_1 / C_2)^{-1/M} + V_{R1}) / (1 - (C_1 / C_2)^{-1/M})$$

- $$M = \log(C_3 / C_2) / \log(V_{R2} / V_{R3})$$

where:

- V_{R1}, V_{R2}, and V_{R3} are the values in the **Reverse bias voltages [VR1 VR2 VR3]** vector.
- C_1, C_2, and C_3 are the values in the **Corresponding capacitances [C1 C2 C3]** vector.

It is not possible to estimate *FC* reliably from tabulated data, so you must specify its value using the **Capacitance coefficient FC** parameter. In the absence of suitable data for this parameter, use a typical value of 0.5.

The reverse bias voltages (defined as positive values) should satisfy $V_{R3} > V_{R2} > V_{R1}$. This means that the capacitances should satisfy $C_1 > C_2 > C_3$ as reverse bias widens the depletion region and hence reduces capacitance. Violating these inequalities results in an error. Voltages V_{R2} and V_{R3} should be well away from the Junction potential *VJ*. Voltage V_{R1} should be less than the Junction potential *VJ*, with a typical value for V_{R1} being 0.1 V.

The voltage-dependent junction is defined in terms of the capacitor charge storage Q_j as:

- For $V < FC \cdot VJ$:

$$Q_j = CJ0 \cdot (VJ / (M-1)) \cdot ((1 - V/VJ)^{1-M} - 1)$$

- For $V \geq FC \cdot VJ$:

$$Q_j = CJ0 \cdot F_1 + (CJ0 / F_2) \cdot (F_3 \cdot (V - FC \cdot VJ) + 0.5(M/VJ) \cdot (V^2 - (FC \cdot VJ)^2))$$

where:

- $$F_1 = (VJ / (1-M)) \cdot (1 - (1-FC)^{1-M}))$$

- $$F_2 = (1-FC)^{1+M}))$$

- $$F_3 = 1 - FC \cdot (1 + M)$$

These equations are the same as used in [2 on page 1-160], except that the temperature dependence of *VJ* and *FC* is not modeled. This model does not include the diffusion capacitance term that affects performance for high frequency switching applications.

For applications such as commutation diodes it can be important to model diode charge dynamics. When a forward-biased diode has a reverse voltage applied across it, it takes time for the charge to dissipate and hence for the diode to turn off. The time taken for the diode to turn off is captured primarily by the transit time parameter. Once the diode is off, any remaining charge then dissipates, the rate at which this happens being determined by the carrier lifetime.

The Diode block uses the model of Lauritzen and Ma [3 on page 1-160] to capture these effects. The three defining equations are:

$$I = \frac{q_E - q_M}{TT}$$

$$\frac{dq_M}{dt} + \frac{q_M}{\tau} - \frac{q_E - q_M}{TT} = 0$$

$$q_E = (\tau + TT) IS \left(\exp\left(\frac{V}{N \cdot V_t} \right) - 1 \right)$$

where:

- *I* is the diode current.
- *V* is the diode voltage.
- *N* is the emission coefficient.
- q_E is the junction charge.
- q_M is the total stored charge.
- *TT* is the transit time.
- τ is the carrier lifetime.

Datasheets do not typically provide values for *TT* and τ. Therefore the Diode block provides an alternative parameterization in terms of **Peak reverse current, Irrm** and

Reverse recovery time, trr. Equivalent values for *TT* and τ are calculated from these values, plus information on the initial forward current and rate of change of current used in the test circuit when measuring I_{rrm} and t_{rr}. The test circuit can consist of a series voltage source, resistor, inductor and the diode. The polarity of the voltage source is switched so as to move the diode from forward conduction to reverse biased. The following figure shows an idealized diode current response.
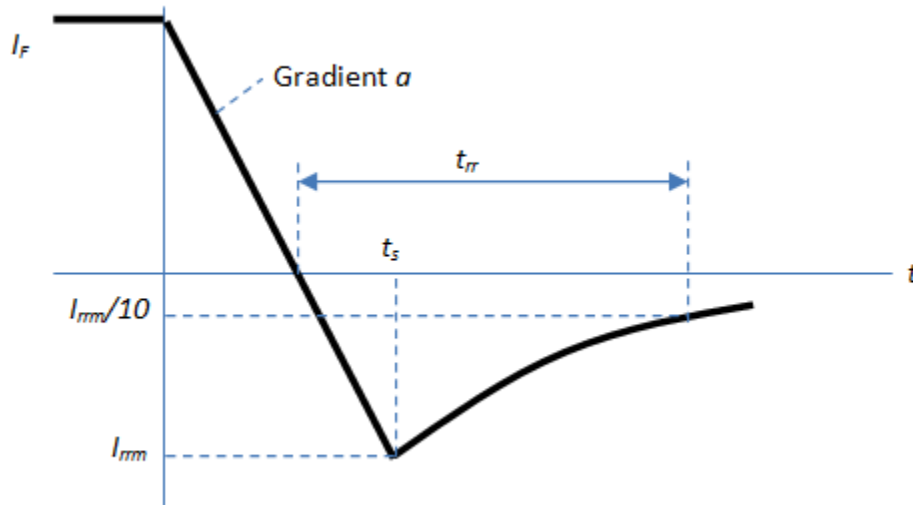
The value of the series resistor and applied voltage value determine the initial current I_F. The value of the series inductance and the applied reverse voltage value determine the current gradient, *a*.

The precise values of peak reverse current and reverse recovery time depend on the test circuit used. Also, junction capacitance has some effect on the current recovery characteristic. However, a junction capacitor value that dominates the response is physically unrealistic.

Only the exponential diode supports modeling of the diode charge dynamics. If you select the Exponential for the **Diode model** parameter, then the **Capacitance** tab contains an additional parameter called **Charge dynamics**. Select between the three options:

- Do not model charge dynamics

- Use peak reverse current and reverse recovery time
- Use transit time and carrier lifetime

The default behavior for the Diode is that dependence on temperature is not modeled, and the device is simulated at the temperature for which you provide block parameters. The Exponential diode model contains several options for modeling the dependence of the diode current-voltage relationship on the temperature during simulation. Temperature dependence of the junction capacitance is not modeled, this being a much smaller effect.

When including temperature dependence, the diode defining equation remains the same. The measurement temperature value, T_{m1}, is replaced with the simulation temperature, T_s. The saturation current, IS, becomes a function of temperature according to the following equation:

$$IS_{Ts} = IS_{Tm1} \cdot (T_s / T_{m1})^{XTI/N} \cdot \exp\left(-\frac{EG}{NkT_s}(1 - T_s / T_{m1})\right)$$

where:

- T_{m1} is the temperature at which the diode parameters are specified, as defined by the **Measurement temperature** parameter value.
- T_s is the simulation temperature.
- IS_{Tm1} is the saturation current at measurement temperature.
- IS_{Ts} is the saturation current at simulation temperature. This is the saturation current value used in the standard diode equation when temperature dependence is modeled.
- EG is the energy gap for the semiconductor type measured in Joules. The value for silicon is usually taken to be 1.11 eV, where 1 eV is 1.602e-19 Joules.
- XTI is the saturation current temperature exponent. This is usually set to 3.0 for pn-junction diodes, and 2.0 for Schottky barrier diodes.
- N is the emission coefficient.
- k is the Boltzmann constant (1.3806503e–23 J/K).

Appropriate values for XTI and EG depend on the type of diode and the semiconductor material used. Default values for particular material types and diode types capture approximate behavior with temperature. The block provides default values for common types of diode.

In practice, the values of *XTI* and *EG* need tuning to model the exact behavior of a particular diode. Some manufacturers quote these tuned values in a SPICE Netlist, and you can read off the appropriate values. Otherwise you can determine improved estimates for *EG* by using a datasheet-defined current-voltage data point at a higher temperature. The block provides a parameterization option for this. It also gives the option of specifying the saturation current at a higher temperature IS_{Tm2} directly.

You can also tune the values of *XTI* and *EG* yourself, to match lab data for your particular device. You can use Simulink® Design Optimization™ software to help tune the values for *XTI* and *EG*.

Caution Device temperature behavior is also dependent on the emission coefficient. An inappropriate value for the emission coefficient can give incorrect temperature dependence, because saturation current is a function of the ratio of *EG* to *N*.

If defining a finite reverse breakdown voltage *BV*, then the value of the reverse breakdown voltage is modulated by the reverse breakdown temperature coefficient *TCV* (specified using the **Reverse breakdown voltage temperature coefficient, dBV/dT** parameter):

$$BV_{Ts} = BV_{Tm1} - TCV \cdot (T_s - T_{m1})$$

Thermal Port

The block has an optional thermal port, hidden by default. To expose the thermal port, right-click the block in your model, and then from the context menu select **Simscape > Block choices > Show thermal port**. This action displays the thermal port H on the block icon, and adds the **Thermal Port** tab to the block dialog box.

Use the thermal port to simulate the effects of generated heat and device temperature. For more information on using thermal ports and on the **Thermal Port** tab parameters, see "Simulating Thermal Effects in Semiconductors".

Variables

Use the **Variables** section of the block interface to set the priority and initial target values for the block variables prior to simulation. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

Basic Assumptions and Limitations

The Exponential diode model has the following limitations:

- When you select `Use I-V curve data points` for the **Parameterization** parameter, choose a pair of voltages near the diode turn-on voltage. Typically, this is in the range from 0.05 to 1 Volt. Using values outside of this region may lead to numerical issues and poor estimates for *IS* and *N*.

- The block does not account for temperature-dependent effects on the junction capacitance.

- You may need to use nonzero ohmic resistance and junction capacitance values to prevent numerical simulation issues, but the simulation may run faster with these values set to zero.

Parameters

- "Main" on page 1-152
- "Breakdown" on page 1-155
- "Capacitance" on page 1-155
- "Temperature Dependence" on page 1-158

Main

Diode model

Select one of the following diode models:

- `Piecewise Linear (Foundation Library)` — Use a piecewise linear model for the diode, as described in "Piecewise Linear" on page 1-144. This is the default method.

- `Piecewise Linear Zener` — Use a piecewise linear model with reverse breakdown characteristics for the diode, as described in "Piecewise Linear Zener" on page 1-144.

- `Exponential` — Use a standard exponential model for the diode, as described in "Exponential" on page 1-145.

Forward voltage

Minimum voltage that needs to be applied for the diode to become forward-biased. This parameter is only visible when you select `Piecewise Linear (Foundation Library)` or `Piecewise Linear Zener` for the **Diode model** parameter. The default value is `0.6` V.

On resistance

The resistance of the diode when it is forward biased. This parameter is only visible when you select `Piecewise Linear (Foundation Library)` or `Piecewise Linear Zener` for the **Diode model** parameter. The default value is `0.3` Ω.

Off conductance

The conductance of the diode when it is reverse biased. This parameter is only visible when you select `Piecewise Linear (Foundation Library)` or `Piecewise Linear Zener` for the **Diode model** parameter. The default value is `1e-08` 1/Ω.

Parameterization

Select one of the following methods for model parameterization:

- `Use two I-V curve data points` — Specify measured data at two points on the diode I-V curve. This is the default method.
- `Use parameters IS and N` — Specify saturation current and emission coefficient.
- `Use an I-V data point and IS` — Specify measured data at a single point on the diode I-V curve in combination with the saturation current.
- `Use an I-V data point and N` — Specify measured data at a single point on the diode I-V curve in combination with the emission coefficient.

This parameter is only visible when you select `Exponential` for the **Diode model** parameter.

Currents [I1 I2]

A vector of the current values at the two points on the diode I-V curve that the block uses to calculate *IS* and *N*. This parameter is only visible when you select `Exponential` for the **Diode model** parameter and `Use two I-V curve data points` for the **Parameterization** parameter. The default value is `[ 0.0137 0.545 ]` A.

Voltages [V1 V2]

A vector of the voltage values at the two points on the diode I-V curve that the block uses to calculate *IS* and *N*. This parameter is only visible when you select

Exponential for the **Diode model** parameter and Use two I-V curve data points for the **Parameterization** parameter. The default value is [0.6 0.7] V.

Current I1

A current value at the point on the diode I-V curve that the block uses for calculations. This parameter is only visible when you select Exponential for the **Diode model** parameter and either Use an I-V data point and IS or Use an I-V data point and N for the **Parameterization** parameter. Depending on the **Parameterization** value, the block uses this parameter to calculate either *N* or *IS*. The default value is 0.07 A.

Voltage V1

A voltage value at the point on the diode I-V curve that the block uses for calculations. This parameter is only visible when you select Exponential for the **Diode model** parameter and either Use an I-V data point and IS or Use an I-V data point and N for the **Parameterization** parameter. Depending on the **Parameterization** value, the block uses this parameter to calculate either *N* or *IS*. The default value is 0.7 V.

Saturation current, IS

The magnitude of the current that the ideal diode equation approaches asymptotically for very large reverse bias levels. This parameter is only visible when you select Exponential for the **Diode model** parameter and either Use parameters IS and N or Use an I-V data point and IS for the **Parameterization** parameter. The default value is 1e-14 A.

Emission coefficient, N

The diode emission coefficient or ideality factor. This parameter is only visible when you select Exponential for the **Diode model** parameter and either Use parameters IS and N or Use an I-V data point and N for the **Parameterization** parameter. The default value is 1.

Ohmic resistance, RS

The series diode connection resistance. This parameter is only visible when you select Exponential for the **Diode model** parameter. The default value is 0.01 Ω.

Measurement temperature

The temperature T_{m1} at which IS or the I-V curve was measured. This parameter is only visible when you select Exponential for the **Diode model** parameter. The default value is 25 °C.

Breakdown

This section is not applicable for Piecewise Linear diode models.

Zener resistance Rz

>   The resistance of the diode when the voltage is less than the **Reverse breakdown voltage Vz** value. This parameter is only visible when you select `Piecewise Linear Zener` for the **Diode model** parameter. The default value is `0.3` Ω.

Reverse breakdown voltage Vz

>   The reverse voltage below which the diode resistance changes to the **Zener resistance Rz** value. This parameter is only visible when you select `Piecewise Linear Zener` for the **Diode model** parameter. The default value is `50` V.

Reverse breakdown voltage BV

>   The reverse voltage below which to model the rapid increase in conductance that occurs at diode breakdown. This parameter is only visible when you select `Exponential` for the **Diode model** parameter. The default value is `Inf` V, which effectively omits reverse breakdown from the model.

Capacitance

Junction capacitance

- When you select `Piecewise Linear (Foundation Library)` or `Piecewise Linear Zener` for the **Diode model** parameter, the **Junction capacitance** parameter is the fixed junction capacitance value. The default value is 5 pF.

- When you select `Exponential` for the **Diode model** parameter, the **Junction capacitance** parameter lets you select one of the following options for modeling the junction capacitance:

  - `Include fixed or zero junction capacitance` — Model the junction capacitance as a fixed value.

  - `Use C-V curve data points` — Specify measured data at three points on the diode C-V curve.

  - `Use parameters CJ0, VJ, M & FC` — Specify zero-bias junction capacitance, junction potential, grading coefficient, and forward-bias depletion capacitance coefficient.

Zero-bias junction capacitance CJ0

The value of the capacitance placed in parallel with the exponential diode term. This parameter is only visible when you select `Exponential` for the **Diode model** parameter and `Include fixed or zero junction capacitance` or `Use parameters CJ0, VJ, M & FC` for the **Junction capacitance** parameter. The default value is 5 pF.

Reverse bias voltages [VR1 VR2 VR3]

A vector of the reverse bias voltage values at the three points on the diode C-V curve that the block uses to calculate *CJ0*, *VJ*, and *M*. This parameter is only visible when you select `Exponential` for the **Diode model** parameter and `Use C-V curve data points` for the **Junction capacitance** parameter. The default value is [0.1 10 100] V.

Corresponding capacitances [C1 C2 C3]

A vector of the capacitance values at the three points on the diode C-V curve that the block uses to calculate *CJ0*, *VJ*, and *M*. This parameter is only visible when you select `Exponential` for the **Diode model** parameter and `Use C-V curve data points` for the **Junction capacitance** parameter. The default value is [3.5 1 0.4] pF.

Junction potential VJ

The junction potential. This parameter is only visible when you select `Exponential` for the **Diode model** parameter and `Use parameters CJ0, VJ, M & FC` for the **Junction capacitance** parameter. The default value is 1 V.

Grading coefficient M

The grading coefficient. This parameter is only visible when you select `Exponential` for the **Diode model** parameter and `Use parameters CJ0, VJ, M & FC` for the **Junction capacitance** parameter. The default value is 0.5.

Capacitance coefficient FC

Fitting coefficient that quantifies the decrease of the depletion capacitance with applied voltage. This parameter is only visible when you select `Exponential` for the **Diode model** parameter and `Use C-V curve data points` or `Use parameters CJ0, VJ, M & FC` for the **Junction capacitance** parameter. The default value is 0.5.

Charge model

Select one of the following methods for charge dynamics parameterization:

- `Do not model charge dynamics` — Do not include charge dynamics modeling. This is the default method.

- `Use peak reverse current and reverse recovery time` — Model charge dynamics by providing values for peak reverse current, I_{rrm}, and reverse recovery time, t_{rr}, plus information on the initial forward current and rate of change of current used in the test circuit when measuring I_{rrm} and t_{rr}. Use this option if the manufacturer datasheet does not provide values for transit time, *TT*, and carrier lifetime, τ.

- `Use transit time and carrier lifetime` — Model charge dynamics by providing values for transit time, *TT*, and carrier lifetime, τ.

Peak reverse current, Irrm

The peak reverse current measured in a test circuit. This parameter is only visible when you select `Exponential` for the **Diode model** parameter and `Use peak reverse current and reverse recovery time` for the **Charge model** parameter. The default value is `7.15` A.

Starting forward current when measuring Irrm

The initial forward current when measuring peak reverse current. This parameter is only visible when you select `Exponential` for the **Diode model** parameter and `Use peak reverse current and reverse recovery time` for the **Charge model** parameter. The default value is `4` A.

Rate of change of current when measuring Irrm

The rate of change of current when measuring peak reverse current. This parameter is only visible when you select `Exponential` for the **Diode model** parameter and `Use peak reverse current and reverse recovery time` for the **Charge model** parameter. The default value is `-750` A/us.

Reverse recovery time, trr

The time between the point where the current initially goes to zero when the diode turns off, and the point where the current falls to less than ten percent of the peak reverse current. This parameter is only visible when you select `Exponential` for the **Diode model** parameter and `Use peak reverse current and reverse recovery time` for the **Charge model** parameter. The default value is `115` ns.

Transit time, TT

A measure of how long it takes carriers to cross the diode junction. This parameter is only visible when you select `Exponential` for the **Diode model** parameter and `Use transit time and carrier lifetime` for the **Charge model** parameter. The default value is `50` ns.

Carrier lifetime, tau

> A measure of how long it takes for the carriers to dissipate once the diode is no longer conducting. This parameter is only visible when you select `Exponential` for the **Diode model** parameter and `Use transit time and carrier lifetime` for the **Charge model** parameter. The default value is `100` ns.

Temperature Dependence

This section is applicable for Exponential diode models only.

Parameterization

> Select one of the following methods for temperature dependence parameterization:
>
> - `None — Simulate at parameter measurement temperature` — Temperature dependence is not modeled, or the model is simulated at the measurement temperature T_{m1} (as specified by the **Measurement temperature** parameter on the **Main** tab). This is the default method.
>
> - `Use an I-V data point at second measurement temperature` — If you select this option, you specify a second measurement temperature T_{m2}, and the current and voltage values at this temperature. The model uses these values, along with the parameter values at the first measurement temperature T_{m1}, to calculate the energy gap value.
>
> - `Specify saturation current at second measurement temperature` — If you select this option, you specify a second measurement temperature T_{m2}, and saturation current value at this temperature. The model uses these values, along with the parameter values at the first measurement temperature T_{m1}, to calculate the energy gap value.
>
> - `Specify the energy gap, EG` — Specify the energy gap value directly.

Current I1 at second measurement temperature

> Specify the diode current *I1* value when the voltage is *V1* at the second measurement temperature. This parameter is only visible when you select `Use an I-V data point at second measurement temperature` for the **Parameterization** parameter. The default value is `0.245` A.

Voltage V1 at second measurement temperature

> Specify the diode voltage *V1* value when the current is *I1* at the second measurement temperature. This parameter is only visible when you select `Use an I-V data`

`point at second measurement temperature` for the **Parameterization** parameter. The default value is `0.5` V.

Saturation current, IS, at second measurement temperature

Specify the saturation current *IS* value at the second measurement temperature. This parameter is only visible when you select `Specify saturation current at second measurement temperature` for the **Parameterization** parameter. The default value is `1.25e-7` A.

Second measurement temperature

Specify the value for the second measurement temperature. This parameter is only visible when you select either `Use an I-V data point at second measurement temperature` or `Specify saturation current at second measurement temperature` for the **Parameterization** parameter. The default value is `125` °C.

Energy gap parameterization

This parameter is only visible when you select `Specify the energy gap, EG` for the **Parameterization** parameter. It lets you select a value for the energy gap from a list of predetermined options, or specify a custom value:

- `Use nominal value for silicon (EG=1.11eV)` — This is the default.
- `Use nominal value for 4H-SiC silicon carbide (EG=3.23eV)`
- `Use nominal value for 6H-SiC silicon carbide (EG=3.00eV)`
- `Use nominal value for germanium (EG=0.67eV)`
- `Use nominal value for gallium arsenide (EG=1.43eV)`
- `Use nominal value for selenium (EG=1.74eV)`
- `Use nominal value for Schottky barrier diodes (EG=0.69eV)`
- `Specify a custom value` — If you select this option, the **Energy gap, EG** parameter appears in the dialog box, to let you specify a custom value for *EG*.

Energy gap, EG

Specify a custom value for the energy gap, *EG*. This parameter is only visible when you select `Specify a custom value` for the **Energy gap parameterization** parameter. The default value is `1.11` eV.

Saturation current temperature exponent parameterization

Select one of the following options to specify the saturation current temperature exponent value:

- Use nominal value for pn-junction diode (XTI=3) — This is the default.
- Use nominal value for Schottky barrier diode (XTI=2)
- Specify a custom value — If you select this option, the **Saturation current temperature exponent, XTI** parameter appears in the dialog box, to let you specify a custom value for *XTI*.

Saturation current temperature exponent, XTI

Specify a custom value for the saturation current temperature exponent, *XTI*. This parameter is only visible when you select Specify a custom value for the **Saturation current temperature exponent parameterization** parameter. The default value is 3.

Reverse breakdown voltage temperature coefficient, dBV/dT

This coefficient modulates the reverse breakdown voltage *BV*. If you define the reverse breakdown voltage *BV* as a positive quantity, a positive value for *TCV* implies that the magnitude of the reverse breakdown voltage decreases with temperature. The default value is 0 V/K.

Device simulation temperature

Specify the value for the temperature T_s, at which the device is to be simulated. The default value is 25 °C.

Ports

The block has the following ports:

+

Electrical conserving port associated with the diode positive terminal

-

Electrical conserving port associated with the diode negative terminal

References

[1] MH. Ahmed and P.J. Spreadbury. *Analogue and digital electronics for engineers*. 2nd Edition, Cambridge University Press, 1984.

[2] G. Massobrio and P. Antognetti. *Semiconductor Device Modeling with SPICE*. 2nd Edition, McGraw-Hill, 1993.

[3] Lauritzen, P.O. and C.L. Ma. "A Simple Diode Model with Reverse Recovery." *IEEE® Transactions on Power Electronics*. Vol. 6, No. 2, April 1991.

DPDT Switch

Model double-pole double-throw switch

Library

Passive Devices/Switches

Description

The DPDT Switch block models a double-pole double-throw switch:

- When the switch is closed, ports c1 and c2 are connected to ports s12 and s22, respectively.
- When the switch is open, ports c1 and c2 are connected to ports s11 and s21, respectively.

Closed connections are modeled by a resistor with value equal to the **Closed resistance** parameter value. Open connections are modeled by a resistor with value equal to the reciprocal of the **Open conductance** parameter value.

If the **Threshold width** parameter is set to zero, the switch is closed if the voltage presented at the vT control port exceeds the value of the **Threshold** parameter.

If the **Threshold width** parameter is greater than zero, then switch conductance G varies smoothly between off-state and on-state values:

$$G = \frac{x}{R_{closed}} + (1 - x)G_{open}$$

$$\lambda = \frac{vT - \text{Threshold}}{\text{Threshold width}}$$

$$x = \begin{cases} 0 & \text{for } \lambda \leq 0 \\ 3\lambda^2 - 2\lambda^3 & \text{for } 0 < \lambda < 1 \\ 1 & \text{for } \lambda \geq 1 \end{cases}$$

The block uses the function $3\lambda^2 - 2\lambda^3$ because its derivative is zero for $\lambda = 0$ and $\lambda = 1$.

Defining a small positive **Threshold width** can help solver convergence in some models, particularly if the control port signal *vT* varies continuously as a function of other network variables. However, defining a nonzero threshold width precludes the solver making use of switched linear optimizations. Therefore, if the rest of your network is switched linear, MathWorks recommends that you set **Threshold width** to zero.

Optionally, you can add a delay between the point at which the voltage at vT passes the threshold and the switch opening or closing. To enable the delay, on the **Dynamics** tab, set the **Model dynamics** parameter to Model turn-on and turn-off times.

Parameters

Main Tab

Closed resistance

Resistance between the c and s electrical ports when the switch is closed. The value must be greater than zero. The default value is 0.01 Ω.

Open conductance

Conductance between the c and s electrical ports when the switch is open. The value must be greater than zero. The default value is 1e-6 S.

Threshold

The threshold voltage for the control physical signal input vT above which the switch will turn on. The default value is 0 V.

Threshold width

The minimum increase in the control signal vT above the threshold value that will move the switch from fully open to fully closed. The default value is 0 V.

Dynamics Tab

Model dynamics

Select whether the block models a switching delay:

- No dynamics — Do not model the delay. This is the default option.
- Model turn-on and turn-off times — Use additional parameters to model a delay between the point at which the voltage at vT passes the threshold and the switch opening or closing.

Turn-on delay

Time between the input voltage exceeding the threshold voltage and the switch closing. This parameter is only visible when you select Model turn-on and turn-off times for the **Model dynamics** parameter. The value must be greater than zero. The default value is 1e-3 seconds.

Turn-off delay

Time between the input voltage falling below the threshold voltage and the switch opening. This parameter is only visible when you select Model turn-on and turn-off times for the **Model dynamics** parameter. The value must be greater than zero. The default value is 1e-3 seconds.

Initial input value, vT

The value of the physical signal input vT at time zero. This value is used to initialize the delayed control voltage parameter internally. This parameter is only visible when you select Model turn-on and turn-off times for the **Model dynamics** parameter. The default value is 0 V.

Ports

This block has the following ports:

vT

Physical signal that opens and closes the switch

c1, c2, s11, s12, s21, s22
> Electrical conserving ports

See Also

DPST Switch | SPDT Switch | SPST Switch | Switch

DPST Switch

Model double-pole single-throw switch

Library

Passive Devices/Switches

Description

The DPST Switch block models a double-pole single-throw switch. When the switch is closed, ports c1 and c2 are connected to ports s1 and s2, respectively.

Closed connections are modeled by a resistor with value equal to the **Closed resistance** parameter value. Open connections are modeled by a resistor with value equal to the reciprocal of the **Open conductance** parameter value.

If the **Threshold width** parameter is set to zero, the switch is closed if the voltage presented at the vT control port exceeds the value of the **Threshold** parameter.

If the **Threshold width** parameter is greater than zero, then switch conductance *G* varies smoothly between off-state and on-state values:

$$G = \frac{x}{R_{closed}} + (1 - x)G_{open}$$

$$\lambda = \frac{vT - \text{Threshold}}{\text{Threshold width}}$$

$$x = \begin{cases} 0 & \text{for } \lambda \leq 0 \\ 3\lambda^2 - 2\lambda^3 & \text{for } 0 < \lambda < 1 \\ 1 & \text{for } \lambda \geq 1 \end{cases}$$

The block uses the function $3\lambda^2 - 2\lambda^3$ because its derivative is zero for $\lambda = 0$ and $\lambda = 1$.

Defining a small positive **Threshold width** can help solver convergence in some models, particularly if the control port signal *vT* varies continuously as a function of other network variables. However, defining a nonzero threshold width precludes the solver making use of switched linear optimizations. Therefore, if the rest of your network is switched linear, MathWorks recommends that you set **Threshold width** to zero.

Optionally, you can add a delay between the point at which the voltage at vT passes the threshold and the switch opening or closing. To enable the delay, on the **Dynamics** tab, set the **Model dynamics** parameter to `Model turn-on and turn-off times`.

Parameters

- "Main Tab" on page 1-167
- "Dynamics Tab" on page 1-168

Main Tab

Closed resistance

Resistance between the c and s electrical ports when the switch is closed. The value must be greater than zero. The default value is `0.01` Ω.

Open conductance

Conductance between the c and s electrical ports when the switch is open. The value must be greater than zero. The default value is `1e-6` S.

Threshold

The threshold voltage for the control physical signal input vT above which the switch will turn on. The default value is `0` V.

Threshold width

The minimum increase in the control signal vT above the threshold value that will move the switch from fully open to fully closed. The default value is `0` V.

Dynamics Tab

Model dynamics

Select whether the block models a switching delay:

- `No dynamics` — Do not model the delay. This is the default option.
- `Model turn-on and turn-off times` — Use additional parameters to model a delay between the point at which the voltage at vT passes the threshold and the switch opening or closing.

Turn-on delay

Time between the input voltage exceeding the threshold voltage and the switch closing. This parameter is only visible when you select `Model turn-on and turn-off times` for the **Model dynamics** parameter. The value must be greater than zero. The default value is `1e-3` seconds.

Turn-off delay

Time between the input voltage falling below the threshold voltage and the switch opening. This parameter is only visible when you select `Model turn-on and turn-off times` for the **Model dynamics** parameter. The value must be greater than zero. The default value is `1e-3` seconds.

Initial input value, vT

The value of the physical signal input vT at time zero. This value is used to initialize the delayed control voltage parameter internally. This parameter is only visible when you select `Model turn-on and turn-off times` for the **Model dynamics** parameter. The default value is `0` V.

Ports

This block has the following ports:

vT

Physical signal that opens and closes the switch

c1, c2, s1, s2

Electrical conserving ports

See Also

DPDT Switch | SPDT Switch | SPST Switch | Switch

Eddy Current

Model eddy current losses
Library:　　　　Simscape / Electronics / Passive Devices

Description

The Eddy Current block models the effect of eddy currents by generating a magnetomotive force (MMF) that opposes changes in the magnetic flux. In addition, the block models parasitic effects using a series reluctance and a parallel leakage permeance.

Eddy currents in magnetic core materials are caused by time-varying magnetic fields. These changing field densities induce voltage potentials within the material, causing current to flow in closed loops. Such currents are usually undesired and cause heating of the magnetic material.

Use this block to add eddy current losses in the magnetic domain to a custom transformer or other magnetic component.

The eddy current component is sometimes referred to as magnetic inductance because it is the magnetic-domain analog to an inductor in the electrical domain.

Equations and Equivalent Circuit

This is the equivalent magnetic circuit for the block, including the eddy current path (bottom) and parallel leakage path (top).

Parasitic parallel
permeance

Parasitic series
reluctance

Ideal eddy
current
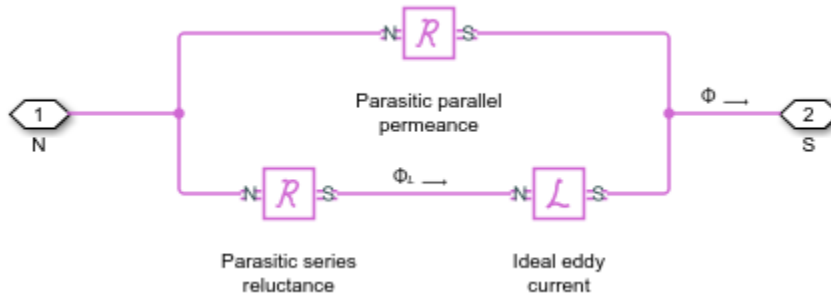
In the diagram:

- Φ is the total flux at the terminals. This flux is the summation of the eddy current loop and the parallel leakage path.

- Φ_L is the flux through the eddy current loop.

The block calculates the terminal MMF, F , and flux, Φ , as:

$$F = G_{\text{eddy}} \frac{d\Phi_L}{dt} + R \; \Phi_L$$
$$\Phi = F P + \Phi_L$$

where:

- G_{eddy} is the conductance of the eddy current loop.

- R is the parasitic series reluctance of the eddy current path.

- P is the parallel permeance.

Because the parasitic series reluctance and parallel permeance are lossless, the total dissipated power over the block is:

$$P_{diss} = G_{\text{eddy}} \left(\frac{d\Phi_L}{dt} \right)^2$$

1-171

Relating Transformer Electrical and Magnetic Models

In the electrical domain, eddy current losses are modeled using a parallel resistance across the primary winding. This is the equivalent circuit for a nonideal two-winding transformer.
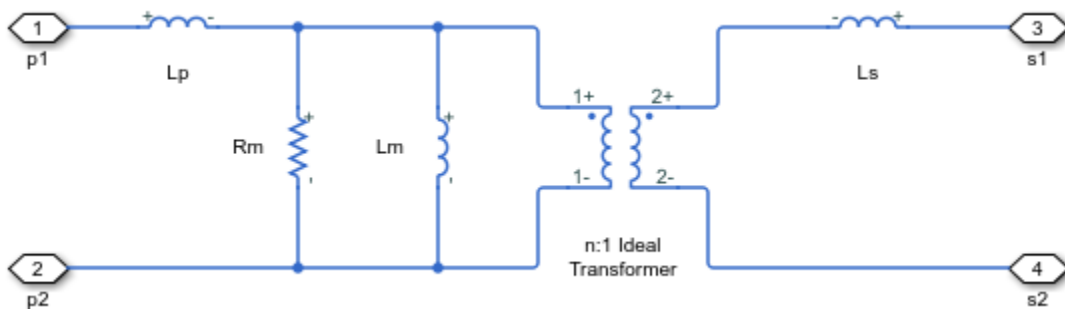
In the diagram:

- L_p and L_s are the self-inductances of the primary and secondary windings, respectively.
- L_m is the mutual inductance between the two windings.
- R_m is the mutual resistance between the two windings, caused by the eddy current losses.

This two-winding transformer can similarly be represented in the magnetic domain. This is the equivalent circuit.
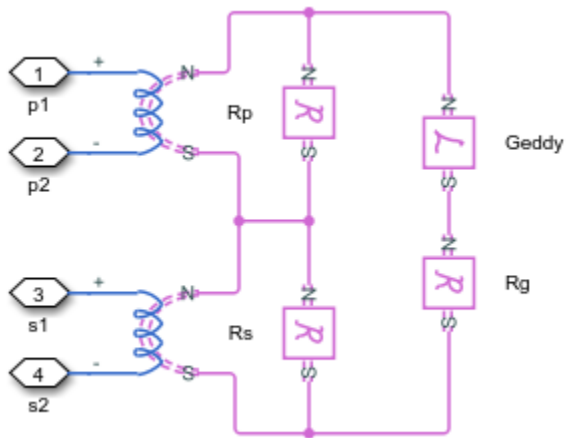
In the diagram:

- R_p and R_s are the reluctances associated with the primary and secondary windings, respectively.

- R_g is the reluctance associated with the magnetic coupling of the two windings.
- n is the turns ratio between the two windings.

The electrical- and magnetic-domain circuits are equivalent if:

$$R_g = \frac{n^2}{L_m}$$

$$R_p = \frac{n^2}{L_p}$$

$$R_s = \frac{1}{L_s}$$

$$G_{\text{eddy}} = \frac{n^2}{R_m}$$

You can use these relationships to calculate the equivalent two-winding transformer properties for one domain from the other.

Ports

Conserving

N — North terminal
magnetic

Magnetic conserving port associated with the north terminal of the block.

S — South terminal
magnetic

Magnetic conserving port associated with the south terminal of the block.

Parameters

Conductance of eddy current loop — Eddy current conductance
1 1/Ohm (default) | positive number

Conductance, G_{eddy}, of the eddy current loop. This component is the magnetic-domain analog of inductance in the electrical domain.

Parasitic series reluctance — Series reluctance
1e-3 1/H (default) | nonnegative number

Parasitic reluctance, R , of the eddy current loop.

Parasitic parallel permeance — Parallel leakage permeance
1e-6 H (default) | nonnegative number

Parasitic permeance, P , of the parallel leakage path. To aid simulation convergence in some circuit topologies, set this parameter to a small value.

References

[1] Brown, A. D., J. N. Ross, and K. G. Nichols. "Time-domain simulation of mixed nonlinear magnetic and electronic systems." *IEEE Transactions on Magnetics.* Vol. 37, Number 1, 2001, pp. 522-532.

See Also
Mutual Inductor | Nonlinear Transformer

Introduced in R2018a

Exponential Current Source

Model exponential pulse current source

Library

SPICE-Compatible Components/Sources

Description

The Exponential Current Source block represents a current source whose output current value is an exponential pulse as a function of time and is independent of the voltage across the terminals of the source. The following equations describe the current through the source as a function of time:

$$I_{out}\left(0 \le Time \le TDR\right)) = I1$$

$$I_{out}\left(TDR < Time \le TDF\right) = I1 + \left(I2 - I1\right) * \left(1 - e^{-(Time-TDR)/TR}\right)$$

$$I_{out}\left(TDF < Time\right) = I1 + \left(I2 - I1\right) * \left(e^{-(Time-TDF)/TF} - e^{-(Time-TDR)/TR}\right)$$

where:

- *I1* is the **Initial value, I1** parameter value.
- *I2* is the **Pulse value, I2** parameter value.
- *TDR* is the **Rise delay time, TDR** parameter value.
- *TR* is the **Rise time, TR** parameter value.
- *TDF* is the **Fall delay time, TDF** parameter value.
- *TF* is the **Fall time, TF** parameter value.

The block uses a small conductance internally to prevent numerical simulation issues. The conductance connects the + and - ports of the device and has a conductance *GMIN*:

- By default, *GMIN* matches the **Minimum conductance GMIN** parameter of the SPICE Environment Parameters block, whose default value is 1e–12.

- To change *GMIN*, add a SPICE Environment Parameters block to your model and set the **Minimum conductance GMIN** parameter to the desired value.

Parameters

Initial value, I1

> The value of the output current at time zero. The default value is `0` A.

Pulse value, I2

> The asymptotic value of the output current when the output is high. The default value is `0` A.

Rise delay time, TDR

> The rise time delay. The default value is `0` s.

Rise time, TR

> The time it takes the output current to rise from the **Initial Value, I1** value to the **Pulse Value, I2** value. The default value is `1e-09` s. The value must be greater than `0`.

Fall delay time, TDR

> The fall time delay. The default value is `0` s, which differs from the SPICE default value.

Fall time, TF

> The time it takes the output current to fall from the **Pulse value, I2** value to the **Initial value, I1** value. The default value is `1e-09` s. The value must be greater than `0`.

Ports

The block has the following ports:

+

    Positive electrical voltage.

-

    Negative electrical voltage.

See Also

Exponential Voltage Source

Exponential Voltage Source

Model exponential pulse voltage source

Library

SPICE-Compatible Components/Sources

Description

The Exponential Voltage Source block represents a voltage source whose output voltage value is an exponential pulse as a function of time and is independent of the current through the source. The following equations describe the output current as a function of time:

$$V_{out}\left(0 \le Time \le TDR\right)) = V1$$

$$V_{out}\left(TDR < Time \le TDF\right) = V1 + \left(V2 - V1\right) * \left(1 - e^{-(Time-TDR)/TR}\right)$$

$$V_{out}\left(TDF < Time\right) = V1 + \left(V2 - V1\right) * \left(e^{-(Time-TDF)/TF} - e^{-(Time-TDR)/TR}\right)$$

where:

- *V1* is the **Initial value, V1** parameter value.
- *V2* is the **Pulse value, V2** parameter value.
- *TDR* is the **Rise delay time, TDR** parameter value.
- *TR* is the **Rise time, TR** parameter value.
- *TDF* is the **Fall delay time, TDF** parameter value.
- *TF* is the **Fall time, TF** parameter value.

Parameters

Initial value, V1

The value of the output voltage at time zero. The default value is 0 V.

Pulse value, V2

The asymptotic value of the output voltage when the output is high. The default value is 0 V.

Rise delay time, TDR

The rise time delay. The default value is 0 s.

Rise time, TR

The time it takes the output voltage to rise from the **Initial value, I1** value to the **Pulse value, I2** value. The default value is 1e-09 s. The value must be greater than 0.

Fall delay time, TDR

The fall time delay. The default value is 0 s.

Fall Time, TF

The time it takes the output voltage to fall from the **Pulse value, I2** value to the **Initial value, I1** value. The default value is 1e-09 s. The value must be greater than 0.

Ports

The block has the following ports:

+

Positive electrical voltage.

-

Negative electrical voltage.

See Also

Exponential Current Source

Fault

Model electrical fault

Library

Passive Devices

Description

The Fault block allows you to represent an electrical fault as an instantaneous change in resistance. You can use it to replicate both open-circuit and short-circuit fault behaviors. The block can trigger fault events:

- At a specific time
- When a predefined voltage range or current range is exceeded
- When an external trigger signal goes high or low

Optionally, the external trigger option also permits the fault to be reset when the trigger signal reverts. You can enable or disable all three trigger mechanisms separately, or use them together if more than one trigger mechanism is required in a simulation.

When no fault is triggered, the resistance between the two electrical ports is defined by the **Unfaulted resistance** parameter value. The default value for this parameter is `inf` ohms, i.e. the ports are open-circuit. When a fault is triggered, the block changes the resistance between the two electrical ports to the **Faulted resistance** value. The default value for this parameter is `1e-3` ohms, i.e. the ports are short-circuited.

You can choose whether to issue an assertion when a fault occurs, by using the **Reporting when a fault occurs** parameter. The assertion can take the form of a warning or an error. By default, the block does not issue an assertion.

The physical output X represents the fault state; it is 1 if the block is faulted, and 0 otherwise. The physical signal input F is the external fault trigger signal and is used only if **Enable external fault trigger** is set to `Yes`.

Parameters

- "Main Tab" on page 1-182
- "Temporal Trigger Tab" on page 1-182
- "Behavioral Trigger Tab" on page 1-183
- "External Trigger Tab" on page 1-183

Main Tab

Unfaulted resistance

Resistance between the + and – ports when there is no fault. The default value is `inf` Ω.

Faulted resistance

Resistance between the + and – ports when the block is in the faulted state. The default value is `1e-3` Ω.

Reporting when a fault occurs

Choose whether to issue an assertion when a fault occurs:

- `None` — The block does not issue an assertion. This is the default.
- `Warn` — The block issues a warning.
- `Error` — Simulation stops with an error.

Temporal Trigger Tab

Enable temporal fault trigger

Select `Yes` to enable time-based fault triggering. The default value is `No`.

Simulation time for fault event

Set the simulation time at which you want the block to enter the fault state. This parameter is visible only if the **Enable temporal fault trigger** parameter is set to `Yes`. The default value is `1` s.

Behavioral Trigger Tab

Enable behavioral fault trigger

Select `Yes` to enable behavioral fault triggering. The default value is `No`.

Permissible voltage range

Specify a vector of length 2 that defines the permissible voltage range. If the voltage exceeds this range for longer than the **Time to fail when exceeding voltage range** parameter value, then the block enters the fault state. This parameter is visible only if the **Enable behavioral fault trigger** parameter is set to `Yes`. The default value is `[-100, 100]` V.

Time to fail when exceeding voltage range

Set the maximum length of time that the voltage can exceed the permissible voltage range without triggering the fault. This parameter is visible only if the **Enable behavioral fault trigger** parameter is set to `Yes`. The default value is `1` s.

Permissible current range

Specify a vector of length 2 that defines the permissible current range. If the current exceeds this range for longer than the **Time to fail when exceeding current range** parameter value, then the block enters the fault state. This parameter is visible only if the **Enable behavioral fault trigger** parameter is set to `Yes`. The default value is `[-1, 1]` A.

Time to fail when exceeding current range

Set the maximum length of time that the current can exceed the permissible current range without triggering the fault. This parameter is visible only if the **Enable behavioral fault trigger** parameter is set to `Yes`. The default value is `1` s.

External Trigger Tab

Enable external fault trigger

Select `Yes` to enable external fault triggering. The physical signal input F provides the external fault trigger signal. The default value is `No`.

External fault trigger

Choose the fault condition:

- `Faulted if F >= Fault threshold` — The fault occurs when the external signal value becomes greater than, or equal to, the **Fault threshold** parameter value. This is the default.

- `Faulted if F <= Fault threshold` — The fault occurs when the external signal value becomes less than, or equal to, the **Fault threshold** parameter value.

This parameter is visible only if the **Enable external fault trigger** parameter is set to `Yes`.

Fault threshold

The threshold value that triggers the fault when the external signal crosses it in the direction, specified by the fault condition. This parameter is visible only if the **Enable external fault trigger** parameter is set to `Yes`. The default value is `0.5`.

Fault resets when fault trigger reverts

Select `Yes` to have the fault reset when the trigger signal reverts. The default value is `No`. This parameter is visible only if the **Enable external fault trigger** parameter is set to `Yes`.

Ports

The block has the following ports:

+

Positive electrical port.

-

Negative electrical port.

F

Physical signal input port that provides the external fault trigger signal.

X

Physical signal output port that indicates the fault state. Outputs 1 if the block is faulted, and 0 otherwise.

See Also

Resistor

Introduced in R2015b

FEM-Parameterized Linear Actuator

Model linear actuator defined in terms of magnetic flux
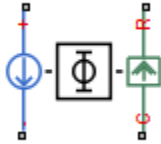
Library

Translational Actuators

Description

The FEM-Parameterized Linear Actuator block implements a model of a linear actuator defined in terms of magnetic flux. Use this block to model custom solenoids and linear motors where magnetic flux depends on both distance and current. You parameterize the block using data from a third-party Finite Element Magnetic (FEM) package.

The block has two options for the electrical equation. The first, `Define in terms of dPhi(i,x)/dx and dPhi(i,x)/di`, defines the current in terms of partial derivatives of the magnetic flux (Φ) with respect to distance (x) and current (i), the equations for which are:

$$\frac{di}{dt} = \left(v - iR - \frac{\partial \Phi}{\partial x} \frac{dx}{dt} \right) / \frac{\partial \Phi}{\partial i}$$

The second option, `Define in terms of Phi(i,x)`, defines the voltage across the component directly in terms of the flux, the equation for which is:

$$v = iR + \frac{d}{dt} \Phi\left(x, i \right)$$

Numerically, defining the electrical equation in terms of flux partial derivatives is better because the back-emf is piecewise continuous. If using the flux directly, using a finer grid size for current and position will improve results, as will selecting cubic or spline interpolation.

In both cases, you have an option to either directly specify the force as a function of current and position, by using the **Force matrix, F(i,x)** parameter, or have the block automatically calculate the force matrix.

If entering the electromagnetic force data directly, you can either use data supplied by the finite element magnetic package (which you used to determine the flux) or calculate the force from the flux with following equation:

$$F = \int\limits_{0}^{i} \frac{\partial \Phi(x,i)}{\partial x} di$$

See the Finite Element Parameterized Solenoid example model and its initialization file `elec_fem_solenoid_ini.m` for an example of how to implement this type of integration in MATLAB.

Alternatively, the block can automatically calculate the force matrix from the flux information that you provide. To select this option, set the **Calculate force matrix?** parameter to `Yes`. The force matrix calculation occurs at model initialization based on current block flux linkage information. The force is calculated by numerically integrating the rate of change of flux linkage with respect to position over current, according to the preceding equation. If the **Electrical model** parameter is set to `Define in terms of Phi(i,x)`, then the block must first estimate the **Flux partial derivative wrt displacement, dPhi(i,x)/dx** parameter value from the flux linkage data. When doing this, the block uses the interpolation method specified by the **Interpolation method** parameter. Typically, the `Smooth` option is most accurate, but the `Linear` option is most robust.

You can define Φ and its partial derivatives for just positive, or positive and negative currents. If defining for just positive currents, then the block assumes that $\Phi(-i,x) = -\Phi(i,x)$. Therefore, if the current vector is positive only:

- The first current value must be zero.
- The flux corresponding to zero current must be zero.
- The partial derivative of flux with respect to displacement must be zero for zero current.

To model a linear motor with a repeated flux pattern, set the **Flux dependence on displacement** parameter to `Cyclic`. When selecting this option, the force and flux (or force and flux partial derivatives depending on the option chosen) must have identical first and last columns.

Thermal Port

The block has an optional thermal port, hidden by default. To expose the thermal port, right-click the block in your model, and then from the context menu select **Simscape > Block choices > Show thermal port**. This action displays the thermal port H on the block icon, and adds the **Temperature Dependence** and **Thermal Port** tabs to the block dialog box.

Use the thermal port to simulate the effects of copper resistance losses that convert electrical power to heat. For more information on using thermal ports and on the **Temperature Dependence** and **Thermal Port** tab parameters, see "Simulating Thermal Effects in Rotational and Translational Actuators".

Basic Assumptions and Limitations

This block has the following limitations:

- It is imperative that you supply a consistent set of force and flux data. There is no checking that the force matrix is consistent with the flux data.
- When driving the FEM-Parameterized Linear Actuator block via a series inductor, you may need to include a parallel conductance in the inductor component.

Parameters

- "Magnetic Force Tab" on page 1-187
- "Mechanical Tab" on page 1-191

Magnetic Force Tab

Electrical model

Select one of the following parameterization options, based on the underlying electrical model:

- `Define in terms of dPhi(i,x)/dx and dPhi(i,x)/di` — Define the current through the block in terms of partial derivatives of the magnetic flux with respect to distance and current. This is the default method.
- `Define in terms of Phi(i,x)` — Define the voltage across the block terminals directly in terms of the flux.

Current vector, i

Specify a vector of monotonically increasing current values corresponding to your force-flux data. If you specify positive currents only, the first element must be zero. The default value is [`0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1`] A.

Displacement vector, x

Specify a vector of monotonically increasing displacement values corresponding to your force-flux data. The default value is [`0 0.05 0.1 0.15 0.2`] m/m.

Flux partial derivative wrt current, dPhi(i,x)/di

Specify a matrix of the flux partial derivatives with respect to current. This parameter is visible only if **Electrical model** is set to `Define in terms of dPhi(i,x)/dx and dPhi(i,x)/di`. The default value, in Wb/A, is:

```
[ 0.104 0.098 0.091 0.085 0.078;
0.095 0.089 0.084 0.079 0.073;
0.085 0.081 0.077 0.073 0.069;
0.076 0.073 0.07 0.067 0.064;
0.067 0.065 0.063 0.061 0.06;
0.057 0.057 0.056 0.056 0.055;
0.048 0.049 0.049 0.05 0.05;
0.038 0.04 0.042 0.044 0.046;
0.029 0.032 0.035 0.038 0.041;
0.02 0.024 0.028 0.033 0.037;
0.01 0.016 0.021 0.027 0.032 ]
```

Flux partial derivative wrt displacement, dPhi(i,x)/dx

Specify a matrix of the flux partial derivatives with respect to displacement. This parameter is visible only if **Electrical model** is set to `Define in terms of dPhi(i,x)/dx and dPhi(i,x)/di`. The default value, in Wb/m, is:

```
[ 0 0 0 0 0;
-11.94 -10.57 -9.19 -7.81 -6.43;
-21.17 -19.92 -18.67 -17.42 -16.16;
-27.99 -26.87 -25.75 -24.62 -23.5;
-32.42 -31.43 -30.43 -29.43 -28.44;
-34.46 -33.59 -32.72 -31.85 -30.98;
```

```
-34.09 -33.35 -32.61 -31.87 -31.12;
-31.33 -30.72 -30.1 -29.49 -28.87;
-26.17 -25.68 -25.2 -24.71 -24.22;
-18.62 -18.26 -17.9 -17.54 -17.18;
-8.66 -8.43 -8.2 -7.97 -7.73 ]
```

Flux linkage matrix, Phi(i,x)

Specify a matrix of the total flux linkage, that is, flux times the number of turns. This parameter is visible only if **Electrical model** is set to `Define in terms of Phi(i,x)`. The default value, in Wb, is:

```
[ 0 0 0 0 0;
0.0085 0.0079 0.0075 0.0071 0.0067;
0.0171 0.016 0.0151 0.0143 0.0137;
0.0254 0.0239 0.0226 0.0215 0.0206;
0.033 0.0312 0.0297 0.0283 0.0271;
0.0396 0.0377 0.036 0.0345 0.0331;
0.0452 0.0433 0.0415 0.0399 0.0384;
0.0495 0.0478 0.0461 0.0446 0.0431;
0.0526 0.0512 0.0498 0.0485 0.0472;
0.0545 0.0537 0.0528 0.0519 0.0508;
0.0554 0.0553 0.0551 0.0548 0.0542 ]
```

Calculate force matrix?

Specify the way of providing the electromagnetic force data:

- `No — specify directly` — Enter the electromagnetic force data directly, by using the **Force matrix, F(i,x)** parameter. This is the default option.

- Yes — The block calculates the force from the flux linkage information, as a function of current and displacement.

Force matrix, F(i,x)

Specify a matrix of the electromagnetic force applied to the plunger or moving part. This parameter is visible only if **Calculate force matrix?** is set to `No — specify directly`. The default value, in N, is:

```
[ 0 0 0 0 0;
-0.6 -0.5 -0.4 -0.3 -0.3;
-2.3 -2 -1.7 -1.4 -1.2;
-4.9 -4.3 -3.7 -3.2 -2.7;
-8.3 -7.3 -6.4 -5.5 -4.7;
-12.2 -10.7 -9.4 -8.2 -7.2;
-16.2 -14.4 -12.7 -11.3 -10;
```

```
-20 -17.9 -15.9 -14.3 -12.9;
-23.3 -20.9 -18.8 -17.1 -15.7;
-25.7 -23.1 -21.1 -19.4 -18.2;
-26.5 -24.1 -22.2 -20.9 -20.1 ]
```

Flux dependence on displacement

Specify the flux pattern:

- Unique — No flux pattern present. This is the default option.
- Cyclic — Select this option to model a linear motor with a repeated flux pattern. The force and flux (or force and flux partial derivatives, depending on the **Electrical model** option chosen) must have identical first and last columns.

Interpolation method

Select one of the following interpolation methods for approximating the output value when the input value is between two consecutive grid points:

- Linear — Select this option to get the best performance.
- Smooth — Select this option to produce a continuous surface with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (2D) block reference page.

Extrapolation method

Select one of the following extrapolation methods for determining the output value when the input value is outside the range specified in the argument list:

- Linear — Select this option to produce a surface with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region.
- Nearest — Select this option to produce an extrapolation that does not go above the highest point in the data or below the lowest point in the data.

For more information on extrapolation algorithms, see the PS Lookup Table (2D) block reference page.

This parameter is not available if you set the **Flux dependence on displacement** parameter to Cyclic.

Winding resistance

Total resistance of the electrical winding. The default value is 14 Ohm.

Mechanical Tab

Damping

Linear damping. The default value is 1 N/(m/s). The value can be zero.

Plunger mass

Mass of the moving part, which corresponds to mechanical translational port R. The default value is 0.05 kg. The value can be zero.

Minimum stroke

The stroke at which the lower mechanical end stop is applied. The default value is 0. The value can be -Inf.

Maximum stroke

The stroke at which the upper mechanical end stop is applied. The default value is 0.2 mm. The value can be Inf.

Initial plunger position

Position of the plunger at the start of the simulation. The default value is 0 mm.

Initial plunger velocity

Speed of the plunger at the start of the simulation. The default value is 0 mm/s.

Contact stiffness

Contact stiffness between plunger and end stops. The default value is 1e8 N/m.

Contact damping

Contact damping between plunger and end stops. The default value is 1e4 N/(m/s).

Ports

This block has the following ports:

+

Positive electrical conserving port

-

Negative electrical conserving port

C

Mechanical translational conserving port connected to the actuator case

R

Mechanical translational conserving port connected to the plunger

Examples

The Solenoid Parameterized with FEM Data example illustrates the use and parameterization options of this block.

See Also

FEM-Parameterized Rotary Actuator | Solenoid

FEM-Parameterized PMSM

Permanent magnet synchronous motor defined in terms of magnetic flux linkage

Library

Rotational Actuators

Description

The FEM-Parameterized PMSM block implements a model of a permanent magnet synchronous motor (PMSM) defined in terms of magnetic flux linkage. You parameterize the block by providing tabulated data of motor magnetic flux as a function of current and rotor angle. This is the way third-party Finite Element Magnetic (FEM) packages usually export flux information. Because of the tabulated form, the flux can vary in a nonlinear way on both rotor angle and current. You can therefore use this block to model PMSM with trapezoidal back-emf profile, sometimes called brushless DC motor, as well as regular PMSM.

The figure shows the equivalent circuit for a wye-connected PMSM. The rotor angle is zero when the permanent magnet flux aligns with the A-phase magnetic axis.
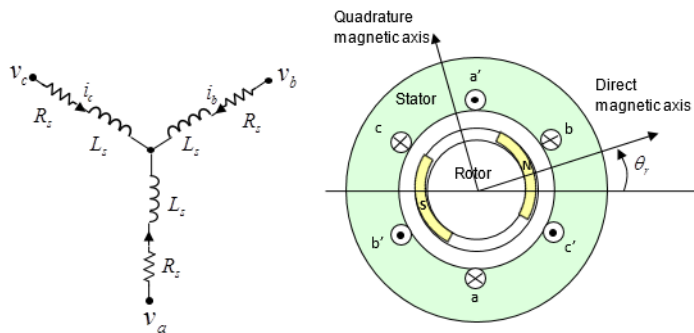
In practice, the flux linking each of the three windings depends on all three currents and rotor angle. Tabulating flux as a function of four independent variables might lead to simulation inefficiency and significant memory requirements to manage the data. The block, therefore, lets you select between the following parameterization methods for flux and torque:

- **2-D data** — 2-D table lookup, with options to tabulate in terms of current and rotor angle, or in terms of d-axis and q-axis currents. The first option assumes constant mutual inductance and supports nonsinusoidal back emf profiles. The second option assumes a sinusoidal back emf and captures saturation effects for interior PMSMs (IPMSMs).

- **3-D A-phase data** — 3-D table lookup, based on direct current, quadrature current, and rotor angle. You provide the flux lookup data for the a phase. The block uses Park transform to map the three stator winding currents to direct and quadrature currents. This method reduces the data complexity, as compared to the 4-D table lookup, and therefore results in improved simulation performance.

- **4-D A-phase data** — 4-D table lookup, based on the three stator winding currents and the rotor angle. You provide the flux lookup data for the a phase. This model has the best fidelity of the three, but also is the most costly in terms of simulation performance and memory requirements.

- **3-D DQ data** — 3-D table lookup, based on direct current, quadrature current, and rotor angle. You provide the flux lookup data for the d and q phases. The block uses Park transform to map the three stator winding currents to direct and quadrature currents. This method reduces the data complexity, as compared to the 4-D table lookup, and therefore results in improved simulation performance.

By default, all of the block variants implement a wye-wound configuration for the stator windings. The 3-D DQ data variant also supports a delta-wound configuration, selectable using the **Winding type** parameter. When in the delta-wound configuration, the *a* phase is connected between ports **a** and **b**, the *b* phase between ports **b** and **c** and the *c* phase between ports **c** and **a**.

To access these parameterization methods, right-click the block in your model, select **Simscape** > **Block choices**, and then select the desired block variant, with or without thermal ports. By default, the thermal ports are not exposed. For more information, see "Thermal Ports" on page 1-201.

2-D Data Model with Constant Mutual Inductance

In this 2-D flux data model, the flux linking each winding is assumed to depend nonlinearly only on the current in that same winding, plus the rotor angle. In practice, this is a reasonable assumption for many permanent magnet synchronous motors; however, it is less accurate for switched reluctance motors. Given this assumption, the fluxes in the three windings are:

$$
\begin{bmatrix} \phi_a \\ \phi_b \\ \phi_c \end{bmatrix} = \begin{bmatrix} 0 & -M_s & -M_s \\ -M_s & 0 & -M_s \\ -M_s & -M_s & 0 \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} \phi\left(i_a,\theta_r\right) \\ \phi\left(i_b,\theta_r - 2\pi / \left(3N\right)\right) \\ \phi\left(i_c,\theta_r - 4\pi / \left(3N\right)\right) \end{bmatrix}
$$

where $\phi\left(\theta_r, i_a\right)$ is the flux linkage for the A-phase winding as a function of rotor angle and A-phase current. $\Theta_r = 0$ corresponds to the rotor *d*-axis aligning with the A-phase positive magnetic flux direction. M_s is the stator-stator mutual inductance.

For improved numerical performance, the equations implemented in the block actually

work with the partial derivatives of flux linkage with respect to current, $\partial\phi(i,\theta_r)/\partial i$, and

rotor angle, $\partial\phi(i,\theta_r)/\partial\theta_r$, rather than the flux directly. If your FEM package does not export these partial derivatives, you can determine them using a MATLAB script. See the Finite Element Parameterized Solenoid example model and its supporting MATLAB script for an example of how to do this.

The electrical equations for the block, defined in terms of flux partial derivatives, are:

$$v_a = \frac{\partial \phi}{\partial i_a}\frac{di_a}{dt} + \frac{\partial \phi}{\partial \theta_r}\frac{d\theta_r}{dt} - M_s\left(\frac{di_b}{dt} + \frac{di_c}{dt}\right) + R_s i_a$$

$$v_b = \frac{\partial \phi}{\partial i_b}\frac{di_b}{dt} + \frac{\partial \phi}{\partial \theta_r}\frac{d\theta_r}{dt} - M_s\left(\frac{di_a}{dt} + \frac{di_c}{dt}\right) + R_s i_b$$

$$v_c = \frac{\partial \phi}{\partial i_c}\frac{di_c}{dt} + \frac{\partial \phi}{\partial \theta_r}\frac{d\theta_r}{dt} - M_s\left(\frac{di_a}{dt} + \frac{di_b}{dt}\right) + R_s i_c$$

where

- v_a, v_b, v_c are the voltages applied to the A, B, and C stator windings.
- i_a, i_b, i_c are the stator currents in each of the three windings.
- R_s is the resistance of each of the stator windings.
- M_s is the stator-stator mutual inductance.
- $\partial \phi / \partial i_a$, $\partial \phi / \partial i_b$, $\partial \phi / \partial i_c$ are the partial derivatives of flux linkage with respect to stator current in each of the three windings.
- $\partial \phi / \partial \theta_r$ is the partial derivative of flux linkage with respect to rotor angle.

The block can automatically calculate the torque matrix from the flux information that you provide. Alternatively, you can set the **Calculate torque matrix?** parameter to No and directly specify the torque as a function of current and rotor angle. See the FEM-Parameterized Rotary Actuator block reference page for more information.

2-D Data Model with Sinusoidal Back EMF

In this 2-D flux data model, the flux linking each winding is assumed to depend nonlinearly on all stator winding currents, plus it is assumed that the permanent magnet flux linkage is sinusoidal. Interior magnet PMSMs (or IPMSMs) usually fit this assumption well. The equations are:

$$\begin{bmatrix} \phi_d \\ \phi_q \end{bmatrix} = \begin{bmatrix} L_d\left(i_d,i_q\right) & \\ & L_q\left(i_d,i_q\right) \end{bmatrix}\begin{bmatrix} i_d \\ i_q \end{bmatrix} + \begin{bmatrix} \phi_m\left(i_d,i_q\right) \end{bmatrix}$$

$$T = \frac{3}{2}N\left(i_q\left(i_d L_d\left(i_d,i_q\right) + \phi_m\left(i_d,i_q\right)\right) - i_d i_q L_q\left(i_d,i_q\right)\right)$$

where

- i_d and i_q are the *d*-axis and *q*-axis currents, respectively.

- ϕ_d and ϕ_q are the *d*-axis and *q*-axis flux linkages, respectively.

- ϕ_m is the permanent magnet flux linkage.

- L_d and L_q are the *d*-axis and *q*-axis inductances, respectively. They are assumed to depend on the *d*-axis and *q*-axis currents.

- *N* is the number of pole pairs.

- *T* is the electrical torque.

3-D A-Phase Data Model Using Park Transform

Working with four-dimensional data has both a simulation performance cost and a memory cost. To reduce the table dimension to three-dimensions, the 3-D data model uses Park transform to map the three currents to direct and quadrature currents:

$$\begin{bmatrix} i_d \\ i_q \end{bmatrix} = \frac{2}{3} \begin{bmatrix} \cos\theta_e & \cos\left(\theta_e - \frac{2\pi}{3}\right) & \cos\left(\theta_e + \frac{2\pi}{3}\right) \\ -\sin\theta_e & -\sin\left(\theta_e - \frac{2\pi}{3}\right) & -\sin\left(\theta_e + \frac{2\pi}{3}\right) \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix}$$

In the general case, Park transform maps to direct, quadrature, and zero-sequence currents. However, the zero-sequence current is typically small under normal operating conditions. Therefore, the model neglects the dependence of the flux linkage terms on zero-sequence current, and determines the flux linkage in terms of just direct and quadrature currents plus rotor angle. The flux equation for the 3-D data model is:

$$\begin{bmatrix} \phi_a \\ \phi_b \\ \phi_c \end{bmatrix} = \begin{bmatrix} \phi\left(i_d, i_q, \theta_r\right) \\ \phi\left(i_d, i_q, \theta_r - 2\pi/(3N)\right) \\ \phi\left(i_d, i_q, \theta_r - 4\pi/(3N)\right) \end{bmatrix}$$

The electrical equations for the block are also defined in terms of flux partial derivatives, similar to the 4-D data model. You can calculate 3-D flux linkage partial derivative data from 4-D flux linkage data using `elec_calculateFluxPartialDerivatives`.

4-D A-Phase Data Model

The flux linking each of the windings is a function of the current in that winding, the currents in the other two windings, and the rotor angle. For full accuracy, the 4-D flux data model assumes that the flux linkage is a function of the three currents and the rotor angle, therefore performing four-dimensional table lookups. The flux equation is:

$$\begin{bmatrix} \phi_a \\ \phi_b \\ \phi_c \end{bmatrix} = \begin{bmatrix} \phi\left(i_a, i_b, i_c, \theta_r\right) \\ \phi\left(i_b, i_c, i_a, \theta_r - 2\pi / \left(3N\right)\right) \\ \phi\left(i_c, i_a, i_b, \theta_r - 4\pi / \left(3N\right)\right) \end{bmatrix}$$

where

- ϕ_a, ϕ_b, ϕ_c are the flux linkages for the A, B, and C stator windings.
- i_a, i_b, i_c are the stator currents in each of the three windings.
- Θ_r is the rotor angle. $\Theta_r = 0$ corresponds to the case where the permanent magnet flux is aligned with the A-phase stator winding flux.
- N is the number of pole pairs.

Flux linkage data is assumed cyclic with Θ_r. If, for example, the motor has six pole pairs, then the range for the data is $0 \leq \Theta_r \leq 60°$. You must provide data both at 0 and 60 degrees, and because the data is cyclic, the flux linkage partial derivatives must be the same at these two end points.

The torque equation is:

$$\tau = T\left(i_a, i_b, i_c, \theta_r\right)$$

The 4-D data model does not have an option for the block to determine torque from flux linkage. Because of the increased numerical overhead in the 4-D case, it is better to precalculate the torque just once, rather than calculate it every time you run the simulation.

For improved numerical performance, the equations implemented in the block actually work with the partial derivatives of flux linkage with respect to the three currents and the rotor angle, rather than the flux directly. If your FEM package does not export these partial derivatives, you can determine them using `elec_calculateFluxPartialDerivatives`.

The electrical equations for the block, defined in terms of flux partial derivatives, are:

$$v_a = \frac{\partial \phi_a}{\partial i_a}\frac{di_a}{dt} + \frac{\partial \phi_a}{\partial i_b}\frac{di_b}{dt} + \frac{\partial \phi_a}{\partial i_c}\frac{di_c}{dt} + \frac{\partial \phi_a}{\partial \theta_r}\frac{d\theta_r}{dt} + R_s i_a$$

$$v_b = \frac{\partial \phi_b}{\partial i_a}\frac{di_a}{dt} + \frac{\partial \phi_b}{\partial i_b}\frac{di_b}{dt} + \frac{\partial \phi_b}{\partial i_c}\frac{di_c}{dt} + \frac{\partial \phi_b}{\partial \theta_r}\frac{d\theta_r}{dt} + R_s i_b$$

$$v_c = \frac{\partial \phi_c}{\partial i_a}\frac{di_a}{dt} + \frac{\partial \phi_c}{\partial i_b}\frac{di_b}{dt} + \frac{\partial \phi_c}{\partial i_c}\frac{di_c}{dt} + \frac{\partial \phi_c}{\partial \theta_r}\frac{d\theta_r}{dt} + R_s i_c$$

where

- v_a, v_b, v_c are the voltages applied to the A, B, and C stator windings.
- i_a, i_b, i_c are the stator currents in each of the three windings.
- R_s is the resistance of each of the stator windings.

3-D DQ Data Model Using Park Transform

Similar to the 3-D A-Phase Data model, the 3-D DQ Data model uses Park transform to map the the stator phase currents to the direct and armature currents.

$$\begin{bmatrix} i_d \\ i_q \end{bmatrix} = \frac{2}{3} \begin{bmatrix} \cos\theta_e & \cos\left(\theta_e - \frac{2\pi}{3}\right) & \cos\left(\theta_e + \frac{2\pi}{3}\right) \\ -\sin\theta_e & -\sin\left(\theta_e - \frac{2\pi}{3}\right) & -\sin\left(\theta_e + \frac{2\pi}{3}\right) \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix}$$

The electrical equations for the block, defined in terms of flux partial derivatives, are:

$$v_d = \frac{\partial \phi_d}{\partial \theta_r}\frac{d\theta_r}{dt} + \frac{\partial \phi_d}{\partial i_d}\frac{di_d}{dt} + \frac{\partial \phi_d}{\partial i_q}\frac{di_q}{dt} + R_s i_d$$

$$v_q = \frac{\partial \phi_q}{\partial \theta_r}\frac{d\theta_r}{dt} + \frac{\partial \phi_q}{\partial i_d}\frac{di_d}{dt} + \frac{\partial \phi_q}{\partial i_q}\frac{di_q}{dt} + R_s i_q$$

Where:

- φ_d and φ_q are the d- and q-axis flux linkages, which you provide as lookup table data.

- v_d and v_q are the voltages applied to the d and q axes.
- i_d and i_q are the d- and q-axis currents.
- R_s is the resistance of each of the stator windings.
- θ_r is the rotor angle.

The inverse Park transform gives the three winding voltages from the direct and quadrature voltages:

$$
\begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} \cos\theta_e & -\sin\theta_e \\ \cos(\theta_e - 2\pi/3) & -\sin(\theta_e - 2\pi/3) \\ \cos(\theta_e + 2\pi/3) & -\sin(\theta_e + 2\pi/3) \end{bmatrix} \begin{bmatrix} v_d \\ v_q \end{bmatrix}
$$

Where:

-
$$
\begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} v_a \\ v_b \\ v_c \end{bmatrix}
$$
for the wye configuration.

-
$$
\begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} v_a - v_b \\ v_b - v_c \\ v_c - v_a \end{bmatrix}
$$
for the delta configuration.

Calculating Iron Losses

Regardless of the parameterization methods for flux and torque, all block variants use the same iron losses model, which is based on the work of Mellor [1]. Iron losses are divided into two terms, one representing the main magnetizing path, and the other representing the cross-tooth tip path that becomes active during field weakened operation.

The term representing the main magnetizing path depends on the induced RMS stator voltage, $V_{m_{rms}}$:

$$
P_{OC}\left(V_{m_{rms}}\right) = \frac{a_h}{k} V_{m_{rms}} + \frac{a_j}{k^2} V_{m_{rms}}^2 + \frac{a_{ex}}{k^{1.5}} V_{m_{rms}}^{1.5}
$$

This is the dominant term during no-load operation. k is the back emf constant relating

RMS volts per Hz. It is defined as $k = V_{m_{rms}} / f$, where f is the electrical frequency. The first term on the right-hand side is the magnetic hysteresis loss, the second is the eddy current loss and the third is the excess loss. The three coefficients appearing on the numerators are derived from the values that you provide for the open-circuit hysteresis, eddy, and excess losses.

The term representing the cross-tooth tip path becomes important when a demagnetizing field is set up and can be determined from a finite element analysis short-circuit test. It

depends on the RMS emf associated with the cross-tooth tip flux, $V_{d_{rms}}^*$:

$$P_{SC}\left(V_{d_{rms}}^*\right) = \frac{b_h}{k} V_{d_{rms}}^* + \frac{b_j}{k^2} V_{d_{rms}}^{*2} + \frac{b_{ex}}{k^{1.5}} V_{d_{rms}}^{*1.5}$$

The three numerator terms are derived from the values you provide for the short-circuit hysteresis, eddy, and excess losses.

Thermal Ports

The block has four optional thermal ports, one for each of the three windings and one for the rotor. These ports are hidden by default. To expose the thermal ports, right-click the block in your model, select **Simscape > Block choices**, and then select the desired block variant with thermal ports: **2-D data | Show thermal port**, **3-D A-phase data | Show thermal port**, **4-D A-phase data | Show thermal port**, or **3-D DQ data | Show thermal port**. This action displays the thermal ports on the block icon, and adds the **Temperature Dependence** and **Thermal Port** tabs to the block dialog box. These tabs are described further on this reference page.

Use the thermal ports to simulate the effects of copper resistance and iron losses that convert electrical power to heat. For more information on using thermal ports in actuator blocks, see "Simulating Thermal Effects in Rotational and Translational Actuators".

Basic Assumptions and Limitations

This block has the following limitations:

- For the 2-D data model, the stator-stator mutual inductance, defined by the **Stator mutual inductance, Ms** parameter value, is constant during simulation and does not vary with rotor angle. This means that the block is suitable for modeling most PMSM and brushless DC motors, but not switched reluctance motors.

- The 3-D and 4-D data models assume symmetry, so that the flux linkage dependency on currents and rotor angle for windings B and C can be determined from that for winding A.

- For the 3-D DQ data model, you do not provide information about zero-sequence inductance. As a result, the block does not expose its neutral port and machine currents are always balanced.

- For the 4-D data model, consider memory requirements when fixing the independent parameter values (three currents and rotor angles). The linear interpolation option uses less memory, but the smooth interpolation option is more accurate for a given independent parameter spacing.

- The iron losses model assumes sinusoidal currents.

Parameters

Electrical (2-D Data Variant)

This configuration of the **Electrical** tab corresponds to the 2-D Data block variants, with or without thermal ports. If you are using the 3-D A-Phase Data, 4-D A-Phase Data, or 3-D DQ Data variant of the block, see "Electrical (3-D A-Phase Data Variant)" on page 1-206, "Electrical (4-D A-Phase Data Variant)" on page 1-207, or "Electrical (3-D DQ Data Variant)" on page 1-209 respectively.

Parameterization

Select the parameterization method:

- `Assume constant mutual inductance - tabulate with phase current and rotor angle` — This method assumes that the flux linking each winding depends nonlinearly only on the current in that same winding, plus the rotor angle.

- `Assume sinusoidal back emf - tabulate with d- and q-axis currents` — This method assumes that the flux linking each winding depends nonlinearly on all stator winding currents. It also assumes that the permanent

magnet flux linkage is sinusoidal. This option is usually a good fit for interior magnet PMSMs (or IPMSMs).

Current vector, i

Vector of currents corresponding to the provided flux linkage partial derivatives. The current vector must be two-sided (have positive and negative values). The default value is `[-2, 0, 2]` A.

This parameter is visible only if **Parameterization** is set to `Assume constant mutual inductance - tabulate with phase current and rotor angle`.

Rotor angle vector, theta

Vector of rotor angles corresponding to the provided flux linkage partial derivatives. The vector must start at zero. This value corresponds to the angle where the A-phase magnetic flux aligns with the rotor permanent magnetic peak flux direction (the direct-axis, or *d*-axis). The last value, Θ_{max}, must be the rotor angle where the flux linkage pattern peaks again. Therefore, the number of pole pairs is $360/\Theta_{max}$ if Θ_{max} is expressed in degrees. The default value is `[0, 20, 40, 60]` deg, which corresponds to a 6 pole-pair motor.

This parameter is visible only if **Parameterization** is set to `Assume constant mutual inductance - tabulate with phase current and rotor angle`.

Flux linkage partial derivative wrt current, dPhi(i,theta)/di

Matrix of the flux linkage partial derivatives with respect to current, defined as a function of current vector and rotor angle vector. Flux linkage is the flux multiplied by the number of winding turns. The default value is `0.0002*ones(3,4)` Wb/A, which corresponds to the special case where stator inductance does not depend on stator current or on rotor angle.

This parameter is visible only if **Parameterization** is set to `Assume constant mutual inductance - tabulate with phase current and rotor angle`.

Flux linkage partial derivative wrt angle, dPhi(i,theta)/dtheta

Matrix of the flux linkage partial derivatives with respect to rotor angle, defined as a function of current vector and rotor angle vector. Flux linkage is the flux multiplied by the number of winding turns. The default value is `[0, -0.16, 0.16, 0; 0, -0.16, 0.16, 0; 0, -0.16, 0.16, 0]` Wb/rad, which corresponds to the special case where stator inductance does not depend on stator current.

This parameter is visible only if **Parameterization** is set to `Assume constant mutual inductance - tabulate with phase current and rotor angle`.

Direct axis current vector, id

Vector of *d*-axis currents corresponding to the provided inductances. The current vector must be two-sided (have positive and negative values). The default value is `[-200, 0, 200]` A.

This parameter is visible only if **Parameterization** is set to `Assume sinusoidal back emf - tabulate with d- and q-axis currents`.

Quadrature axis current vector, iq

Vector of *q*-axis currents corresponding to the provided inductances. The current vector must be two-sided (have positive and negative values). The default value is `[-200, 0, 200]` A.

This parameter is visible only if **Parameterization** is set to `Assume sinusoidal back emf - tabulate with d- and q-axis currents`.

Ld matrix, Ld(id,iq)

Matrix of the *d*-axis inductances with respect to current, defined as a function of *d*-axis and *q*-axis current vectors. The default value is `0.0002*ones(3,3)` H.

This parameter is visible only if **Parameterization** is set to `Assume sinusoidal back emf - tabulate with d- and q-axis currents`.

Lq matrix, Lq(id,iq)

Matrix of the *q*-axis inductances with respect to current, defined as a function of *d*-axis and *q*-axis current vectors. The default value is `0.0002*ones(3,3)` H.

This parameter is visible only if **Parameterization** is set to `Assume sinusoidal back emf - tabulate with d- and q-axis currents`.

Permanent magnet flux linkage, PM(id,iq)

Matrix of the permanent magnet flux linkages with respect to current, defined as a function of *d*-axis and *q*-axis current vectors. Flux linkage is the flux multiplied by the number of winding turns. The default value is `0.1*ones(3,3)` Wb.

This parameter is visible only if **Parameterization** is set to `Assume sinusoidal back emf - tabulate with d- and q-axis currents`.

Number of pole pairs

Number of the permanent magnet motor pole pairs. The default value is `6`.

This parameter is visible only if **Parameterization** is set to `Assume sinusoidal back emf - tabulate with d- and q-axis currents`.

Calculate torque matrix?

Specify the way of providing the electromagnetic torque data:

- Yes — The block calculates the torque from the flux linkage information, as a function of current and rotor angle. This is the default option.

- `No — specify directly` — Enter the electromagnetic torque data directly, by using the **Torque matrix, T(i,theta)** parameter.

This parameter is visible only if **Parameterization** is set to `Assume constant mutual inductance - tabulate with phase current and rotor angle`. If **Parameterization** is set to `Assume sinusoidal back emf - tabulate with d- and q-axis currents`, the equation for torque is explicit in terms of the provided matrices.

Torque matrix, T(i,theta)

Specify a matrix of the electromagnetic torque applied to the rotor, as a function of current and rotor angle. This parameter is visible only if **Calculate torque matrix?** is set to `No — specify directly`. The default value is `[0, 0.3, -0.3, 0; 0, 0, 0, 0; 0, -0.3, 0.3, 0]` N*m.

Interpolation method

Select one of the following interpolation methods for approximating the output value when the input value is between two consecutive grid points:

- `Linear` — Uses an extension of linear algorithm for multidimensional interpolation. Select this option to get the best performance.

- `Smooth` — Uses a modified Akima interpolation algorithm. Select this option to produce a continuous surface with continuous first-order derivatives.

For more information on interpolation methods, see the PS Lookup Table (2D) block reference page.

Stator resistance per phase, Rs

Resistance of each of the stator windings. The default value is `0.013` Ohm.

Stator mutual inductance, Ms

Stator-stator mutual inductance, which is assumed to be independent of both current and rotor angle. The default value is `0.00002` H.

This parameter is visible only if **Parameterization** is set to `Assume constant mutual inductance - tabulate with phase current and rotor angle`.

Electrical (3-D A-Phase Data Variant)

This configuration of the **Electrical** tab corresponds to the 3-D A-Phase Data block variants, with or without thermal ports. If you are using the 2-D Data, 4-D A-Phase Data, or 3-D DQ Data variant of the block, see "Electrical (2-D Data Variant)" on page 1-202, "Electrical (4-D A-Phase Data Variant)" on page 1-207, or "Electrical (3-D DQ Data Variant)" on page 1-209 respectively.

Direct-axis current vector, iD

Vector of direct-axis currents corresponding to the provided flux linkage partial derivatives. The current vector must be two-sided (have positive and negative values). The default value is `[-200, 0, 200]` A.

Quadrature-axis current vector, iQ

Vector of quadrature-axis currents corresponding to the provided flux linkage partial derivatives. The current vector must be two-sided (have positive and negative values). The default value is `[-200, 0, 200]` A.

Rotor angle vector, theta

Vector of rotor angles corresponding to the provided flux linkage partial derivatives. The vector must start at zero. This value corresponds to the angle where the A-phase magnetic flux aligns with the rotor permanent magnetic peak flux direction (the direct-axis, or *d*-axis). The last value, Θ_{max}, must be the rotor angle where the flux linkage pattern peaks again. Therefore, the number of pole pairs is $360/\Theta_{max}$ if Θ_{max} is expressed in degrees. The default value is `[0, 20, 40, 60]` deg, which corresponds to a 6 pole-pair motor.

A-phase flux linkage partial derivative wrt iA, dPhiA(iD,iQ,theta)/diA

Matrix of the A-phase flux linkage partial derivatives with respect to current in winding A, defined as a function of the two current vectors and the rotor angle vector. Flux linkage is the flux multiplied by the number of winding turns. The default value is `zeros(3, 3, 4)` Wb/A.

A-phase flux linkage partial derivative wrt iB, dPhiA(iD,iQ,theta)/diB

Matrix of the A-phase flux linkage partial derivatives with respect to current in winding B, defined as a function of the two current vectors and the rotor angle vector. Flux linkage is the flux multiplied by the number of winding turns. The default value is `zeros(3, 3, 4)` Wb/A.

A-phase flux linkage partial derivative wrt iC, dPhiA(iD,iQ,theta)/diC

Matrix of the A-phase flux linkage partial derivatives with respect to current in winding C, defined as a function of the two current vectors and the rotor angle vector.

Flux linkage is the flux multiplied by the number of winding turns. The default value is `zeros(3, 3, 4)` Wb/A.

A-phase flux linkage partial derivative wrt angle, dPhiA(iD,iQ,theta)/dtheta

Matrix of the A-phase flux linkage partial derivatives with respect to rotor angle, defined as a function of the two current vectors and the rotor angle vector. Flux linkage is the flux multiplied by the number of winding turns. The default value is `zeros(3, 3, 4)` Wb/rad.

Torque matrix, T(iD,iQ,theta)

Specify a matrix of the electromagnetic torque applied to the rotor, as a function of the two currents and the rotor angle. The default value is `zeros(3, 3, 4)` N*m.

Interpolation method

Select one of the following interpolation methods for approximating the output value when the input value is between two consecutive grid points:

- `Linear` — Uses an extension of linear algorithm for multidimensional interpolation. Select this option to get the best performance.
- `Smooth` — Uses a modified Akima interpolation algorithm. Select this option to produce a continuous surface with continuous first-order derivatives.

For more information on interpolation methods, see the PS Lookup Table (3D) block reference page.

Stator resistance per phase, Rs

Resistance of each of the stator windings. The default value is `0.013` Ohm.

Electrical (4-D A-Phase Data Variant)

This configuration of the **Electrical** tab corresponds to the 4-D A-Phase Data block variants, with or without thermal ports. If you are using the 2-D Data, 3-D A-Phase Data, or 3-D DQ Data variant of the block, see "Electrical (2-D Data Variant)" on page 1-202, "Electrical (3-D A-Phase Data Variant)" on page 1-206, or "Electrical (3-D DQ Data Variant)" on page 1-209 respectively.

A-phase current vector, iA

Vector of A-phase currents corresponding to the provided flux linkage partial derivatives. The current vector must be two-sided (have positive and negative values). The default value is `[-200, 0, 200]` A.

B-phase current vector, iB

Vector of B-phase currents corresponding to the provided flux linkage partial derivatives. The current vector must be two-sided (have positive and negative values). The default value is `[-200, 0, 200]` A.

C-phase current vector, iC

Vector of C-phase currents corresponding to the provided flux linkage partial derivatives. The current vector must be two-sided (have positive and negative values). The default value is `[-200, 0, 200]` A.

Rotor angle vector, theta

Vector of rotor angles corresponding to the provided flux linkage partial derivatives. The vector must start at zero. This value corresponds to the angle where the A-phase magnetic flux aligns with the rotor permanent magnetic peak flux direction (the direct-axis, or *d*-axis). The last value, Θ_{max}, must be the rotor angle where the flux linkage pattern peaks again. Therefore, the number of pole pairs is $360/\Theta_{max}$ if Θ_{max} is expressed in degrees. The default value is `[0, 20, 40, 60]` deg, which corresponds to a 6 pole-pair motor.

A-phase flux linkage partial derivative wrt iA, dPhiA(iA,iB,iC,theta)/diA

Matrix of the A-phase flux linkage partial derivatives with respect to current in winding A, defined as a function of the three current vectors and the rotor angle vector. Flux linkage is the flux multiplied by the number of winding turns. The default value is `zeros(3, 3, 3, 4)` Wb/A.

A-phase flux linkage partial derivative wrt iB, dPhiA(iA,iB,iC,theta)/diB

Matrix of the A-phase flux linkage partial derivatives with respect to current in winding B, defined as a function of the three current vectors and the rotor angle vector. Flux linkage is the flux multiplied by the number of winding turns. The default value is `zeros(3, 3, 3, 4)` Wb/A.

A-phase flux linkage partial derivative wrt iC, dPhiA(iA,iB,iC,theta)/diC

Matrix of the A-phase flux linkage partial derivatives with respect to current in winding C, defined as a function of the three current vectors and the rotor angle vector. Flux linkage is the flux multiplied by the number of winding turns. The default value is `zeros(3, 3, 3, 4)` Wb/A.

A-phase flux linkage partial derivative wrt angle, dPhiA(iA,iB,iC,theta)/dtheta

Matrix of the A-phase flux linkage partial derivatives with respect to rotor angle, defined as a function of three current vectors and the rotor angle vector. Flux linkage is the flux multiplied by the number of winding turns. The default value is `zeros(3, 3, 3, 4)` Wb/rad.

Torque matrix, T(iA,iB,iC,theta)

Specify a matrix of the electromagnetic torque applied to the rotor, as a function of the three currents and the rotor angle. The default value is `zeros(3, 3, 3, 4)` N*m.

Interpolation method

Select one of the following interpolation methods for approximating the output value when the input value is between two consecutive grid points:

- `Linear` — Uses an extension of linear algorithm for multidimensional interpolation. Select this option to get the best performance.

- `Smooth` — Uses a modified Akima interpolation algorithm. Select this option to produce a continuous surface with continuous first-order derivatives.

For more information on interpolation methods, see the PS Lookup Table (4D) block reference page.

Stator resistance per phase, Rs

Resistance of each of the stator windings. The default value is `0.013` Ohm.

Electrical (3-D DQ Data Variant)

This configuration of the **Electrical** tab corresponds to the 3-D DQ Data block variants, with or without thermal ports. If you are using the 2-D Data, 3-D A-Phase Data, or 4-D A-Phase Data variant of the block, see "Electrical (2-D Data Variant)" on page 1-202, "Electrical (3-D A-Phase Data Variant)" on page 1-206, or "Electrical (4-D A-Phase Data Variant)" on page 1-207, respectively.

Winding type

Select the configuration for the stator windings:

- `Wye-wound` — The stator windings are wye-wound. This is the default value.

- `Delta-wound` — The stator windings are delta-wound. The *a*-phase is connected between ports **a** and **b**, the *b*-phase between ports **b** and **c** and the *c*-phase between ports **c** and **a**.

Number of pole pairs

Number of the permanent magnet motor pole pairs. The default value is 4.

Park's convention for tabulated data

Select the order and reference angle for the Park transform mapping the given *dq* data to the three windings.

- `Q leads D, rotor angle measured from A-phase to D-axis` — Quadrature-direct transformation with angle measured with respect to *d* axis.
- `Q leads D, rotor angle measured from A-phase to Q-axis` — Quadrature-direct transformation with angle measured with respect to *q* axis.
- `D leads Q, rotor angle measured from A-phase to D-axis` — Direct-quadrature transformation with angle measured with respect to *d* axis.
- `D leads Q, rotor angle measured from A-phase to Q-axis` — Direct-quadrature transformation with angle measured with respect to *q* axis.

The default value is `D leads Q, rotor angle measured from A-phase to D-axis`.

Direct-axis current vector, iD

Vector of direct-axis currents corresponding to the provided flux linkage partial derivatives. The current vector must be two-sided (have positive and negative values). The default value is `[-200, 0, 200]` A.

Quadrature-axis current vector, iQ

Vector of quadrature-axis currents corresponding to the provided flux linkage partial derivatives. The current vector must be two-sided (have positive and negative values). The default value is `[-200, 0, 200]` A.

Rotor angle vector, theta

Vector of rotor angles corresponding to the provided flux linkage partial derivatives. The vector must start at zero. This value corresponds to the angle where the A-phase magnetic flux aligns with the rotor permanent magnetic peak flux direction (the direct-axis, or *d*-axis). The last value, Θ_{max}, must be the rotor angle where the flux linkage pattern peaks again. Therefore, the number of pole pairs is $360/\Theta_{max}$ if Θ_{max} is expressed in degrees. The default value is `[0, 10, 20, 30]` deg, which corresponds to a 4 pole-pair motor.

D-axis flux linkage, Fd(iD,iQ,theta)

Matrix of the *d*-axis flux linkage, defined as a function of the *dq* currents, and the rotor angle vector. Flux linkage is the flux multiplied by the number of winding turns. The default value is `zeros(3, 3, 4)` Wb.

If your flux data is given in a different order, you can use the `permute` function to reorder it. For an example of this reordering, see the associated MATLAB script in "Import IPMSM Flux Linkage Data from ANSYS Maxwell".

Q-axis flux linkage, Fq(iD,iQ,theta)

Matrix of the *q*-axis flux linkage, defined as a function of the *dq* currents, and the rotor angle vector. Flux linkage is the flux multiplied by the number of winding turns. The default value is `zeros(3, 3, 4)` Wb.

If your flux data is given in a different order, you can use the `permute` function to reorder it. For an example of this reordering, see the associated MATLAB script in "Import IPMSM Flux Linkage Data from ANSYS Maxwell".

Torque matrix, T(iD,iQ,theta)

Specify a matrix of the electromagnetic torque applied to the rotor, as a function of the *dq* currents and the rotor angle. The default value is `zeros(3, 3, 4)` N*m.

If your flux data is given in a different order, you can use the `permute` function to reorder it. For an example of this reordering, see the associated MATLAB script in "Import IPMSM Flux Linkage Data from ANSYS Maxwell".

Interpolation method

Select one of the following interpolation methods for approximating the output value when the input value is between two consecutive grid points:

- `Linear` — Uses an extension of linear algorithm for multidimensional interpolation. Select this option to get the best performance.
- `Smooth` — Uses a modified Akima interpolation algorithm. Select this option to produce a continuous surface with continuous first-order derivatives.

For more information on interpolation methods, see the PS Lookup Table (4D) block reference page.

Stator resistance per phase, Rs

Resistance of each of the stator windings. The default value is `0.013` Ohm.

Iron Losses

Open-circuit iron losses, [P_hysteresis P_eddy P_excess]

Row vector, of length 3, of the open-circuit iron losses due to hysteresis, Eddy, and excess losses, respectively, at the frequency specified by **Electrical frequency at which losses determined**. The default value is `[0.0, 0.0, 0.0]` W.

Short-circuit iron losses, [P_hysteresis P_eddy P_excess]

Row vector, of length 3, of the short-circuit iron losses due to hysteresis, Eddy, and excess losses, respectively, at the frequency specified by **Electrical frequency at which losses determined**. The default value is `[0.0, 0.0, 0.0]` W.

Electrical frequency at which losses determined

Electrical frequency at which the open-circuit and short-circuit iron losses were measured. The default value is `60` Hz.

Short-circuit RMS current for short-circuit iron losses

The resulting short-circuit RMS phase current when measuring the short-circuit losses. The default value is `95` A.

Mechanical

Rotor inertia

Inertia of the rotor attached to mechanical translational port R. The default value is `0.01` kg*m^2. The value can be zero.

Rotor damping

Rotary damping. The default value is `0` N*m/(rad/s).

Temperature Dependence

This tab appears only for blocks with exposed thermal ports. For more information, see "Thermal Ports" on page 1-201.

Measurement temperature

The temperature for which motor parameters are quoted. The default value is `298.15` K.

Resistance temperature coefficient

Coefficient α in the equation relating resistance to temperature, as described in "Thermal Model for Actuator Blocks". The default value, `3.93e-3` 1/K, is for copper.

Permanent magnet flux temperature coefficient

The fractional rate of change of permanent magnet flux density with temperature. It is used to linearly reduce the torque and the induced back EMF as temperature rises. The value is usually negative. The default value is `-0.001`.

Thermal Port

This tab appears only for blocks with exposed thermal ports. For more information, see "Thermal Ports" on page 1-201.

Thermal mass for each stator winding

The thermal mass value for the A, B, and C windings. The thermal mass is the energy required to raise the temperature by one degree. The default value is 100 J/K.

Initial stator winding temperatures

A 1-by-3 row vector defining the temperature of the A, B, and C thermal ports at the start of simulation. The default value is [298.15, 298.15, 298.15] K.

Rotor thermal mass

The thermal mass of the rotor, that is, the energy required to raise the temperature of the rotor by one degree. The default value is 200 J/K.

Rotor initial temperature

The temperature of the rotor at the start of simulation. The default value is 298.15 K.

Percentage of main flux path iron losses associated with the rotor

The percentage of the main flux path iron losses associated with the magnetic path through the rotor. It determines how much of the iron loss heating is attributed to the rotor thermal port HR, and how much is attributed to the three winding thermal ports HA, HB, and HC. The default value is 90%.

Percentage of cross-tooth flux path iron losses associated with the rotor

The percentage of the cross-tooth flux path iron losses associated with the magnetic path through the rotor. It determines how much of the iron loss heating is attributed to the rotor thermal port HR, and how much is attributed to the three winding thermal ports HA, HB, and HC. The default value is 30%.

Ports

This block has the following ports:

a

A-phase electrical connection.

b

B-phase electrical connection.

c

C-phase electrical connection.

n

Electrical conserving port associated with the neutral phase.

C

Mechanical rotational conserving port connected to the motor case.

R

Mechanical rotational conserving port connected to the rotor.

HA

Winding A thermal port. For more information, see "Thermal Ports" on page 1-201.

HB

Winding B thermal port. For more information, see "Thermal Ports" on page 1-201.

HC

Winding C thermal port. For more information, see "Thermal Ports" on page 1-201.

HR

Rotor thermal port. For more information, see "Thermal Ports" on page 1-201.

References

[1] Mellor, P.H., R. Wrobel, and D. Holliday. "A computationally efficient iron loss model for brushless AC machines that caters for rated flux and field weakened operation." *IEEE Electric Machines and Drives Conference*. May 2009.

See Also

FEM-Parameterized Linear Actuator | FEM-Parameterized Rotary Actuator | `elec_calculateFluxPartialDerivatives`

Topics

"Import IPMSM Flux Linkage Data from ANSYS Maxwell"
HEV PMSM Drive Test Harness
PMSM Iron Losses
PMSM with Thermal Model

Introduced in R2015b

FEM-Parameterized Rotary Actuator

Model rotary actuator defined in terms of magnetic flux
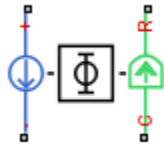
Library

Rotational Actuators

Description

The FEM-Parameterized Rotary Actuator block implements a model of a rotary actuator defined in terms of magnetic flux. Use this block to model custom rotary actuators and motors where magnetic flux depends on both rotor angle and current. You parameterize the block using data from a third-party Finite Element Magnetic (FEM) package.

The block has two options for the electrical equation. The first, `Define in terms of dPhi(i,theta)/dtheta and dPhi(i,theta)/di`, defines the current in terms of partial derivatives of the magnetic flux (Φ) with respect to rotor angle (θ) and current (i), the equations for which are:

$$\frac{di}{dt} = \left(v - iR - \frac{\partial \Phi}{\partial \theta}\frac{d\theta}{dt} \right) \bigg/ \frac{\partial \Phi}{\partial i}$$

The second option, `Define in terms of Phi(i,theta)`, defines the voltage across the component directly in terms of the flux, the equation for which is:

$$v = iR + \frac{d}{dt}\Phi\left(\theta, i\right)$$

Numerically, defining the electrical equation in terms of flux partial derivatives is better because the back-emf is piecewise continuous. If using the flux directly, using a finer grid size for current and position will improve results, as will selecting cubic or spline interpolation.

In both cases, you have an option to either directly specify the torque as a function of current and rotor angle, by using the **Torque matrix, T(i,theta)** parameter, or have the block automatically calculate the torque matrix.

If entering the electromagnetic torque data directly, you can either use data supplied by the finite element magnetic package (which you used to determine the flux) or calculate the torque from the flux with following equation:

$$T = \int\limits_{0}^{i} \frac{\partial \Phi(\theta,i)}{\partial \theta} di$$

See the Finite Element Parameterized Solenoid example model and its initialization file `elec_fem_solenoid_ini.m` for an example of how to implement this type of integration in MATLAB.

Alternatively, the block can automatically calculate the torque matrix from the flux information that you provide. To select this option, set the **Calculate torque matrix?** parameter to `Yes`. The torque matrix calculation occurs at model initialization based on current block flux linkage information. The torque is calculated by numerically integrating the rate of change of flux linkage with respect to angle over current, according to the preceding equation. If the **Electrical model** parameter is set to `Define in terms of Phi(i,theta)`, then the block must first estimate the **Flux partial derivative wrt angle, Phi(i,theta)/dtheta** parameter value from the flux linkage data. When doing this, the block uses the interpolation method specified by the **Interpolation method** parameter. Typically, the `Smooth` option is most accurate, but the `Linear` option is most robust.

You can define Φ and its partial derivatives for just positive, or positive and negative currents. If defining for just positive currents, then the block assumes that $\Phi(-i,x) = -\Phi(i,x)$. Therefore, if the current vector is positive only:

- The first current value must be zero.
- The flux corresponding to zero current must be zero.
- The partial derivative of flux with respect to rotor angle must be zero for zero current.

1-217

To model a rotary motor with a repeated flux pattern, set the **Flux dependence on displacement** parameter to `Cyclic`. When selecting this option, the torque and flux (or torque and flux partial derivatives depending on the option chosen) must have identical first and last columns.

Thermal Port

The block has an optional thermal port, hidden by default. To expose the thermal port, right-click the block in your model, and then from the context menu select **Simscape > Block choices > Show thermal port**. This action displays the thermal port H on the block icon, and adds the **Temperature Dependence** and **Thermal Port** tabs to the block dialog box.

Use the thermal port to simulate the effects of copper resistance losses that convert electrical power to heat. For more information on using thermal ports and on the **Temperature Dependence** and **Thermal Port** tab parameters, see "Simulating Thermal Effects in Rotational and Translational Actuators".

Basic Assumptions and Limitations

This block has the following limitations:

*   It is imperative that you supply a consistent set of torque and flux data. There is no checking that the torque matrix is consistent with the flux data.
*   When driving the FEM-Parameterized Rotary Actuator block via a series inductor, you may need to include a parallel conductance in the inductor component.

Parameters

*   "Magnetic Force Tab" on page 1-218
*   "Mechanical Tab" on page 1-221

Magnetic Force Tab

Electrical model

Select one of the following parameterization options, based on the underlying electrical model:

- `Define in terms of dPhi(i,theta)/dtheta and dPhi(i,theta)/di` — Define the current through the block in terms of partial derivatives of the magnetic flux with respect to rotor angle and current. This is the default method.

- `Define in terms of Phi(i,theta)` — Define the voltage across the block terminals directly in terms of the flux.

Current vector, i

Specify a vector of monotonically increasing current values corresponding to your torque-flux data. If you specify positive currents only, the first element must be zero. The default value is `[ 0 0.2 0.4 0.6 0.8 1 ]` A.

Angle vector, theta

Specify a vector of monotonically increasing rotor angle values corresponding to your torque-flux data. The default value is `[ 0 10 20 30 40 50 60 70 80 90 100 110 120 130 140 150 160 170 180 ]` deg.

Flux partial derivative wrt current, dPhi(i,theta)/di

Specify a matrix of the flux partial derivatives with respect to current. This parameter is visible only if **Electrical model** is set to `Define in terms of dPhi(i,theta)/dtheta and dPhi(i,theta)/di`. The default value, in Wb/A, is:

```
[ 0.002 0.0024 0.0035 0.0052 0.0074 0.0096 0.0118 0.0135 0.0146 ...
       0.015 0.0146 0.0135 0.0118 0.0096 0.0074 0.0052 0.0035 0.0024 0.002;
  0.002 0.0024 0.0035 0.0052 0.0074 0.0096 0.0118 0.0135 0.0146 ...
       0.015 0.0146 0.0135 0.0118 0.0096 0.0074 0.0052 0.0035 0.0024 0.002;
  0.002 0.0024 0.0035 0.0052 0.0074 0.0096 0.0118 0.0135 0.0146 ...
       0.015 0.0146 0.0135 0.0118 0.0096 0.0074 0.0052 0.0035 0.0024 0.002;
  0.002 0.0024 0.0035 0.0052 0.0074 0.0096 0.0118 0.0135 0.0146 ...
       0.015 0.0146 0.0135 0.0118 0.0096 0.0074 0.0052 0.0035 0.0024 0.002;
  0.002 0.0024 0.0035 0.0052 0.0074 0.0096 0.0118 0.0135 0.0146 ...
       0.015 0.0146 0.0135 0.0118 0.0096 0.0074 0.0052 0.0035 0.0024 0.002;
  0.002 0.0024 0.0035 0.0052 0.0074 0.0096 0.0118 0.0135 0.0146 ...
       0.015 0.0146 0.0135 0.0118 0.0096 0.0074 0.0052 0.0035 0.0024 0.002; ]
```

Flux partial derivative wrt angle, dPhi(i,theta)/dtheta

Specify a matrix of the flux partial derivatives with respect to rotor angle. This parameter is visible only if **Electrical model** is set to `Define in terms of dPhi(i,theta)/dtheta and dPhi(i,theta)/di`. The default value, in Wb/rad, is:

```
[ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;
  0 9e-4 0.0017 0.0023 0.0026 0.0026 0.0023 0.0017 9e-4 ...
       0 -9e-4 -0.0017 -0.0023 -0.0026 -0.0026 -0.0023 -0.0017 -9e-4 0;
  0 0.0018 0.0033 0.0045 0.0051 0.0051 0.0045 0.0033 0.0018 ...
       0 -0.0018 -0.0033 -0.0045 -0.0051 -0.0051 -0.0045 -0.0033 -0.0018 0;
  0 0.0027 0.005 0.0068 0.0077 0.0077 0.0068 0.005 0.0027 ...
       0 -0.0027 -0.005 -0.0068 -0.0077 -0.0077 -0.0068 -0.005 -0.0027 0;
  0 0.0036 0.0067 0.009 0.0102 0.0102 0.009 0.0067 0.0036 ...
       0 -0.0036 -0.0067 -0.009 -0.0102 -0.0102 -0.009 -0.0067 -0.0036 0;
```

```
        0 0.0044 0.0084 0.0113 0.0128 0.0128 0.0113 0.0084 0.0044 ...
             0 -0.0044 -0.0084 -0.0113 -0.0128 -0.0128 -0.0113 -0.0084 -0.0044 0 ]
```

Flux linkage matrix, Phi(i,theta)

Specify a matrix of the total flux linkage, that is, flux times the number of turns. This parameter is visible only if **Electrical model** is set to `Define in terms of Phi(i,theta)`. The default value, in Wb, is:

```
[ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;
  4e-4 4.8e-4 7e-4 0.00105 0.00147 0.00193 0.00235 0.0027 0.00292 ...
        0.003 0.00292 0.0027 0.00235 0.00193 0.00147 0.00105 7e-4 4.8e-4 4e-4;
  8e-4 9.6e-4 0.00141 0.0021 0.00295 0.00385 0.0047 0.00539 0.00584 ...
        0.006 0.00584 0.00539 0.0047 0.00385 0.00295 0.0021 0.00141 9.6e-4 8e-4;
  0.0012 0.00144 0.00211 0.00315 0.00442 0.00578 0.00705 0.00809 0.00876 ...
        0.009 0.00876 0.00809 0.00705 0.00578 0.00442 0.00315 0.00211 0.00144 0.0012;
  0.0016 0.00191 0.00282 0.0042 0.0059 0.0077 0.0094 0.01078 0.01169 ...
        0.012 0.01169 0.01078 0.0094 0.0077 0.0059 0.0042 0.00282 0.00191 0.0016;
  0.002 0.00239 0.00352 0.00525 0.00737 0.00963 0.01175 0.01348 0.01461 ...
        0.015 0.01461 0.01348 0.01175 0.00963 0.00737 0.00525 0.00352 0.00239 0.002 ]
```

Calculate torque matrix?

Specify the way of providing the electromagnetic torque data:

- `No — specify directly` — Enter the electromagnetic torque data directly, by using the **Torque matrix, T(i,theta)** parameter. This is the default option.

- Yes — The block calculates the torque from the flux linkage information, as a function of current and rotor angle.

Torque matrix, T(i,theta)

Specify a matrix of the electromagnetic torque applied to the rotor. This parameter is visible only if **Calculate torque matrix?** is set to `No — specify directly`. The default value, in mN*m, is:

```
[ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;
  0 0.0889 0.1671 0.2252 0.2561 0.2561 0.2252 0.1671 0.0889 ...
        0 -0.0889 -0.1671 -0.2252 -0.2561 -0.2561 -0.2252 -0.1671 -0.0889 0;
  0 0.3557 0.6685 0.9007 1.0242 1.0242 0.9007 0.6685 0.3557 ...
        0 -0.3557 -0.6685 -0.9007 -1.0242 -1.0242 -0.9007 -0.6685 -0.3557 0;
  0 0.8003 1.5041 2.0265 2.3045 2.3045 2.0265 1.5041 0.8003 ...
        0 -0.8003 -1.5041 -2.0265 -2.3045 -2.3045 -2.0265 -1.5041 -0.8003 0;
  0 1.4228 2.674 3.6027 4.0968 4.0968 3.6027 2.674 1.4228 ...
        0 -1.4228 -2.674 -3.6027 -4.0968 -4.0968 -3.6027 -2.674 -1.4228 0;
  0 2.2231 4.1781 5.6292 6.4013 6.4013 5.6292 4.1781 2.2231 ...
        0 -2.2231 -4.1781 -5.6292 -6.4013 -6.4013 -5.6292 -4.1781 -2.2231 0 ]
```

Flux dependence on displacement

Specify the flux pattern:

- `Unique` — No flux pattern present. This is the default option.

- Cyclic — Select this option to model a rotary motor with a repeated flux pattern. The torque and flux (or torque and flux partial derivatives, depending on the **Electrical model** option chosen) must have identical first and last columns.

Interpolation method

Select one of the following interpolation methods for approximating the output value when the input value is between two consecutive grid points:

- Linear — Select this option to get the best performance.
- Smooth — Select this option to produce a continuous surface with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (2D) block reference page.

Extrapolation method

Select one of the following extrapolation methods for determining the output value when the input value is outside the range specified in the argument list:

- Linear — Select this option to produce a surface with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region.
- Nearest — Select this option to produce an extrapolation that does not go above the highest point in the data or below the lowest point in the data.

For more information on extrapolation algorithms, see the PS Lookup Table (2D) block reference page.

This parameter is not available if you set the **Flux dependence on displacement** parameter to Cyclic.

Winding resistance

Total resistance of the electrical winding. The default value is 10 Ohm.

Mechanical Tab

Damping

Rotary damping. The default value is 1e-4 N*m/(rad/s). The value can be zero.

1-221

Rotor inertia

Inertia of the rotor attached to mechanical translational port R. The default value is `5e-5` kg*m^2. The value can be zero.

Minimum rotor angle

The rotor angle at which the lower mechanical end stop is applied. The default value is `-Inf`.

Maximum rotor angle

The rotor angle at which the upper mechanical end stop is applied. The default value is `Inf`.

Initial rotor position

Position of the rotor at the start of the simulation. The default value is `0` deg.

Initial rotor velocity

Angular velocity of the rotor at the start of the simulation. The default value is `0` deg/s.

Contact stiffness

Contact stiffness between rotor and end stops. The default value is `1e8` N*m/rad.

Contact damping

Contact damping between rotor and end stops. The default value is `1e4` N*m/(rad/s).

Ports

This block has the following ports:

+

Positive electrical conserving port

-

Negative electrical conserving port

C

Mechanical rotational conserving port connected to the actuator case

R

Mechanical rotational conserving port connected to the rotor

See Also

FEM-Parameterized Linear Actuator | FEM-Parameterized PMSM | Solenoid

Finite-Gain Op-Amp

Gain-limited operational amplifier model with optional noise
Library: Simscape / Electronics / Integrated Circuits

Description

The Finite-Gain Op-Amp block models a gain-limited operational amplifier. If the voltages at the positive and negative ports are V_p and V_m, respectively, the output voltage is:

$$V_{out} \; = \; A\left(V_p - V_m\right) - I_{out} * R_{out}$$

where:

- A is the gain.
- R_{out} is the output resistance.
- I_{out} is the output current.

The input current is:

$$\frac{V_p - V_m}{R_{in}}$$

where R_{in} is the input resistance.

The output voltage is limited by the minimum and maximum output values you specify in the block dialog box.

Thermal Noise

The Finite-Gain Op-Amp block can generate thermal noise. If you set the **Noise mode** parameter to `Enabled`, then the equivalent circuit for the block includes a noise current source attached to each of the inputs, and a noise voltage source attached to the noninverting input. These three noise sources are independent and uncorrelated.
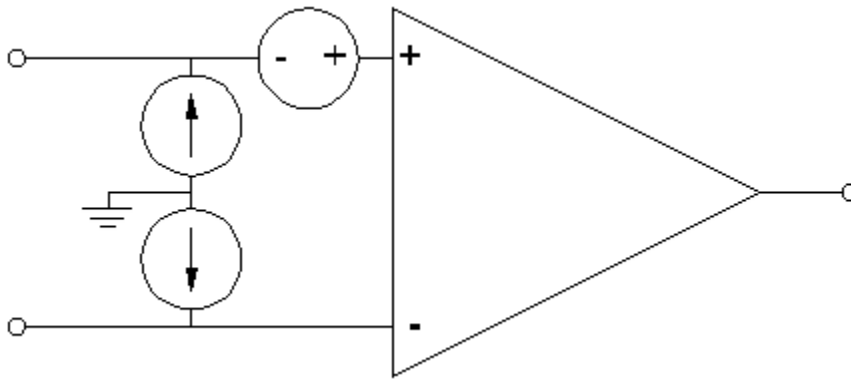
The block generates noise voltage and current according to:

$$v_{noise} = \frac{v_{density}}{\sqrt{2h}} N(0,1)$$

$$i_{noise} = \frac{i_{density}}{\sqrt{2h}} N(0,1)$$

where:

- v_{noise} is the noise voltage.
- $v_{density}$ is the single-sided, spectral amplitude density of the voltage noise.
- i_{noise} is the noise current at an input.
- $i_{density}$ is the single-sided, spectral amplitude density of the current noise applied to that input.
- h is the sampling time.
- N is a Gaussian random number with zero mean and standard deviation of one.

The block generates Gaussian noise by using the Random Number source in the Simscape Foundation library. You can control the random number seed by setting the **Repeatability** parameter:

- `Not repeatable` — Every time you simulate your model, the block resets the random seed using the MATLAB random number generator:

```
seed = randi(2^32-1);
```

- `Repeatable` — The block automatically generates a seed value and stores it inside the block, to always start the simulation with the same random number. This auto-generated seed value is set when you add a Finite-Gain Op-Amp block from the block library to the model. When you make a new copy of the Finite-Gain Op-Amp block from an existing one in a model, a new seed value is generated. The block sets the value using the MATLAB random number generator command shown above.

- `Specify seed` — If you select this option, additional parameters let you directly specify the random number seed values for input voltage, noninverting input current, and inverting input current.

Ports

Conserving

+ — Positive electrical voltage
electrical

Electrical conserving port associated with the op-amp noninverting input.

- — Negative electrical voltage
electrical

Electrical conserving port associated with the op-amp inverting input.

`out` — Output voltage
electrical

Electrical conserving port associated with the op-amp output. The port name is hidden on the block icon, but you can see it in simulation data logs.

Parameters

Main

`Gain, A` — Open-loop gain
1000 (default)

The open-loop gain of the operational amplifier.

Input resistance, Rin — Resistance at block input
1e6 Ω (default)

The resistance at the input of the operational amplifier that the block uses to calculate the input current.

Output resistance, Rout — Resistance at block output
100 Ω (default)

The resistance at the output of the operational amplifier that the block uses to calculate the drop in output voltage due to output current.

Minimum output, Vmin — Output voltage lower limit
-15 V (default)

The lower limit on the operational amplifier output voltage.

Maximum output, Vmax — Output voltage upper limit
15 V (default)

The upper limit on the operational amplifier output voltage.

Noise

Noise mode — Select whether op-amp generates thermal noise
Disabled (default) | Enabled

Select whether to model thermal noise effects:

- Disabled — Op-amp does not generate thermal noise.
- Enabled — Op-amp generates thermal noise voltage and current, and the associated parameters become visible in the **Noise** section.

Input noise voltage density — Density of voltage noise
30e-9 V/Hz^0.5 (default)

Single-sided, spectral amplitude density of the voltage noise applied to the noninverting input.

1-227

Dependencies

Enabled when the **Noise mode** parameter is set to `Enabled`.

`Noise current parameterization` — Select whether noise current density is different or the same for both inputs
`Apply same density function to both inputs` (default) | `Apply different density function to each input`

Select whether current density values applied to block inputs are different or the same.

Dependencies

Enabled when the **Noise mode** parameter is set to `Enabled`.

`Input noise current density` — Current noise at both inputs
0.5e-12 A/Hz^0.5 (default)

Single-sided, spectral amplitude density of the current noise applied to both inputs. Note that even though the density function is the same for both inputs, the actual noise current also depends on the random number seed. If the seeds used for the random number generation are different, then the actual noise currents at the inputs are also different.

Dependencies

Enabled when the **Noise current parameterization** parameter is set to `Apply same density function to both inputs`.

`Noninverting input noise current density` — Current noise at noninverting input
0.5e-12 A/Hz^0.5 (default)

Single-sided, spectral amplitude density of the current noise applied to noninverting input.

Dependencies

Enabled when the **Noise current parameterization** parameter is set to `Apply different density function to each input`.

`Inverting input noise current density` — Current noise at inverting input
0.5e-12 A/Hz^0.5 (default)

Single-sided, spectral amplitude density of the current noise applied to inverting input.

Dependencies

Enabled when the **Noise current parameterization** parameter is set to `Apply different density function to each input`.

`Sample time` — Rate at which the noise source is sampled
1e-3 s (default)

Defines the rate at which the noise source is sampled. Choose it to reflect the frequencies of interest in your model. Making the sample time too small will unnecessarily slow down your simulation.

Dependencies

Enabled when the **Noise mode** parameter is set to `Enabled`.

Repeatability — Select the noise control option
`Not repeatable` (default) | `Repeatable` | `Specify seed`

Select the noise control option:

- `Not repeatable` — The random sequence used for noise generation is not repeatable.
- `Repeatable` — The random sequence used for noise generation is repeatable, with a system-generated seed.
- `Specify seed` — The random sequence used for noise generation is repeatable, and you control the seed by using the seed parameters. You specify seed values separately for input noise voltage, noninverting input noise current, and inverting input noise current.

Dependencies

Enabled when the **Noise mode** parameter is set to `Enabled`.

`Input noise voltage auto-generated seed used for repeatable option` — Auto-generated random number seed for voltage noise
random real number

Random number seed stored inside the block to make the random sequence repeatable. The parameter value is automatically generated using the MATLAB random number generator command. You can modify this parameter value, but it gets overwritten by a new random value if you copy the block to another block in the model. Therefore, if you

want to control the seed of the random sequence, use the `Specify seed` option for the **Repeatability** parameter and specify the desired seed value using the **Input noise voltage seed** parameter.

Dependencies

Enabled when the **Repeatability** parameter is set to `Repeatable`.

`Noninverting input noise current auto-generated seed used for repeatable option` — Auto-generated random number seed for current noise at noninverting input
random real number

Random number seed stored inside the block to make the random sequence repeatable. The parameter value is automatically generated using the MATLAB random number generator command. You can modify this parameter value, but it gets overwritten by a new random value if you copy the block to another block in the model. Therefore, if you want to control the seed of the random sequence, use the `Specify seed` option for the **Repeatability** parameter and specify the desired seed value using the **Noninverting input noise current seed** parameter.

Dependencies

Enabled when the **Repeatability** parameter is set to `Repeatable`.

`Inverting input noise current auto-generated seed used for repeatable option` — Auto-generated random number seed for current noise at inverting input
random real number

Random number seed stored inside the block to make the random sequence repeatable. The parameter value is automatically generated using the MATLAB random number generator command. You can modify this parameter value, but it gets overwritten by a new random value if you copy the block to another block in the model. Therefore, if you want to control the seed of the random sequence, use the `Specify seed` option for the **Repeatability** parameter and specify the desired seed value using the **Inverting input noise current seed** parameter.

Dependencies

Enabled when the **Repeatability** parameter is set to `Repeatable`.

`Input noise voltage seed` — Random number seed for voltage noise
0 (default)

Seed used by the noise random number generator.

Dependencies

Enabled when the **Repeatability** parameter is set to `Specify seed`.

Noninverting input noise current seed — Random number seed for current noise at noninverting input
0 (default)

Seed used by the noise random number generator.

Dependencies

Enabled when the **Repeatability** parameter is set to `Specify seed`.

Inverting input noise current seed — Random number seed for current noise at inverting input
0 (default)

Seed used by the noise random number generator.

Dependencies

Enabled when the **Repeatability** parameter is set to `Specify seed`.

See Also

Band-Limited Op-Amp | Fully Differential Op-Amp | Op-Amp

Introduced in R2008b

Fully Differential Op-Amp

Model operational amplifier with fully differential output, that is, not referenced to ground

Library

Integrated Circuits

Description

The Fully Differential Op-Amp block models a fully differential operational amplifier. Differential signal transmission is better than single-ended transmission due to reduced susceptibility to external noise sources. Applications include data acquisition where inputs are differential, for example, sigma-delta converters.

The following diagram shows the internal representation of the amplifier.
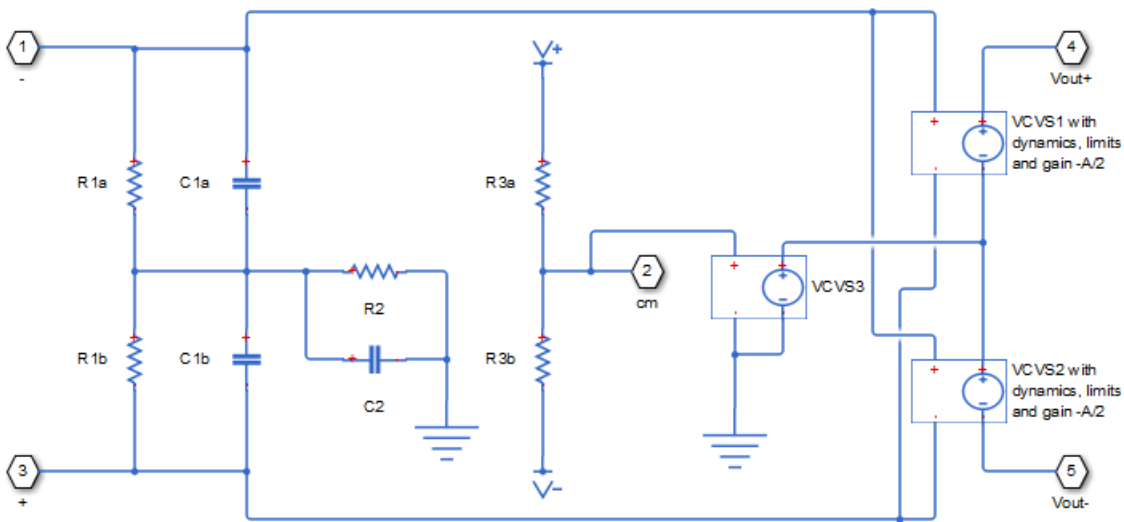
Parameters for the circuit components are derived from the block parameters that you provide. The gain of the two voltage-controlled voltage sources (VCVS1 and VCVS2) is set to half of the differential gain value. Similarly the slew rate of each of the voltage sources is set to half of the differential maximum slew rate value. The voltages of the two output ports `Vout+` and `Vout-` are both limited to be within the minimum and maximum output voltages that you specify.

The output voltage for zero differential input voltage is controlled by the common-mode port, `cm`. If no current is drawn from the `cm` port by the external circuit, then the output voltage is set to be the average of the positive and negative supply voltages by the resistor ladder of R3a and R3b. Note that the negative supply voltage can be zero, which corresponds to operation when a split supply is not available. The values for the minimum and maximum output voltages that you provide must be consistent with the values for the supply voltages that you provide. So, for example, the maximum output high voltage will be less than the positive supply voltage, the difference corresponding to the number of p-n junction voltage drops in the circuit.

Note Physical Network block diagrams do not allow unconnected Conserving ports. If you want to leave pin `cm` open-circuit, connect it to an Open Circuit block from the Simscape Foundation library.

1-233

Basic Assumptions and Limitations

This block provides a behavioral model of a fully differential operational amplifier. It does not represent nonlinear effects, such as variation in gain with output voltage amplitude, and the nonlinear nature of the output voltage-current relationship for large load currents.

Parameters

- "Gain Tab" on page 1-234
- "Input Impedance Tab" on page 1-234
- "Output Limits Tab" on page 1-235
- "Output Bias Tab" on page 1-235
- "Initial Conditions Tab" on page 1-236

Gain Tab

Differential gain

The gain applied to a voltage difference between the + and – inputs. The default value is `1000`.

Bandwidth

The frequency at which the differential voltage gain drops by 3 dB from its dc value. The default value is `1.5` GHz.

Input Impedance Tab

Differential input resistance

The input resistance seen by a voltage source applied across the + and – inputs. The default value is `1.3` MOhm.

Differential input capacitance

The input capacitance seen by a current source applied across the + and – inputs. The default value is `1.8` pF.

Common-mode input resistance

The input resistance seen by a voltage source applied between ground and the + input, or between ground and the – input. The default value is 1 MOhm.

Common-mode input capacitance

The input capacitance seen by a current source applied between ground and the + input, or between ground and the – input. The default value is 2.3 pF.

Output Limits Tab

Output resistance

The output resistance of either of the outputs with respect to the common-mode voltage reference. Differential output resistance is therefore twice the value of the output resistance R_out. The default value is 1 Ohm.

Minimum output voltage low

The minimum output voltage for either of the two output pins with respect to ground. The default value is -1.4 V.

Maximum output voltage low

The maximum output voltage for either of the two output pins with respect to ground. The default value is 1.4 V.

Differential maximum slew rate

The maximum slew rate of the differential output voltage. The default value is 5000 V/µs.

Output Bias Tab

Common-mode port input resistance

The input resistance seen by a voltage source applied between ground and the common mode port. The default value is 23 kOhm.

Negative supply voltage

The value of the negative supply voltage connected to common-mode bias resistor R3b (see diagram). The default value is -5 V.

Positive supply voltage

The value of the positive supply voltage connected to common-mode bias resistor R3a (see diagram). The default value is 5 V.

Initial Conditions Tab

Initial differential output voltage

The initial differential voltage across the two outputs if the output current is zero. The default value is 0 V.

Ports

The block has the following ports:

+

Noninverting input.

-

Inverting input.

cm

Common-mode port. If you want to leave this pin open-circuit, connect a voltage sensor between the cm port and a reference port.

Vout+

Noninverting output.

Vout-

Inverting output.

See Also

Band-Limited Op-Amp | Finite-Gain Op-Amp | Op-Amp

Fuse

Model fuse that protects against excessive current
Library: Simscape / Electronics / Passive Devices

Description

The Fuse block breaks the circuit, to protect it against excessive current. The block provides the following modeling options:

- `Opening time is independent of current` — The fuse opens when the current through the fuse exceeds the threshold current continuously for longer than the specified amount of time. This option assumes that the time to open is independent of the current. It also implicitly assumes that heat in the fuse is instantaneously dissipated once the current drops below a certain threshold.

- `Opening time depends on I^2*t` — Makes fuse opening dependent on the energy dissipated in the fuse. The nominal melting value (I^2*t) is usually available on manufacturer datasheets. If you select this option, the fuse blows when:

$$\int_{t_{over}}^{t} i^2 dt \geq I2t$$

where t_{over} is the time when the current crosses over the threshold current, *i* is instantaneous current through the fuse, and *I2t* is the nominal melting value from the datasheet. Use this option for high-current range of operation.

- `Opening time is tabulated` — Makes fuse opening dependent on the average time-current tabulated data. This data indicates the time that it takes for the fuse to open given a certain current, and is available on manufacturer datasheets as time-current curves.

  For example, these are the time-current curves from a Littelfuse® 218 Series fuse datasheet.
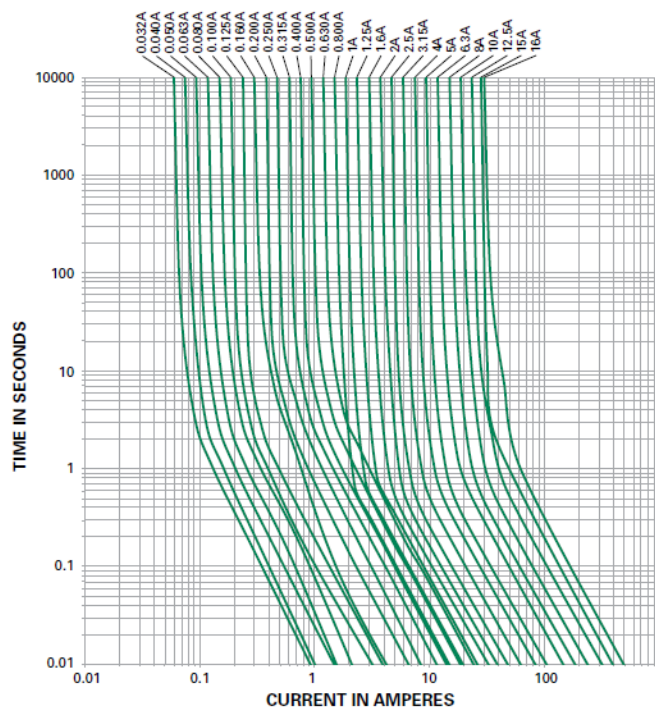
Select the appropriate curve and enter the corresponding current and time values into the **Tabulated currents** and **Tabulated opening time** parameter vectors, respectively. Use this option for small currents.

Ports

Conserving

+ — Positive terminal
electrical

Electrical conserving port associated with the fuse positive terminal.

- — Negative terminal
electrical

Electrical conserving port associated with the fuse negative terminal.

Parameters

`Parameterization` — Model dependence of opening time on current
Opening time is independent of current (default) | Opening time depends on I^2*t | Opening time is tabulated

Select whether opening time depends on current:

- `Opening time is independent of current` — The fuse opens when the current through the fuse exceeds the threshold current continuously for longer than the specified amount of time.
- `Opening time depends on I^2*t` — Makes fuse opening dependent on the energy dissipated in the fuse. Use this option for high-current range of operation.
- `Opening time is tabulated` — Makes fuse opening dependent on the average time-current tabulated data. Use this option for small currents.

`Rated current, I_rated` — Nominal operating current
1 A (default)

The fuse rating from the datasheet, or nominal operating current.

Dependencies

Enabled when the **Parameterization** parameter is set to Opening time is independent of current or Opening time depends on I^2*t.

`Ratio of minimum melting current to rated current, I_melt/I_rated` — Defined threshold current for melting
1 (default)

Minimum melting current defines the fuse operating threshold. If the current through the fuse does not exceed this threshold, the fuse stays in low-ohmic state. With the default parameter value, 1, the threshold current is the same as the rated current. In most cases, this parameter value should be greater than 1. You can also set it to a value less than 1, for situations where the fuse may blow at a current level below the rated current (for example, due to high temperature, or due to repetitive pulses). The parameter value must be greater than 0.

Dependencies

Enabled when the **Parameterization** parameter is set to `Opening time is independent of current` or `Opening time depends on I^2*t`.

`Time to fuse` — Time needed for the fuse to blow
0 s (default)

The amount of time needed for the fuse to blow. If the current through the fuse exceeds the threshold current continuously for longer than this amount of time, the fuse enters the high-ohmic state and breaks the circuit. If the current falls below the threshold before the time elapsed, the fuse stays in the low-ohmic state. If the current exceeds the threshold again, the time counter restarts. With the default parameter value, `0`, the fuse blows immediately as soon as the current exceeds the minimum melting current. The parameter value must be greater than or equal to 0.

Dependencies

Enabled when the **Parameterization** parameter is set to `Opening time is independent of current`.

`Nominal melting, I^2*t` — Dissipated energy needed for the fuse to blow
1 s*A^2 (default)

The amount of energy dissipated in the fuse that is needed for the fuse to blow. This is the nominal melting, or I2t, parameter from the datasheet. The value must be greater than or equal to 0.

Dependencies

Enabled when the **Parameterization** parameter is set to `Opening time depends on I^2*t`.

`Tabulated currents` — Currents data from the time-current curves
[1.9, 2, 2.5, 30] A (default)

The tabulated current values from a time-current curve on the manufacturer datasheet. Every element in this vector must be greater than 0, and the vector must be strictly ascending.

Dependencies

Enabled when the **Parameterization** parameter is set to `Opening time is tabulated`.

Tabulated opening times — Times data from the time-current curves
[10000, 1, 0.3, 0.001] s (default)

The tabulated time values from a time-current curve on the manufacturer datasheet. Every element in this vector must be greater than or equal to 0, and the vector must be strictly descending. The vector must be of the same size as the **Tabulated currents** vector.

Dependencies

Enabled when the **Parameterization** parameter is set to Opening time is tabulated.

Fuse resistance R — Resistance of the fuse
0.01 Ω (default)

The fuse resistance in low-ohmic state. The parameter value must be greater than 0.

Open-circuit conductance G — Open-circuit fuse conductance
1e-8 1/Ω (default)

The open-circuit fuse conductance in high-ohmic state, when the fuse has blown. The parameter value must be greater than 0.

See Also

Introduced in R2008b

Gate Driver

Abstracted representation of gate driver integrated circuit
Library: Simscape / Electronics / Actuators & Drivers / Drivers

Description

The Gate Driver block provides an abstracted representation of a gate driver integrated circuit. The block models input hysteresis, propagation delay, and turn-on/turn-off dynamics. Unless modeling a gate driver circuit explicitly, always use this block or the Half-Bridge Driver block to set gate-source voltage on a MOSFET block or gate-emitter voltage on an IGBT block. Do not connect a controlled voltage source directly to a semiconductor gate, because this omits the gate driver output impedance that determines switching dynamics.

The Gate Driver block has two modeling variants, accessible by right-clicking the block in your block diagram and then selecting the appropriate option from the context menu, under **Simscape > Block choices**:

- **Physical signal input** — The driver output state is controlled by a physical signal input u. Use this variant if all of your controller, including PWM waveform generation, is determined by Simulink blocks. This modeling variant is the default.

- **Electrical input ports** — The driver output state is controlled by two electrical input connections, PWM and REF. Use this variant if your model has upstream analog components, such as the Controlled PWM Voltage source.

When the input rises above the logic 1 input level, the transition of the output state from off to on is initiated after a delay equal to the turn-on propagation delay. The demanded output voltage across the G and S ports steps in value from the off-state output voltage to the on-state output voltage, but the actual output voltage is set by the RC time constant associated with the **On-state gate drive resistance** value and the total load capacitance. Similarly, when the input falls below the logic 0 input value, the transition of the output state from on to off is initiated after a delay equal to the turn-off propagation delay and with dynamics now set by the **Off-state gate drive resistance** value.

Faults

You can insert a fault into the output of the gate driver at a specified simulation time, to make the connected semiconductor device either permanently off or permanently on. Use this feature to represent a failed semiconductor device as failed at open-circuit or at normal on-state conditions.

Ports

Input

u — Control signal, unitless
physical signal

Input physical signal that specifies the input control value.

Dependencies

Enabled for the **Physical signal input** variant of the block.

Conserving

PWM — Pulse-width modulated signal
electrical

Electrical conserving port associated with the pulse-width modulated signal.

Dependencies

Enabled for the **Electrical input ports** variant of the block.

REF — Floating zero volt reference
electrical

Electrical conserving port associated with the floating zero volt reference.

Dependencies

Enabled for the **Electrical input ports** variant of the block.

G — Gate connection
electrical

Electrical conserving port associated with the gate. Connect this port to the gate of a MOSFET or IGBT block.

S — Source or emitter connection
electrical

Electrical conserving port associated with the source or emitter. Connect this port to the source of a MOSFET block or the emitter of an IGBT block.

Parameters

Input Logic

`Logic 1 input value` — Signal value for logic 1
0.7 (default)

Value of the input signal corresponding to the logic 1 level.

Dependencies

Enabled for the **Physical signal input** variant of the block.

`Logic 0 input value` — Signal value for logic 0
0.3 (default)

Value of the input signal corresponding to the logic 0 level.

Dependencies

Enabled for the **Physical signal input** variant of the block.

`Logic 1 input voltage` — Voltage value for logic 1
2.0 V (default)

Value of the input voltage corresponding to the logic 1 level.

Dependencies

Enabled for the **Electrical input ports** variant of the block.

Logic 0 input voltage — Voltage value for logic 0
0.8 V (default)

Value of the input voltage corresponding to the logic 0 level.

Dependencies

Enabled for the **Electrical input ports** variant of the block.

Outputs

On-state gate-source voltage — Demanded output voltage for on state
15 V (default)

Demanded output voltage when the driver is in on state.

Off-state gate-source voltage — Demanded output voltage for off state
0 V (default)

Demanded output voltage when the driver is in off state.

Timing

Propagation delay (logic 0->logic 1) — Turn-on propagation delay
50 ns (default)

When the input rises above the logic 1 input level, the transition of the output state from off to on is initiated after a delay equal to the turn-on propagation delay.

Propagation delay (logic 1->logic 0) — Turn-off propagation delay
50 ns (default)

When the input falls below the logic 0 input value, the transition of the output state from on to off is initiated after a delay equal to the turn-off propagation delay.

Dynamics

Parameterization — Select driver parameterization
Output impedance (default) | Rise and fall times

Select the type of driver parameterization:

- `Output impedance` — Specify on-state and off-state gate drive resistances.
- `Rise and fall times` — Specify rise time, fall time, and load capacitance.

On-state gate drive resistance — Gate drive resistance for on state
2 Ω (default)

Gate drive resistance when the driver is in on state.

Dependencies

Enabled when the **Parameterization** parameter is set to `Output impedance`.

Off-state gate drive resistance — Gate drive resistance for off state
2 Ω (default)

Gate drive resistance when the driver is in off state.

Dependencies

Enabled when the **Parameterization** parameter is set to `Output impedance`.

Rise time — Driver rise time
20 ns (default)

Driver rise time from 10% to 90%.

Dependencies

Enabled when the **Parameterization** parameter is set to `Rise and fall times`.

Fall time — Driver fall time
20 ns (default)

Driver fall time from 90% to 10%.

Dependencies

Enabled when the **Parameterization** parameter is set to `Rise and fall times`.

Load capacitance for rise and fall times — Driver load capacitance
10 nF (default)

Driver load capacitance.

Dependencies

Enabled when the **Parameterization** parameter is set to `Rise and fall times`.

Faults

`Enable faults` — Select Yes to enable faults modeling
No (default) | Yes

Select `Yes` to enable faults modeling. The associated parameters in the **Faults** section become visible to let you specify time to fail and the failure mode.

`Simulation time for fault event` — Time before entering faulted state
1 s (default)

Set the simulation time at which you want the block to enter the faulted state.

Dependencies

Enabled when the **Enable faults** parameter is set to `Yes`.

`Failure mode` — State of driver after failure
Fail off (default) | Fail on

Select whether driver fails by making the connected semiconductor device permanently turned off or on.

Dependencies

Enabled when the **Enable faults** parameter is set to `Yes`.

See Also

Controlled PWM Voltage | Half-Bridge Driver | N-Channel IGBT | N-Channel MOSFET | P-Channel MOSFET

Introduced in R2017b

Generic Linear Actuator

Model generic linear actuator driven from DC voltage source or PWM driver
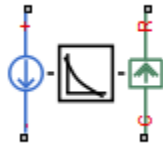
Library

Translational Actuators

Description

The Generic Linear Actuator block implements a model of a generic linear actuator designed to be driven from a DC voltage source or a PWM driver. Define force-speed characteristics in terms of tabulated values for powering the motor at the rated voltage. This functionality enables you to model a motor without referencing an equivalent circuit.

The motor or actuator architecture determines the way in which electrical losses depend on force. For example, a DC motor has losses that are proportional to the square of the current. As force is proportional to current, losses are also proportional to mechanical force. Most motors have an electrical loss term that is proportional to the square of mechanical force. The Generic Linear Actuator block calculates this loss term using the **Motor efficiency (percent)** and **Speed at which efficiency is measured** parameters that you provide.

Some motors also have a loss term that is independent of force. An example is a shunt motor where the field winding draws a constant current regardless of load. The **Force-independent electrical losses** parameter accounts for this effect.

The motor efficiency is the mechanical power divided by the sum of the mechanical power and both electrical loss terms. The block assumes that the speed at which the motor efficiency is defined is in the motoring quadrant and, therefore, positive.

You can operate the block in the reverse direction by changing the sign of the voltage applied. The H-Bridge block, for example, reverses motor direction if the voltage at the REV port is greater than the **Reverse threshold voltage** parameter. However, if you are using the block in reverse, specify the force-speed data for forward operation:

- Positive forces and positive speeds in the motoring quadrant.
- Positive force and negative speeds in the generating counterclockwise quadrant.
- Negative force and positive speed in the generating clockwise quadrant.

Thermal Port

The block has an optional thermal port, hidden by default. To expose the thermal port, right-click the block in your model, and then from the context menu select **Simscape > Block choices > Show thermal port**. This action displays the thermal port H on the block icon, and adds the **Temperature Dependence** and **Thermal Port** tabs to the block dialog box.

Use the thermal port to simulate the effects of copper resistance losses that convert electrical power to heat. For more information on using thermal ports and on the **Temperature Dependence** and **Thermal Port** tab parameters, see "Simulating Thermal Effects in Rotational and Translational Actuators".

Basic Assumptions and Limitations

This block has the following limitations:

- The force-speed curve data corresponds only to the rated voltage, so the block produces accurate results only when driven by plus or minus the rated voltage.
- The block requires you to provide force-speed data for the full range over which you use the actuator. To use the actuator in the generating and braking regions, provide additional data outside of the normal motoring region.
- Model behavior is sensitive to force-speed data. For example, no-load speed is correctly defined and finite only when the data crosses the speed axis.
- To drive the block from the H-Bridge block:

  - Do not place any other blocks between the H-Bridge and the Generic Linear Actuator blocks.

- In the H-Bridge block dialog box, set the **Freewheeling mode** to `Via one semiconductor switch and one freewheeling diode`. Selecting `Via two freewheeling diodes` does not set the bridge output voltage to zero when the PWM input signal is low.
- In the H-Bridge, Generic Linear Actuator, and Controlled PWM Voltage block dialog boxes, ensure that the **Simulation mode** is the same for all three blocks.

Parameters

- "Electrical Force Tab" on page 1-250
- "Mechanical Tab" on page 1-251

Electrical Force Tab

Speed values

Specify a vector of speeds, including their units, for your force-speed data. The default value is `[ -15 -10 -5 0 5 10 15 20 25 30 ]` m/s.

Force values

Specify a vector of forces, including their units, for your force-speed data. The default value is `[ 4 3.5 3 2.5 2 1.5 1 0.5 0 -0.5 ]` N.

Rated voltage

Indicate the voltage for which the device you are modeling is rated. The default value is 12 V.

Motor efficiency (percent)

Efficiency that the block uses to calculate force-dependent electrical losses. The default value is 70.

Speed at which efficiency is measured

Speed that the block uses to calculate force-dependent electrical losses. The default value is 20 m/s.

Force-independent electrical losses

Fixed electrical loss associated with the actuator when the force is zero. The default value is 2 W.

Simulation mode

If you set the **Simulation mode** parameter to `PWM`, apply a PWM waveform switching between zero and rated volts to the block electrical terminals. The current drawn

from the electrical supply is equal to the amount required to deliver the mechanical power and to compensate for electrical losses. If the applied voltage exceeds the rated voltage, the resultant force scales proportionately. However, applying anything other than the rated voltage can provide unrepresentative results. PWM is the default setting.

If you set the **Simulation mode** parameter to Averaged, the force generated in response to an applied voltage V_{av} is

$$\frac{V_{av}}{V_{rated}} \times F(v)$$

where $F(v)$ is the force value at speed v. The current drawn from the supply is such that the product of the current and V_{av} is equal to the average power that is consumed.

Mechanical Tab

Plunger mass

Mass of the moving part of the motor. The default value is 0.1 kg. The value can be zero.

Linear damping

Linear damping. The default value is 1e-05 N/(m/s). The value can be zero.

Initial plunger speed

Speed of the plunger at the start of the simulation. The default value is 0 m/s.

Ports

This block has the following ports:

+

Positive electrical conserving port

-

Negative electrical conserving port

C

Mechanical translational conserving port connected to the actuator case

R

Mechanical translational conserving port connected to the plunger

See Also

Generic Rotary Actuator | H-Bridge

Generic Rotary Actuator

Model generic rotary actuator driven from DC voltage source or PWM driver
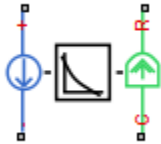
Library

Rotational Actuators

Description

The Generic Rotary Actuator block implements a model of a generic rotary actuator designed to be driven from a DC voltage source or PWM driver. You define torque-speed characteristics in terms of tabulated values for powering the motor at the rated voltage. This functionality allows you to model a motor without referencing an equivalent circuit.

The motor or actuator architecture determines the way in which electrical losses depend on torque. For example, a DC motor has losses that are proportional to the square of the current. As torque is proportional to current, losses are also proportional to mechanical torque. Most motors have an electrical loss term that is proportional to the square of mechanical torque. The Generic Rotary Actuator block calculates this loss term using the **Motor efficiency (percent)** and **Speed at which efficiency is measured** parameters that you provide.

Some motors also have a loss term that is independent of torque. An example is a shunt motor where the field winding draws a constant current regardless of load. The **Torque-independent electrical losses** parameter accounts for this effect.

The motor efficiency is the mechanical power divided by the sum of the mechanical power and both electrical loss terms. The block assumes that the speed at which the motor efficiency is defined is in the motoring quadrant and, therefore, positive.

1-253

You can operate the block in the reverse direction by changing the sign of the voltage that you apply. The H-Bridge block, for example, reverses motor direction if the voltage at the REV port is greater than the **Reverse threshold voltage** parameter. However, if you are using the block in reverse, specify the torque-speed data for forward operation:

- Positive torques and positive speeds in the motoring quadrant.
- Positive torque and negative speeds in the generating counterclockwise quadrant.
- Negative torque and positive speed in the generating clockwise quadrant.

Thermal Port

The block has an optional thermal port, hidden by default. To expose the thermal port, right-click the block in your model, and then from the context menu select **Simscape > Block choices > Show thermal port**. This action displays the thermal port H on the block icon, and adds the **Temperature Dependence** and **Thermal Port** tabs to the block dialog box.

Use the thermal port to simulate the effects of copper resistance losses that convert electrical power to heat. For more information on using thermal ports and on the **Temperature Dependence** and **Thermal Port** tab parameters, see "Simulating Thermal Effects in Rotational and Translational Actuators".

Basic Assumptions and Limitations

This block has the following limitations:

- The torque-speed curve data corresponds only to the rated voltage, so the block produces accurate results only when driven by plus or minus the rated voltage.
- In this block requires, you must provide torque-speed data for the full range over which you use the actuator. To use the actuator in the generating and braking regions, provide additional data outside of the normal motoring region.
- Model behavior is sensitive to torque-speed data. For example, no-load speed is correctly defined and finite only when the data crosses the speed axis.
- To drive the block from the H-Bridge block:

  - Do not place any other blocks between the H-Bridge and the Generic Rotary Actuator blocks.

- In the H-Bridge block dialog box, set the **Freewheeling mode** to `Via one semiconductor switch and one freewheeling diode`. Selecting `Via two freewheeling diodes` does not set the bridge output voltage to zero when the PWM input signal is low.
- In the H-Bridge, Generic Rotary Actuator, and Controlled PWM Voltage block dialog boxes, ensure that the **Simulation mode** is the same for all three blocks.

Parameters

- "Electrical Torque Tab" on page 1-255
- "Mechanical Tab" on page 1-256

Electrical Torque Tab

Speed values

Specify a vector of speeds, including their units, for your torque-speed data. The default value is `[ -1.5e+03 -1000 -500 0 500 1000 1.5e+03 2e+03 2.5e+03 3e+03 ]` rpm.

Torque values

Specify a vector of torques, including their units, for your torque-speed data. The default value is `[ 0.04 0.035 0.03 0.025 0.02 0.015 0.01 0.005 0 -0.005 ]` Nm.

Rated voltage

Indicate the voltage for which the device you are modeling is rated. The default value is 12 V.

Motor efficiency (percent)

The efficiency that the block uses to calculate torque-dependent electrical losses. The default value is `80`.

Speed at which efficiency is measured

The speed that the block uses to calculate torque-dependent electrical losses. The default value is `2e+03` rpm.

Torque-independent electrical losses

Fixed electrical loss associated with the actuator when the torque is zero. The default value is `0.1` W.

Simulation mode

If you set the **Simulation mode** parameter to `PWM`, apply a PWM waveform switching between zero and rated volts to the block electrical terminals. The current drawn from the electrical supply is equal to the amount required to deliver the mechanical power and to compensate for electrical losses. If the applied voltage exceeds the rated voltage, the resultant torque scales proportionately. However, applying anything other than the rated voltage can provide unrepresentative results. `PWM` is the default setting.

If you set the **Simulation mode** parameter to `Averaged`, the torque generated in response to an applied voltage V_{av} is

$$\frac{V_{av}}{V_{rated}} \times T(\omega)$$

where $T(\omega)$ is the torque value at speed ω. The current drawn from the supply is such that the product of the current and V_{av} is equal to the average power that is consumed.

Mechanical Tab

Rotor inertia

Rotor resistance to change in motor motion. The default value is `1e-04` kg*m^2. The value can be zero.

Rotor damping

Rotor damping. The default value is `1e-08` N*m/(rad/s). The value can be zero.

Initial rotor speed

Speed of the rotor at the start of the simulation. The default value is `0` rpm.

Ports

This block has the following ports:

+

Positive electrical conserving port

\-

Negative electrical conserving port

C

Mechanical rotational conserving port

R

Mechanical rotational conserving port

See Also

Generic Linear Actuator | H-Bridge

Gyro

Behavioral model of MEMS gyro
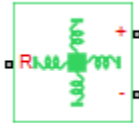
Library

Sensors

Description

The Gyro block implements a behavioral model of a MicroElectroMechanical Systems (MEMS) gyro. The gyro provides an output voltage that is proportional to the angular rotation rate presented at the mechanical rotational physical port R. The output voltage is limited according to the values that you provide for maximum and minimum output voltage.

Optionally, you can model sensor dynamics by setting the **Dynamics** parameter to `Model sensor bandwidth`. Including dynamics adds a first-order lag between the angular rate presented at port R and the corresponding voltage applied to the electrical + and - ports.

If running your simulation with a fixed-step solver, or generating code for hardware-in-the-loop testing, MathWorks recommends that you set the **Dynamics** parameter to `No dynamics — Suitable for HIL`, because this avoids the need for a small simulation time step if the sensor bandwidth is high.

Variables

Use the **Variables** section of the block interface to set the priority and initial target values for the block variables prior to simulation. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

The **Measured angular rate** variable target specifies the initial output for the sensor.

Parameters

Sensitivity

The change in output voltage level per unit change in rotation rate when the output is not being limited. The default value is `12.5` mV/(deg/s).

Output voltage for zero rotation

The output voltage from the sensor when the rotation rate is zero. The default value is `2.5` V.

Maximum output voltage

The maximum output voltage from the sensor, which determines the sensor maximum measured rotational rate. The default value is `4` V.

Minimum output voltage

The minimum output voltage from the sensor, which determines the sensor minimum measured rotational rate. The default value is `1` V.

Dynamics

Select one of the following options for modeling sensor dynamics:

- `No dynamics — Suitable for HIL` — Do not model sensor dynamics. Use this option when running your simulation fixed step or generating code for hardware-in-the-loop testing, because this avoids the need for a small simulation time step if the sensor bandwidth is high. This is the default option.

- `Model sensor bandwidth` — Model sensor dynamics with a first-order lag approximation, based on the **Bandwidth** parameter value. You can control the initial condition for the lag by specifying the **Measured angular rate** variable target.

Bandwidth

Specifies the 3dB bandwidth for the measured rotational rate assuming a first-order time constant. This parameter is only visible when you select `Model sensor bandwidth` for the **Dynamics** parameter. The default value is `3` kHz.

Ports

The block has the following ports:

R

    Mechanical rotational port

+

    Positive electrical port

-

    Negative electrical port

H-Bridge

Model H-bridge motor driver

Library

Drivers

Description

The H-Bridge block represents an H-bridge motor driver. The block has the following two **Simulation mode** options:

- PWM — The H-Bridge output is a controlled voltage that depends on the input signal at the PWM port. If the input signal has a value greater than the **Enable threshold voltage** parameter value, the H-Bridge output is on and has a value equal to the value of the **Output voltage amplitude** parameter. If it has a value less than the **Enable threshold voltage** parameter value, the block maintains the load circuit using one of the following three **Freewheeling mode** options:

  - Via one semiconductor switch and one freewheeling diode
  - Via two freewheeling diodes
  - Via two semiconductor switches and one freewheeling diode

  The first and third options are sometimes referred to as synchronous operation.

  The signal at the REV port determines the polarity of the output. If the value of the signal at the REV port is less than the value of the **Reverse threshold voltage** parameter, the output has positive polarity; otherwise, it has negative polarity.

- Averaged — This mode has two **Load current characteristics** options:

- `Smoothed`
- `Unsmoothed or discontinuous`

The `Smoothed` option assumes that the current is practically continuous due to load inductance. In this case, the H-Bridge output is:

$$\frac{V_O V_{PWM}}{A_{PWM}} - I_{OUT} R_{ON}$$

where:

- V_O is the value of the **Output voltage amplitude** parameter.
- V_{PWM} is the value of the voltage at the PWM port.
- A_{PWM} is the value of the **PWM signal amplitude** parameter.
- I_{OUT} is the value of the output current.
- R_{ON} is the **Bridge on resistance** parameter.

The current will be smooth if the PWM frequency is large enough. Synchronous operation where freewheeling is via a bridge arm back to the supply also helps smooth the current. For cases where the current is not smooth, or possibly discontinuous (that is, it goes to zero between PWM cycles), use the `Unsmoothed or discontinuous` option. For this option, you must also provide values for the **Total load series resistance**, **Total load series inductance** and **PWM frequency**. During simulation, the block uses these values to calculate a more accurate value for H-bridge output voltage that achieves the same average current as would be present if simulating in PWM mode.

Set the **Simulation mode** parameter to `Averaged` to speed up simulations when driving the H-Bridge block with a Controlled PWM Voltage block. You must also set the **Simulation mode** parameter of the Controlled PWM Voltage block to `Averaged` mode. This applies the average of the demanded PWM voltage to the motor. The accuracy of the `Averaged` mode simulation results relies on the validity of your assumption about the load current. If you specify that the current is `Unsmoothed or discontinuous`, then the accuracy also depends on the values you provide for load resistance and inductance being representative. This mode also makes some simplifying assumptions about the underlying equations for the case when current is discontinuous. For typical motor and bridge parameters, accuracy should be within a few percent. To verify `Averaged` mode accuracy, run the simulation using the `PWM` mode and compare the results to those obtained from using the `Averaged` mode.

Braking mode is invoked when the voltage presented at the BRK port is larger than the **Braking threshold voltage**. Regardless of whether in `PWM` or `Averaged` mode, when in braking mode the H-bridge is modeled by a series combination of two resistances *R1* and *R2* where:

- *R1* is the resistance of a single bridge arm, that is, half the value of the **Total bridge on resistance** parameter.

- *R2* is the resistance of a single bridge arm in parallel with a diode resistance, that is, *R1 · Rd / (R1 + Rd)*, where *Rd* is the diode resistance.

Thermal Port

The block has an optional thermal port, hidden by default. To expose the thermal port, right-click the block in your model, and then from the context menu select **Simscape > Block choices > Show thermal port**. This action displays the thermal port H on the block icon, and adds the **Temperature Dependence** and **Thermal Port** tabs to the block dialog box. These tabs are described further on this reference page.

When the thermal port is visible:

- The heat generated by the bridge on-resistance and freewheeling diodes is added to the thermal port. The thermal port has an associated thermal mass and initial temperature that you can set from the **Thermal Port** tab.

- The bridge on-resistance and freewheeling diode resistance become functions of temperature. You can define the values for these resistances and the second measurement temperature from the **Temperature Dependence** tab. Resistance is assumed to vary linearly between the two measurement temperatures. Extrapolation is used for temperatures outside of this range, except for when simulating in averaged mode with discontinuous load current characteristics.

Basic Assumptions and Limitations

The model has the following limitations:

- If you are linearizing your model, set the **Simulation mode** parameter to `Averaged` and ensure that you have specified the operating point correctly. You can only linearize the H-Bridge block for inputs that are greater than zero and less than the PWM signal amplitude.

- No forward voltage is modeled for the freewheeling diodes. They are approximated as ideal resistances when forward biased, with resistance equal to the **Freewheeling diode on** resistance parameter value.

- In `Averaged` mode, and with the `Unsmoothed or discontinuous` choice for **Load current characteristics**, you must provide representative values for load inductance and resistance. If driving a DC Motor, then the resistance is the armature resistance, and the inductance is the sum of the armature inductance plus series smoothing inductor (if present). For a Universal Motor, total resistance is the sum of the armature and field windings, and total inductance is the sum of armature and field inductances plus any series smoothing inductance. For a Shunt Motor, MathWorks recommends that you draw a Thévenin equivalent circuit to determine appropriate values.

Parameters

- "Simulation Mode & Load Assumptions Tab" on page 1-264
- "Input Thresholds Tab" on page 1-266
- "Bridge Parameters Tab" on page 1-267
- "Temperature Dependence Tab" on page 1-267
- "Thermal Port Tab" on page 1-268

Simulation Mode & Load Assumptions Tab

Simulation mode

Select one of the following options for the type of output voltage:

- `PWM` — The output voltage is a pulse-width modulated signal. This is the default option.
- `Averaged` — The output voltage is a constant whose value is equal to the average value of the PWM signal.

Freewheeling mode

Select one of the following options for the type of H-Bridge dissipation circuit:

- `Via one semiconductor switch and one freewheeling diode` — In this mode, the block controls the load by maintaining one high-side bridge arm permanently on and using the PWM signal to modulate the corresponding low-side bridge arm. This means that the block uses only one of the freewheeling diodes in

completing the dissipation circuit when the bridge turns off. This option is the default.

- `Via two freewheeling diodes` — In this mode, all bridge arms are off during the bridge off-state. This means that the block dissipates the load current across the power supply by two freewheeling diodes.

- `Via two semiconductor switches and one freewheeling diode` — In this mode, the block controls the load by maintaining one high-side bridge arm permanently on and using the PWM signal to toggle between enabling the corresponding low-side bridge arm and the opposite high-side bridge arm. This means that the block uses a freewheeling diode in parallel with a bridge arm, plus another series bridge arm, to complete the dissipation circuit when the bridge turns off.

This parameter is only visible when you select `PWM` for the **Simulation mode** parameter, or when you select `Averaged` for the **Simulation mode** parameter and `Unsmoothed or discontinuous` for the **Load current characteristics** parameter.

Load current characteristics

Select one of the following options for the type of load current:

- `Smoothed` — Assumes that the current is practically continuous due to load inductance. This option is the default.

- `Unsmoothed or discontinuous` — Use this option for cases where the current is not smooth, or possibly discontinuous (that is, it goes to zero between PWM cycles). For this option, you must also provide values for the **Total load series resistance**, **Total load series inductance**, and **PWM frequency** parameters. During simulation, the block uses these values to calculate a more accurate value for H-bridge output voltage that achieves the same average current as would be present if simulating in PWM mode.

This parameter is only visible when you select `Averaged` for the **Simulation mode** parameter.

Load total series resistance

The total load series resistance seen by the H-bridge. The default value is `10` Ω.

This parameter is only visible when you select `Averaged` for the **Simulation mode** parameter and `Unsmoothed or discontinuous` for the **Load current characteristics** parameter.

Load total series inductance

The total load series inductance seen by the H-bridge. As well as motor inductance, you should include any series inductance added external to the motor to smooth current. The default value is `1e-5` H.

This parameter is only visible when you select `Averaged` for the **Simulation mode** parameter and `Unsmoothed or discontinuous` for the **Load current characteristics** parameter.

PWM frequency

The PWM frequency at which the H-bridge is driven. For consistency, this should be the same value as the PWM frequency specified by the Controlled PWM Voltage block driving the H-Bridge. The default value is `10` kHz.

This parameter is only visible when you select `Averaged` for the **Simulation mode** parameter and `Unsmoothed or discontinuous` for the **Load current characteristics** parameter.

Input Thresholds Tab

Enable threshold voltage

Threshold above which the voltage at the PWM port must rise to enable the H-Bridge output. This parameter is used only when the **Simulation mode** parameter on the **Simulation Mode & Load Assumptions** tab is set to `PWM`. The default value is `2.5` V.

PWM signal amplitude

The amplitude of the signal at the PWM input. The H-Bridge block uses this parameter only when the **Simulation mode** parameter on the **Simulation Mode & Load Assumptions** tab is set to `Averaged`. The default value is `5` V.

Reverse threshold voltage

When the voltage at the REV port is greater than this threshold, the output polarity becomes negative. The default value is `2.5` V.

Braking threshold voltage

When the voltage at the BRK port is greater than this threshold, the H-Bridge output terminals are short-circuited via the following series of devices:

- One bridge arm

- One bridge arm in parallel with a conducting freewheeling diode

The default value is 2.5 V.

Bridge Parameters Tab

Output voltage amplitude

The amplitude of the voltage across the H-Bridge output ports when the output is on. The default value is 12 V.

Total bridge on resistance

The total effective resistance of the two semiconductor switches that connect the load to the two power rails when the voltage at the PWM port is greater than the value of the **Enable threshold voltage** parameter on the **Input Thresholds** tab. The default value is 0.1 Ω.

Freewheeling diode on resistance

The total resistance in the freewheeling diodes that dissipate the current that flows through the motor when the voltage at the PWM port is less than the value of the **Enable threshold voltage** parameter on the **Input Thresholds** tab. The default value is 0.05 Ω.

Measurement temperature

The temperature for which for which the resistance values on the **Bridge Parameters** tab are specified. This parameter appears only for blocks with exposed thermal port. For more information, see "Thermal Port" on page 1-263. The default value is 298.15 K.

Temperature Dependence Tab

This tab appears only for blocks with exposed thermal port. For more information, see "Thermal Port" on page 1-263.

Total bridge on resistance at second measurement temperature

The total effective resistance of the two semiconductor switches that connect the load to the two power rails (as described in the **Total bridge on resistance** parameter definition), quoted at the **Second measurement temperature**. The default value is 0.1 Ω.

Freewheeling diode on resistance at second measurement temperature

The total resistance in the freewheeling diodes that dissipate the current that flows through the motor (as described in the **Total bridge on resistance** parameter

definition), quoted at the **Second measurement temperature**. The default value is `0.05` Ω.

Second measurement temperature

The temperature for which for which the resistance values on the **Temperature Dependence** tab are specified. The default value is `398.15` K.

Thermal Port Tab

This tab appears only for blocks with exposed thermal port. For more information, see "Thermal Port" on page 1-263.

Thermal mass

Thermal mass associated with the thermal port H. It represents the energy required to raise the temperature of the thermal port by one degree. The default value is `100` J/K.

Initial temperature

The temperature of the thermal port at the start of simulation. The default value is `298.15` K.

Ports

The block has the following ports:

`+ref`

Positive electrical output voltage.

`-ref`

Negative electrical output voltage.

`PWM`

Pulse-width modulated signal. The voltage is defined relative to the REF port.

`REF`

Floating zero volt reference.

`REV`

Voltage that controls when to reverse the polarity of the H-Bridge output. The voltage is defined relative to the REF port.

BRK

> Voltage that controls when to short circuit the H-Bridge output. The voltage is defined relative to the REF port.

H

> Thermal port. For more information, see "Thermal Port" on page 1-263.
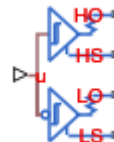
Examples

See the PWM-Controlled DC Motor and Linear Electrical Actuator with Control examples.

Half-Bridge Driver

Abstracted representation of half-bridge driver integrated circuit
Library:　　　　　　Simscape / Electronics / Actuators & Drivers / Drivers

Description

The Half-Bridge Driver block provides an abstracted representation of an integrated circuit for driving MOSFET and IGBT half-bridges. The block models input hysteresis, propagation delay, and turn-on/turn-off dynamics. Unless modeling a gate driver circuit explicitly, always use this block or the Gate Driver block to set gate-source voltage on a MOSFET block or gate-emitter voltage on an IGBT block. Do not connect a controlled voltage source directly to a semiconductor gate, because this omits the gate driver output impedance that determines switching dynamics.

The Half-Bridge Driver block has two modeling variants, accessible by right-clicking the block in your block diagram and then selecting the appropriate option from the context menu, under **Simscape > Block choices**:

- **Physical signal input** — The driver output state is controlled by a physical signal input u. Use this variant if all of your controller, including PWM waveform generation, is determined by Simulink blocks. This modeling variant is the default.

- **Electrical input ports** — The driver output state is controlled by two electrical input connections, PWM and REF. Use this variant if your model has upstream analog components, such as the Controlled PWM Voltage source.

The first pair of output electrical ports, HO and HS, behave in the same way as the G and S ports of the Gate Driver block. Connect these ports to the high-side MOSFET or IGBT of the half-bridge. The second pair of ports, LO and LS, connect to the low-side MOSFET or IGBT of the half-bridge. They behave in a similar way, except that their logic is inverted with respect to that of the high side.

The diagram shows the timing properties for the half-bridge driver, where:

- t_{pLH} is low-side propagation delay when the input logic goes from 0 to 1.
- t_{dLH} is high-side dead time when the input logic goes from 0 to 1.
- t_{pHL} is high-side propagation delay when the input logic goes from 1 to 0.
- t_{dHL} is low-side dead time when the input logic goes from 1 to 0.
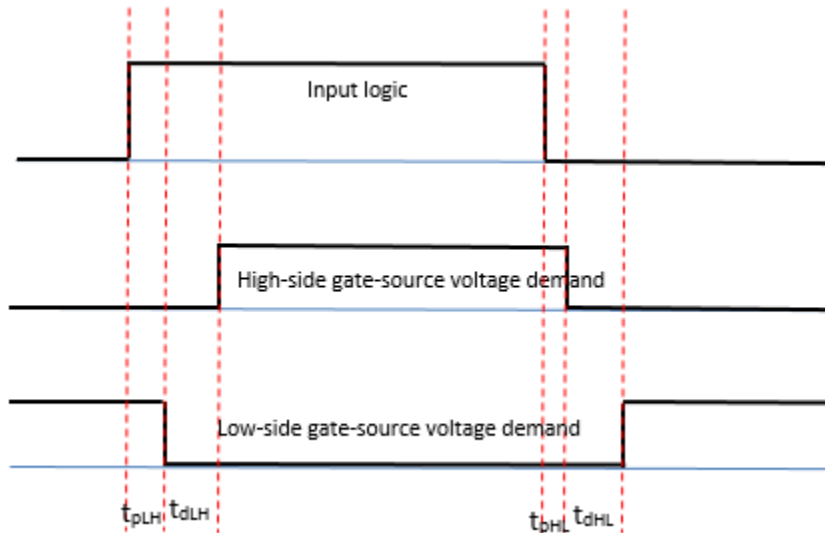
Faults

You can insert a fault into one or both of the outputs at a specified simulation time. The fault options are:

- Fail input fixed at logic 0
- Fail input fixed at logic 1
- Fail high side off
- Fail high side on
- Fail low side off
- Fail low side on
- Fail high and low sides off

- Fail high and low sides on

Ports

Input

u — Control signal, unitless
physical signal

Input physical signal that specifies the input control value.

Dependencies

Enabled for the **Physical signal input** variant of the block.

Conserving

PWM — Pulse-width modulated signal
electrical

Electrical conserving port associated with the pulse-width modulated signal.

Dependencies

Enabled for the **Electrical input ports** variant of the block.

REF — Floating zero volt reference
electrical

Electrical conserving port associated with the floating zero volt reference.

Dependencies

Enabled for the **Electrical input ports** variant of the block.

H0 — High-side gate connection
electrical

Electrical conserving port associated with the gate on the high side of the half bridge. Connect this port to the gate of a MOSFET or IGBT block.

HS — High-side source or emitter connection
electrical

Electrical conserving port associated with the source or emitter on the high side of the half bridge. Connect this port to the source of a MOSFET block or the emitter of an IGBT block.

L0 — Low-side gate connection
electrical

Electrical conserving port associated with the gate on the low side of the half bridge. Connect this port to the gate of a MOSFET or IGBT block.

LS — Low-side source or emitter connection
electrical

Electrical conserving port associated with the source or emitter on the low side of the half bridge. Connect this port to the source of a MOSFET block or the emitter of an IGBT block.

Parameters

Input Logic

`Logic 1 input value` — Signal value for logic 1
0.7 (default)

Value of the input signal corresponding to the logic 1 level.

Dependencies

Enabled for the **Physical signal input** variant of the block.

`Logic 0 input value` — Signal value for logic 0
0.3 (default)

Value of the input signal corresponding to the logic 0 level.

Dependencies

Enabled for the **Physical signal input** variant of the block.

Logic 1 input voltage — Voltage value for logic 1
2.0 V (default)

Value of the input voltage corresponding to the logic 1 level.

Dependencies

Enabled for the **Electrical input ports** variant of the block.

Logic 0 input voltage — Voltage value for logic 0
0.8 V (default)

Value of the input voltage corresponding to the logic 0 level.

Dependencies

Enabled for the **Electrical input ports** variant of the block.

Outputs

On-state gate-source voltage — Demanded output voltage for on state
15 V (default)

Demanded output voltage when the driver is in on state.

Off-state gate-source voltage — Demanded output voltage for off state
0 V (default)

Demanded output voltage when the driver is in off state.

Timing

Low-side propagation delay (logic 0->logic 1) — t_{pLH} in timing diagram
50 ns (default)

Low-side propagation delay when the input logic goes from 0 to 1.

High-side dead time (logic 0->logic 1) — t_{dLH} in timing diagram
100 ns (default)

High-side dead time when the input logic goes from 0 to 1.

1-274

High-side propagation delay (logic 1->logic 0) — t_{pHL} in timing diagram
50 ns (default)

High-side propagation delay when the input logic goes from 1 to 0.

Low-side dead time (logic 1->logic 0) — t_{dHL} in timing diagram
100 ns (default)

Low-side dead time when the input logic goes from 1 to 0.

Dynamics

Parameterization — Select driver parameterization
Output impedance (default) | Rise and fall times

Select the type of driver parameterization:

- Output impedance — Specify on-state and off-state gate drive resistances.
- Rise and fall times — Specify rise time, fall time, and load capacitance.

On-state gate drive resistance — Demanded output voltage for on state
2 Ω (default)

Demanded output voltage when the driver is in on state.

Dependencies

Enabled when the **Parameterization** parameter is set to Output impedance.

Off-state gate drive resistance — Demanded output voltage for off state
2 Ω (default)

Demanded output voltage when the driver is in off state.

Dependencies

Enabled when the **Parameterization** parameter is set to Output impedance.

Rise time — Driver rise time
20 ns (default)

Driver rise time from 10% to 90%.

Dependencies

Enabled when the **Parameterization** parameter is set to `Rise and fall times`.

Fall time — Driver fall time
20 ns (default)

Driver fall time from 90% to 10%.

Dependencies

Enabled when the **Parameterization** parameter is set to `Rise and fall times`.

Load capacitance for rise and fall times — Driver load capacitance
10 nF (default)

Driver load capacitance.

Dependencies

Enabled when the **Parameterization** parameter is set to `Rise and fall times`.

Faults

Enable faults — Select Yes to enable faults modeling
No (default) | Yes

Select `Yes` to enable faults modeling. The associated parameters in the **Faults** section become visible to let you specify time to fail and the failure mode.

Simulation time for fault event — Time before entering faulted state
1 s (default)

Set the simulation time at which you want the block to enter the faulted state.

Dependencies

Enabled when the **Enable faults** parameter is set to `Yes`.

Failure mode — State of driver after failure
Fail input fixed at logic 0 (default) | Fail input fixed at logic 1 | Fail high side off | Fail high side on | Fail low side off | Fail low side on | Fail high and low sides off | Fail high and low sides on

Select the driver state after the failure.

Dependencies

Enabled when the **Enable faults** parameter is set to `Yes`.

See Also

Controlled PWM Voltage | Gate Driver | N-Channel IGBT | N-Channel MOSFET | P-Channel MOSFET

Introduced in R2017b

Incandescent Lamp

Model incandescent lamp, with resistance depending on temperature

Library

Passive Devices

Description

The Incandescent Lamp block models an incandescent lamp, the key characteristic of which is that the resistance increases as the filament warms up.

Under the simplifying assumption that the rate of heat loss from the filament is proportional to temperature difference to ambient, the temperature of the filament is governed by

$$kt_c \frac{dT}{dt} = i^2 R - kT$$

and the filament resistance is governed by the following equation

$$R = R_0 \left(1 + \alpha T\right)$$

where:

- R_0 is the initial resistance at turn-on (when filament is at ambient temperature).
- T is the filament temperature relative to ambient temperature.
- α is the resistance temperature coefficient.

- t_c is the thermal time constant.
- k is the heat transfer coefficient.
- R is the filament resistance.
- i is the filament current.

There are two parameterization options:

- If you select `Specify resistance values directly`, the block uses values that you provide for filament resistance when on and at turn-on to determine the value for the heat transfer coefficient.
- If you select `Specify currents`, the block uses values that you provide for filament current when on and at turn-on to determine the value for the heat transfer coefficient.

Optionally you can specify a simulation time at which the lamp fails by providing a finite value for the **Time at which lamp goes open circuit** parameter on the **Faults** tab. When in the open-circuit state, the lamp resistance is set to be the value of the **Open-circuit resistance** parameter.

Parameters

- "Resistance Tab" on page 1-279
- "Dynamics Tab" on page 1-280
- "Faults Tab" on page 1-281

Resistance Tab

Parameterization

Select one of the following methods for block parameterization:

- `Specify resistance values directly` — Provide the values for filament resistance at turn-on and when on in steady state. The block determines the value for the heat transfer coefficient based on these values. This is the default option.
- `Specify currents` — Provide the values for filament current at turn-on and when on in steady state. The block determines the value for the heat transfer coefficient based on these values.

Initial resistance at turn-on

The resistance seen by the external circuit when the lamp is initially turned on. This parameter is only visible when you select `Specify resistance values directly` for the **Parameterization** parameter. The default value is `0.15` Ω.

Steady-state resistance when on

The resistance seen by the external circuit when the lamp is on and in steady state. This parameter is only visible when you select `Specify resistance values directly` for the **Parameterization** parameter. This resistance should be greater than the **Initial resistance at turn-on**. The default value is `1` Ω.

Inrush current at turn-on

The current through the lamp when it is initially turned on. This parameter is only visible when you select `Specify currents` for the **Parameterization** parameter. The default value is `70` A.

Steady-state current when on

The current through the lamp when it is on and in steady state. This parameter is only visible when you select `Specify currents` for the **Parameterization** parameter. This current should be less than the **Inrush current at turn-on**. The default value is `10` A.

Rated voltage

The rated voltage for the lamp, and the voltage value for which the resistance or current values are provided in the on and turn-on states. The default value is `12` V.

Resistance temperature coefficient

The fractional increase in resistance per unit increase in temperature. The default value is `0.004` 1/K.

Dynamics Tab

Thermal time constant

The first-order thermal time constant for filament temperature when the lamp is turned on or off. The default value is `25` ms.

Initial lamp state

Select between `On` and `Off`. The default is `Off`.

Faults Tab

Time at which lamp goes open circuit

For simulation times greater than this parameter value the filament resistance becomes equal to the **Open-circuit resistance**. The default value is `inf` seconds. Specifying a finite value for this parameter lets you simulate the fault dynamics when the bulb burns out.

Open-circuit resistance

The value of the filament resistance used when the lamp goes open-circuit. The default value is `1e6` Ω.

Ports

The block has the following ports:

+

Positive electrical port

-

Negative electrical port

Incremental Shaft Encoder

Model device that converts information about angular shaft position into electrical pulses
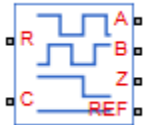
Library

Sensors

Description

The Incremental Shaft Encoder block represents a device that converts information about the angular position of a shaft into electrical pulses. The block produces N pulses on ports A and B per shaft revolution, where N is the value you specify for the **Pulses per revolution** parameter. Pulses A and B are 90 degrees out of phase. If the shaft rotates in a positive direction, then A leads B. The block produces a single index pulse on port Z once per revolution. The Z pulse positive transition always coincides with an A pulse positive transition, and Z pulse length is equal to that for the A and B pulses. The voltages at output ports A, B, and Z are defined relative to the REF reference port voltage.

Use this block if you need to model the shaft encoder signals, either to support development of a decoding algorithm, or to include the quantization effects. Otherwise, use the Simscape Ideal Rotational Motion Sensor block.

Basic Assumptions and Limitations

The Incremental Shaft Encoder block has the following limitations:

- The Incremental Shaft Encoder block is not linearizable. Use the Simscape Ideal Rotational Motion Sensor block for control design studies where you need to linearize your model.

Parameters

Pulses per revolution

The number of pulses produced on each of the A and B phases per revolution of the shaft. The default value is 2.

Output voltage amplitude

The amplitude of the shaft encoder output voltage when the output is high. The default value is 5 V.

Index pulse offset relative to shaft initial angle

The offset of the index pulse Z relative to the angle of the shaft at the start of the simulation. This parameter lets you set the initial location of the index pulse. The default value is 0°.

Ports

The block has the following ports:

R

Mechanical rotational conserving port associated with the sensor positive probe

C

Mechanical rotational conserving port associated with the sensor negative (reference) probe

A

Encoded electrical output

B

Encoded electrical output

Z

Index, or synchronization, electrical output

REF

Floating zero volt reference

See Also

Ideal Rotational Motion Sensor

Topics
Linear Electric Actuator with Control

Induction Motor

Model induction motor powered by ideal AC supply
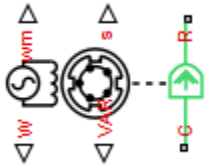
Library

Rotational Actuators

Description

The Induction Motor block represents the electrical and torque characteristics of an induction motor powered by an ideal AC supply. The following figure shows the equivalent circuit model of the Induction Motor block.
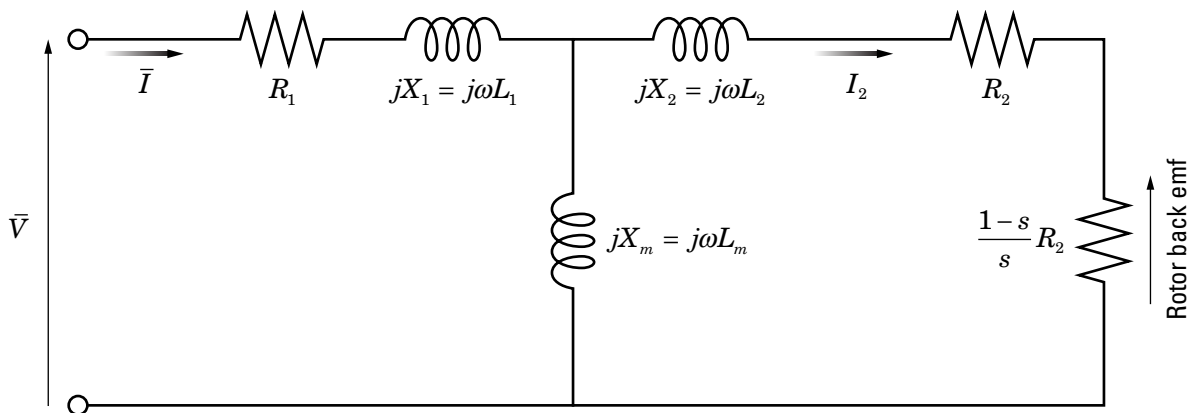
In the figure:

- R_1 is the stator resistance.
- R_2 is the rotor resistance with respect to the stator.
- L_1 is the stator inductance.
- L_2 is the rotor inductance with respect to the stator.
- L_m is magnetizing inductance.
- s is the rotor slip.
- $\overline{V}$ and $\overline{I}$ are the sinusoidal supply voltage and current phasors.

Rotor slip s is defined in terms of the mechanical rotational speed ω_m, the number of pole pairs p, and the electrical supply frequency ω by

$$s = 1 - \frac{p\omega_m}{\omega}$$

This means that the slip is one when starting, and zero when running synchronously with the supply frequency.

For an *n*-phase induction motor the torque-speed relationship is given by:

$$T = \frac{npR_2}{s\omega} \frac{V_{rms}^{\,2}}{\left(R_1 + R_2 + \dfrac{1-s}{s}R_2\right)^2 + \left(X_1 + X_2\right)^2}$$

where:

- V_{rms} is the line-neutral supply voltage for a star-configuration induction motor, and the line-to-line voltage for a delta-configuration induction motor.
- *n* is the number of phases.

You can parameterize this block in terms of the preceding equivalent circuit model parameters or in terms of the motor ratings the block uses to derive these parameters.

This block produces a positive torque acting from the mechanical C to R ports.

Thermal Ports

The block has two optional thermal ports, one per winding, hidden by default. To expose the thermal ports, right-click the block in your model, and then from the context menu select **Simscape > Block choices > Show thermal port**. This action displays the thermal ports on the block icon, and adds the **Temperature Dependence** and **Thermal Port** tabs to the block dialog box. These tabs are described further on this reference page.

Use the thermal ports to simulate the effects of copper resistance losses that convert electrical power to heat. For more information on using thermal ports in actuator blocks, see "Simulating Thermal Effects in Rotational and Translational Actuators".

Basic Assumptions and Limitations

The model is based on the following assumptions:

- The block does not model the starting mechanism for a single-phase induction motor.
- When you parameterize the block by motor ratings, the block derives the equivalent circuit model parameters by assuming that the effect of the magnetizing inductance L_m is negligible, and the magnetizing inductance is not included in the simulated component.

Parameters

- "Electrical Torque Tab" on page 1-287
- "Power Supply Tab" on page 1-290
- "Mechanical Tab" on page 1-290
- "Temperature Dependence Tab" on page 1-290
- "Thermal Port Tab" on page 1-291

Electrical Torque Tab

Model parameterization

 Select one of the following methods for block parameterization:

- By `motor ratings` — Provide electrical torque parameters that the block converts to an equivalent circuit model of the motor assuming that the effect of the magnetizing inductance L_m is negligible. This is the default method.
- By `equivalent circuit parameters` — Provide electrical parameters for an equivalent circuit model of the motor.

Stator resistance R1

Resistance of the stator winding. The default value is $1 \, \Omega$. This parameter is only visible when you select By `equivalent circuit parameters` for the **Model parameterization** parameter.

Rotor resistance R2

Resistance of the rotor, specified with respect to the stator. The default value is $1 \, \Omega$. This parameter is only visible when you select By `equivalent circuit parameters` for the **Model parameterization** parameter.

Stator inductance L1

Inductance of the stator winding. The default value is `0.02` H. This parameter is only visible when you select By `equivalent circuit parameters` for the **Model parameterization** parameter.

Rotor inductance L2

Inductance of the rotor, specified with respect to the stator. The default value is `0.02` H. This parameter is only visible when you select By `equivalent circuit parameters` for the **Model parameterization** parameter.

Magnetizing inductance Lm

Magnetizing inductance of the stator. This parameter is only visible when you select By `equivalent circuit parameters` for the **Model parameterization** parameter. Its value is hard to estimate from motor parameters, but the effect is usually small. If you do not know its value, use a typical value of 25 times the **Stator inductance L1** value. The default value is `0.5` H.

Rated mechanical power

Mechanical power the motor delivers when running at the rated speed. The default value is `825` W. This parameter is only visible when you select By `motor ratings` for the **Model parameterization** parameter.

Rated speed

Speed at which the motor delivers the specified **Rated mechanical power** value. The default value is `3.5e+03` rpm. This parameter is only visible when you select By `motor ratings` for the **Model parameterization** parameter.

Rated RMS line-to-line voltage

Line-to-line voltage at which the motor ratings are specified. The default value is `200` V. This parameter is only visible when you select `By motor ratings` for the **Model parameterization** parameter.

Rated supply frequency

Frequency of the AC supply voltage at which the motor ratings are specified. The default value is `60` hertz. This parameter is only visible when you select `By motor ratings` for the **Model parameterization** parameter.

Rated RMS line current

Line current at which the motor delivers the specified **Rated mechanical power** value. The default value is `2.7` A. This parameter is only visible when you select `By motor ratings` for the **Model parameterization** parameter.

R1 parameterization

Select one of the following parameterizations for the equivalent circuit resistance, R_1, of the motor:

- `From motor efficiency` — Calculate R_1 from the motor efficiency. This is the default method.
- `From power factor` — Calculate R_1 from the motor power factor.
- `Use measured stator resistance R1` — Measure R_1 directly.

This parameter is only visible when you select `By motor ratings` for the **Model parameterization** parameter.

Motor efficiency (percent)

the percentage of input power to the motor that gets delivered to the mechanical load when running at the **Rated speed** value. The default value is `95`. This parameter is only visible when you select `By motor ratings` for the **Model parameterization** parameter and `From motor efficiency` for the **R1 parameterization** parameter.

Motor power factor

The cosine of the angle by which the supply current lags the supply voltage when running at the **Rated mechanical power** value. The default value is `0.93`. This parameter is only visible when you select `By motor ratings` for the **Model parameterization** parameter and `From power factor` for the **R1 parameterization** parameter.

Measured stator resistance R1

the measured stator resistance. The default value is `1` Ω. This parameter is only visible when you select `By motor ratings` for the **Model parameterization**

parameter and `Use measured stator resistance R1` for the **R1 parameterization** parameter.

Number of pole pairs

Total number of pole pairs for the motor. The default value is `1`.

Number of phases

Number of supply phases. The default value is `3`.

Stator connections

Select one of the following motor configurations:

- `Delta configuration` — Connect the motor stator windings in delta configuration. This is the default method.
- `Star configuration` — Connect the motor stator windings in star configuration.

Power Supply Tab

Supply RMS line-to-line voltage

The line-to-line voltage that supplies the motor. The default value is `200` V.

Supply frequency

Frequency of the AC supply voltage. The default value is `60` hertz.

Mechanical Tab

Rotor inertia

Rotor inertia. The default value is `0.1` kg*m^2. The value can be zero.

Rotor damping

Rotor damping. The default value is `2e-06` N*m/(rad/s). The value can be zero.

Initial rotor speed

Speed of the rotor at the start of the simulation. The default value is `0` rpm.

Temperature Dependence Tab

This tab appears only for blocks with exposed thermal ports. For more information, see "Thermal Ports" on page 1-287.

Resistance temperature coefficients, [alpha_1 alpha_2]

A 1 by 2 row vector defining the coefficient α in the equation relating resistance to temperature, as described in "Thermal Model for Actuator Blocks". The first element corresponds to the stator, and the second to rotor. The default value is for copper, and is [0.00393 0.00393] 1/K.

Measurement temperature

The temperature for which motor parameters are defined. The default value is 25 °C.

Thermal Port Tab

This tab appears only for blocks with exposed thermal ports. For more information, see "Thermal Ports" on page 1-287.

Thermal masses, [M_1 M_2]

A 1 by 2 row vector defining the thermal mass for the stator and rotor windings. The thermal mass is the energy required to raise the temperature by one degree. The default value is [100 100] J/K.

Initial temperatures, [T_A T_B]

A 1 by 2 row vector defining the temperature of the stator and rotor thermal ports at the start of simulation. The default value is [25 25] °C.

Ports

The block has the following ports:

W

Real power.

wm

Mechanical speed.

VAR

Imaginary power.

s

Motor slip.

C

Mechanical rotational conserving port.

R

Mechanical rotational conserving port.

H1

Stator thermal port. For more information, see "Thermal Ports" on page 1-287.

H2

Rotor thermal port. For more information, see "Thermal Ports" on page 1-287.

References

[1] S.E. Lyshevski. *Electromechanical Systems, Electric Machines, and Applied Mechatronics*, CRC, 1999.

See Also

DC Motor | Servomotor | Shunt Motor | Universal Motor

Inductor

Inductor model including tolerance, operational limits, and fault behavior
Library: Simscape / Electronics / Passive Devices

Description

The Inductor block lets you model linear inductors, including the following effects:

- "Tolerances" on page 1-294
- "Operating Limits" on page 1-294
- "Faults" on page 1-295

You can turn these modeling options on and off independently of each other. When all the additional options are turned off, the component behavior is identical to the Simscape Foundation library Inductor block.

In its simplest form, the Inductor block models a linear inductor, described with the following equation:

$$V = L \frac{dI}{dt}$$

where:

- V is voltage.
- L is inductance.
- I is current.
- t is time.

To model a nonlinear inductor, use the Nonlinear Inductor block.

Tolerances

You can apply tolerances to the nominal value you provide for the **Inductance** parameter. Datasheets typically provide a tolerance percentage for a given inductor type. The table shows how the block applies tolerances and calculates inductance based on the selected **Tolerance application** option.

Option	Inductance Value
`None — use nominal value`	L
`Random tolerance`	Uniform distribution: $L \cdot (1 - tol + 2 \cdot tol \cdot$ `rand`$)$ Gaussian distribution: $L \cdot (1 + tol \cdot$ `randn` $/ nSigma)$
`Apply maximum tolerance value`	$L \cdot (1 + tol)$
`Apply minimum tolerance value`	$L \cdot (1 - tol)$

In the table,

- L is the **Inductance** parameter value, nominal inductance.
- tol is fractional tolerance, **Inductance tolerance (%)** /100.
- $nSigma$ is the value you provide for the **Number of standard deviations for quoted tolerance** parameter.
- `rand` and `randn` are standard MATLAB functions for generating uniform and normal distribution random numbers.

Operating Limits

Inductors are typically rated with a particular saturation current, and possibly with a maximum allowable power dissipation. You can specify operating limits in terms of these values, to generate warnings or errors if the inductor is driven outside its specification.

When an operating limit is exceeded, the block can either generate a warning or stop the simulation with an error. For more information, see the "Operating Limits" on page 1-298 parameters section.

Faults

Instantaneous changes in inductor parameters are unphysical. Therefore, when the Inductor block enters the faulted state, short-circuit and open-circuit voltages transition to their faulted values over a period of time based on this formula:

$$CurrentValue = FaultedValue – (FaultedValue – UnfaultedValue) \cdot \text{sech}(\Delta t / \tau)$$

where:

- Δt is time since the onset of the fault condition.
- τ is user-defined time constant associated with the fault transition.

For short-circuit faults, the conductance of the short-circuit path also changes according to the $\text{sech}(\Delta t / \tau)$ function from a small value (representing an open-circuit path) to a large value.

The block can trigger the start of fault transition:

- At a specific time
- After voltage exceeds the maximum permissible value a certain number of times
- When current exceeds the maximum permissible value for longer than a specific time interval

You can enable or disable these trigger mechanisms separately, or use them together if more than one trigger mechanism is required in a simulation. When more than one mechanism is enabled, the first mechanism to trigger the fault transition takes precedence. In other words, a component fails no more than once per simulation.

You can also choose whether to issue an assertion when a fault occurs, by using the **Reporting when a fault occurs** parameter. The assertion can take the form of a warning or an error. By default, the block does not issue an assertion.

Faultable inductors often require that you use the fixed-step local solver rather than the variable-step solver. In particular, if you model transitions to a faulted state that include short circuits, MathWorks recommends that you use the fixed-step local solver. For more information, see "Making Optimal Solver Choices for Physical Simulation" (Simscape).

Variables

Use the **Variables** section of the block interface to set the priority and initial target values for the block variables prior to simulation. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

The **Inductor current** variable lets you specify a high-priority target for the initial inductor current at the start of simulation.

Ports

Conserving

+ — Positive terminal
electrical

Electrical conserving port associated with the inductor positive terminal.

- — Negative terminal
electrical

Electrical conserving port associated with the inductor negative terminal.

Parameters

Main

Inductance — Nominal inductance value
1 µH (default)

The nominal inductance value. Inductance value must be greater than zero.

Tolerance (%) — Inductor tolerance, in percent
20 (default)

The inductor tolerance as defined on the manufacturer datasheet.

Tolerance application — Select how to apply tolerance during simulation
None — use nominal value (default) | Random tolerance | Apply maximum tolerance value | Apply minimum tolerance value

Select how to apply tolerance during simulation:

- None — use nominal value — The block does not apply tolerance, it uses the nominal inductance value.
- Random tolerance — The block applies random offset to the inductance value, within the tolerance value limit. You can choose Uniform or Gaussian distribution for calculating the random number by using the **Tolerance distribution** parameter.
- Apply maximum tolerance value — The inductance is increased by the specified tolerance percent value.
- Apply minimum tolerance value — The inductance is decreased by the specified tolerance percent value.

Tolerance distribution — Select the distribution type
Uniform (default) | Gaussian

Select the distribution type for random tolerance:

- Uniform — Uniform distribution
- Gaussian — Gaussian distribution

Dependencies

Enabled when the **Tolerance application** parameter is set to Random tolerance.

Number of standard deviations for quoted tolerance — Used for calculating the Gaussian random number
4 (default)

Number of standard deviations for calculating the Gaussian random number.

Dependencies

Enabled when the **Tolerance distribution** parameter is set to Gaussian.

Series resistance — Equivalent series resistance of the inductor
0 Ω (default)

Equivalent series resistance (ESR) of the inductor, as sometimes specified on manufacturer datasheets. The default value is consistent with the Simscape Foundation library Inductor block. If you model faults, specify a positive value for this parameter.

`Parallel conductance` — Parallel leakage path associated with the inductor
1e-9 1/Ω (default)

Parallel leakage path associated with the inductor. Simulation of some circuits may require the presence of a small parallel conductance. You can also use this parameter to model the inductor core losses.

Operating Limits

`Enable operating limits` — Select Yes to enable reporting when the operational limits are exceeded
No (default) | Yes

Select `Yes` to enable reporting when the operational limits are exceeded. The associated parameters in the **Operating Limits** section become visible to let you select the reporting method and specify the operating limits in terms of power and current.

`Reporting when operating limits exceeded` — Select the reporting method
Warn (default) | Error

Select what happens when an operating limit is exceeded:

- `Warn` — The block issues a warning.
- `Error` — Simulation stops with an error.

Dependencies

Enabled when the **Enable operating limits** parameter is set to `Yes`.

`Saturation current` — Inductor saturation current
1 A (default)

Inductor saturation current, as defined in the manufacturer datasheets. If the current exceeds this value, the core material enters saturation.

Dependencies

Enabled when the **Enable operating limits** parameter is set to `Yes`.

Power rating — Maximum power dissipation in the inductor
1 W (default)

Maximum instantaneous power dissipation in the resistance and conductance elements associated with the inductor.

Dependencies

Enabled when the **Enable operating limits** parameter is set to Yes.

Faults

Enable faults — Select Yes to enable faults modeling
No (default) | Yes

Select Yes to enable faults modeling. The associated parameters in the **Faults** section become visible to let you select the reporting method and specify the trigger mechanism (temporal or behavioral). You can enable these trigger mechanisms separately or use them together.

Reporting when a fault occurs — Choose whether to issue an assertion when a fault occurs
None (default) | Warn | Error

Choose whether to issue an assertion when a fault occurs:

- None — The block does not issue an assertion.
- Warn — The block issues a warning.
- Error — Simulation stops with an error.

Dependencies

Enabled when the **Enable faults** parameter is set to Yes.

Location of fault node (% of total turns from - terminal) — Percentage of turns in the subinductor that is in contact with the - port of the block
50 (default)

In practice, faults are enabled by segmenting the inductor into two coupled subinductors, connected in a series. The inductance is proportional to the square of the number of turns

1-299

in the respective segment, and the series resistance of each subinductor is proportional to the number of turns in each segment. The parallel conductance spans both segments.

This parameter indicates the percentage of turns that are assigned to the subinductor that is in contact with the – port of the block. The remaining turns are assigned to the other subinductor. The default value is 50, which means that the overall inductance is divided into two equal, coupled subinductors.

Dependencies

Enabled when the **Enable faults** parameter is set to Yes.

Faulted coupling factor — Mutual coupling between the two subinductors
0.9999 (default)

The faulted value for the mutual coupling between the two subinductors. The differential equations governing such a construction break down in the limit of perfect coupling, so the coupling should be less than unity. A value of 0 corresponds to no coupling at all between the subinductors. Physically, this corresponds to a fault that affects the flux within the inductor core. This could be a crack in the core material, or windings coming away from the core.

The default value of this parameter is also the internal value the block uses when computing a faultable inductor in the unfaulted state. For an unfaultable inductor, there is only a single equation being solved, and this corresponds to the ideal case of perfect mutual coupling.

Dependencies

Enabled when the **Enable faults** parameter is set to Yes.

Short-circuit turns — Select whether fault results in one of the segments being short-circuited
No (default) | To negative terminal | To positive terminal

Select whether the fault results in one of the subinductor segments being short-circuited:

- No — The fault does not produce a short circuit.
- To negative terminal — The fault short-circuits the subinductor that is in contact with the – port of the block.
- To positive terminal — The fault short-circuits the subinductor that is in contact with the + port of the block.

Dependencies

Enabled when the **Enable faults** parameter is set to `Yes`.

`Open-circuit at fault node` — Select whether to apply an open-circuit fault between the segments
No (default) | Yes

Select whether to apply an open-circuit fault between the two subinductor segments. The default is `No`. Even with an open-circuit fault, the characteristics of the subinductors may still be related, depending on the value of the **Faulted coupling factor** parameter:

- If the coupling factor is not zero, the subinductors are galvanically isolated from each other, but they are still magnetically coupled. Physically, this corresponds to a break in the winding.

- With zero coupling factor, the subinductors are galvanically and magnetically isolated.

Dependencies

Enabled when the **Enable faults** parameter is set to `Yes`.

`Ground fault` — Select whether fault results in one of the segments being short-circuited
No (default) | Negative terminal side of fault node | Positive terminal side of fault node

Select whether, in case of fault, there is a path for current to flow towards the ground node:

- `No` — The fault does not result in a connection to ground.

- `Negative terminal side of fault node` — The side that is in contact with the – port of the block is connected to ground.

- `Positive terminal side of fault node` — The side that is in contact with the + port of the block is connected to ground.

If the **Open-circuit at fault node** parameter is set to `Yes`, you need to specify which side (negative or positive) is connected to ground. If there is no open circuit, the two options behave similarly. Physically, this corresponds to a breakdown in the insulation between the windings and the grounded core or chassis.

Dependencies

Enabled when the **Enable faults** parameter is set to `Yes`.

`Conductance of faulted ground path` — Mutual coupling between the two subinductors
0 1/Ω (default)

If there is a ground fault, this parameter represents the conductance of the current path to ground. For example, if the path to ground is through the core material, then specify a small conductance value depending on the core material being used. For highly conductive core material or for chassis-shorts, specify a higher conductance value.

Dependencies

Enabled when the **Ground fault** parameter is set to `Negative terminal side of fault node` or `Positive terminal side of fault node`.

`Fault transition time constant` — Time constant for the transition to faulted state
1 ms (default)

Time constant associated with the transition to the faulted state, as described in "Faults" on page 1-299.

Dependencies

Enabled when the **Enable faults** parameter is set to `Yes`.

`Enable temporal fault trigger` — Select Yes to enable time-based fault triggering
No (default) | Yes

Select `Yes` to enable time-based fault triggering. You can enable the temporal and behavioral trigger mechanisms separately or use them together.

Dependencies

Enabled when the **Enable faults** parameter is set to `Yes`.

`Simulation time for a fault event` — Time before entering faulted state
1 s (default)

Set the simulation time at which you want the block to start entering the fault state.

Dependencies

Enabled when the **Enable temporal fault trigger** parameter is set to Yes.

Enable behavioral fault trigger — Select Yes to enable behavioral fault triggering
No (default) | Yes

Select Yes to enable behavioral fault triggering. You can enable the temporal and behavioral trigger mechanisms separately or use them together.

Dependencies

Enabled when the **Enable faults** parameter is set to Yes.

Maximum permissible voltage — Voltage threshold to fault transition
100 V (default)

Define the voltage threshold to a fault transition. If the voltage value exceeds this threshold a certain number of times, specified by the **Number of events to fail when exceeding voltage** parameter value, then the block starts entering the fault state.

Dependencies

Enabled when the **Enable behavioral fault trigger** parameter is set to Yes.

Number of events to fail when exceeding voltage — Maximum number of times the voltage exceeds the threshold
1 (default)

Since the physical mechanism underlying voltage-based failures depends on one or more partial discharge events occurring, this parameter allows you to set the number of voltage overshoots that the inductor can withstand before the fault transition begins. Note that the block does not check the time spent in the overvoltage condition, only the number of transitions.

Dependencies

Enabled when the **Enable behavioral fault trigger** parameter is set to Yes.

Maximum permissible current — Current threshold to fault transition
1 A (default)

Define the current threshold to a fault transition. If the current value exceeds this threshold for longer than the **Time to fail when exceeding current** parameter value, then the block starts entering the fault state.

Dependencies

Enabled when the **Enable behavioral fault trigger** parameter is set to Yes.

Time to fail when exceeding current — Maximum length of time the current exceeds the threshold
1 s (default)

Set the maximum length of time that the current can exceed the maximum permissible value without triggering the fault.

Dependencies

Enabled when the **Enable behavioral fault trigger** parameter is set to Yes.

See Also

Capacitor | Fault | Mutual Inductor | Nonlinear Inductor | Resistor

Introduced in R2016a

Light-Emitting Diode

Model light-emitting diode as exponential diode and current sensor in series

Library

Sensors

Description

The Light-Emitting Diode block represents a light-emitting diode as an exponential diode in series with a current sensor. The optical power presented at the signal port W is equal to the product of the current flowing through the diode and the **Optical power per unit current** parameter value.

The exponential diode model provides the following relationship between the diode current *I* and the diode voltage *V*:

$$I = IS \cdot \left(e^{\frac{qV}{NkT_{m1}}} - 1 \right)$$

where:

- *q* is the elementary charge on an electron (1.602176e–19 Coulombs).
- *k* is the Boltzmann constant (1.3806503e–23 J/K).
- *N* is the emission coefficient.
- *IS* is the saturation current.
- T_{m1} is the temperature at which the diode parameters are specified, as defined by the **Measurement temperature** parameter value.

When $(qV / NkT_{m1}) > 80$, the block replaces $e^{\frac{qV}{NkT_{m1}}}$ with $(qV / NkT_{m1} - 79)e^{80}$, which matches the gradient of the diode current at $(qV / NkT_{m1}) = 80$ and extrapolates linearly.

When $(qV / NkT_{m1}) < -79$, the block replaces $e^{\frac{qV}{NkT_{m1}}}$ with $(qV / NkT_{m1} + 80)e^{-79}$, which also matches the gradient and extrapolates linearly. Typical electrical circuits do not reach these extreme values. The block provides this linear extrapolation to help convergence when solving for the constraints during simulation.

When you select `Use parameters IS and N` for the **Parameterization** parameter, you specify the diode in terms of the **Saturation current IS** and **Emission coefficient N** parameters. When you select `Use I-V curve data points` for the **Parameterization** parameter, you specify two voltage and current measurement points on the diode I-V curve and the block derives the *IS* and *N* values. When you specify current and voltage measurements, the block calculates *IS* and *N* as follows:

*
$$\text{N} = ((V_1 - V_2) / V_t) / (\log(I_1) - \log(I_2))$$

*
$$\text{IS} = \left(I_1 / (\exp(V_1 / (\text{N}V_t)) - 1) + I_2 / (\exp(V_2 / (\text{N}V_t)) - 1) \right) / 2$$

where:

* $V_t = kT_{m1} / q$.
* V_1 and V_2 are the values in the **Voltages [V1 V2]** vector.
* I_1 and I_2 are the values in the **Currents [I1 I2]** vector.

The exponential diode model provides the option to include a junction capacitance:

* When you select `Fixed or zero junction capacitance` for the **Junction capacitance** parameter, the capacitance is fixed.
* When you select `Use parameters CJO, VJ, M & FC` for the **Junction capacitance** parameter, the block uses the coefficients *CJO*, *VJ*, *M*, and *FC* to calculate a junction capacitance that depends on the junction voltage.
* When you select `Use C-V curve data points` for the **Junction capacitance** parameter, the block uses three capacitance values on the C-V capacitance curve to estimate *CJO*, *VJ*, and *M* and uses these values with the specified value of *FC* to calculate a junction capacitance that depends on the junction voltage. The block calculates *CJO*, *VJ*, and *M* as follows:

- $$CJ0 = C_1((V_{R2} - V_{R1}) / (V_{R2} - V_{R1}(C_2 / C_1)^{-1/M}))^M$$

- $$VJ = -(-V_{R2}(C_1 / C_2)^{-1/M} + V_{R1}) / (1 - (C_1 / C_2)^{-1/M})$$

- $$M = \log(C_3 / C_2) / \log(V_{R2} / V_{R3})$$

where:

- V_{R1}, V_{R2}, and V_{R3} are the values in the **Reverse bias voltages [VR1 VR2 VR3]** vector.
- C_1, C_2, and C_3 are the values in the **Corresponding capacitances [C1 C2 C3]** vector.

It is not possible to estimate *FC* reliably from tabulated data, so you must specify its value using the **Capacitance coefficient FC** parameter. In the absence of suitable data for this parameter, use a typical value of 0.5.

The reverse bias voltages (defined as positive values) should satisfy $V_{R3} > V_{R2} > V_{R1}$. This means that the capacitances should satisfy $C_1 > C_2 > C_3$ as reverse bias widens the depletion region and hence reduces capacitance. Violating these inequalities results in an error. Voltages V_{R2} and V_{R3} should be well away from the Junction potential *VJ*. Voltage V_{R1} should be less than the Junction potential *VJ*, with a typical value for V_{R1} being 0.1 V.

The voltage-dependent junction is defined in terms of the capacitor charge storage Q_j as:

- For $V < FC \cdot VJ$:

$$Q_j = CJ0 \cdot (VJ / (M - 1)) \cdot ((1 - V / VJ)^{1-M} - 1)$$

- For $V \geq FC \cdot VJ$:

$$Q_j = CJ0 \cdot F_1 + (CJ0 / F_2) \cdot (F_3 \cdot (V - FC \cdot VJ) + 0.5(M / VJ) \cdot (V^2 - (FC \cdot VJ)^2))$$

where:

- $$F_1 = (VJ / (1 - M)) \cdot (1 - (1 - FC)^{1-M}))$$

- $$F_2 = (1 - FC)^{1+M}))$$

- $$F_3 = 1 - FC \cdot (1 + M)$$

These equations are the same as used in [2], except that the temperature dependence of *VJ* and *FC* is not modeled. This model does not include the diffusion capacitance term that affects performance for high frequency switching applications.

The Light-Emitting Diode block contains several options for modeling the dependence of the diode current-voltage relationship on the temperature during simulation. Temperature dependence of the junction capacitance is not modeled, this being a much smaller effect. For details, see the Diode reference page.

Thermal Port

The block has an optional thermal port, hidden by default. To expose the thermal port, right-click the block in your model, and then from the context menu select **Simscape > Block choices > Show thermal port**. This action displays the thermal port H on the block icon, and adds the **Thermal Port** tab to the block dialog box.

Use the thermal port to simulate the effects of generated heat and device temperature. For more information on using thermal ports and on the **Thermal Port** tab parameters, see "Simulating Thermal Effects in Semiconductors".

Variables

Use the **Variables** section of the block interface to set the priority and initial target values for the block variables prior to simulation. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

Basic Assumptions and Limitations

The Light-Emitting Diode block has the following limitations:

- When you select `Use I-V curve data points` for the **Parameterization** parameter, choose a pair of voltages near the diode turn-on voltage. Typically this is in the range from 0.05 to 1 Volt. Using values outside of this region may lead to numerical issues and poor estimates for *IS* and *N*.

- You may need to use nonzero ohmic resistance and junction capacitance values to prevent numerical simulation issues, but the simulation may run faster with these values set to zero.

Parameters

- "Main" on page 1-309
- "Junction Capacitance" on page 1-310
- "Temperature Dependence" on page 1-311

Main

Optical power per unit current

The amount of optical power the light-emitting diode generates per unit of current flowing through the diode. The default value is `0.005` W/A.

Parameterization

Select one of the following methods for model parameterization:

- `Use I-V curve data points` — Specify measured data at two points on the diode I-V curve. This is the default method.
- `Use parameters IS and N` — Specify saturation current and emission coefficient.

Currents [I1 I2]

A vector of the current values at the two points on the diode I-V curve that the block uses to calculate *IS* and *N*. This parameter is only visible when you select `Use I-V curve data points` for the **Parameterization** parameter. The default value is `[ 0.0017 0.003 ]` A.

Voltages [V1 V2]

A vector of the voltage values at the two points on the diode I-V curve that the block uses to calculate *IS* and *N*. This parameter is only visible when you select `Use I-V curve data points` for the **Parameterization** parameter. The default value is `[ 0.9 1.05 ]` V.

Saturation current, IS

The magnitude of the current that the ideal diode equation approaches asymptotically for very large reverse bias levels. This parameter is only visible when you select `Use`

parameters IS and N for the **Parameterization** parameter. The default value is 5e-5 A.

Emission coefficient, N

The diode emission coefficient or ideality factor. This parameter is only visible when you select Use parameters IS and N for the **Parameterization** parameter. The default value is 10.

Ohmic resistance, RS

The series diode connection resistance. The default value is 0.1 Ω.

Measurement temperature

The temperature at which IS or the I-V curve was measured. The default value is 25 °C.

Junction Capacitance

Junction capacitance

Select one of the following options for modeling the junction capacitance:

- Fixed or zero junction capacitance — Model the junction capacitance as a fixed value.
- Use C-V curve data points — Specify measured data at three points on the diode C-V curve.
- Use parameters CJ0, VJ, M & FC — Specify zero-bias junction capacitance, junction potential, grading coefficient, and forward-bias depletion capacitance coefficient.

Zero-bias junction capacitance, CJ0

The value of the capacitance placed in parallel with the exponential diode term. This parameter is only visible when you select Fixed or zero junction capacitance or Use parameters CJ0, VJ, M & FC for the **Junction capacitance** parameter. The default value is 20 pF.

Reverse bias voltages [VR1 VR2 VR3]

A vector of the reverse bias voltage values at the three points on the diode C-V curve that the block uses to calculate *CJ0*, *VJ*, and *M*. This parameter is only visible when you select Use C-V curve data points for the **Junction capacitance** parameter. The default value is [0.1 10 100] V.

Corresponding capacitances [C1 C2 C3]

A vector of the capacitance values at the three points on the diode C-V curve that the block uses to calculate *CJ0, VJ*, and *M*. This parameter is only visible when you select `Use C-V curve data points` for the **Junction capacitance** parameter. The default value is `[ 15 10 2 ]` pF.

Junction potential, VJ

The junction potential. This parameter is only visible when you select `Use parameters CJ0, VJ, M & FC` for the **Junction capacitance** parameter. The default value is `1` V.

Grading coefficient, M

The grading coefficient. This parameter is only visible when you select `Use parameters CJ0, VJ, M & FC` for the **Junction capacitance** parameter. The default value is `0.5`.

Capacitance coefficient, FC

Fitting coefficient that quantifies the decrease of the depletion capacitance with applied voltage. This parameter is only visible when you select `Use C-V curve data points` or `Use parameters CJ0, VJ, M & FC` for the **Junction capacitance** parameter. The default value is `0.5`.

Temperature Dependence

Parameterization

Select one of the following methods for temperature dependence parameterization:

- `None — Simulate at parameter measurement temperature` — Temperature dependence is not modeled, or the model is simulated at the measurement temperature T_{m1} (as specified by the **Measurement temperature** parameter on the **Main** tab). This is the default method.

- `Use an I-V data point at second measurement temperature T2` — If you select this option, you specify a second measurement temperature T_{m2}, and the current and voltage values at this temperature. The model uses these values, along with the parameter values at the first measurement temperature T_{m1}, to calculate the energy gap value.

- `Specify saturation current at second measurement temperature T2` — If you select this option, you specify a second measurement temperature T_{m2}, and saturation current value at this temperature. The model uses these values,

1-311

along with the parameter values at the first measurement temperature T_{m1}, to calculate the energy gap value.

- `Specify the energy gap EG` — Specify the energy gap value directly.

Current I1 at second measurement temperature

Specify the diode current *I1* value when the voltage is *V1* at the second measurement temperature. This parameter is only visible when you select `Use an I-V data point at second measurement temperature` T2 for the **Parameterization** parameter. The default value is `0.0034` A.

Voltage V1 at second measurement temperature

Specify the diode voltage *V1* value when the current is *I1* at the second measurement temperature. This parameter is only visible when you select `Use an I-V data point at second measurement temperature` T2 for the **Parameterization** parameter. The default value is `1.05` V.

Saturation current, IS, at second measurement temperature

Specify the saturation current *IS* value at the second measurement temperature. This parameter is only visible when you select `Specify saturation current at second measurement temperature` T2 for the **Parameterization** parameter. The default value is `1.8e-4` A.

Second measurement temperature

Specify the value for the second measurement temperature. This parameter is only visible when you select either `Use an I-V data point at second measurement temperature` T2 or `Specify saturation current at second measurement temperature` T2 for the **Parameterization** parameter. The default value is `125` °C.

Energy gap parameterization

This parameter is only visible when you select `Specify the energy gap EG` for the **Parameterization** parameter. It lets you select a value for the energy gap from a list of predetermined options, or specify a custom value:

- `Use nominal value for silicon (EG=1.11eV)` — This is the default.
- `Use nominal value for 4H-SiC silicon carbide (EG=3.23eV)`
- `Use nominal value for 6H-SiC silicon carbide (EG=3.00eV)`
- `Use nominal value for germanium (EG=0.67eV)`
- `Use nominal value for gallium arsenide (EG=1.43eV)`

- Use nominal value for selenium (EG=1.74eV)
- Use nominal value for Schottky barrier diodes (EG=0.69eV)
- Specify a custom value — If you select this option, the **Energy gap, EG** parameter appears in the dialog box, to let you specify a custom value for *EG*.

Energy gap, EG

Specify a custom value for the energy gap, *EG*. This parameter is only visible when you select Specify a custom value for the **Energy gap parameterization** parameter. The default value is 1.11 eV.

Saturation current temperature exponent parameterization

Select one of the following options to specify the saturation current temperature exponent value:

- Use nominal value for pn-junction diode (XTI=3) — This is the default.
- Use nominal value for Schottky barrier diode (XTI=2)
- Specify a custom value — If you select this option, the **Saturation current temperature exponent, XTI** parameter appears in the dialog box, to let you specify a custom value for *XTI*.

Saturation current temperature exponent, XTI

Specify a custom value for the saturation current temperature exponent, *XTI*. This parameter is only visible when you select Specify a custom value for the **Saturation current temperature exponent parameterization** parameter. The default value is 3.

Device simulation temperature

Specify the value for the temperature T_s, at which the device is to be simulated. The default value is 25 °C.

Ports

The block has the following ports:

W

Optical output power

+

 Electrical conserving port associated with the diode positive terminal

-

 Electrical conserving port associated with the diode negative terminal

References

[1] H. Ahmed and P.J. Spreadbury. *Analogue and digital electronics for engineers*. 2nd Edition, Cambridge University Press, 1984.

[2] G. Massobrio and P. Antognetti. *Semiconductor Device Modeling with SPICE*. 2nd Edition, McGraw-Hill, 1993.

See Also

Diode | Optocoupler | Photodiode

Multiplier

Model integrated circuit multiplier
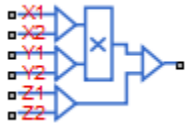
Library

Integrated Circuits

Description

The Multiplier block models an integrated circuit multiplier. The block implements the following equation, which defines the voltage applied to the output port:

$$V_{out} = A\left(\frac{(X_1 - X_2)(Y_1 - Y_2)}{K} - (Z_1 - Z_2) \right)$$

where X_1, X_2, Y_1, Y_2, Z_1, Z_2 are the voltages presented at the input ports, A is the gain, and K is the scale factor.

In a typical multiplication circuit, the output is fed back into input Z1, which results in the following gain (assuming that A is large):

$$V_{out} = \left(\frac{(X_1 - X_2)(Y_1 - Y_2)}{K} + Z_2 \right)$$

The value of the scale factor K is usually altered by an external resistor bias network. The Multiplier block implements K as an internal gain, and the external bias network is not necessary for system simulation. A typical value for K is 10, with a typical adjustment down to 3.

You can use the Multiplier block to implement a number of other functions, as well as multiplication. Examples include division, squares, and square roots. For example circuits, consult manufacturer datasheets.

The following figure shows the internal model structure of the Multiplier block. It includes the Band-Limited Op-Amp block to model finite bandwidth and slew-rate limiting.
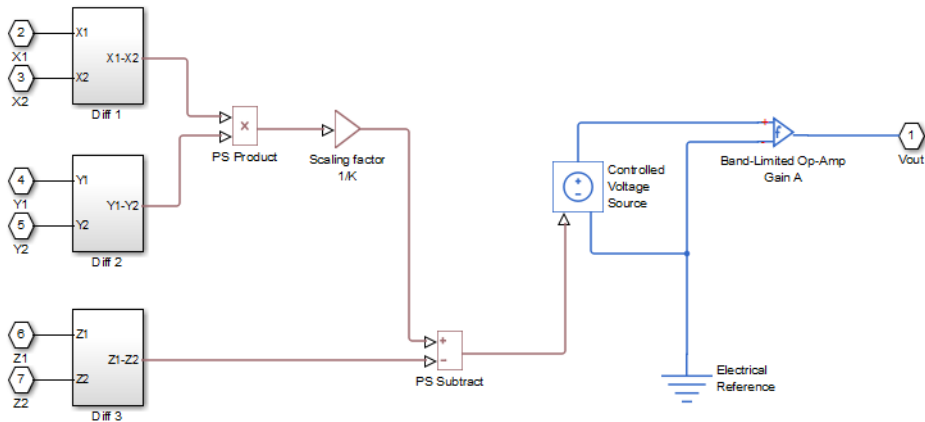
The next figure shows one of the differential subsystem blocks. All three differential subsystem blocks are identical in structure.
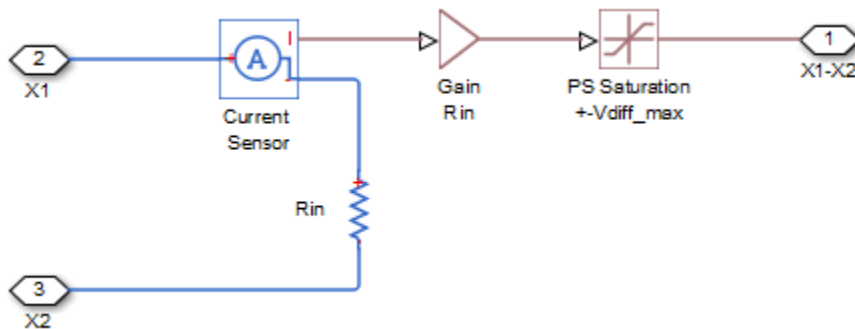
Basic Assumptions and Limitations

The Multiplier block has the following limitations:

- Only differential limiting of the inputs is implemented. You must ensure that the absolute values of the inputs you use keep the actual device operating in its linear region.

- Output current is such that the integrated circuit is operating in the linear I-V region, which can be approximated by a voltage source plus a series output resistance.

- Input offset voltage is not modeled, and the input voltage-current relationship is treated as linear within the differential signal voltage range.

Parameters

- "Main Tab" on page 1-317
- "Inputs Tab" on page 1-317
- "Outputs Tab" on page 1-318

Main Tab

Scaling factor, K

> The scaling factor K in the equation that defines output voltage. Datasheets sometimes refer to it as the scale factor, or SF. The default value is `10` V.

Gain, A

> The gain of the internal operational amplifier, corresponding to the gain A in the equation that defines output voltage. The default value is `3e3`.

Inputs Tab

Differential resistance, Rin

> Each of the differential inputs is approximated as a linear resistor with value *Rin*. Set this value to the datasheet value for differential resistance. The default value is `1e7` Ω.

Differential signal voltage range

This value, *Vdiff_max*, is used to limit the magnitude of each of the three differential input voltages. Set this value to the datasheet value for differential signal voltage range. The default value is `10` V.

Outputs Tab

Output resistance, Rout

The multiplier output stage is modeled as a voltage source plus series resistor inside the Band-Limited Op-Amp block. This parameter specifies the value of this series resistor. The default value is `0.1` Ω.

Minimum output, Vmin

The lower limit of the output voltage. The default value is `-11` V.

Maximum output, Vmax

The upper limit of the output voltage. The default value is `11` V.

Maximum slew rate, Vdot

The maximum positive or negative rate of change of output voltage magnitude. The default value is `20` V/µs.

Bandwidth, f

The bandwidth of the Band-Limited Op-Amp block. The default value is `1` MHz.

Initial output voltage, V0

The value of the initial Multiplier block output if the **Start simulation from steady-state** option is not selected in the Solver block. The default value is `0` V.

Ports

The block has six electrical conserving ports that serve as signal input ports and one electrical conserving port that outputs the multiplied signal.

Mutual Inductor

Mutual inductor model with nominal inductance tolerances for each winding, operating limits, and faults

Library: Simscape / Electronics / Passive Devices
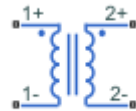
Description

The Mutual Inductor block lets you model a mutual inductor (two-winding transformer) with nominal inductance tolerances for each winding. The model includes the following effects:

- "Tolerances" on page 1-320
- "Operating Limits" on page 1-321
- "Faults" on page 1-321

You can turn these modeling options on and off independently of each other.

In the unfaulted state, the following equations describe the Mutual Inductor block behavior:

$$v_1 = L_1 \frac{di_{L1}}{dt} + M \frac{di_{L2}}{dt} + i_{L1} R_1$$

$$v_2 = L_2 \frac{di_{L2}}{dt} + M \frac{di_{L1}}{dt} + i_{L2} R_2$$

$$M = k \sqrt{L_1 L_2}$$

where:

- v_1 and v_2 are voltages across the primary and secondary winding, respectively.

- L_1 and L_2 are inductances of the primary and secondary winding.

- R_1 and R_2 are series resistances of the primary and secondary winding.

- M is mutual inductance.

- k is coefficient of coupling. To reverse one of the winding directions, use a negative value.

- t is time.

A parallel conductance is placed across the + and – terminals of the primary and secondary windings, so that $i_{L1} = i_1 – G_1 v_1$, where G_1 is the parallel conductance of the primary winding, and i_1 is the terminal current into the primary. Similar definitions and equation apply to i_{L2}.

Tolerances

You can apply tolerances separately for each winding. Datasheets typically provide a tolerance percentage for a given inductor type. Therefore, this value is the same for both windings. The table shows how the block applies tolerances to the nominal inductance value and calculates inductance based on the selected tolerance application option for the winding, **L1 tolerance application** or **L2 tolerance application**.

Option	Inductance Value
`None — use nominal value`	L
`Random tolerance`	Uniform distribution: $L \cdot (1 – tol + 2 \cdot tol \cdot$ `rand`$)$ Gaussian distribution: $L \cdot (1 + tol \cdot$ `randn` $/ nSigma)$
`Apply maximum tolerance value`	$L \cdot (1 + tol)$
`Apply minimum tolerance value`	$L \cdot (1 – tol)$

In the table:

- L is nominal inductance for the primary or secondary winding, **Inductance L1** or **Inductance L2** parameter value.

- tol is fractional tolerance, **Tolerance (%)** /100.

- *nSigma* is the value you provide for the **Number of standard deviations for quoted tolerance** parameter.
- `rand` and `randn` are standard MATLAB functions for generating uniform and normal distribution random numbers.

Operating Limits

Inductors are typically rated with a particular saturation current, and possibly with a maximum allowable power dissipation. You can specify operating limits in terms of these values, to generate warnings or errors if the inductor is driven outside its specification.

When an operating limit is exceeded, the block can either generate a warning or stop the simulation with an error. For more information, see the "Operating Limits" on page 1-326 parameters section.

Faults

Instantaneous changes in inductor parameters are unphysical. Therefore, when the Mutual Inductor block enters the faulted state, short-circuit and open-circuit voltages transition to their faulted values over a period of time based on this formula:

CurrentValue = FaultedValue – (FaultedValue – UnfaultedValue) · sech(Δt / τ)

where:

- Δt is time since the onset of the fault condition.
- τ is user-defined time constant associated with the fault transition.

For short-circuit faults, the conductance of the short-circuit path also changes according to the sech(Δt / τ) function from a small value (representing an open-circuit path) to a large value.

The Mutual Inductor block lets you select whether the faults occur in the primary or secondary winding. The block models the faulted winding as a faulted inductor. The unfaulted winding is coupled to the faulted winding. As a result, the actual equations involve a total of three coupled windings: two for the faulted winding and one for the unfaulted winding. The coupling between the primary and secondary windings is defined by the **Coefficient of coupling** parameter.

The block can trigger the start of fault transition:

1-321

- At a specific time
- After voltage exceeds the maximum permissible value a certain number of times
- When current exceeds the maximum permissible value for longer than a specific time interval

You can enable or disable these trigger mechanisms separately, or use them together if more than one trigger mechanism is required in a simulation. When more than one mechanism is enabled, the first mechanism to trigger the fault transition takes precedence. In other words, a component fails no more than once per simulation.

You can also choose whether to issue an assertion when a fault occurs by using the **Reporting when a fault occurs** parameter. The assertion can take the form of a warning or an error. By default, the block does not issue an assertion.

Faultable inductors often require that you use the fixed-step local solver, rather than the variable-step solver. In particular, if you model transitions to a faulted state that include short circuits, MathWorks recommends that you use the fixed-step local solver. For more information, see "Making Optimal Solver Choices for Physical Simulation" (Simscape).

Variables

Use the **Variables** section of the block interface to set the priority and initial target values for the block variables prior to simulation. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

The **Primary current** and **Secondary current** variables let you specify a high-priority target for the initial inductor current in the respective winding at the start of simulation.

Ports

Conserving

1+ — Positive terminal of the primary winding
electrical

Electrical conserving port associated with the primary winding positive terminal.

1- — Negative terminal of the primary winding
electrical

Electrical conserving port associated with the primary winding negative terminal.

2+ — Positive terminal of the secondary winding
electrical

Electrical conserving port associated with the secondary winding positive terminal.

2- — Negative terminal of the secondary winding
electrical

Electrical conserving port associated with the secondary winding negative terminal.

Parameters

Main

Inductance L1 — Nominal inductance value in the primary winding
10 H (default)

The nominal inductance value in the primary winding. Inductance value must be greater than zero.

Inductance L2 — Nominal inductance value in the secondary winding
0.1 H (default)

The nominal inductance value in the secondary winding. Inductance value must be greater than zero.

Coefficient of coupling — Mutual inductance coupling between windings
0.9 (default)

The coupling between the primary and secondary windings. This coefficient defines the mutual inductance. To reverse one of the winding directions, use a negative value.

Tolerance (%) — Inductor tolerance, in percent
20 (default)

The inductor tolerance as defined on the manufacturer datasheet. Datasheets typically provide a tolerance percentage for a given inductor type. Therefore, this value is the same for both windings.

L1 tolerance application — Select how to apply tolerance to primary winding
None — use nominal value (default) | Random tolerance | Apply maximum tolerance value | Apply minimum tolerance value

Select how to apply tolerance during simulation to the primary winding:

- None — use nominal value — The block does not apply tolerance, it uses the nominal inductance value.
- Random tolerance — The block applies random offset to the inductance value, within the tolerance value limit. You can choose Uniform or Gaussian distribution for calculating the random number by using the **Tolerance distribution** parameter.
- Apply maximum tolerance value — The inductance is increased by the specified tolerance percent value.
- Apply minimum tolerance value — The inductance is decreased by the specified tolerance percent value.

L1 tolerance distribution — Select the distribution type for primary winding
Uniform (default) | Gaussian

Select the distribution type for random tolerance:

- Uniform — Uniform distribution
- Gaussian — Gaussian distribution

Dependencies

Enabled when the **L1 tolerance application** parameter is set to Random tolerance.

L1 number of standard deviations for quoted tolerance — Used for calculating the Gaussian random number for primary winding
4 (default)

Number of standard deviations for calculating the Gaussian random number.

Dependencies

Enabled when the **L1 tolerance distribution** parameter is set to Gaussian.

L2 tolerance application — Select how to apply tolerance to secondary winding
None — use nominal value (default) | Random tolerance | Apply maximum tolerance value | Apply minimum tolerance value

Select how to apply tolerance during simulation to the secondary winding:

- `None — use nominal value` — The block does not apply tolerance, it uses the nominal inductance value.
- `Random tolerance` — The block applies random offset to the inductance value, within the tolerance value limit. You can choose Uniform or Gaussian distribution for calculating the random number by using the **Tolerance distribution** parameter.
- `Apply maximum tolerance value` — The inductance is increased by the specified tolerance percent value.
- `Apply minimum tolerance value` — The inductance is decreased by the specified tolerance percent value.

L2 tolerance distribution — Select the distribution type for secondary winding
`Uniform` (default) | `Gaussian`

Select the distribution type for random tolerance:

- `Uniform` — Uniform distribution
- `Gaussian` — Gaussian distribution

Dependencies

Enabled when the **L2 tolerance application** parameter is set to `Random tolerance`.

L2 number of standard deviations for quoted tolerance — Used for calculating the Gaussian random number for secondary winding
4 (default)

Number of standard deviations for calculating the Gaussian random number.

Dependencies

Enabled when the **L2 tolerance distribution** parameter is set to `Gaussian`.

Resistance

Series resistance, [R_primary R_secondary] — Equivalent series resistance of the primary and secondary winding
[0.001, 0.001] Ω (default)

Equivalent series resistance of the primary and secondary winding, specified as a two-element vector. The first number corresponds to the primary winding, the second number to the secondary winding. For a faulted winding, the block allocates the resistance to each segment in proportion to the number of turns in that segment.

Parallel conductance, [G_primary G_secondary] — Parallel leakage path associated with the primary and secondary winding
[1e-9,1e-9] 1/Ω (default)

Parallel leakage path associated with the primary and secondary winding, specified as a two-element vector. The first number corresponds to the primary winding, the second number to the secondary winding. The parallel conductances are placed directly across the + and – terminals of the primary and secondary winding, respectively.

Operating Limits

Enable operating limits — Select Yes to enable reporting when the operational limits are exceeded
No (default) | Yes

Select Yes to enable reporting when the operational limits are exceeded. The associated parameters in the **Operating Limits** section become visible to let you select the reporting method and specify the operating limits in terms of power and current.

Reporting when operating limits exceeded — Select the reporting method
Warn (default) | Error

Select what happens when an operating limit is exceeded:

- Warn — The block issues a warning.
- Error — Simulation stops with an error.

Dependencies

Enabled when the **Enable operating limits** parameter is set to Yes.

Saturation current — Inductor saturation current
1 A (default)

Inductor saturation current, as defined in the manufacturer datasheets. If the net current into the primary and secondary windings exceeds this value, the core material enters

saturation, and the block reports an operating limits violation. That is, the block compares the limit against $|i_1 + i_2|$, where currents are defined as being positive when they are into the + nodes.

Dependencies

Enabled when the **Enable operating limits** parameter is set to `Yes`.

Power rating — Maximum power dissipation in the inductor
1 W (default)

Maximum instantaneous (total) power dissipation in the resistance and conductance elements associated with the mutual inductor. If the total power (including both primary and secondary winding) exceeds this number, the block reports an operating limits violation.

Dependencies

Enabled when the **Enable operating limits** parameter is set to `Yes`.

Faults

Enable faults — Select Yes to enable faults modeling
No (default) | Yes

Select `Yes` to enable faults modeling. The associated parameters in the **Faults** section become visible to let you select the reporting method and specify the trigger mechanism (temporal or behavioral). You can enable these trigger mechanisms separately or use them together.

Reporting when a fault occurs — Choose whether to issue an assertion when a fault occurs
None (default) | Warn | Error

Choose whether to issue an assertion when a fault occurs:

- `None` — The block does not issue an assertion.
- `Warn` — The block issues a warning.
- `Error` — Simulation stops with an error.

Dependencies

Enabled when the **Enable faults** parameter is set to `Yes`.

1-327

Faulted winding — Select winding to use for fault modeling
Primary (default) | Secondary

Select whether the faults occur in the primary or secondary winding.

Dependencies

Enabled when the **Enable faults** parameter is set to Yes.

Location of fault node (% of total turns from - terminal) — Percentage of turns in the subinductor that is in contact with the – port of the faulted winding
50 (default)

In practice, faults are enabled by segmenting the faulted winding into two coupled subinductors, connected in a series. The inductance is proportional to the square of the number of turns in the respective segment, and the series resistance of each subinductor is proportional to the number of turns in each segment. The parallel conductance spans both segments.

This parameter indicates the percentage of turns that are assigned to the subinductor that is in contact with the – port of the faulted winding. The remaining turns are assigned to the other subinductor. The default value is 50, which means that the overall inductance of the faulted winding is divided into two equal, coupled subinductors.

Dependencies

Enabled when the **Enable faults** parameter is set to Yes.

Short-circuit turns — Select whether fault results in one of the segments being short-circuited
No (default) | To negative terminal | To positive terminal

Select whether the fault results in one of the subinductor segments being short-circuited:

- No — The fault does not produce a short circuit.

- To negative terminal — The fault short-circuits the subinductor that is in contact with the – port of the block.

- To positive terminal — The fault short-circuits the subinductor that is in contact with the + port of the block.

Dependencies

Enabled when the **Enable faults** parameter is set to Yes.

`Open-circuit at fault node` — Select whether to apply an open-circuit fault between the segments
No (default) | Yes

Select whether to apply an open-circuit fault between the two subinductor segments. The default is No. Even with an open-circuit fault, the characteristics of the subinductors are still related because they are magnetically coupled even in the faulted state.

Dependencies

Enabled when the **Enable faults** parameter is set to Yes.

`Ground fault` — Select whether fault results in one of the segments being short-circuited
No (default) | Negative terminal side of fault node | Positive terminal side of fault node

Select whether, in case of fault, there is a path for current to flow towards the ground node:

- No — The fault does not result in a connection to ground.
- Negative terminal side of fault node — The side that is in contact with the – port of the block is connected to ground.
- Positive terminal side of fault node — The side that is in contact with the + port of the block is connected to ground.

If the **Open-circuit at fault node** parameter is set to Yes, you need to specify which side (negative or positive) is connected to ground. If there is no open circuit, the two options behave similarly. Physically, this corresponds to a breakdown in the insulation between the windings and the grounded core or chassis.

Dependencies

Enabled when the **Enable faults** parameter is set to Yes.

`Conductance of faulted ground path` — Mutual coupling between the two subinductors
0 1/Ω (default)

If there is a ground fault, this parameter represents conductance of the current path to ground. For example, if the path to ground is through the core material, then specify a small conductance value depending on the core material being used. For highly conductive core material or for chassis-shorts, specify a higher conductance value.

Dependencies

Enabled when the **Ground fault** parameter is set to `Negative terminal side of fault node` or `Positive terminal side of fault node`.

Fault transition time constant — Time constant for the transition to faulted state
1 ms (default)

Time constant associated with the transition to the faulted state, as described in "Faults" on page 1-321.

Dependencies

Enabled when the **Enable faults** parameter is set to `Yes`.

Enable temporal fault trigger — Select Yes to enable time-based fault triggering
No (default) | Yes

Select `Yes` to enable time-based fault triggering. You can enable the temporal and behavioral trigger mechanisms separately or use them together.

Dependencies

Enabled when the **Enable faults** parameter is set to `Yes`.

Simulation time for a fault event — Time before entering faulted state
1 s (default)

Set the simulation time at which you want the block to start entering the fault state.

Dependencies

Enabled when the **Enable temporal fault trigger** parameter is set to `Yes`.

Enable behavioral fault trigger — Select Yes to enable behavioral fault triggering
No (default) | Yes

Select Yes to enable behavioral fault triggering. You can enable the temporal and behavioral trigger mechanisms separately or use them together.

Dependencies

Enabled when the **Enable faults** parameter is set to Yes.

Maximum permissible voltage — Voltage threshold to fault transition
100 V (default)

Define the voltage threshold to a fault transition. If the voltage value exceeds this threshold a certain number of times, specified by the **Number of events to fail when exceeding voltage** parameter value, then the block starts entering the fault state.

Dependencies

Enabled when the **Enable behavioral fault trigger** parameter is set to Yes.

Number of events to fail when exceeding voltage — Maximum number of times the voltage exceeds the threshold
1 (default)

Because the physical mechanism underlying voltage-based failures depends on one or more partial discharge events occurring, this parameter allows you to set the number of voltage overshoots that the inductor can withstand before the fault transition begins. Note that the block does not check the time spent in the overvoltage condition, only the number of transitions.

Dependencies

Enabled when the **Enable behavioral fault trigger** parameter is set to Yes.

Maximum permissible current — Current threshold to fault transition
1 A (default)

Define the current threshold to a fault transition. If the current value exceeds this threshold for longer than the **Time to fail when exceeding current** parameter value, then the block starts entering the fault state.

Dependencies

Enabled when the **Enable behavioral fault trigger** parameter is set to Yes.

1-331

`Time to fail when exceeding current` — Maximum length of time the current exceeds the threshold
1 s (default)

Set the maximum length of time that the current can exceed the maximum permissible value without triggering the fault.

Dependencies

Enabled when the **Enable behavioral fault trigger** parameter is set to `Yes`.

See Also

Fault | Inductor

Introduced in R2017a

N-Channel IGBT

Model N-Channel IGBT

Library

Semiconductor Devices

Description

The N-Channel IGBT block models an Insulated Gate Bipolar Transistor (IGBT). The block provides two main modeling variants, accessible by right-clicking the block in your block diagram and then selecting the appropriate option from the context menu, under **Simscape > Block choices**:

- **Full I-V and capacitance characteristics** — This variant is a detailed component model suitable for simulating detailed switching characteristics and predicting component losses. This variant, in turn, provides two ways of modeling an IGBT:

  - As an equivalent circuit based on a PNP bipolar transistor and N-channel MOSFET. For more information on using this model, see "Representation by Equivalent Circuit" on page 1-334, "Fine-Tuning the Current-Voltage Characteristics" on page 1-342, and "Modeling Temperature Dependence" on page 1-342.

  - By a 2-D lookup table approximation to the I-V (current-voltage) curve. For details, see "Representation by 2-D Lookup Table" on page 1-337.

  - By a 3-D lookup table approximation to the I-V (current-voltage) curve that includes temperature data. For details, see "Representation by 3-D Lookup Table" on page 1-337.

The gate junction capacitance in the detailed model is represented as a fixed gate-emitter capacitance C_{GE} and either a fixed or a nonlinear gate-collector capacitance C_{GC}. For details, see "Charge Model" on page 1-338.

- **Simplified I-V characteristics and event-based timing** — This variant models the IGBT more simply by using just the on-state I-V data corresponding to the gate voltage used in your circuit. Switching between states is achieved by linearly ramping the collector-emitter voltage. This simplified model is suitable when approximate dynamic characteristics are sufficient, and simulation speed is of paramount importance. For details, see "Event-Based IGBT Variant" on page 1-343.

Together with the thermal port variants (see "Thermal Port" on page 1-344), the block therefore provides you with four choices. To select the desired variant, right-click the block in your model. From the context menu, select **Simscape > Block choices**, and then one of the following options:

- **Full I-V and capacitance characteristics | No thermal port** — Detailed model that does not simulate the effects of generated heat and device temperature. This is the default.
- **Full I-V and capacitance characteristics | Show thermal port** — Detailed model with exposed thermal port.
- **Simplified I-V characteristics and event-based timing| No thermal port** — Simplified event-based model, which also does not simulate the effects of generated heat and device temperature.
- **Simplified I-V characteristics and event-based timing | Show thermal port** — Simplified event-based model with exposed thermal port.

Representation by Equivalent Circuit

The equivalent circuit of the detailed block variant consists of a PNP Bipolar Transistor block driven by an N-Channel MOSFET block, as shown in the following figure:
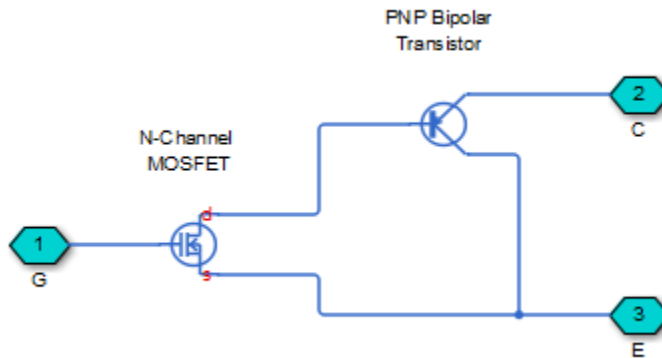
The MOSFET source is connected to the bipolar transistor collector, and the MOSFET drain is connected to the bipolar transistor base. The MOSFET uses the equations shown in the N-Channel MOSFET block reference page. The bipolar transistor uses the equations shown in the PNP Bipolar Transistor block reference page, but with the addition of an emission coefficient parameter N that scales kT/q.

The N-Channel IGBT block uses the on and off characteristics you specify in the block dialog box to estimate the parameter values for the underlying N-Channel MOSFET and PNP bipolar transistor.

The block uses the off characteristics to calculate the base-emitter voltage, V_{be}, and the saturation current, I_S.

When the transistor is off, the gate-emitter voltage is zero and the IGBT base-collector voltage is large, so the PNP base and collector current equations simplify to:

$$I_b = 0 = -I_s \left[\frac{1}{\beta_F} (e^{-qV_{be}/(NkT)} - 1) - \frac{1}{\beta_R} \right]$$

$$I_c = -I_s \left[e^{-qV_{be}/(NkT)} \left(1 + \frac{V_{bc}}{V_{AF}} \right) + \frac{1}{\beta_R} \right]$$

where N is the **Emission coefficient, N** parameter value, V_{AF} is the forward Early voltage, and I_c and I_b are defined as positive flowing into the collector and base, respectively. See the PNP Bipolar Transistor reference page for definitions of the remaining variables. The first equation can be solved for V_{be}.

The base current is zero in the off-condition, and hence $I_c = -I_{ces}$, where I_{ces} is the Zero gate voltage collector current. The base-collector voltage, V_{bc}, is given by $V_{bc} = V_{ces} + V_{ces}$, where V_{ces} is the voltage at which I_{ces} is measured. Hence we can rewrite the second equation as follows:

$$I_{ces} = I_s \left[e^{-qV_{be}/(NkT)} \left(1 + \frac{V_{ces} + V_{be}}{V_{AF}} \right) + \frac{1}{\beta_R} \right]$$

The block sets β_R and β_F to typical values of 1 and 50, so these two equations can be used to solve for V_{be} and I_S:

$$V_{be} = \frac{-NkT}{q} \log \left(1 + \frac{\beta_F}{\beta_R} \right)$$

$$I_s = \frac{I_c}{e^{-qV_{be}/(NkT)} + \frac{1}{\beta_R}}$$

Note The block does not require an exact value for β_F because it can adjust the MOSFET gain K to ensure the overall device gain is correct.

The block parameters **Collector-emitter saturation voltage, Vce(sat)** and **Collector current at which Vce(sat) is defined** are used to determine $V_{be(sat)}$ by solving the following equation:

$$I_{ce(sat)} = I_s \left[e^{-qV_{be(sat)}/(NkT)} \left(1 + \frac{V_{ce(sat)} + V_{be(sat)}}{V_{AF}} \right) + \frac{1}{\beta_R} \right]$$

Given this value, the block calculates the MOSFET gain, K, using the following equation:

$$I_{ds} = I_b = K \left[(V_{GE(sat)} - V_{th})V_{ds} - \frac{V_{ds}^2}{2} \right]$$

where V_{th} is the **Gate-emitter threshold voltage, Vge(th)** parameter value and $V_{GE(sat)}$ is the **Gate-emitter voltage at which Vce(sat) is defined** parameter value.

V_{ds} is related to the transistor voltages as $V_{ds} = V_{ce} - V_{be}$. The block substitutes this relationship for V_{ds}, sets the base-emitter voltage and base current to their saturated values, and rearranges the MOSFET equation to give

$$K = \frac{I_{b(sat)}}{\left[\left(V_{GE(sat)} - V_{th}\right)\left(V_{be(sat)} + V_{ce(sat)}\right) - \frac{\left(V_{be(sat)} + V_{ce(sat)}\right)^2}{2} \right]}$$

where $V_{ce(sat)}$ is the **Collector-emitter saturation voltage, Vce(sat)** parameter value.

These calculations ensure the zero gate voltage collector current and collector-emitter saturation voltage are exactly met at these two specified conditions. However, the current-voltage plots are very sensitive to the emission coefficient N and the precise value of V_{th}. If the manufacturer datasheet gives current-voltage plots for different V_{GE} values, then the N and V_{th} can be tuned by hand to improve the match.

Representation by 2-D Lookup Table

For the lookup table representation of the detailed block variant, you provide tabulated values for collector current as a function of gate-emitter voltage and collector-emitter voltage. The main advantage of using this option is simulation speed. It also lets you parameterize the device from either measured data or from data obtained from another simulation environment. To generate your own data from the equivalent circuit representation, you can use a test harness, such as shown in the IGBT Characteristics example.

The lookup table representation combines all of the equivalent circuit components (PNP transistor, N-channel MOSFET, collector resistor and emitter resistor) into one equivalent lookup table. Therefore, if you use this option, the **Advanced** tab has no parameters.

Representation by 3-D Lookup Table

For the temperature-dependent lookup table representation of the detailed block variant, you provide tabulated values for collector current as a function of gate-emitter voltage, collector-emitter voltage, and temperature.

The lookup table representation combines all of the equivalent circuit components (PNP transistor, N-channel MOSFET, collector resistor and emitter resistor) into one equivalent lookup table. Therefore, if you use this option, the **Advanced** tab has no parameters.

If the block thermal port is not exposed, then the **Device simulation temperature** parameter on the **Temperature Dependence** tab lets you specify the simulation temperature. If the block has an exposed thermal port, the **Temperature Dependence** tab is empty because the temperature variable T on the thermal port sets the device simulation temperature.

Charge Model

The detailed variant of the block models junction capacitances either by fixed capacitance values, or by tabulated values as a function of the collector-emitter voltage. In either case, you can either directly specify the gate-emitter and gate-collector junction capacitance values, or let the block derive them from the input and reverse transfer capacitance values. Therefore, the **Parameterization** options for charge model on the **Junction Capacitance** tab are:

- `Specify fixed input, reverse transfer and output capacitance` — Provide fixed parameter values from datasheet and let the block convert the input and reverse transfer capacitance values to junction capacitance values, as described below. This is the default method.

- `Specify fixed gate-emitter, gate-collector and collector-emitter capacitance` — Provide fixed values for junction capacitance parameters directly.

- `Specify tabulated input, reverse transfer and output capacitance` — Provide tabulated capacitance and collector-emitter voltage values based on datasheet plots. The block converts the input and reverse transfer capacitance values to junction capacitance values, as described below.

- `Specify tabulated gate-emitter, gate-collector and collector-emitter capacitance` — Provide tabulated values for junction capacitances and collector-emitter voltage.

Use one of the tabulated capacitance options (`Specify tabulated input, reverse transfer and output capacitance` or `Specify tabulated gate-emitter, gate-collector and collector-emitter capacitance`) when the datasheet provides a plot of junction capacitances as a function of collector-emitter voltage. Using tabulated capacitance values will give more accurate dynamic characteristics, and avoids the need to iteratively tune parameters to fit the dynamics.

If you use the `Specify fixed gate-emitter, gate-collector and collector-emitter capacitance` or `Specify tabulated gate-emitter, gate-collector and collector-emitter capacitance` option, the **Junction Capacitance** tab lets you specify the **Gate-emitter junction capacitance**, **Gate-collector junction capacitance**, and **Collector-emitter junction capacitance** parameter values (fixed or tabulated) directly. Otherwise, the block derives them from the **Input capacitance, Cies**, **Reverse transfer capacitance, Cres**, and **Output capacitance, Coes** parameter values. These two parameterization methods are related as follows:

- $C_{GE} = Cres$
- $C_{GC} = Cies - Cres$
- $C_{CE} = Coes - Cres$

The two fixed capacitance options (`Specify fixed input, reverse transfer and output capacitance` or `Specify fixed gate-emitter, gate-collector and collector-emitter capacitance`) let you model gate junction capacitance as a fixed gate-emitter capacitance C_{GE} and either a fixed or a nonlinear gate-collector capacitance C_{GC}. If you select the `Gate-collector charge function is nonlinear` option for the **Charge-voltage linearity** parameter, then the gate-collector charge relationship is defined by the piecewise-linear function shown in the following figure.

Nonlinear gate-collector charge function
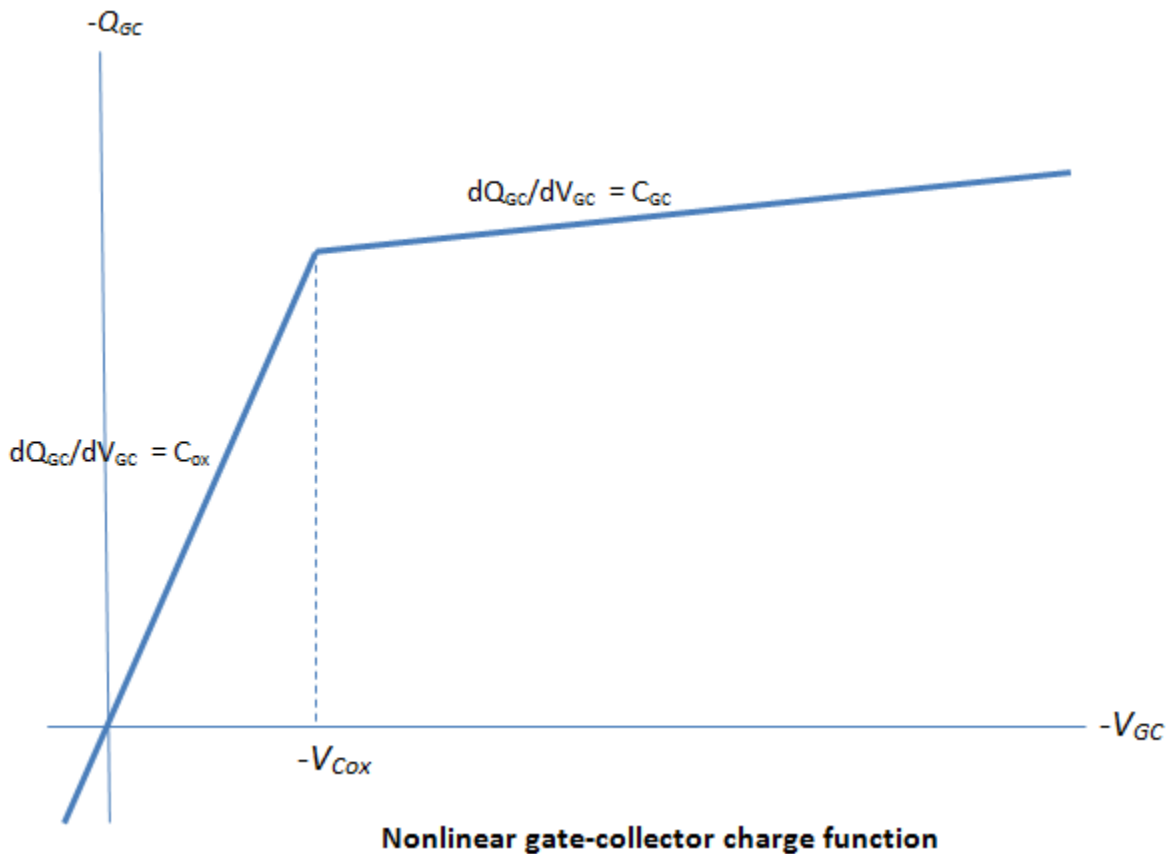
With this nonlinear capacitance, the gate-emitter and collector-emitter voltage profiles take the form shown in the next figure, where the collector-emitter voltage fall has two regions (labeled 2 and 3) and the gate-emitter voltage has two time-constants (before and after the threshold voltage V_{th}):
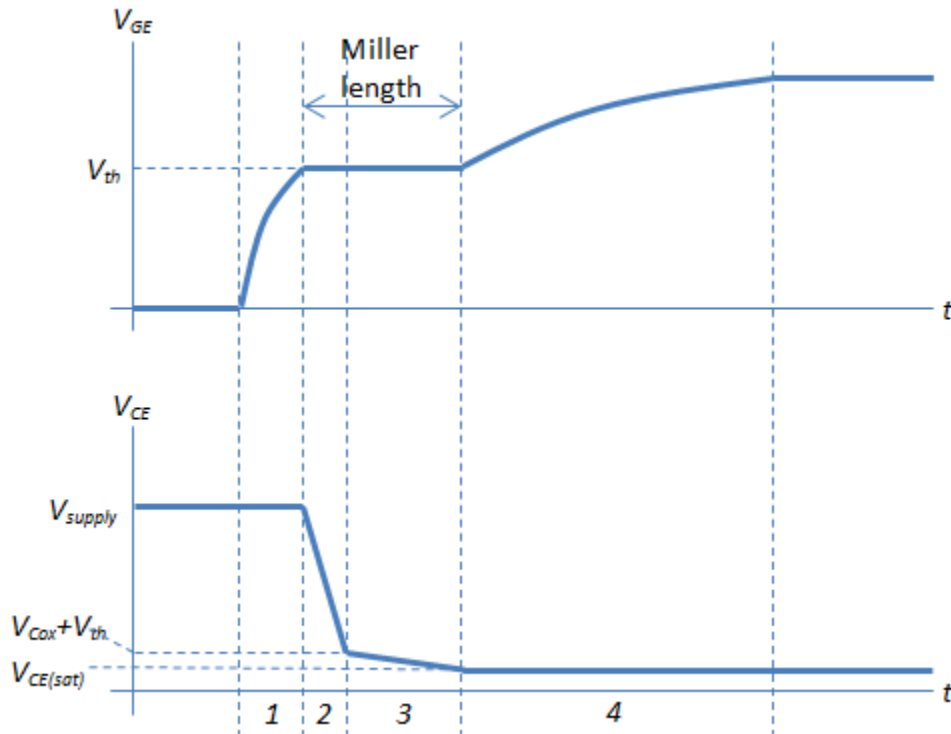
You can determine the capacitor values for *Cies*, *Cres*, and C_{ox} as follows, assuming that the IGBT gate is driven through an external resistance R_G:

1 Set *Cies* to get correct time-constant for V_{GE} in Region 1. The time constant is defined by the product of *Cies* and R_G. Alternatively, you can use a datasheet value for *Cies*.

2 Set *Cres* so as to achieve the correct V_{CE} gradient in Region 2. The gradient is given by $(V_{GE} - V_{th})/(Cres \cdot R_G)$.

3 Set V_{Cox} to the voltage at which the V_{CE} gradient changes minus the threshold voltage *V*th.

4 Set C_{ox} to get correct Miller length and time constant in Region 4.

Because the underlying model is a simplification of an actual charge distribution, some iteration of these four steps may be required to get a best overall fit to measured data.

The collector current tail when the IGBT is turned off is determined by the **Total forward transit time** parameter.

Note Because this block implementation includes a charge model, you must model the impedance of the circuit driving the gate to obtain representative turn-on and turn-off dynamics. Therefore, if you are simplifying the gate drive circuit by representing it as a controlled voltage source, you must include a suitable series resistor between the voltage source and the gate.

Fine-Tuning the Current-Voltage Characteristics

For the equivalent circuit representation of the detailed model, use the parameters on the **Advanced** tab to fine-tune the current-voltage characteristics of the modeled device. To use these additional parameters effectively, you will need a manufacturer datasheet that provides plots of the collector current versus collector-emitter voltage for different values of gate-emitter voltage. The parameters on the **Advanced** tab have the following effects:

- The **Emission coefficient, N** parameter controls the shape of the current-voltage curves around the origin.

- The **Collector resistance, RC** and **Emitter resistance, RE** parameters affect the slope of the current-voltage curve at higher currents, and when fully turned on by a high gate-emitter voltage.

- The **Forward Early voltage, VAF** parameter affects the shape of the current-voltage curves for gate-emitter voltages around the **Gate-emitter threshold voltage, Vge(th)**.

Modeling Temperature Dependence

For the 2-D lookup table representation, the electrical equations do not depend on temperature. However, you can model temperature dependence by either using the 3-D lookup table representation, or by using the equivalent circuit representation of the detailed model.

For the equivalent circuit representation, temperature dependence is modeled by the temperature dependence of the constituent components. See the N-Channel MOSFET and PNP Bipolar Transistor block reference pages for further information on the defining equations.

Some datasheets do not provide information on the zero gate voltage collector current, *Ices*, at a higher measurement temperature. In this case, you can alternatively specify the energy gap, *EG*, for the device, using a typical value for the semiconductor type. For silicon, the energy gap is usually 1.11 eV.

Event-Based IGBT Variant

This implementation has much simpler equations than that with full I-V and capacitance characteristics. Use the event-based variant when the focus of the analysis is to understand overall circuit behavior rather than to verify the precise IGBT timing or losses characteristics.

The device is always in one of the following four states:

- Off
- Turning on
- On
- Turning off

In the off state, the relationship between collector current (i_c) and collector-emitter voltage (v_{ce}) is

$$i_c = G_{off}v_{ce}$$

In the on state, the relationship between collector current (i_c) and collector-emitter voltage (v_{ce}) is

$$v_{ce} = \text{tablelookup}(i_c)$$

When turning on, the collector-emitter voltage is ramped down to zero over the rise time, the device moving into the on state when the voltage falls below the tabulated on-state value. Similarly when turning off, the collector-emitter voltage is ramped up over the (current) fall time to the specified blocking voltage value.

The following figure shows the resulting voltage and current profiles when driving a resistive load.
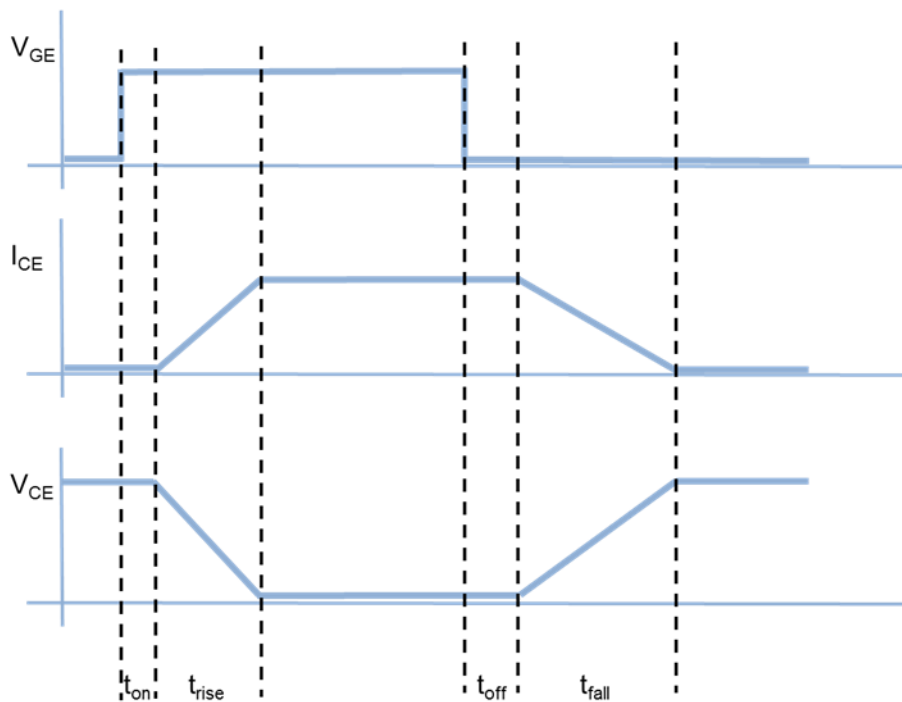
Thermal Port

The block has an optional thermal port, hidden by default. To expose the thermal port, right-click the block in your model, and select the appropriate block variant:

- For a detailed model, select **Simscape** > **Block choices** > **Full I-V and capacitance characteristics | Show thermal port**. This action displays the thermal port H on the block icon, and adds the **Thermal Port** tab to the block dialog box.

- For a simplified event-based model, select **Simscape** > **Block choices** > **Simplified I-V characteristics and event-based timing | Show thermal port**. This action displays the thermal port H on the block icon, adds the **Thermal Port** tab to the block dialog box, and displays additional parameters on the **Main** tab. To simulate thermal effects, you must provide additional tabulated data for turn-on and turn-off losses and define the collector-emitter on-state voltage as a function of both current and temperature.

Use the thermal port to simulate the effects of generated heat and device temperature. For more information on using thermal ports and on the **Thermal Port** tab parameters, see "Simulating Thermal Effects in Semiconductors".

Basic Assumptions and Limitations

The detailed model is based on the following assumptions:

- This block does not allow you to specify initial conditions on the junction capacitances. If you select the **Start simulation from steady state** option in the Solver Configuration block, the block solves the initial voltages to be consistent with the calculated steady state. Otherwise, voltages are zero at the start of the simulation.

- You may need to use nonzero junction capacitance values to prevent numerical simulation issues, but the simulation may run faster with these values set to zero.

- The block does not account for temperature-dependent effects on the junction capacitances.

The simplified, event-based model is based on the following assumptions:

- When you use a pair of IGBTs in a bridge arm, normally the gate drive circuitry will prevent a device turning on until the corresponding device has turned off, thereby implementing a minimum dead band. If you need to simulate the case where there is no minimum dead band and both devices are momentarily partially on, use the detailed IGBT model variant (**Full I-V and capacitance characteristics**). The assumption used by the event-based variant that the collector-emitter voltages can be ramped between on and off states is not valid for such cases.

- A minimum pulse width is applied when turning on or off; at the point where the gate-collector voltage rises above the threshold, any subsequent gate voltage changes are ignored for a time equal to the sum of the turn-on delay and current rise time. Similarly at the point where the gate collector voltage falls below the threshold, any subsequent gate voltage changes are ignored for a time equal to the sum of the turn-off delay and current fall time. This feature is normally implemented in the gate drive circuitry.

- This model does not account for charge. Hence there is no current tail when turning off an inductive load.

- Representative modeling of the current spike during turn-on of an inductive load with preexisting freewheeling current requires tuning of the **Miller resistance** parameter.

- The tabulated turn-on switching loss uses the previous on-state current, not the current value (which is not known until the device reaches the final on state).
- Due to high model stiffness that can arise from the simplified equations, you may get minimum step size violation warnings when using this block. Open the Solver pane of the Configuration Parameters dialog box and increase the **Number of consecutive min steps** parameter value as necessary to remove these warnings.

Parameters

Main Tab (Default Block Variant)

This configuration of the **Main** tab corresponds to the detailed block variant, which is the default. If you are using the simplified, event-based variant of the block, see "Main Tab (Event-Based Block Variant)" on page 1-354.

I-V characteristics defined by

Select the IGBT representation:

- `Fundamental nonlinear equations` — Use an equivalent circuit based on a PNP bipolar transistor and N-channel MOSFET. This is the default.
- `Lookup table (2-D, temperature independent)` — Use 2-D table lookup for collector current as a function of gate-emitter voltage and collector-emitter voltage.
- `Lookup table (3-D, temperature dependent)` — Use 3-D table lookup for collector current as a function of gate-emitter voltage, collector-emitter voltage, and temperature.

Zero gate voltage collector current, Ices

The collector current that flows when the gate-emitter voltage is set to zero, and a large collector-emitter voltage is applied, that is, the device is in the off-state. The

value of the large collector-emitter voltage is defined by the parameter **Voltage at which Ices is defined**. The default value is 2 mA. This parameter is visible only when you select `Fundamental nonlinear equations` for the **I-V characteristics defined by** parameter.

Voltage at which Ices is defined

The voltage used when measuring the **Zero gate voltage collector current, Ices**. The default value is `600` V. This parameter is visible only when you select `Fundamental nonlinear equations` for the **I-V characteristics defined by** parameter.

Gate-emitter threshold voltage, Vge(th)

The threshold voltage used in the MOSFET equations. The default value is `6` V. This parameter is visible only when you select `Fundamental nonlinear equations` for the **I-V characteristics defined by** parameter.

Collector-emitter saturation voltage, Vce(sat)

The collector-emitter voltage for a typical on-state as specified by the manufacturer. The default value is `2.8` V. This parameter is visible only when you select `Fundamental nonlinear equations` for the **I-V characteristics defined by** parameter.

Collector current at which Vce(sat) is defined

The collector-emitter current when the gate-emitter voltage is $V_{ge(sat)}$ and collector-emitter voltage is $V_{ce(sat)}$. The default value is `400` A. This parameter is visible only when you select `Fundamental nonlinear equations` for the **I-V characteristics defined by** parameter.

Gate-emitter voltage at which Vce(sat) is defined

The gate voltage used when measuring $V_{ce(sat)}$ and $I_{ce(sat)}$. The default value is `10` V. This parameter is visible only when you select `Fundamental nonlinear equations` for the **I-V characteristics defined by** parameter.

Measurement temperature

The temperature for which the parameters are quoted (T_{m1}). The default value is 25 °C. This parameter is visible only when you select `Fundamental nonlinear equations` for the **I-V characteristics defined by** parameter.

Vector of gate-emitter voltages, Vge

The vector of gate-emitter voltages, to be used for table lookup. The vector values must be strictly increasing. The values can be nonuniformly spaced. The default values, in V, are `[-2 6 7 8 10 12 15 20]`. This parameter is visible only when you select `Lookup table (2-D, temperature independent)` or `Lookup table`

(3-D, temperature dependent) for the **I-V characteristics defined by** parameter.

Vector of collector-emitter voltages, Vce

The vector of collector-emitter voltages, to be used for table lookup. The vector values must be strictly increasing. The values can be nonuniformly spaced. The default values, in V, are [-1 0 0.5 1 1.5 2 2.5 3 3.5 4]. This parameter is visible only when you select Lookup table (2-D, temperature independent) or Lookup table (3-D, temperature dependent) for the **I-V characteristics defined by** parameter.

Vector of temperatures, T

The vector of temperatures, to be used for table lookup. The vector values must be strictly increasing. The values can be nonuniformly spaced. The default values, in degC, are [25 125]. This parameter is visible only when you select Lookup table (3-D, temperature dependent) for the **I-V characteristics defined by** parameter.

Tabulated collector currents, Ic=fcn(Vge,Vce)

Tabulated values for collector current as a function of gate-emitter voltage and collector-emitter voltage, to be used for 2-D table lookup. Each value in the matrix specifies the collector current for a specific combination of gate-emitter voltage and collector-emitter voltage. The matrix size must match the dimensions defined by the gate-emitter voltage and collector-emitter voltage vectors. The default values, in A, are:

```
[-1.015e-5 1.35e-8 4.7135e-4 5.092e-4 5.105e-4 5.1175e-4 5.1299e-4 5.1423e-4 5.1548e-4 5.1672e-4;
 -9.9869e-6 1.35e-8 4.7135e-4 5.092e-4 5.105e-4 5.1175e-4 5.1299e-4 5.1423e-4 5.1548e-4 5.1672e-4;
 -9.955e-6 1.35e-8 0.0065225 3.3324 48.154 93.661 105.52 105.72 105.93 106.14;
 -9.955e-6 1.35e-8 0.0065235 3.5783 70.264 166.33 252.4 317.67 353.38 357.39;
 -9.955e-6 1.35e-8 0.006524 3.7206 89.171 228.09 371.63 511.02 642.69 764.04;
 -9.9549e-6 1.35e-8 0.0065242 3.7716 97.793 256.21 424.27 592.92 759.2 921.52;
 -9.9549e-6 1.35e-8 0.0065243 3.8067 104.52 278.11 464.6 654.37 844.57 1.0339e+3;
 -9.9549e-6 1.35e-8 0.0065244 3.8324 109.92 295.67 496.54 702.28 909.96 1.1183e+3]
```

This parameter is visible only when you select Lookup table (2-D, temperature independent) for the **I-V characteristics defined by** parameter.

Tabulated collector currents, Ic=fcn(Vge,Vce,T)

Tabulated values for collector current as a function of gate-emitter voltage, collector-emitter voltage, and temperature, to be used for 3-D table lookup. Each value in the matrix specifies the collector current for a specific combination of gate-emitter voltage and collector-emitter voltage at a specific temperature. The matrix size must match the dimensions defined by the gate-emitter voltage, collector-emitter voltage, and temperature vectors. The default value, in A, is zeros(8, 10, 2).

This parameter is visible only when you select `Lookup table (3-D, temperature dependent)` for the **I-V characteristics defined by** parameter.

Junction Capacitance Tab (Default Block Variant)

Parameterization

Select one of the following methods for block parameterization:

- `Specify fixed input, reverse transfer and output capacitance` — Provide fixed parameter values from datasheet and let the block convert the input, output, and reverse transfer capacitance values to junction capacitance values, as described in "Charge Model" on page 1-338. This is the default method.

- `Specify fixed gate-emitter, gate-collector and collector-emitter capacitance` — Provide fixed values for junction capacitance parameters directly.

- `Specify tabulated input, reverse transfer and output capacitance` — Provide tabulated capacitance and collector-emitter voltage values based on datasheet plots. The block converts the input, output, and reverse transfer capacitance values to junction capacitance values, as described in "Charge Model" on page 1-338.

- `Specify tabulated gate-emitter, gate-collector and collector-emitter capacitance` — Provide tabulated values for junction capacitances and collector-emitter voltage.

Input capacitance, Cies

The gate-emitter capacitance with the collector shorted to the source. This parameter is visible only for the following two values for the **Parameterization** parameter:

- If you select `Specify fixed input, reverse transfer and output capacitance`, the default value is `26.4` nF.

- If you select `Specify tabulated input, reverse transfer and output capacitance`, the default value is `[80 40 32 28 27.5 27 26.5 26.5 26.5]` nF.

Reverse transfer capacitance, Cres

The collector-gate capacitance with the emitter connected to ground. This parameter is visible only for the following two values for the **Parameterization** parameter:

- If you select `Specify fixed input, reverse transfer and output capacitance`, the default value is 2.7 nF.
- If you select `Specify tabulated input, reverse transfer and output capacitance`, the default value is [55 9 5.5 3.1 2.5 2.1 1.9 1.8 1.7] nF.

Output capacitance, Coes

The collector-gate capacitance with the gate and emitter shorted. This parameter is visible only for the following two values for the **Parameterization** parameter:

- If you select `Specify fixed input, reverse transfer and output capacitance`, the default value is 0 nF.
- If you select `Specify tabulated input, reverse transfer and output capacitance`, the default value is [60 20 12 8 6 4.8 4 3.5 3.1] nF.

Gate-emitter junction capacitance

The value of the capacitance placed between the gate and the emitter. This parameter is visible only for the following two values for the **Parameterization** parameter:

- If you select `Specify fixed gate-emitter, gate-collector and collector-emitter capacitance`, the default value is 23.7 nF.
- If you select `Specify tabulated gate-emitter, gate-collector and collector-emitter capacitance`, the default value is [25 31 26.5 24.9 25 24.9 24.6 24.7 24.8] nF.

Gate-collector junction capacitance

The value of the capacitance placed between the gate and the collector. This parameter is visible only for the following two values for the **Parameterization** parameter:

- If you select `Specify fixed gate-emitter, gate-collector and collector-emitter capacitance`, the default value is 2.7 nF.
- If you select `Specify tabulated gate-emitter, gate-collector and collector-emitter capacitance`, the default value is [55 9 5.5 3.1 2.5 2.1 1.9 1.8 1.7] nF.

Collector-emitter junction capacitance

The value of the capacitance placed between the collector and the emitter. This parameter is visible only for the following two values for the **Parameterization** parameter:

- If you select `Specify fixed gate-emitter, gate-collector and collector-emitter capacitance`, the default value is 0 nF.

- If you select `Specify tabulated gate-emitter, gate-collector and collector-emitter capacitance`, the default value is [5 11 6.5 4.9 3.5 2.7 2.1 1.7 1.4] nF.

Corresponding collector-emitter voltages

The collector-emitter voltages corresponding to the tabulated capacitance values. This parameter is visible only for tabulated capacitance models (`Specify tabulated input, reverse transfer and output capacitance` or `Specify tabulated gate-emitter, gate-collector and output capacitance`). The default value is [0 1 2 5 10 15 20 25 30] V.

Charge-voltage linearity

Select whether gate-drain capacitance is fixed or nonlinear:

- `Gate-collector capacitance is constant` — The capacitance value is constant and defined according to the selected parameterization option, either directly or derived from a datasheet. This is the default method.

- `Gate-collector charge function is nonlinear` — The gate-collector charge relationship is defined according to the piecewise-nonlinear function described in "Charge Model" on page 1-338. Two additional parameters appear to let you define the gate-collector charge function.

This parameter is visible only for fixed capacitance models (`Specify fixed input, reverse transfer and output capacitance` or `Specify fixed gate-emitter, gate-collector and output capacitance`).

Gate-collector oxide capacitance

The gate-collector capacitance when the device is on and the collector-gate voltage is small. This parameter is visible only when you select `Gate-collector charge function is nonlinear` for the **Charge-voltage linearity** parameter. The default value is 20 nF.

Collector-gate voltage below which oxide capacitance becomes active

The collector-gate voltage at which the collector-gate capacitance switches between off-state (C_{GC}) and on-state (C_{ox}) capacitance values. This parameter is visible only when you select `Gate-collector charge function is nonlinear` for the **Charge-voltage linearity** parameter. The default value is -5 V.

1-351

Total forward transit time

> The forward transit time for the PNP transistor used as part of the underlying IGBT model. It affects how quickly charge is removed from the channel when the IGBT is turned off. The default value is 0 μs.

Advanced Tab (Default Block Variant)

The lookup table representation combines all the equivalent circuit components into one lookup table, and therefore this tab is empty. If you use the equivalent circuit representation, this tab has the following parameters.

Emission coefficient, N

> The emission coefficient or ideality factor of the bipolar transistor. The default value is 1.

Forward Early voltage, VAF

> The forward Early voltage for the PNP transistor used in the IGBT model. See the PNP Bipolar Transistor block reference page for more information. The default value is 200 V.

Collector resistance, RC

> Resistance at the collector. The default value is 0.001 Ohm.

Emitter resistance, RE

> Resistance at the emitter. The default value is 0.001 Ohm.

Internal gate resistance, RG

> The value of the internal gate resistor at the measurement temperature. Note that this is not the value of the external circuit series gate resistance, which you should model externally to the IGBT. The default value is 0.001 Ohm.

Forward current transfer ratio, BF

> Ideal maximum forward current gain for the PNP transistor used in the IGBT model. See the PNP Bipolar Transistor block reference page for more information. The default value is 50.

Temperature Dependence Tab (Default Block Variant)

For the 2-D lookup table representation, the electrical equations do not depend on temperature and therefore this tab is empty. For the 3-D lookup table representation with exposed thermal port, this tab is also empty because the 3-D matrix on the **Main** tab

captures the temperature dependence. If the block thermal port is not exposed for the 3-D lookup table representation, then this tab contains only the **Device simulation temperature** parameter. If you use the equivalent circuit representation, this tab has the following parameters.

Parameterization

Select one of the following methods for temperature dependence parameterization:

- `None — Simulate at parameter measurement temperature` — Temperature dependence is not modeled, and none of the other parameters on this tab are visible. This is the default method.

- `Specify Ices and Vce(sat) at second measurement temperature` — Model temperature-dependent effects by providing values for the zero gate voltage collector current, *Ices*, and collector-emitter voltage, $V_{ce(sat)}$, at the second measurement temperature.

- `Specify Vce(sat) at second measurement temperature plus the energy gap, EG` — Use this option when the datasheet does not provide information on the zero gate voltage collector current, *Ices*, at a higher measurement temperature.

Energy gap, EG

Energy gap value. This parameter is visible only when you select `Specify Vce(sat) at second measurement temperature plus the energy gap, EG` for the **Parameterization** parameter. The default value is `1.11` eV.

Zero gate voltage collector current, Ices, at second measurement temperature

The zero gate collector current value at the second measurement temperature. This parameter is visible only when you select `Specify Ices and Vce(sat) at second measurement temperature` for the **Parameterization** parameter. The default value is `100` mA.

Collector-emitter saturation voltage, Vce(sat), at second measurement temperature

The collector-emitter saturation voltage value at the second measurement temperature, and when the collector current and gate-emitter voltage are as defined by the corresponding parameters on the **Main** tab. The default value is `3` V.

Second measurement temperature

Second temperature T_{m2} at which **Zero gate voltage collector current, Ices, at second measurement temperature** and **Collector-emitter saturation voltage,**

Vce(sat), at second measurement temperature are measured. The default value is 125 °C.

Saturation current temperature exponent, XTI

The saturation current exponent value for your device type. If you have graphical data for the value of *Ices* as a function of temperature, you can use it to fine-tune the value of *XTI*. The default value is 3.

Mobility temperature exponent, BEX

Mobility temperature coefficient value. You can use the default value for most devices. If you have graphical data for $V_{ce(sat)}$ at different temperatures, you can use it to fine-tune the value of *BEX*. The default value is -1.5.

Internal gate resistance temperature coefficient

Represents the fractional rate of change (α) of internal gate resistance (RG) with temperature. Thus the gate resistance is $R = R_{meas}(1 + \alpha\ (T_s - T_{m1}))$, where R_{meas} is the **Internal gate resistance, RG** parameter value. The default value is 0 1/K.

Device simulation temperature

Temperature T_s at which the device is simulated. The default value is 25 °C.

Main Tab (Event-Based Block Variant)

This configuration of the **Main** tab corresponds to the simplified, event-based block variant. If you are using the detailed variant of the block, see "Main Tab (Default Block Variant)" on page 1-346.

Vector of temperatures, Tj

Temperature values at which the collector-emitter and turn-on/turn-off losses are quoted. The default values, in K, are [298.15, 398.15]. This parameter is visible only if your block has an exposed thermal port.

Vector of collector currents, Ic

Collector currents for which the on-state collector-emitter voltages are defined. The first element must be zero. The default values, in A, are [0, 10, 50, 100, 200, 400, 600].

Corresponding on-state collector-emitter voltages

Collector-emitter voltages corresponding to the vector of collector currents. The first element must be zero. The default values, in V, are [0, 1.1, 1.3, 1.45, 1.75, 2.25, 2.7]. If your block has an exposed thermal port, this parameter is replaced

with the **Collector-emitter on-state voltages, Vce=fcn(Tj,Ic)** parameter, which defines the voltages in terms of both temperature and current.

Collector-emitter on-state voltages, Vce=fcn(Tj,Ic)

Collector-emitter voltages when in the on state, defined as a function of both temperature and current. The default values, in V, are `[0, 1.1, 1.3, 1.45, 1.75, 2.25, 2.7; 0, 1.0, 1.15, 1.35, 1.7, 2.35, 3.0]`. This parameter is visible only if your block has an exposed thermal port.

Turn-on switching losses, Eon=fcn(Tj,Ic)

Energy loss when turning the device on, defined as a function of temperature and final on-state current. The default values, in J, are `[0, 0.2, 1, 2, 4, 8, 15; 0, 0.3, 1.3, 2.5, 5, 11, 18]*1e-3`. This parameter is visible only if your block has an exposed thermal port.

Turn-off switching losses, Eoff=fcn(Tj,Ic)

Energy loss when turning the device off, defined as a function of temperature and final on-state current. The default values, in J, are `[0, 0.3, 1.5, 3, 6, 15, 25; 0, 0.7, 3.3, 6.5, 13, 25, 35]*1e-3`. This parameter is visible only if your block has an exposed thermal port.

Miller resistance

When the device turns on, it has a constant-value Miller resistance in series with the demanded voltage ramp. This resistance represents the partial conductance path through the device during turn on, and can be used to match the voltage spike observed when reconnecting a current-carrying inductor and corresponding freewheeling diode. A typical value is 10 to 50 times the effective on-state resistance. The default value is `0.1` Ohm.

Off-state conductance

Conductance when the device is in the off state. The default value is `1e-5` 1/Ohm.

Threshold voltage, Vth

The gate-emitter voltage must be greater than this value for the device to turn on. The default value is `6` V.

Dynamics Tab (Event-Based Block Variant)

Turn-on delay

Time before which the device starts to ramp on. The default value is `0.07` us.

Current rise time

Time taken for the current to ramp up when driving a resistive load. The default value is `0.7` us.

Turn-off delay

Time before which the device starts to ramp off. The default value is `0.2` us.

Current fall time

Time taken for the current to ramp down when driving a resistive load. The default value is `0.5` us.

Off-state voltage for rise and fall times

Off-state collector-emitter voltage used when specifying the rise and fall times. The default value is `300` V. If your block has an exposed thermal port, this parameter is replaced with the **Off-state voltage for timing and losses data** parameter, which defines the voltage used when specifying the rise and fall times and also the losses data, also with the default value of `300` V.

Ports

The block has the following ports:

`C`

Electrical conserving port associated with the PNP emitter terminal

`G`

Electrical conserving port associated with the IGBT gate terminal

`E`

Electrical conserving port associated with the PNP collector terminal

N-Channel JFET

Model N-Channel JFET

Library

Semiconductor Devices

Description

The N-Channel JFET block uses the Shichman and Hodges equations to represent an N-Channel JFET using a model with the following structure:

G is the transistor gate, D is the transistor drain, and S is the transistor source. The drain current, I_D, depends on the region of operation and whether the transistor is operating in normal or inverse mode.

- In normal mode ($V_{DS} \geq 0$), the block provides the following relationship between the drain current I_D and the drain-source voltage V_{DS}.

Region	Applicable Range of V_{GS} and V_{DS} Values	Corresponding I_D Equation
Off	$V_{GS} - V_{t0} \leq 0$	$I_D = 0$
Linear	$0 < V_{DS} < V_{GS} - V_{t0}$	$I_D = \beta V_{DS}(2(V_{GS} - V_{t0}) - V_{DS})(1 + \lambda V_{DS})$
Saturated	$0 < V_{GS} - V_{t0} \leq V_{DS}$	$I_D = \beta (V_{GS} - V_{t0})^2 (1 + \lambda V_{DS})$

- In inverse mode ($V_{DS} < 0$), the block provides the following relationship between the drain current I_D and the drain-source voltage V_{DS}.

Region	Applicable Range of V_{GS} and V_{DS} Values	Corresponding I_D Equation
Off	$V_{GD} - V_{t0} \leq 0$	$I_D = 0$
Linear	$0 < -V_{DS} < V_{GD} - V_{t0}$	$I_D = \beta V_{DS}(2(V_{GD} - V_{t0}) + V_{DS})(1 - \lambda V_{DS})$
Saturated	$0 < V_{GD} - V_{t0} \leq -V_{DS}$	$I_D = \beta (V_{GD} - V_{t0})^2 (1 - \lambda V_{DS})$

In the preceding equations:

- V_{GS} is the gate-source voltage.

- V_{GD} is the gate-drain voltage.

- V_{t0} is the threshold voltage. If you select `Specify using equation parameters directly` for the **Parameterization** parameter, V_{t0} is the **Threshold voltage** parameter value. Otherwise, the block calculates V_{t0} from the datasheet parameters you specify.

- β is the transconductance parameter. If you select `Specify using equation parameters directly` for the **Parameterization** parameter, β is the **Transconductance parameter** parameter value. Otherwise, the block calculates β from the datasheet parameters you specify.

- λ is the channel-length modulation parameter. If you select `Specify using equation parameters directly` for the **Parameterization** parameter, λ is the **Channel-length modulation** parameter value. Otherwise, the block calculates λ from the datasheet parameters you specify.

The currents in each of the diodes satisfy the exponential diode equation

$$I_{GD} = IS \cdot \left(e^{\frac{qV_{GD}}{kT_{m1}}} - 1 \right)$$

$$I_{GS} = IS \cdot \left(e^{\frac{qV_{GS}}{kT_{m1}}} - 1 \right)$$

where:

- IS is the saturation current. If you select `Specify using equation parameters directly` for the **Parameterization** parameter, IS is the **Saturation current** parameter value. Otherwise, the block calculates IS from the datasheet parameters you specify.
- q is the elementary charge on an electron (1.602176e–19 Coulombs).
- k is the Boltzmann constant (1.3806503e–23 J/K).
- T_{m1} is the measurement temperature. The value comes from the **Measurement temperature** parameter.

The block models gate junction capacitance as a fixed gate-drain capacitance C_{GD} and a fixed gate-source capacitance C_{GS}. If you select `Specify using equation parameters directly` for the **Parameterization** parameter, you specify these values directly using the **Gate-drain junction capacitance** and **Gate-source junction capacitance** parameters. Otherwise, the block derives them from the **Input capacitance Ciss** and **Reverse transfer capacitance Crss** parameter values. The two parameterizations are related as follows:

- $C_{GD} = Crss$
- $C_{GS} = Ciss - Crss$

Modeling Temperature Dependence

The default behavior is that dependence on temperature is not modeled, and the device is simulated at the temperature for which you provide block parameters. You can optionally include modeling the dependence of the transistor static behavior on temperature during simulation. Temperature dependence of the junction capacitances is not modeled, this being a much smaller effect.

When including temperature dependence, the transistor defining equations remain the same. The measurement temperature value, T_{m1}, is replaced with the simulation temperature, T_s. The transconductance, β, and the threshold voltage, V_{t0}, become a function of temperature according to the following equations:

$$\beta_{Ts} = \beta_{Tm1}\left(\frac{T_s}{T_{m1}}\right)^{BEX}$$

$$V_{t0s} = V_{t01} + \alpha \ (T_s - T_{m1})$$

where:

- T_{m1} is the temperature at which the transistor parameters are specified, as defined by the **Measurement temperature** parameter value.

- T_s is the simulation temperature.

- β_{Tm1} is JFET transconductance at the measurement temperature.

- β_{Ts} is JFET transconductance at the simulation temperature. This is the transconductance value used in the JFET equations when temperature dependence is modeled.

- V_{t01} is the threshold voltage at measurement temperature.

- V_{t0s} is the threshold voltage at simulation temperature. This is the threshold voltage value used in the JFET equations when temperature dependence is modeled.

- *BEX* is the mobility temperature exponent. A typical value of *BEX* is -1.5.

- α is the gate threshold voltage temperature coefficient, dV_{th}/dT.

For most JFETS, you can use the default value of `-1.5` for *BEX*. Some datasheets quote the value for α, but most typically they provide the temperature dependence for the saturated drain current, *I_dss*. Depending on the block parameterization method, you have two ways of specifying α:

- If you parameterize the block from a datasheet, you have to provide *I_dss* at a second measurement temperature. The block then calculates the value for α based on this data.

- If you parameterize by specifying equation parameters, you have to provide the value for α directly.

If you have more data comprising drain current as a function of gate-source voltage for fixed drain-source voltage plotted at more than one temperature, then you can also use Simulink Design Optimization software to help tune the values for α and *BEX*.

In addition, the saturation current term, *IS*, in the gate-drain and gate-source current equations depends on temperature

$$IS_{Ts} = IS_{Tm1} \cdot (T_s / T_{m1})^{XTI} \cdot \exp\left(-\frac{EG}{kT_s}(1 - T_s / T_{m1})\right)$$

where:

- IS_{Tm1} is the saturation current at the measurement temperature.
- IS_{Ts} is the saturation current at the simulation temperature. This is the saturation current value used in the bipolar transistor equations when temperature dependence is modeled.
- *EG* is the energy gap.
- *k* is the Boltzmann constant (1.3806503e–23 J/K).
- *XTI* is the saturation current temperature exponent.

Similar to α, you have two ways of specifying *EG* and *XTI*:

- If you parameterize the block from a datasheet, you have to specify the gate reverse current, *I_gss*, at a second measurement temperature. The block then calculates the value for *EG* based on this data and assuming a p-n junction nominal value of 3 for *XTI*.
- If you parameterize by specifying equation parameters, you have to provide the values for *EG* and *XTI* directly. This option gives you most flexibility to match device behavior, for example, if you have a graph of *I_gss* as a function of temperature. With this data you can use Simulink Design Optimization software to help tune the values for *EG* and *XTI*.

Thermal Port

The block has an optional thermal port, hidden by default. To expose the thermal port, right-click the block in your model, and then from the context menu select **Simscape** > **Block choices** > **Show thermal port**. This action displays the thermal port H on the block icon, and adds the **Thermal Port** tab to the block dialog box.

Use the thermal port to simulate the effects of generated heat and device temperature. For more information on using thermal ports and on the **Thermal Port** tab parameters, see "Simulating Thermal Effects in Semiconductors".

Basic Assumptions and Limitations

The model is based on the following assumptions:

- This block does not allow you to specify initial conditions on the junction capacitances. If you select the **Start simulation from steady state** option in the Solver Configuration block, the block solves the initial voltages to be consistent with the calculated steady state. Otherwise, voltages are zero at the start of the simulation.

- You may need to use nonzero ohmic resistance and junction capacitance values to prevent numerical simulation issues, but the simulation may run faster with these values set to zero.

- The block does not account for temperature-dependent effects on the junction capacitances.

- When you specify *I_dss* at a second measurement temperature, it must be quoted for the same working point (that is, the same drain current and gate-source voltage) as for the *I_dss* value on the **Main** tab. Inconsistent values for *I_dss* at the higher temperature will result in unphysical values for α and unrepresentative simulation results.

- You may need to tune the value of *BEX* to replicate the I_D-V_{GS} relationship (if available) for a given device. The value of *BEX* affects whether the I_D-V_{GS} curves for different temperatures cross each other, or not, for the ranges of I_D and V_{GS} considered.

Parameters

- "Main Tab" on page 1-363
- "Ohmic Resistance Tab" on page 1-364
- "Junction Capacitance Tab" on page 1-364
- "Temperature Dependence Tab" on page 1-365

Main Tab

Parameterization

Select one of the following methods for block parameterization:

- `Specify from a datasheet` — Provide parameters that the block converts to equations that describe the transistor. This is the default method.
- `Specify using equation parameters directly` — Provide equation parameters β, *IS*, V_{t0}, and λ.

Gate reverse current, I_gss

The reverse current that flows in the diode when the drain and source are short-circuited and a large negative gate-source voltage is applied. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is `-1` nA.

Saturated drain current, I_dss

The current that flows when a large positive drain-source voltage is applied for a specified gate-source voltage. For a depletion-mode device, this gate-source voltage may be zero, in which case *I_dss* may be referred to as the zero-gate voltage drain current. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is 3 mA.

I_dss measurement point, [V_gs V_ds]

A vector of the values of V_{GS} and V_{DS} at which *I_dss* is measured. Normally V_{GS} is zero. V_{DS} should be greater than zero. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is [`0 15`] V.

Small-signal parameters, [g_fs g_os]

A vector of the values of *g_fs* and *g_os*. *g_fs* is the forward transfer conductance, that is, the conductance for a fixed drain-source voltage. *g_os* is the output conductance, that is, the conductance for a fixed gate-source voltage. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is [`3e+03 10`] uS.

Small-signal measurement point, [V_gs V_ds]

A vector of the values of V_{GS} and V_{DS} at which *g_fs* and *g_os* are measured. V_{DS} should be greater than zero. For depletion-mode devices, V_{GS} is typically zero. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is [`0 15`] V.

Transconductance parameter

The derivative of drain current with respect to gate voltage. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter. The default value is `1e-04` A/V^2.

Saturation current

The magnitude of the current that the ideal diode equation approaches asymptotically for very large reverse bias levels. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter. The default value is `1e-14` A.

Threshold voltage

The gate-source voltage above which the transistor produces a nonzero drain current. For an enhancement device, Vt0 should be positive. For a depletion mode device, Vt0 should be negative. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter. The default value is `-2` V.

Channel-length modulation

The channel-length modulation. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter. The default value is `0` 1/V.

Measurement temperature

The temperature for which the datasheet parameters are quoted. The default value is 25 °C.

Ohmic Resistance Tab

Source ohmic resistance

The transistor source resistance. The default value is `1e-4` Ω. The value must be greater than or equal to `0`.

Drain ohmic resistance

The transistor drain resistance. The default value is `0.01` Ω. The value must be greater than or equal to `0`.

Junction Capacitance Tab

Parameterization

Select one of the following methods for block parameterization:

- `Specify from a datasheet` — Provide parameters that the block converts to junction capacitance values. This is the default method.
- `Specify using equation parameters directly` — Provide junction capacitance parameters directly.

Input capacitance, Ciss

The gate-source capacitance with the drain shorted to the source. This parameter is only visible when you select `Specify from a datasheet` for the **Model junction capacitance** parameter. The default value is 4.5 pF.

Reverse transfer capacitance, Crss

The drain-gate capacitance with the source connected to ground. This parameter is only visible when you select `Specify from a datasheet` for the **Model junction capacitance** parameter. The default value is 1.5 pF.

Gate-source junction capacitance

The value of the capacitance placed between the gate and the source. This parameter is only visible when you select `Specify using equation parameters directly` for the **Model junction capacitance** parameter. The default value is 3 pF.

Gate-drain junction capacitance

The value of the capacitance placed between the gate and the drain. This parameter is only visible when you select `Specify using equation parameters directly` for the **Model junction capacitance** parameter. The default value is 1.5 pF.

Temperature Dependence Tab

Parameterization

Select one of the following methods for temperature dependence parameterization:

- `None — Simulate at parameter measurement temperature` — Temperature dependence is not modeled. This is the default method.
- `Model temperature dependence` — Model temperature-dependent effects. You also have to provide a set of additional parameters depending on the block parameterization method. If you parameterize the block from a datasheet, you have to provide values for *I_gss* and *I_dss* at second measurement temperature. If you parameterize by directly specifying equation parameters, you have to provide the values for *EG*, *XTI*, and the gate threshold voltage temperature coefficient, dV_{t0}/dT. Regardless of the block parameterization method, you also have to provide values for *BEX* and for the simulation temperature, T_s.

Gate reverse current, I_gss, at second measurement temperature

The value of the gate reverse current, *I_gss*, at the second measurement temperature. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter on the **Main** tab. It must be quoted for the same working point (drain current and gate-source voltage) as the **Drain-source on resistance, R_DS(on)** parameter on the **Main** tab. The default value is -200 nA.

Saturated drain current, I_dss, at second measurement temperature

The value of the saturated drain current, *I_dss*, at the second measurement temperature, and when the *I_dss* measurement point is the same as defined by the **I_dss measurement point, [V_gs V_ds]** parameter on the **Main** tab. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter on the **Main** tab. The default value is 2.5 mA.

Second measurement temperature

Second temperature T_{m2} at which **Gate reverse current, I_gss, at second measurement temperature** and **Saturated drain current, I_dss, at second measurement temperature** are measured. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter on the **Main** tab. The default value is 125 °C.

Energy gap, EG

Energy gap value. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter on the **Main** tab. The default value is 1.11 eV.

Saturation current temperature exponent, XTI

Saturation current temperature coefficient value. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter on the **Main** tab. The default value is 3.

Gate threshold voltage temperature coefficient, dVt0/dT

The rate of change of gate threshold voltage with temperature. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter on the **Main** tab. The default value is -6 mV/K.

Mobility temperature exponent, BEX

Mobility temperature coefficient value. You can use the default value for most JFETs. See the "Basic Assumptions and Limitations" on page 1-362 section for additional considerations. The default value is -1.5.

Device simulation temperature

Temperature T_s at which the device is simulated. The default value is 25 °C.

Ports

The block has the following ports:

G

    Electrical conserving port associated with the transistor gate terminal

D

    Electrical conserving port associated with the transistor drain terminal

S

    Electrical conserving port associated with the transistor source terminal

References

[1] H. Shichman and D. A. Hodges, *Modeling and simulation of insulated-gate field-effect transistor switching circuits*. IEEE J. Solid State Circuits, SC-3, 1968.

[2] G. Massobrio and P. Antognetti. *Semiconductor Device Modeling with SPICE*. 2nd Edition, McGraw-Hill, 1993. Chapter 2.

See Also

P-Channel JFET

N-Channel LDMOS FET

Model N-Channel LDMOS or VDMOS transistors suitable for high voltage

Library

Semiconductor Devices

Description

The N-Channel LDMOS FET block lets you model LDMOS (or VDMOS) transistors suitable for high voltage. The model is based on surface potential and includes effects due to an extended drain (drift) region:

- Nonlinear capacitive effects associated with the drift region
- Surface scattering and velocity saturation in the drift region
- Velocity saturation and channel-length modulation in the channel region
- Charge conservation inside the model, so you can use the model for charge sensitive simulations
- The intrinsic body diode
- Reverse recovery in the body diode model
- Temperature scaling of physical parameters
- For the thermal variant (see "Thermal Port" on page 1-374), dynamic self-heating

The physical structure of the model is shown in the following figure.
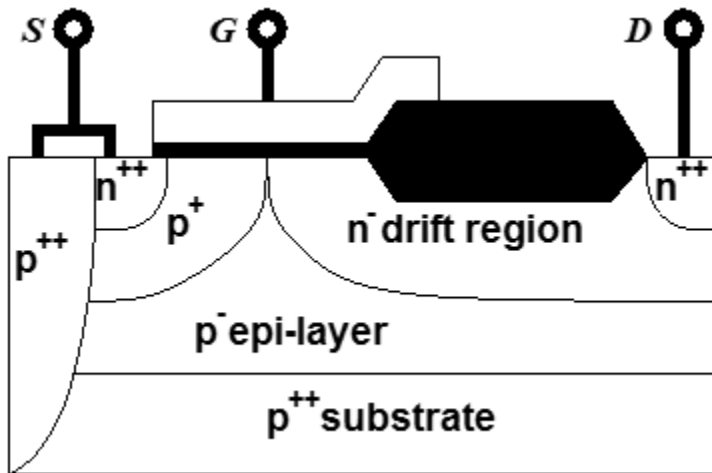
The channel region is in the p+ region, from the heavily n-doped source well to the end of the p+ region. The drift region is a lightly doped drain extension. Further down, there is a p-type epi-layer, and then the entire structure is on a heavily p-doped substrate. The gate oxide is thin over the entire channel region and over part of the drift region. Further into the drift region, the gate oxide has a greater thickness in the local-oxidation-of-silicon (LOCOS) region.

The next figure shows the equivalent circuit of the model.
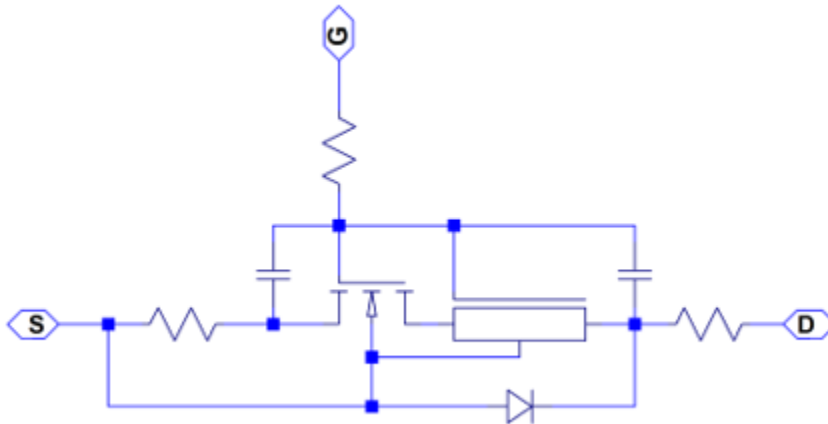
The modeling approach is similar to [1]. The overlaps of the gate contact with the source and drain n-wells are modeled as lumped linear capacitances. The channel (p+) region is modeled using the surface-potential-based MOSFET model. The pn-junction between the source/bulk and drain is modeled using an ideal diode, including both junction and diffusion capacitances. The drift region underneath the thin gate oxide is modeled according to a surface-potential formulation, which includes:

- The current due to the accumulation layer at the semiconductor-oxide interface
- The current due to the electrons flowing towards the drain deeper inside the drift region

The space-charge region between the epi-layer and the drift region is represented using a pinching effect on the current flowing through the bulk of the drift region. The LOCOS part of the drift region is modeled as a lumped, series resistor, and there are also series resistances added to the source and gate contacts.

For detailed description of the channel model, see the surface-potential-based model of the N-Channel MOSFET block. The drift region model is similarly derived from the surface potential using the Poisson equation. For an n-type semiconductor under the gradual-channel approximation, the defining equations are:

$$\frac{\partial^2 \psi}{\partial y^2} \approx \frac{q N_D}{\varepsilon_{Si}} \left[-1 - \exp\left(\frac{-\psi - 2\phi_B}{\phi_T} \right) + \exp\left(\frac{\psi - V_{CB}}{\phi_T} \right) \right]$$

$$\phi_T = \frac{k_B T}{q}$$

where:

- ψ is the electrostatic potential.
- q is the magnitude of the electronic charge.
- N_D is the doping density of the drift region.
- ε_{Si} is the dielectric permittivity of the semiconductor material (for example, silicon).
- ϕ_B is the difference between the intrinsic Fermi level and the Fermi level deep in the drift region.
- V_{CB} is the quasi-Fermi potential of the drift region referenced to the bulk.
- ϕ_T is the thermal voltage.

- k_B is Boltzmann's constant.

- T is temperature.

If we neglect inversion for the DC current model, we obtain the following current expression:

$$I_D = \frac{1}{1 + \theta_{sat} V_{DK}} \left[\frac{f_{lin}}{R_D} V_{DK} + \frac{\beta}{2} \cdot \frac{V_{GK}^2 - V_{GD}^2}{1 + \frac{\theta_{surf}}{2}(V_{GK} + V_{GD})} \right]$$

where:

- I_D is the drain current.

- θ_{sat} is the velocity saturation.

- V_{ij} is the voltage difference between nodes i and j, where subscripts D and K refer to the drain and to the junction of the channel and drift regions, respectively, and subscript G refers to the gate with a correction due to the flatband voltage being applied.

- f_{lin}/R_D represents the conductance of the bulk of the drift region, including the effect of pinching due to depletion from the epi-drift interface.

- β is the gain of the accumulation layer at the interface between the drift region and the thin gate oxide.

- θ_{surf} is the parameter that accounts for scattering in the accumulation layer due to the vertical electric field.

The pinching off of the bulk part of the drift region is described by

$$f_{lin} = 1 - \lambda_D \frac{\sqrt{V_{bi} + V_{SB}} - \sqrt{V_{bi}}}{\sqrt{V_{bi}}}$$

where:

- λ_D is the parameter representing the n-side vertical depth of the space-charge region along the epi-drift interface at zero bias divided by the vertical depth of the undepleted part of the drift region at zero bias.

1-371
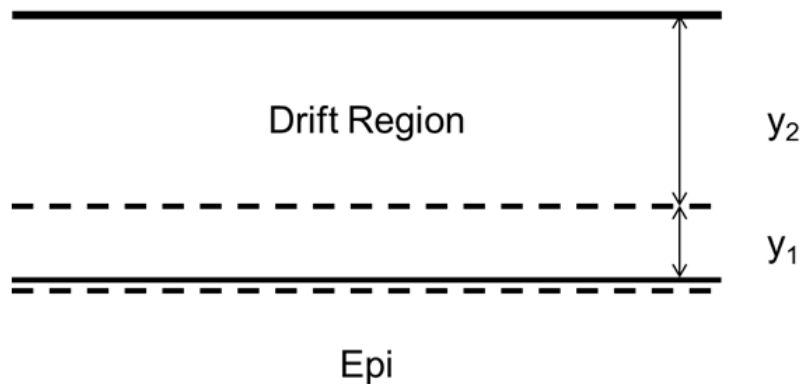
In the figure, the top solid line is the semiconductor surface. The lower solid line is the junction between the drift region and the epi layer. The dashed lines show the extent of the space-charge region around the drift-epi interface. λ_D is y_1/y_2 at zero bias.

- V_{bi} is the built-in voltage for the epi-drift diode.

- V_{SB} is the source-body voltage, used as an approximation to the bias applied across the epi-drift diode. Using this voltage instead of V_{KB} is more numerically stable, and is justified because most of the drain-source voltage drops across the drift region in the transistor on-state.

The charge model is similar to that of the surface-potential-based MOSFET model, with additional expressions to account for the charge in the drift region. The block uses the derived equations as described in [1], which include both inversion and accumulation in the drift region.

Modeling Body Diode

The block models the body diode as an ideal, exponential diode with both junction and diffusion capacitances:

$$I_{dio} = I_s \left[\exp\left(-\frac{V_{DB}}{n\phi_T} \right) - 1 \right]$$

$$C_j = \frac{C_{j0}}{\sqrt{1 + \dfrac{V_{DB}}{V_{bi}}}}$$

$$C_{diff} = \frac{\tau I_s}{n\phi_T} \exp\left(-\frac{V_{DB}}{n\phi_T}\right)$$

where:

- I_{dio} is the current through the diode.
- I_s is the reverse saturation current.
- V_{DB} is the drain-body voltage.
- n is the ideality factor.
- ϕ_T is the thermal voltage.
- C_j is the junction capacitance of the diode.
- C_{j0} is the zero-bias junction capacitance.
- V_{bi} is the built-in voltage.
- C_{diff} is the diffusion capacitance of the diode.
- τ is the transit time.

The capacitances are defined through an explicit calculation of charges, which are then differentiated to give the capacitive expressions above. The block computes the capacitive diode currents as time derivatives of the relevant charges, similar to the computation in the surface-potential-based MOSFET model.

Modeling Temperature Dependence

The default behavior is that dependence on temperature is not modeled, and the device is simulated at the temperature for which you provide block parameters. To model the dependence on temperature during simulation, select `Model temperature dependence` for the **Parameterization** parameter on the **Temperature Dependence** tab.

The model includes temperature effects on the capacitance characteristics, as well as modeling the dependence of the transistor static behavior on temperature during simulation.

The **Measurement temperature** parameter on the **Main** tab specifies temperature T_{m1} at which the other device parameters have been extracted. The **Temperature Dependence** tab provides the simulation temperature, T_s, and the temperature-scaling coefficients for the other device parameters. For more information, see "Temperature Dependence Tab" on page 1-378.

Thermal Port

The block has an optional thermal port, hidden by default. To expose the thermal port, right-click the block in your model, and then from the context menu select **Simscape > Block choices > Show thermal port**. This action displays the thermal port H on the block icon, and adds the **Thermal Port** tab to the block dialog box.

Use the thermal port to simulate the effects of generated heat and device temperature. For more information on using thermal ports and on the **Thermal Port** tab parameters, see "Simulating Thermal Effects in Semiconductors".

The thermal variant of the block includes dynamic self-heating, that is, lets you simulate the effect of self-heating on the electrical characteristics of the device.

Parameters

- "Main Tab" on page 1-374
- "Ohmic Resistance Tab" on page 1-377
- "Capacitances Tab" on page 1-377
- "Body Diode Tab" on page 1-378
- "Temperature Dependence Tab" on page 1-378

Main Tab

Gain, [channel drift_region]

The gain, β, of the MOSFET regions. The parameter value is a two-element vector, with the first element corresponding to the channel, and the second — to the drift region. This parameter primarily defines the linear region of operation on an I_D–V_{DS} characteristic. The values of both elements must be greater than 0. The default value is `[11.6, 0.01]` A/V^2.

Flatband voltage, [channel drift_region]

The flatband voltage, V_{FB}, defines the gate bias that must be applied in order to achieve the flatband condition at the surface of the silicon. The parameter value is a two-element vector, with the first element corresponding to the channel, and the second — to the drift region. The default value is [-1.05, -0.1] V. You can also use this parameter to arbitrarily shift the threshold voltage due to material work function differences, and to trapped interface or oxide charges. In practice, however, it is usually recommended to modify the threshold voltage by using the **Body factor** and **Surface potential at strong inversion** parameters first, and only use this parameter for fine-tuning.

The threshold voltage for the channel region, for a short-circuited source-bulk connection, is approximately

$$V_T = V_{FB} + 2\phi_B + 2\phi_T + \gamma\sqrt{2\phi_B + 2\phi_T}$$

where $2\phi_B$ is the surface potential at strong inversion and γ is the body factor, both at the channel region.

Body factor, [channel drift_region]

Body factor, γ, in the surface-potential equation. The parameter value is a two-element vector, with the first element corresponding to the channel, and the second — to the drift region. The default value is [3.4, 2.5] $V^{1/2}$.

For the channel region, the body factor is

$$\gamma = \frac{\sqrt{2q\varepsilon_{Si}N_A}}{C_{ox}}$$

See the N-Channel MOSFET block reference page for details on this equation. The drift region equation is similar, except that N_A is replaced by the doping density, N_D. The channel-region parameter value primarily impacts the threshold voltage. For the drift region, this parameter primarily affects the charge model, and also has a minor effect on the pinch-off behavior of the bulk current through the drift region.

Surface potential at strong inversion, [channel drift_region]

The $2\phi_B$ term in the surface-potential equation. The parameter value is a two-element vector, with the first element corresponding to the channel, and the second — to the drift region. The default value is [0.95, 0.95] V.

The channel-region parameter value also primarily impacts the threshold voltage. For the drift region, this parameter affects the charge model only.

Velocity saturation factor, [channel drift_region]

Velocity saturation, θ_{sat}, in the drain-current equation. Use this parameter in cases where a good fit to linear operation leads to a saturation current that is too high. By increasing this parameter value, you reduce the saturation current. The parameter value is a two-element vector, with the first element corresponding to the channel, and the second — to the drift region. The default value is `[0.0, 0.1]` 1/V, which means that velocity saturation in the channel region is off by default.

Drift region surface scattering factor

Surface scattering factor, θ_{surf}, in the drain-current equation. This parameter applies to the drift region only and accounts for scattering in the accumulation layer due to the vertical electric field. The default value is `0` 1/V.

Channel-length modulation factor

The factor, α, multiplying the logarithmic term in the $G_{\Delta L}$ equation. See the N-Channel MOSFET block reference page for details on this equation. This parameter describes the onset of channel-length modulation. For device characteristics that exhibit a positive conductance in saturation, increase the parameter value to fit this behavior. This parameter applies to the channel region only. The default value is `0`, which means that channel-length modulation is off by default.

Channel-length modulation voltage

The voltage V_{p} in the $G_{\Delta L}$ equation. See the N-Channel MOSFET block reference page for details on this equation. This parameter controls the drain-voltage at which channel-length modulation starts to become active. This parameter applies to the channel region only. The default value is `50` mV.

Linear-to-saturation transition coefficient

This parameter controls how smoothly the MOSFET transitions from linear into saturation, particularly when velocity saturation is enabled. This parameter can usually be left at its default value, but you can use it to fine-tune the knee of the I_{D}–V_{DS} characteristic. This parameter applies both to the channel and drift regions. The expected range for this parameter value is between 2 and 8. The default value is `8`.

Measurement temperature

Temperature T_{m1} at which the block parameters are measured. If the **Device simulation temperature** parameter on the **Temperature Dependence** tab differs from this value, then device parameters will be scaled from their defined values according to the simulation and reference temperatures. For more information, see "Temperature Dependence Tab" on page 1-378. The default value is 25 °C.

Ohmic Resistance Tab

Source ohmic resistance

The transistor source resistance, that is, the series resistance associated with the source contact. The default value is `1e-4` Ω. The value must be greater than or equal to `0`.

Drain ohmic resistance

The transistor drain resistance, that is, the series resistance associated with the drain contact and with the LOCOS part of the drift region, which is not heavily impacted by the applied gate voltage. The default value is `0.07` Ω. The value must be greater than or equal to `0`.

Gate ohmic resistance

The transistor gate resistance, that is, the series resistance associated with the gate contact. The default value is `8.4` Ω. The value must be greater than or equal to `0`.

Drift region low-bias resistance for gated region

Resistance R_D in the drain-current equation. It represents the resistance of the bulk part of the drift region in the absence of depletion from the top and bottom interfaces. The default value is `0.1` Ω. The value must be greater than or equal to `0`.

Drift region depletion layer thickness factor

Parameter λ_D in the drain-current equation. It is the ratio of vertical depths y_1 and y_2 at zero bias, where y_1 represents the space-charge region and y_2 represents the undepleted part of the drift region. The default value is `0.2`.

Capacitances Tab

Oxide capacitance

The parallel plate gate-channel and gate-drift-region capacitance. The parameter value is a two-element vector, with the first element corresponding to the channel, and the second — to the drift region. The default value is `[1600.0, 1000.0]` pF.

Gate-source overlap capacitance

The fixed, linear capacitance associated with the overlap of the gate electrode with the source well. The default value is `15` pF.

Gate-drain overlap capacitance

The fixed, linear capacitance associated with the overlap of the gate electrode with the drain well. The default value is `15` pF.

Body Diode Tab

Reverse saturation current

The current designated by the I_s symbol in the body-diode equations. The default value is `1e-13` A.

Built-in voltage

The built-in voltage of the diode, designated by the V_{bi} symbol in the body-diode equations. The default value is `0.6` V.

Ideality factor

The factor designated by the n symbol in the body-diode equations. The default value is `1`.

Zero-bias junction capacitance

The capacitance between the drain and bulk contacts at zero-bias due to the body diode alone. It is designated by the C_{j0} symbol in the body-diode equations. The default value is `1800` pF.

Transit time

The time designated by the τ symbol in the body-diode equations. The default value is `50` ns.

Temperature Dependence Tab

Parameterization

Select one of the following methods for temperature dependence parameterization:

- `None — Simulate at parameter measurement temperature` — Temperature dependence is not modeled. This is the default method.

- `Model temperature dependence` — Model temperature-dependent effects. Provide a value for the device simulation temperature, T_s, and the temperature-scaling coefficients for other block parameters.

Device simulation temperature

Temperature T_s at which the device is simulated. The default value is 25 °C.

Gain temperature exponent, [channel drift_region]

The parameter value is a two-element vector, with the first element corresponding to the channel, and the second — to the drift region. Both in the channel and the drift region, the MOSFET gain, β, is assumed to scale exponentially with temperature, $\beta =$

$\beta_{m1}(T_{m1}/T_s)^\wedge \eta_\beta$. β_{m1} is the value of the channel or drift region gain, as specified by the **Gain, [channel drift_region]** parameter from the **Main** tab. η_β is the corresponding element of the **Gain temperature exponent, [channel drift_region]** parameter. The default value is [1.3, 1.3].

Flatband voltage temperature coefficient, [channel drift_region]

The parameter value is a two-element vector, with the first element corresponding to the channel, and the second — to the drift region. The flatband voltage, V_{FB}, is assumed to scale linearly with temperature, $V_{FB} = V_{FBm1} + (T_s - T_{m1})S_{T,V_{FB}}$. V_{FBm1} is the value of the channel or drift region flatband voltage, as specified by the **Flatband voltage, [channel drift_region]** parameter from the **Main** tab. $S_{T,V_{FB}}$ is the corresponding element of the **Flatband voltage temperature coefficient, [channel drift_region]** parameter. The default value is [0.0005, 0.0005] V/K.

Surface potential at strong inversion temperature coefficient

The surface potential at strong inversion, $2\phi_B$, is assumed to scale linearly with temperature, $2\phi_B = 2\phi_{Bm1} + (T_s - T_{m1})S_{T,\phi_B}$. $2\phi_{Bm1}$ is the value of the **Surface potential at strong inversion** parameter from the **Main** tab and S_{T,ϕ_B} is the **Surface potential at strong inversion temperature coefficient**. The default value is -8.5e-4 V/K.

Velocity saturation temperature exponent, [channel drift_region]

The parameter value is a two-element vector, with the first element corresponding to the channel, and the second — to the drift region. The velocity saturation, θ_{sat}, is assumed to scale exponentially with temperature, $\theta_{sat} = \theta_{sat,m1}(T_{m1}/T_s)^\wedge \eta_\theta$. $\theta_{sat,m1}$ is the value of the channel or drift region velocity saturation factor, as specified by the **Velocity saturation factor, [channel drift_region]** parameter from the **Main** tab. η_θ is the corresponding element of the **Velocity saturation temperature exponent, [channel drift_region]** parameter. The default value is [1.04, 1.04].

Ohmic resistance temperature exponent

The series resistances are assumed to correspond to semiconductor resistances. Therefore, they decrease exponentially with increasing temperature. $R_i = R_{i,m1}(T_{m1}/T_s)^\wedge \eta_R$, where i is S, D, or G, for the source, drain, or gate series resistance, respectively. $R_{i,m1}$ is the value of the corresponding series resistance parameter from the **Ohmic Resistance** tab and η_R is the **Ohmic resistance temperature exponent**. The default value is 0.95.

Drift region low-bias resistance temperature exponent for gated portion

Resistance R_D, the low-bias resistance of the bulk part of the drift region, scales similarly to the other series resistances. A separate value of the temperature

exponent for this resistance provides an extra degree of freedom. The default value is 0.95.

Body diode reverse saturation current temperature exponent

The reverse saturation current for the body diode is assumed to be proportional to the square of the intrinsic carrier concentration, $n_i = N_C \exp(-E_G/2k_B T)$. N_C is the temperature-dependent effective density of states and E_G is the temperature-dependent bandgap for the semiconductor material. To avoid introducing another temperature-scaling parameter, the block neglects the temperature dependence of the bandgap and uses the bandgap of silicon at 300K (1.12eV) for all device types. Therefore, the temperature-scaled reverse saturation current is given by

$$I_s = I_{s,m1} \left(\frac{T_s}{T_{m1}} \right)^{\eta_{Is}} \cdot \exp\left(\frac{E_G}{k_B} \cdot \left(\frac{1}{T_{m1}} - \frac{1}{T_s} \right) \right).$$

$I_{s,m1}$ is the value of the **Reverse saturation current** parameter from the **Body Diode** tab, k_B is Boltzmann's constant (8.617x10-5eV/K), and η_{Is} is the **Body diode reverse saturation current temperature exponent**. The default value is 3, because N_C for silicon is roughly proportional to $T^{3/2}$. You can remedy the effect of neglecting the temperature-dependence of the bandgap by a pragmatic choice of η_{Is}.

Ports

The block has the following ports:

G

Electrical conserving port associated with the transistor gate terminal

D

Electrical conserving port associated with the transistor drain terminal

S

Electrical conserving port associated with the transistor source terminal

References

[1] Aarts, A., N. D'Halleweyn, and R. Van Langevelde. "A Surface-Potential-Based High-Voltage Compact LDMOS Transistor Model." *IEEE Transactions on Electron Devices*. 52(5):999 - 1007. June 2005.

[2] Van Langevelde, R., A. J. Scholten, and D. B .M. Klaassen. "Physical Background of MOS Model 11. Level 1101." *Nat.Lab. Unclassified Report 2003/00239*. April 2003.

[3] Oh, S-Y., D. E. Ward, and R. W. Dutton. "Transient analysis of MOS transistors." *IEEE J. Solid State Circuits*. SC-15, pp. 636-643, 1980.

See Also

N-Channel MOSFET | P-Channel LDMOS FET

Topics

Interactive Generation of LDMOS Characteristics

Introduced in R2016b

N-Channel MOSFET

Model N-Channel MOSFET using either Shichman-Hodges equation or surface-potential-based model

Library

Semiconductor Devices

Description

The N-Channel MOSFET block provides two main modeling variants:

- Based on threshold voltage — Uses the Shichman-Hodges equation to represent the device. This modeling approach, based on threshold voltage, has the benefits of simple parameterization and simple current-voltage expressions. However, these models have difficulty in accurately capturing transitions across the threshold voltage and lack some important effects, such as velocity saturation. For details, see "Threshold-Based Model" on page 1-383.

- Based on surface potential — Uses the surface-potential equation to represent the device. This modeling approach provides a greater level of model fidelity than the simple square-law (threshold-voltage-based) models can provide. The trade-off is that there are more parameters that require extraction. For details, see "Surface-Potential-Based Model" on page 1-386.

Together with the thermal port variants (see "Thermal Port" on page 1-392), the block therefore provides you with four choices. To select the desired variant, right-click the block in your model. From the context menu, select **Simscape > Block choices**, and then one of the following options:

- **Threshold-based** — Basic model, which represents the device using the Shichman-Hodges equation (based on threshold voltage) and does not simulate thermal effects. This is the default.
- **Threshold-based with thermal** — Model based on threshold voltage and with exposed thermal port.
- **Surface-potential-based** — Model based on surface potential. This model does not simulate thermal effects.
- **Surface-potential-based with thermal** — Thermal variant of the model based on surface potential.

Threshold-Based Model

The threshold-based variant of the block uses the Shichman and Hodges equations [1] for an insulated-gate field-effect transistor to represent an N-Channel MOSFET.

The drain-source current, I_{DS}, depends on the region of operation:

- In the off region ($V_{GS} < V_{th}$), the drain-source current is:

$$I_{DS} = 0$$

- In the linear region ($0 < V_{DS} < V_{GS} - V_{th}$), the drain-source current is:

$$I_{DS} = K\left((V_{GS} - V_{th})V_{DS} - V_{DS}^{2}/2\right)\left(1 + \lambda\,|V_{DS}|\right)$$

- In the saturated region ($0 < V_{GS} - V_{th} < V_{DS}$), the drain-source current is:

$$I_{DS} = (K/2)(V_{GS} - V_{th})^{2}\left(1 + \lambda\,|V_{DS}|\right)$$

In the preceding equations:

- K is the transistor gain.
- V_{DS} is the positive drain-source voltage.
- V_{GS} is the gate-source voltage.
- V_{th} is the threshold voltage.
- λ is the channel modulation.

Charge Model for Threshold-Based Variant

The block models junction capacitances either by fixed capacitance values, or by tabulated values as a function of the drain-source voltage. In either case, you can either directly specify the gate-source and gate-drain junction capacitance values, or let the block derive them from the input and reverse transfer capacitance values. Therefore, the **Parameterization** options for charge model on the **Junction Capacitance** tab are:

- `Specify fixed input, reverse transfer and output capacitance` — Provide fixed parameter values from datasheet and let the block convert the input and reverse transfer capacitance values to junction capacitance values, as described below. This is the default method.

- `Specify fixed gate-source, gate-drain and drain-source capacitance` — Provide fixed values for junction capacitance parameters directly.

- `Specify tabulated input, reverse transfer and output capacitance` — Provide tabulated capacitance and drain-source voltage values based on datasheet plots. The block converts the input and reverse transfer capacitance values to junction capacitance values, as described below.

- `Specify tabulated gate-source, gate-drain and drain-source capacitance` — Provide tabulated values for junction capacitances and drain-source voltage.

Use one of the tabulated capacitance options (`Specify tabulated input, reverse transfer and output capacitance` or `Specify tabulated gate-source, gate-drain and drain-source capacitance`) when the datasheet provides a plot of junction capacitances as a function of drain-source voltage. Using tabulated capacitance values gives more accurate dynamic characteristics and avoids the need for interactive tuning of parameters to fit the dynamics.

If you use the `Specify fixed gate-source, gate-drain and drain-source capacitance` or `Specify tabulated gate-source, gate-drain and drain-source capacitance` option, the **Junction Capacitance** tab lets you specify the **Gate-drain junction capacitance**, **Gate-source junction capacitance**, and **Drain-source junction capacitance** parameter values (fixed or tabulated) directly. Otherwise, the block derives them from the **Input capacitance, Ciss**, **Reverse transfer capacitance, Crss**, and **Output capacitance, Coss** parameter values. These two parameterization methods are related as follows:

- $C_{GD} = Crss$

- $C_{GS} = Ciss - Crss$
- $C_{DS} = Coss - Crss$

The two fixed capacitance options (`Specify fixed input, reverse transfer and output capacitance` or `Specify fixed gate-source, gate-drain and drain-source capacitance`) let you model gate junction capacitance as a fixed gate-source capacitance C_{GS} and either a fixed or a nonlinear gate-drain capacitance C_{GD}. If you select the `Gate-drain charge function is nonlinear` option for the **Charge-voltage linearity** parameter, then the gate-drain charge relationship is defined by the piecewise-linear function shown in the following figure.

Nonlinear gate-drain charge function

For instructions on how to map a time response to device capacitance values, see the N-Channel IGBT block reference page. However, this mapping is only approximate because the Miller voltage typically varies more from the threshold voltage than in the case for the IGBT.

Note Because this block implementation includes a charge model, you must model the impedance of the circuit driving the gate to obtain representative turn-on and turn-off dynamics. Therefore, if you are simplifying the gate drive circuit by representing it as a controlled voltage source, you must include a suitable series resistor between the voltage source and the gate.

Surface-Potential-Based Model

The surface-potential-based variant of the block provides a greater level of model fidelity than the simple square-law (threshold-voltage-based) model. The surface-potential-based block variant includes the following effects:

- Fully nonlinear capacitance model (including the nonlinear Miller capacitance)
- Charge conservation inside the model, so you can use the model for charge sensitive simulations
- Velocity saturation and channel-length modulation
- The intrinsic body diode
- Reverse recovery in the body diode model
- Temperature scaling of physical parameters
- For the thermal variant, dynamic self-heating (that is, you can simulate the effect of self-heating on the electrical characteristics of the device)

This model is a minimal version of the world-standard PSP model (see http://nsti.org/Nanotech2005/WCM2005/WCM2005-GGildenblat.pdf), including only certain effects from the PSP model in order to strike a balance between model fidelity and complexity. For details of the physical background to the phenomena included in this model, see [2].

The basis of the model is Poisson equation:

$$\frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} = \frac{qN_A}{\varepsilon_{Si}} \left[1 - \exp\left(\frac{-\psi}{\phi_T}\right) + \exp\left(\frac{\psi - 2\phi_B - V_{CB}}{\phi_T}\right) \right]$$

$$\phi_T = \frac{k_B T}{q}$$

where:

- ψ is the electrostatic potential.
- q is the magnitude of the electronic charge.
- N_A is the density of acceptors in the substrate.
- ε_{Si} is the dielectric permittivity of the semiconductor material (for example, silicon).
- ϕ_B is the difference between the intrinsic Fermi level and the Fermi level in the bulk silicon.
- V_{CB} is the quasi-Fermi potential of the surface layer referenced to the bulk.
- ϕ_T is the thermal voltage.
- k_B is Boltzmann's constant.
- T is temperature.

Poisson equation is used to derive the surface-potential equation:

$$\left(V_{GB} - V_{FB} - \psi_s\right)^2 = \gamma^2 \left[\psi_s + \phi_T\left(\exp\left(\frac{-\psi_s}{\phi_T}\right) - 1\right) + \phi_T \exp\left(-\frac{2\phi_B + V_{CB}}{\phi_T}\right)\left(\exp\left(\frac{\psi_s}{\phi_T}\right) - 1\right)\right]$$

where:

- V_{GB} is the applied gate-body voltage.
- V_{FB} is the flatband voltage.
- ψ_s is the surface potential.
- γ is the body factor,

$$\gamma = \frac{\sqrt{2q\varepsilon_{Si}N_A}}{C_{ox}}$$

- C_{ox} is the capacitance per unit area.

The block uses an explicit approximation to the surface-potential equation, to avoid the need for numerical solution to this implicit equation.

Once the surface potential is known, the drain current I_D is given by

$$I_D = \frac{W\mu_0}{LG_{\Delta L}G_{mob}\sqrt{1+\left(\theta_{sat}\Delta\psi\right)^2}}\left[-\bar{Q}_{inv}\Delta\psi + \phi_T\left(Q_{invL} - Q_{inv0}\right)\right]$$

where:

- W is the device width.
- L is the channel length.
- μ_0 is the low-field mobility.
- θ_{sat} is the velocity saturation.
- $\Delta\psi$ is the difference in the surface potential between the drain and the source.
- Q_{inv0} and Q_{invL} are the inversion charge densities at the source and drain, respectively.
- $\bar{Q}_{inv}$ is the average inversion charge density across the channel.
- G_{mob} is the mobility reduction factor. For more information, see the **Surface roughness scattering factor** parameter description in the "Main Tab (Surface-Potential-Based Variant)" on page 1-395 section.
- $G_{\Delta L}$ is the channel-length modulation.

$$G_{\Delta L} = 1 - \frac{\Delta L}{L} = 1 - \alpha \ln\left[\frac{V_{DB} - V_{DB,eff} + \sqrt{\left(V_{DB} - V_{DB,eff}\right)^2 + V_p^2}}{V_p}\right]$$

where:

- α is the channel-length modulation factor.
- V_{DB} is the drain-body voltage.
- $V_{\text{DB,eff}}$ is the drain-body voltage clipped to a maximum value corresponding to velocity saturation or pinch-off (whichever occurs first).
- V_{p} is the channel-length modulation voltage.

The block computes the inversion charge densities directly from the surface potential.

The block also computes the nonlinear capacitances from the surface potential. Source and drain charge contributions are assigned via a bias-dependent Ward-Dutton charge-

partitioning scheme, as described in [3]. These charges are computed explicitly, so this model is charge-conserving. The capacitive currents are computed by taking the time derivatives of the relevant charges. In practice, the charges within the simulation are normalized to the oxide capacitance and computed in units of volts.

The MOSFET gain, β, is given by

$$\beta = \frac{W \mu_0 C_{ox}}{L}$$

The threshold voltage for a short-circuited source-bulk connection is approximately given by

$$V_T = V_{FB} + 2\phi_B + 2\phi_T + \gamma \sqrt{2\phi_B + 2\phi_T}$$

where:

- $2\phi_B$ is the surface potential at strong inversion.

The overall model consists of an intrinsic MOSFET defined by the surface-potential formulation, a body diode, series resistances, and fixed overlap capacitances, as shown in the schematic.

Modeling Body Diode

The block models the body diode as an ideal, exponential diode with both junction and diffusion capacitances:

$$I_{dio} = I_s \left[\exp\left(-\frac{V_{DB}}{n\phi_T} \right) - 1 \right]$$

$$C_j = \frac{C_{j0}}{\sqrt{1 + \dfrac{V_{DB}}{V_{bi}}}}$$

$$C_{diff} = \frac{\tau I_s}{n\phi_T} \exp\left(-\frac{V_{DB}}{n\phi_T} \right)$$

where:

- I_{dio} is the current through the diode.
- I_s is the reverse saturation current.
- V_{DB} is the drain-body voltage.
- n is the ideality factor.
- ϕ_T is the thermal voltage.
- C_j is the junction capacitance of the diode.
- C_{j0} is the zero-bias junction capacitance.
- V_{bi} is the built-in voltage.
- C_{diff} is the diffusion capacitance of the diode.
- τ is the transit time.

The capacitances are defined through an explicit calculation of charges, which are then differentiated to give the capacitive expressions above. The block computes the capacitive diode currents as time derivatives of the relevant charges, similar to the computation in the surface-potential-based MOSFET model.

Modeling Temperature Dependence

The default behavior is that dependence on temperature is not modeled, and the device is simulated at the temperature for which you provide block parameters. To model the dependence on temperature during simulation, select `Model temperature dependence` for the **Parameterization** parameter on the **Temperature Dependence** tab.

Threshold-Based Model

For threshold-based variant, you can include modeling the dependence of the transistor static behavior on temperature during simulation. Temperature dependence of the junction capacitances is not modeled, this being a much smaller effect.

When including temperature dependence, the transistor defining equations remain the same. The gain, K, and the threshold voltage, V_{th}, become a function of temperature according to the following equations:

$$K_{Ts} = K_{Tm1}\left(\frac{T_s}{T_{m1}}\right)^{BEX}$$

$$V_{ths} = V_{th1} + \alpha \left(T_s - T_{m1}\right)$$

where:

- T_{m1} is the temperature at which the transistor parameters are specified, as defined by the **Measurement temperature** parameter value.

- T_s is the simulation temperature.

- K_{Tm1} is the transistor gain at the measurement temperature.

- K_{Ts} is the transistor gain at the simulation temperature. This is the transistor gain value used in the MOSFET equations when temperature dependence is modeled.

- V_{th1} is the threshold voltage at the measurement temperature.

- V_{ths} is the threshold voltage at the simulation temperature. This is the threshold voltage value used in the MOSFET equations when temperature dependence is modeled.

- BEX is the mobility temperature exponent. A typical value of BEX is -1.5.

- α is the gate threshold voltage temperature coefficient, dV_{th}/dT.

For most MOSFETS, you can use the default value of `-1.5` for *BEX*. Some datasheets quote the value for α, but most typically they provide the temperature dependence for drain-source on resistance, $R_{DS}(on)$. Depending on the block parameterization method, you have two ways of specifying α:

- If you parameterize the block from a datasheet, you have to provide $R_{DS}(on)$ at a second measurement temperature. The block then calculates the value for α based on this data.

- If you parameterize by specifying equation parameters, you have to provide the value for α directly.

If you have more data comprising drain current as a function of gate-source voltage for more than one temperature, then you can also use Simulink Design Optimization software to help tune the values for α and *BEX*.

Surface-Potential-Based Model

The surface-potential-based model includes temperature effects on the capacitance characteristics, as well as modeling the dependence of the transistor static behavior on temperature during simulation.

The **Measurement temperature** parameter on the **Main** tab specifies temperature T_{m1} at which the other device parameters have been extracted. The **Temperature Dependence** tab provides the simulation temperature, T_s, and the temperature-scaling coefficients for the other device parameters. For more information, see "Temperature Dependence Tab (Surface-Potential-Based Variant)" on page 1-403.

Thermal Port

The block has an optional thermal port, hidden by default. To expose the thermal port, right-click the block in your model, and select the appropriate block variant:

- For a model based on threshold voltage and with exposed thermal port, select **Simscape** > **Block choices** > **Threshold-based with thermal**.

- For a thermal variant of the model based on surface potential, select **Simscape** > **Block choices** > **Surface-potential-based with thermal**.

This action displays the thermal port H on the block icon, and adds the **Thermal Port** tab to the block dialog box.

Use the thermal port to simulate the effects of generated heat and device temperature. For more information on using thermal ports and on the **Thermal Port** tab parameters, see "Simulating Thermal Effects in Semiconductors".

Basic Assumptions and Limitations

When modeling temperature dependence for threshold-based block variant, consider the following:

- The block does not account for temperature-dependent effects on the junction capacitances.

- When you specify $R_{DS}(on)$ at a second measurement temperature, it must be quoted for the same working point (that is, the same drain current and gate-source voltage) as for the other $R_{DS}(on)$ value. Inconsistent values for $R_{DS}(on)$ at the higher temperature will result in unphysical values for α and unrepresentative simulation results. Typically $R_{DS}(on)$ increases by a factor of about 1.5 for a hundred degree increase in temperature.

- You may need to tune the values of *BEX* and threshold voltage, V_{th}, to replicate the I_{DS}–V_{GS} relationship (if available) for a given device. Increasing V_{th} moves the I_{DS}–V_{GS} plots to the right. The value of *BEX* affects whether the I_{DS}–V_{GS} curves for different temperatures cross each other, or not, for the ranges of V_{DS} and V_{GS} considered. Therefore, an inappropriate value can result in the different temperature curves appearing to be reordered. Quoting $R_{DS}(on)$ values for higher currents, preferably close to the current at which it will operate in your circuit, will reduce sensitivity to the precise value of *BEX*.

Parameters

- "Main Tab (Threshold-Based Variant)" on page 1-394
- "Main Tab (Surface-Potential-Based Variant)" on page 1-395
- "Ohmic Resistance Tab" on page 1-397
- "Junction Capacitance Tab" on page 1-397
- "Channel Capacitances Tab" on page 1-400
- "Body Diode Tab" on page 1-401
- "Temperature Dependence Tab (Threshold-Based Variant)" on page 1-401

- "Temperature Dependence Tab (Surface-Potential-Based Variant)" on page 1-403

Main Tab (Threshold-Based Variant)

This configuration of the **Main** tab corresponds to the threshold-based block variant, which is the default. If you are using the surface-potential-based variant of the block, see "Main Tab (Surface-Potential-Based Variant)" on page 1-395.

Parameterization

Select one of the following methods for block parameterization:

- `Specify from a datasheet` — Provide the drain-source on resistance and the corresponding drain current and gate-source voltage. The block calculates the transistor gain for the Shichman and Hodges equations from this information. This is the default method.

- `Specify using equation parameters directly` — Provide the transistor gain.

Drain-source on resistance, R_DS(on)

The ratio of the drain-source voltage to the drain current for specified values of drain current and gate-source voltage. $R_{DS}(on)$ should have a positive value. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is 0.025 Ω.

Drain current, Ids, for R_DS(on)

The drain current the block uses to calculate the value of the drain-source resistance. I_{DS} should have a positive value. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is 6 A.

Gate-source voltage, Vgs, for R_DS(on)

The gate-source voltage the block uses to calculate the value of the drain-source resistance. V_{GS} should have a positive value. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is 10 V.

Gain, K

Positive constant gain coefficient for the Shichman and Hodges equations. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter. The default value is 5 A/V^2.

Gate-source threshold voltage, Vth

Gate-source threshold voltage V_{th} in the Shichman and Hodges equations. For an enhancement device, V_{th} should be positive. For a depletion mode device, V_{th} should be negative. The default value is `1.7` V.

Channel modulation, L

The channel-length modulation, usually denoted by the mathematical symbol λ. When in the saturated region, it is the rate of change of drain current with drain-source voltage. The effect on drain current is typically small, and the effect is neglected if calculating transistor gain K from drain-source on-resistance, $R_{DS}(on)$. A typical value is 0.02, but the effect can be ignored in most circuit simulations. However, in some circuits a small nonzero value may help numerical convergence. The default value is `0` 1/V.

Measurement temperature

Temperature T_{m1} at which **Drain-source on resistance, R_DS(on)** is measured. The default value is `25` °C.

Main Tab (Surface-Potential-Based Variant)

This configuration of the **Main** tab corresponds to the surface-potential-based block variant. If you are using the threshold-based variant of the block, based on the Shichman and Hodges equations, see "Main Tab (Threshold-Based Variant)" on page 1-394.

Gain

The MOSFET gain, β. This parameter primarily defines the linear region of operation on an I_D–V_{DS} characteristic. The value must be greater than 0. The default value is `18` A/V^2.

Flatband voltage

The flatband voltage, V_{FB}, defines the gate bias that must be applied in order to achieve the flatband condition at the surface of the silicon. The default value is `-1.1` V. You can also use this parameter to arbitrarily shift the threshold voltage due to material work function differences, and to trapped interface or oxide charges. In practice, however, it is usually recommended to modify the threshold voltage by using the **Body factor** and **Surface potential at strong inversion** parameters first, and only use this parameter for fine-tuning.

Body factor

Body factor, γ, in the surface-potential equation. This parameter primarily impacts the threshold voltage. The default value is `3.5` V$^{1/2}$.

Surface potential at strong inversion

The $2\phi_B$ term in the surface-potential equation. This parameter also primarily impacts the threshold voltage. The default value is 1 V.

Velocity saturation factor

Velocity saturation, θ_{sat}, in the drain-current equation. Use this parameter in cases where a good fit to linear operation leads to a saturation current that is too high. By increasing this parameter value, you reduce the saturation current. For high-voltage devices, it is often the case that a good fit to linear operation leads to a saturation current that is too low. In such a case, either increase both the gain and the drain ohmic resistance or use an N-Channel LDMOS FET block instead. The default value is 0.4 1/V.

Channel-length modulation factor

The factor, α, multiplying the logarithmic term in the $G_{\Delta L}$ equation. This parameter describes the onset of channel-length modulation. For device characteristics that exhibit a positive conductance in saturation, increase the parameter value to fit this behavior. The default value is 0, which means that channel-length modulation is off by default.

Channel-length modulation voltage

The voltage V_p in the $G_{\Delta L}$ equation. This parameter controls the drain-voltage at which channel-length modulation starts to become active. The default value is 50 mV.

Surface roughness scattering factor

Indicates the strength of the mobility reduction. The mobility is $\mu = \mu_0/G_{mob}$, where μ_0 is the low-field mobility without the effect of surface scattering. The mobility

reduction factor, G_{mob}, is given by $G_{mob} = \sqrt{1 + \left(\theta_{sr} V_{eff}\right)^4}$, where θ_{sr} is the surface roughness scattering factor and V_{eff} is a voltage that is indicative of the effective vertical electric field strength in the channel, E_{eff}. For high vertical electric fields, the mobility is roughly proportional to E_{eff}^2 for electrons. The default parameter value is 0 1/V.

Linear-to-saturation transition coefficient

This parameter controls how smoothly the MOSFET transitions from linear into saturation, particularly when velocity saturation is enabled. This parameter can usually be left at its default value, but you can use it to fine-tune the knee of the I_D–V_{DS} characteristic. The expected range for this parameter value is between 2 and 8. The default value is 8.

Measurement temperature

Temperature T_{m1} at which the block parameters are measured. If the **Device simulation temperature** parameter on the **Temperature Dependence** tab differs from this value, then device parameters will be scaled from their defined values according to the simulation and reference temperatures. For more information, see "Temperature Dependence Tab (Surface-Potential-Based Variant)" on page 1-403. The default value is 25 °C.

Ohmic Resistance Tab

Source ohmic resistance

The transistor source resistance, that is, the series resistance associated with the source contact. The value must be greater than or equal to `0`. The default value for threshold-based variants is `1e-4` Ω. The default value for surface-potential-based variants is `2e-3` Ω.

Drain ohmic resistance

The transistor drain resistance, that is, the series resistance associated with the drain contact. The value must be greater than or equal to `0`. The default value for threshold-based variants is `0.01` Ω. The default value for surface-potential-based variants is `0.17` Ω.

Gate ohmic resistance

The transistor gate resistance, that is, the series resistance associated with the gate contact. This parameter is visible only for the surface-potential-based block variants. The value must be greater than or equal to `0`. The default value is `8.4` Ω.

Junction Capacitance Tab

This tab is visible only for the threshold-based variant of the block.

Parameterization

Select one of the following methods for capacitance parameterization:

- `Specify fixed input, reverse transfer and output capacitance` — Provide fixed parameter values from datasheet and let the block convert the input, output, and reverse transfer capacitance values to junction capacitance values, as described in "Charge Model for Threshold-Based Variant" on page 1-384. This is the default method.

- Specify fixed gate-source, gate-drain and drain-source capacitance — Provide fixed values for junction capacitance parameters directly.

- Specify tabulated input, reverse transfer and output capacitance — Provide tabulated capacitance and drain-source voltage values based on datasheet plots. The block converts the input, output, and reverse transfer capacitance values to junction capacitance values, as described in "Charge Model for Threshold-Based Variant" on page 1-384.

- Specify tabulated gate-source, gate-drain and drain-source capacitance — Provide tabulated values for junction capacitances and drain-source voltage.

Input capacitance, Ciss

The gate-source capacitance with the drain shorted to the source. This parameter is visible only for the following two values for the **Parameterization** parameter:

- If you select Specify fixed input, reverse transfer and output capacitance, the default value is 350 pF.

- If you select Specify tabulated input, reverse transfer and output capacitance, the default value is [720 700 590 470 390 310] pF.

Reverse transfer capacitance, Crss

The drain-gate capacitance with the source connected to ground, also known as the Miller capacitance. This parameter is visible only for the following two values for the **Parameterization** parameter:

- If you select Specify fixed input, reverse transfer and output capacitance, the default value is 80 pF.

- If you select Specify tabulated input, reverse transfer and output capacitance, the default value is [450 400 300 190 95 55] pF.

Output capacitance, Coss

The drain-source capacitance with the gate and source shorted. This parameter is visible only for the following two values for the **Parameterization** parameter:

- If you select Specify fixed input, reverse transfer and output capacitance, the default value is 0 pF.

- If you select Specify tabulated input, reverse transfer and output capacitance, the default value is [900 810 690 420 270 170] pF.

Gate-source junction capacitance

The value of the capacitance placed between the gate and the source. This parameter is visible only for the following two values for the **Parameterization** parameter:

- If you select `Specify fixed gate-source, gate-drain and drain-source capacitance`, the default value is 270 pF.

- If you select `Specify tabulated gate-source, gate-drain and drain-source capacitance`, the default value is [270 300 290 280 295 255] pF.

Gate-drain junction capacitance

The value of the capacitance placed between the gate and the drain. This parameter is visible only for the following two values for the **Parameterization** parameter:

- If you select `Specify fixed gate-source, gate-drain and drain-source capacitance`, the default value is 80 pF.

- If you select `Specify tabulated gate-source, gate-drain and drain-source capacitance`, the default value is [450 400 300 190 95 55] pF.

Drain-source junction capacitance

The value of the capacitance placed between the drain and the source. This parameter is visible only for the following two values for the **Parameterization** parameter:

- If you select `Specify fixed gate-source, gate-drain and drain-source capacitance`, the default value is 0 pF.

- If you select `Specify tabulated gate-source, gate-drain and drain-source capacitance`, the default value is [450 410 390 230 175 115] pF.

Corresponding drain-source voltages

The drain-source voltages corresponding to the tabulated capacitance values. This parameter is visible only for tabulated capacitance models (`Specify tabulated input, reverse transfer and output capacitance` or `Specify tabulated gate-source, gate-drain and output capacitance`). The default value is [0.1 0.3 1 3 10 30] V.

Gate-source voltage, Vgs, for tabulated capacitances

For tabulated capacitance models, this parameter controls the voltage dependence of the **Reverse transfer capacitance, Crss** or the **Gate-drain junction capacitance** parameter (depending on the selected parameterization option). These capacitances are a function of the drain-gate voltage. The block calculates drain-gate voltages by

subtracting this gate-source voltage value from the values specified for the **Corresponding drain-source voltages** parameter. The default value is `0` V. This parameter is visible only for tabulated capacitance models (`Specify tabulated input, reverse transfer and output capacitance` or `Specify tabulated gate-source, gate-drain and output capacitance`).

Charge-voltage linearity

The two fixed capacitance options (`Specify fixed input, reverse transfer and output capacitance` or `Specify fixed gate-source, gate-drain and drain-source capacitance`) let you model gate junction capacitance as a fixed gate-source capacitance C_{GS} and either a fixed or a nonlinear gate-drain capacitance C_{GD}. Select whether the gate-drain capacitance is fixed or nonlinear:

- `Gate-drain capacitance is constant` — The capacitance value is constant and defined according to the selected parameterization option, either directly or derived from a datasheet. This is the default method.

- `Gate-drain charge function is nonlinear` — The gate-drain charge relationship is defined according to the piecewise-nonlinear function described in "Charge Model for Threshold-Based Variant" on page 1-384. Two additional parameters appear to let you define the gate-drain charge function.

Gate-drain oxide capacitance

The gate-drain capacitance when the drain-gate voltage is less than the **Drain-gate voltage at which oxide capacitance becomes active** parameter value. This parameter is visible only when you select `Gate-drain charge function is nonlinear` for the **Charge-voltage linearity** parameter. The default value is `200` pF.

Drain-gate voltage at which oxide capacitance becomes active

The drain-gate voltage at which the drain-gate capacitance switches between off-state (C_{GD}) and on-state (C_{ox}) capacitance values. This parameter is only visible when you select `Gate-drain charge function is nonlinear` for the **Charge-voltage linearity** parameter. The default value is `-0.5` V.

Channel Capacitances Tab

This tab is visible only for the surface-potential-based variant of the block.

Oxide capacitance

The parallel plate gate-channel capacitance. The default value is `1500` pF.

Gate-source overlap capacitance

The fixed, linear capacitance associated with the overlap of the gate electrode with the source well. The default value is 100 pF.

Gate-drain overlap capacitance

The fixed, linear capacitance associated with the overlap of the gate electrode with the drain well. The default value is 14 pF.

Body Diode Tab

Reverse saturation current

The current designated by the I_s symbol in the body-diode equations. The default value for threshold-based variant is 0 A. The default value for surface-potential-based variant is 5.2e-13 A.

Built-in voltage

The built-in voltage of the diode, designated by the V_{bi} symbol in the body-diode equations. Built-in voltage has an impact only on the junction capacitance equation. It does not affect the conduction current. The default value is 0.6 V.

Ideality factor

The factor designated by the n symbol in the body-diode equations. The default value is 1.

Zero-bias junction capacitance

The capacitance between the drain and bulk contacts at zero-bias due to the body diode alone. It is designated by the C_{j0} symbol in the body-diode equations. The default value for threshold-based variant is 0 pF. The default value for surface-potential-based variant is 480 pF.

Transit time

The time designated by the τ symbol in the body-diode equations. The default value is 50 ns.

Temperature Dependence Tab (Threshold-Based Variant)

This configuration of the **Temperature Dependence** tab corresponds to the threshold-based block variant, which is the default. If you are using the surface-potential-based variant of the block, see "Temperature Dependence Tab (Surface-Potential-Based Variant)" on page 1-403

Parameterization

Select one of the following methods for temperature dependence parameterization:

- `None — Simulate at parameter measurement temperature` — Temperature dependence is not modeled. This is the default method.

- `Model temperature dependence` — Model temperature-dependent effects. Provide a value for simulation temperature, T_s, a value for *BEX*, and a value for the measurement temperature T_{m1} (using the **Measurement temperature** parameter on the **Main** tab). You also have to provide a value for α using one of two methods, depending on the value of the **Parameterization** parameter on the **Main** tab. If you parameterize the block from a datasheet, you have to provide $R_{DS}(on)$ at a second measurement temperature, and the block will calculate α based on that. If you parameterize by specifying equation parameters, you have to provide the value for α directly.

Drain-source on resistance, R_DS(on), at second measurement temperature

The ratio of the drain-source voltage to the drain current for specified values of drain current and gate-source voltage at second measurement temperature. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter on the **Main** tab. It must be quoted for the same working point (drain current and gate-source voltage) as the **Drain-source on resistance, R_DS(on)** parameter on the **Main** tab. The default value is `0.037` Ω.

Second measurement temperature

Second temperature T_{m2} at which **Drain-source on resistance, R_DS(on), at second measurement temperature** is measured. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter on the **Main** tab. The default value is `125` °C.

Gate threshold voltage temperature coefficient, dVth/dT

The rate of change of gate threshold voltage with temperature. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter on the **Main** tab. The default value is `-6` mV/K.

Mobility temperature exponent, BEX

Mobility temperature coefficient value. You can use the default value for most MOSFETs. See the "Basic Assumptions and Limitations" on page 1-393 section for additional considerations. The default value is `-1.5`.

Body diode reverse saturation current temperature exponent

The reverse saturation current for the body diode is assumed to be proportional to the square of the intrinsic carrier concentration, $n_i = N_C \exp(-E_G/2k_B T)$. N_C is the

temperature-dependent effective density of states and E_G is the temperature-dependent bandgap for the semiconductor material. To avoid introducing another temperature-scaling parameter, the block neglects the temperature dependence of the bandgap and uses the bandgap of silicon at 300K (1.12eV) for all device types. Therefore, the temperature-scaled reverse saturation current is given by

$$I_s = I_{s,m1}\left(\frac{T_s}{T_{m1}}\right)^{\eta_{Is}} \cdot \exp\left(\frac{E_G}{k_B}\cdot\left(\frac{1}{T_{m1}} - \frac{1}{T_s}\right)\right).$$

$I_{s,m1}$ is the value of the **Reverse saturation current** parameter from the **Body Diode** tab, k_B is Boltzmann's constant (8.617x10-5eV/K), and η_{Is} is the **Body diode reverse saturation current temperature exponent**. The default value is 3, because N_C for silicon is roughly proportional to $T^{3/2}$. You can remedy the effect of neglecting the temperature-dependence of the bandgap by a pragmatic choice of η_{Is}.

Device simulation temperature

Temperature T_s at which the device is simulated. The default value is 25 °C.

Temperature Dependence Tab (Surface-Potential-Based Variant)

This configuration of the **Temperature Dependence** tab corresponds to the surface-potential-based block variant. If you are using the threshold-based variant of the block, see "Temperature Dependence Tab (Threshold-Based Variant)" on page 1-401

Parameterization

Select one of the following methods for temperature dependence parameterization:

- `None — Simulate at parameter measurement temperature` — Temperature dependence is not modeled. This is the default method.

- `Model temperature dependence` — Model temperature-dependent effects. Provide a value for the device simulation temperature, T_s, and the temperature-scaling coefficients for other block parameters.

Gain temperature exponent

The MOSFET gain, β, is assumed to scale exponentially with temperature, $\beta = \beta_{m1}(T_{m1}/T_s)^{\wedge}\eta_\beta$. β_{m1} is the value of the **Gain** parameter from the **Main** tab and η_β is the **Gain temperature exponent**. The default value is `1.3`.

Flatband voltage temperature coefficient

The flatband voltage, V_{FB}, is assumed to scale linearly with temperature, $V_{FB} = V_{FBm1} + (T_s - T_{m1})S_{T,V_{FB}}$. V_{FBm1} is the value of the **Flatband voltage** parameter from the **Main** tab and $S_{T,V_{FB}}$ is the **Flatband voltage temperature coefficient**. The default value is `5e-4` V/K.

Surface potential at strong inversion temperature coefficient

The surface potential at strong inversion, $2\phi_B$, is assumed to scale linearly with temperature, $2\phi_B = 2\phi_{Bm1} + (T_s - T_{m1})S_{T,\phi_B}$. $2\phi_{Bm1}$ is the value of the **Surface potential at strong inversion** parameter from the **Main** tab and S_{T,ϕ_B} is the **Surface potential at strong inversion temperature coefficient**. The default value is `-8.5e-4` V/K.

Velocity saturation temperature exponent

The velocity saturation, θ_{sat}, is assumed to scale exponentially with temperature, $\theta_{sat} = \theta_{sat,m1}(T_{m1}/T_s)^{\wedge}\eta_\theta$. $\theta_{sat,m1}$ is the value of the **Velocity saturation factor** parameter from the **Main** tab and η_θ is the **Velocity saturation temperature exponent**. The default value is `1.04`.

Surface roughness scattering temperature exponent

This parameter leads to a temperature-dependent reduction in the MOSFET transconductance at high gate voltage. The surface roughness scattering, θ_{sr}, is assumed to scale exponentially with temperature, $\theta_{sr} = \theta_{sr,m1}(T_{m1}/T_s)^{\wedge}\eta_{sr}$. $\theta_{sr,m1}$ is the value of the **Surface roughness scattering factor** parameter from the **Main** tab and η_{sr} is the **Surface roughness scattering temperature exponent**. The default value is `0.65`.

Resistance temperature exponent

The series resistances are assumed to correspond to semiconductor resistances. Therefore, they decrease exponentially with increasing temperature. $R_i = R_{i,m1}(T_{m1}/T_s)^{\wedge}\eta_R$, where i is S, D, or G, for the source, drain, or gate series resistance, respectively. $R_{i,m1}$ is the value of the corresponding series resistance parameter from the **Ohmic Resistance** tab and η_R is the **Resistance temperature exponent**. The default value is `0.95`.

Body diode reverse saturation current temperature exponent

The reverse saturation current for the body diode is assumed to be proportional to the square of the intrinsic carrier concentration, $n_i = N_C \exp(-E_G/2k_B T)$. N_C is the temperature-dependent effective density of states and E_G is the temperature-dependent bandgap for the semiconductor material. To avoid introducing another temperature-scaling parameter, the block neglects the temperature dependence of

the bandgap and uses the bandgap of silicon at 300K (1.12eV) for all device types. Therefore, the temperature-scaled reverse saturation current is given by

$$I_s = I_{s,m1} \left(\frac{T_s}{T_{m1}} \right)^{\eta_{Is}} \cdot \exp\left(\frac{E_G}{k_B} \cdot \left(\frac{1}{T_{m1}} - \frac{1}{T_s} \right) \right).$$

$I_{s,m1}$ is the value of the **Reverse saturation current** parameter from the **Body Diode** tab, k_B is Boltzmann's constant (8.617x10-5eV/K), and η_{Is} is the **Body diode reverse saturation current temperature exponent**. The default value is 3, because N_C for silicon is roughly proportional to $T^{3/2}$. You can remedy the effect of neglecting the temperature-dependence of the bandgap by a pragmatic choice of η_{Is}.

Device simulation temperature

Temperature T_s at which the device is simulated. The default value is 25 °C.

Ports

The block has the following ports:

G

Electrical conserving port associated with the transistor gate terminal

D

Electrical conserving port associated with the transistor drain terminal

S

Electrical conserving port associated with the transistor source terminal

References

[1] Shichman, H. and D. A. Hodges. "Modeling and simulation of insulated-gate field-effect transistor switching circuits." *IEEE J. Solid State Circuits*. SC-3, 1968.

[2] Van Langevelde, R., A. J. Scholten, and D. B .M. Klaassen. "Physical Background of MOS Model 11. Level 1101." *Nat.Lab. Unclassified Report 2003/00239*. April 2003.

[3] Oh, S-Y., D. E. Ward, and R. W. Dutton. "Transient analysis of MOS transistors." *IEEE J. Solid State Circuits*. SC-15, pp. 636-643, 1980.

See Also

P-Channel MOSFET

Topics

Interactive Generation of MOSFET Characteristics

Negative Supply Rail

Model ideal negative supply rail

Library

Sources

Description

The Negative Supply Rail block represents an ideal negative supply rail. Use this block instead of the Simscape DC Voltage Source block to define the output voltage relative to the Simscape Electrical Reference block that must appear in each model.

Note Do not attach more than one Negative Supply Rail block to any connected line.

Parameters

Constant voltage

The voltage at the output port relative to the Electrical Reference block ground port. The default value is -1 V.

Ports

The block has the following ports:

-

Negative electrical voltage

See Also

DC Voltage Source | Positive Supply Rail

Nonlinear Inductor

Model inductor with nonideal core

Library

Passive Devices

Description

The Nonlinear Inductor block represents an inductor with a nonideal core. A core may be nonideal due to its magnetic properties and dimensions. The block provides the following parameterization options:

- Single inductance (linear) on page 1-409
- Single saturation point on page 1-410
- Magnetic flux versus current characteristic on page 1-411
- Magnetic flux density versus magnetic field strength characteristic on page 1-412
- Magnetic flux density versus magnetic field strength characteristic with hysteresis on page 1-413

Single Inductance (Linear)

The relationships between voltage, current and flux are defined by the following equations:

$$i = i_L + v G_p$$

$$v = N_w \frac{d\Phi}{dt}$$

$$\Phi = \frac{L}{N_w} i_L$$

where:

- v is the terminal voltage.
- i is the terminal current.
- i_L is the current through inductor.
- G_p is the parasitic parallel conductance.
- N_w is the number of winding turns.
- Φ is the magnetic flux.
- L is the unsaturated inductance.

Single Saturation Point

The relationships between voltage, current and flux are defined by the following equations:

$$i = i_L + v G_p$$

$$v = N_w \frac{d\Phi}{dt}$$

$$\Phi = \frac{L}{N_w} i_L \text{ (for unsaturated)}$$

$$\Phi = \frac{L_{sat}}{N_w} i_L \pm \Phi_{offset} \text{ (for saturated)}$$

where:

- v is the terminal voltage.
- i is the terminal current.
- i_L is the current through inductor.

- G_p is the parasitic parallel conductance.

- N_w is the number of winding turns.

- Φ is the magnetic flux.

- Φ_{offset} is the magnetic flux saturation offset.

- L is the unsaturated inductance.

- L_{sat} is the saturated inductance.

Magnetic Flux Versus Current Characteristic

The relationships between voltage, current and flux are defined by the following equations:

$$i = i_L + v G_p$$

$$v = N_w \frac{d\Phi}{dt}$$

$$\Phi = f(i_L)$$

where:

- v is the terminal voltage.

- i is the terminal current.

- i_L is the current through inductor.

- G_p is the parasitic parallel conductance.

- N_w is the number of winding turns.

- Φ is the magnetic flux.

Magnetic flux is determined by one-dimensional table lookup, based on the vector of current values and the vector of corresponding magnetic flux values that you provide. You can construct these vectors using either negative and positive data, or positive data only. If using positive data only, the vector must start at 0, and the negative data will be automatically calculated by rotation about (0,0).

Magnetic Flux Density Versus Magnetic Field Strength Characteristic

The relationships between voltage, current and flux are defined by the following equations:

$$i = i_L + vG_p$$

$$v = N_w \frac{d\Phi}{dt}$$

$$\Phi = B \cdot A_e$$

$$B = f(H)$$

$$H = \frac{N_w}{l_e} i_L$$

where:

- v is the terminal voltage.
- i is the terminal current.
- i_L is the current through inductor.
- G_p is the parasitic parallel conductance.
- N_w is the number of winding turns.
- Φ is the magnetic flux.
- B is the magnetic flux density.
- H is the magnetic field strength.
- l_e is the effective core length.
- A_e is the effective core cross-sectional area.

Magnetic flux density is determined by one-dimensional table lookup, based on the vector of magnetic field strength values and the vector of corresponding magnetic flux density values that you provide. You can construct these vectors using either negative and

positive data, or positive data only. If using positive data only, the vector must start at 0, and the negative data will be automatically calculated by rotation about (0,0).

Magnetic Flux Density Versus Magnetic Field Strength Characteristic with Hysteresis

The relationships between voltage, current and flux are defined by the following equations:

$$i = i_L + v G_p$$

$$v = N_w \frac{d\Phi}{dt}$$

$$\Phi = B \cdot A_e$$

$$B = \mu_0 \left(H + M \right)$$

$$H = \frac{N_w}{l_e} i_L$$

where:

- v is the terminal voltage.
- i is the terminal current.
- i_L is the current through inductor.
- G_p is the parasitic parallel conductance.
- N_w is the number of winding turns.
- Φ is the magnetic flux.
- B is the magnetic flux density.
- μ_0 is the magnetic constant, permeability of free space.
- H is the magnetic field strength.
- M is the magnetization of the inductor core.
- l_e is the effective core length.

- A_e is the effective core cross-sectional area.

The magnetization acts to increase the magnetic flux density, and its value depends on both the current value and the history of the field strength H. The Jiles-Atherton [1 on page 1-422, 2 on page 1-422] equations are used to determine M at any given time. The figure below shows a typical plot of the resulting relationship between B and H.

In this case the magnetization starts as zero, and hence the plot starts at $B = H = 0$. As the field strength increases, the plot tends to the positive-going hysteresis curve; then on reversal the rate of change of H, it follows the negative-going hysteresis curve. The difference between positive-going and negative-going curves is due to the dependence of M on the trajectory history. Physically the behavior corresponds to magnetic dipoles in the core aligning as the field strength increases, but not then fully recovering to their original position as field strength decreases.

The starting point for the Jiles-Atherton equation is to split the magnetization effect into two parts, one that is purely a function of effective field strength (H_{eff}) and the other an irreversible part that depends on past history:

$$M = cM_{an} + (1-c)M_{irr}$$

The M_{an} term is called the anhysteretic magnetization because it exhibits no hysteresis. It is described by the following function of the current value of the effective field strength, H_{eff}:

$$M_{an} = M_s \left(\coth\left(\frac{H_{eff}}{\alpha} \right) - \frac{\alpha}{H_{eff}} \right)$$

This function defines a saturation curve with limiting values $\pm M_s$ and point of saturation determined by the value of α, the anhysteretic shape factor. It can be approximately thought of as describing the average of the two hysteretic curves. In the Nonlinear

Inductor block, you provide values for dM_{an} / dH_{eff} when $H_{eff} = 0$ and a point $[H_1, B_1]$ on the anhysteretic B-H curve, and these are used to determine values for α and M_s.

The parameter c is the coefficient for reversible magnetization, and dictates how much of the behavior is defined by M_{an} and how much by the irreversible term M_{irr}. The Jiles-Atherton model defines the irreversible term by a partial derivative with respect to field strength:

$$\frac{dM_{irr}}{dH} = \frac{M_{an} - M_{irr}}{K\delta - \alpha(M_{an} - M_{irr})}$$
$$\delta = \begin{cases} 1 & \text{if } H \geq 0 \\ -1 & \text{if } H < 0 \end{cases}$$

Comparison of this equation with a standard first order differential equation reveals that as increments in field strength, H, are made, the irreversible term M_{irr} attempts to track

the reversible term M_{an}, but with a variable tracking gain of $1 / \left(K\delta - \alpha(M_{an} - M_{irr}) \right)$. The tracking error acts to create the hysteresis at the points where δ changes sign. The main parameter that shapes the irreversible characteristic is K, which is called the bulk coupling coefficient. The parameter α is called the inter-domain coupling factor, and is also used to define the effective field strength used when defining the anhysteretic curve:

$$H_{eff} = H + \alpha M$$

The value of α affects the shape of the hysteresis curve, larger values acting to increase

the B-axis intercepts. However, notice that for stability the term $K\delta - \alpha(M_{an} - M_{irr})$

1-415

must be positive for $\delta > 0$ and negative for $\delta < 0$. Therefore not all values of α are permissible, a typical maximum value being of the order 1e-3.

Procedure for Finding Approximate Values for Jiles-Atherton Equation Coefficients

You can determine representative parameters for the equation coefficients by using the following procedure:

1 Provide a value for the **Anhysteretic B-H gradient when H is zero** parameter

(dM_{an} / dH_{eff} when $H_{eff} = 0$) plus a data point $[H_1, B_1]$ on the anhysteretic B-H curve. From these values, the block initialization determines values for α and M_s.

2 Set the **Coefficient for reversible magnetization, c** parameter to achieve correct initial B-H gradient when starting a simulation from [H B] = [0 0]. The value of c is approximately the ratio of this initial gradient to the **Anhysteretic B-H gradient when H is zero**. The value of c must be greater than 0 and less than 1.

3 Set the **Bulk coupling coefficient, K** parameter to the approximate magnitude of H when $B = 0$ on the positive-going hysteresis curve.

4 Start with α very small, and gradually increase to tune the value of B when crossing $H = 0$ line. A typical value is in the range of 1e-4 to 1e-3. Values that are too large will cause the gradient of the B-H curve to tend to infinity, which is nonphysical and generates a run-time assertion error.

Sometimes you need to iterate on these four steps to get a good match against a predefined B-H curve.

Parameters

Main Tab

Parameterized by

Select one of the following methods for block parameterization:

- `Single inductance (linear)` — Provide the values for number of turns, unsaturated inductance, and parasitic parallel conductance.

- `Single saturation point` — Provide the values for number of turns, unsaturated and saturated inductances, saturation magnetic flux, and parasitic parallel conductance. This is the default option.

- `Magnetic flux versus current characteristic` — In addition to the number of turns and the parasitic parallel conductance value, provide the current vector and the magnetic flux vector, to populate the magnetic flux versus current lookup table.

- `Magnetic flux density versus magnetic field strength characteristic` — In addition to the number of turns and the parasitic parallel conductance value, provide the values for effective core length and cross-sectional area, as well as the magnetic field strength vector and the magnetic flux density vector, to populate the magnetic flux density versus magnetic field strength lookup table.

- `Magnetic flux density versus magnetic field strength characteristic with hysteresis` — In addition to the number of turns and the effective core length and cross-sectional area, provide the values for the initial anhysteretic B-H curve gradient, the magnetic flux density and field strength at a certain point on the B-H curve, as well as the coefficient for the reversible magnetization, bulk coupling coefficient, and inter-domain coupling factor, to define magnetic flux density as a function of both the current value and the history of the field strength.

Number of turns

The total number of turns of wire wound around the inductor core. The default value is `10`.

Unsaturated inductance

The value of inductance used when the inductor is operating in its linear region. This parameter is visible only when you select `Single inductance (linear)` or `Single saturation point` for the **Parameterized by** parameter. The default value is `2e-4` H.

Saturated inductance

The value of inductance used when the inductor is operating beyond its saturation point. This parameter is visible only when you select `Single saturation point` for the **Parameterized by** parameter. The default value is `1e-4` H.

Saturation magnetic flux

The value of magnetic flux at which the inductor saturates. This parameter is visible only when you select `Single saturation point` for the **Parameterized by** parameter. The default value is `1.3e-5` Wb.

Current, i

The current data used to populate the magnetic flux versus current lookup table. This parameter is visible only when you select `Magnetic flux versus current characteristic` for the **Parameterized by** parameter. The default value is `[ 0 0.64 1.28 1.92 2.56 3.20 ]` A.

Magnetic flux vector, phi

The magnetic flux data used to populate the magnetic flux versus current lookup table. This parameter is visible only when you select `Magnetic flux versus current characteristic` for the **Parameterized by** parameter. The default value is `[0 1.29 2.00 2.27 2.36 2.39 ].*1e-5` Wb.

Magnetic field strength vector, H

The magnetic field strength data used to populate the magnetic flux density versus magnetic field strength lookup table. This parameter is visible only when you select `Magnetic flux density versus magnetic field strength characteristic` for the **Parameterized by** parameter. The default value is `[ 0 200 400 600 800 1000 ]` A/m.

Magnetic flux density vector, B

The magnetic flux density data used to populate the magnetic flux density versus magnetic field strength lookup table. This parameter is visible only when you select `Magnetic flux density versus magnetic field strength characteristic` for the **Parameterized by** parameter. The default value is `[ 0 0.81 1.25 1.42 1.48 1.49 ]` T.

Effective length

The effective core length, that is, the average distance of the magnetic path. This parameter is visible only when you select `Magnetic flux density versus magnetic field strength characteristic` or `Magnetic flux density versus magnetic field strength characteristic with hysteresis` for the **Parameterized by** parameter. The default value is `0.032` m.

Effective cross-sectional area

The effective core cross-sectional area, that is, the average area of the magnetic path. This parameter is visible only when you select `Magnetic flux density versus magnetic field strength characteristic` or `Magnetic flux density`

versus magnetic field strength characteristic with hysteresis for the **Parameterized by** parameter. The default value is `1.6e-5 m^2`.

Anhysteretic B-H gradient when H is zero

The gradient of the anhysteretic (no hysteresis) B-H curve around zero field strength. Set it to the average gradient of the positive-going and negative-going hysteresis curves. This parameter is visible only when you select `Magnetic flux density versus magnetic field strength characteristic with hysteresis` for the **Parameterized by** parameter. The default value is `0.005 m*T/A`.

Flux density point on anhysteretic B-H curve

Specify a point on the anhysteretic curve by providing its flux density value. Picking a point at high field strength where the positive-going and negative-going hysteresis curves align is the most accurate option. This parameter is visible only when you select `Magnetic flux density versus magnetic field strength characteristic with hysteresis` for the **Parameterized by** parameter. The default value is `1.49 T`.

Corresponding field strength

The corresponding field strength for the point that you define by the **Flux density point on anhysteretic B-H curve** parameter. This parameter is visible only when you select `Magnetic flux density versus magnetic field strength characteristic with hysteresis` for the **Parameterized by** parameter. The default value is `1000 A/m`.

Coefficient for reversible magnetization, c

The proportion of the magnetization that is reversible. The value should be greater than zero and less than one. This parameter is visible only when you select `Magnetic flux density versus magnetic field strength characteristic with hysteresis` for the **Parameterized by** parameter. The default value is `0.1`.

Bulk coupling coefficient, K

The Jiles-Atherton parameter that primarily controls the field strength magnitude at which the B-H curve crosses the zero flux density line. This parameter is visible only when you select `Magnetic flux density versus magnetic field strength characteristic with hysteresis` for the **Parameterized by** parameter. The default value is `200 A/m`.

Inter-domain coupling factor, alpha

The Jiles-Atherton parameter that primarily affects the points at which the B-H curves intersect the zero field strength line. Typical values are in the range of 1e-4 to 1e-3. This parameter is visible only when you select `Magnetic flux density versus`

1-419

`magnetic field strength characteristic with hysteresis` for the **Parameterized by** parameter. The default value is `1e-4`.

Averaging period for power logging

Averaging period for the hysteresis losses calculation. These losses are proportional to the area enclosed by the B-H trajectory. If the block is excited at a known, fixed frequency, you can set this value to the corresponding excitation period to calculate the hysteresis loss. In this case, the block logs the hysteresis loss once per AC cycle to the variable `power_dissipated`. If you are using a fixed-step solver, this value must be an integer multiple of the simulation step size.

If the block is not excited at a known, fixed frequency, set this parameter to `0`. In this case, the block sets `power_dissipated` to zero, and you can calculate the actual hysteresis loss by post-processing the logged variable `power_instantaneous`.

The default value is `0` s. This parameter is visible only when you select `Magnetic flux density versus magnetic field strength characteristic with hysteresis` for the **Parameterized by** parameter.

Parasitic parallel conductance

Use this parameter to represent small parasitic effects. A small parallel conductance may be required for the simulation of some circuit topologies. The default value is `1e-9` 1/Ω.

Interpolation option

The lookup table interpolation option. This parameter is visible only when you select `Magnetic flux versus current characteristic` or `Magnetic flux density versus magnetic field strength characteristic` for the **Parameterized by** parameter. Select one of the following interpolation methods:

- `Linear` — Select this option to get the best performance.
- `Smooth` — Select this option to produce a continuous curve with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Initial Conditions Tab

Specify initial state by

Select the appropriate initial state specification option:

- `Current` — Specify the initial state of the inductor by the initial current through the inductor (i_L). This is the default option.
- `Magnetic flux` — Specify the initial state of the inductor by the magnetic flux.

This parameter is not visible when you select `Magnetic flux density versus magnetic field strength characteristic with hysteresis` for the **Parameterized by** parameter on the **Main** tab.

Initial current

The initial current value used to calculate the value of magnetic flux at time zero. This is the current passing through the inductor. Component current consists of current passing through the inductor and current passing through the parasitic parallel conductance. This parameter is visible only when you select `Current` for the **Specify initial state by** parameter. The default value is `0` A.

Initial magnetic flux

The value of magnetic flux at time zero. This parameter is visible only when you select `Magnetic flux` for the **Specify initial state by** parameter. The default is `0` Wb.

Initial magnetic flux density

The value of magnetic flux density at time zero. This parameter is visible only when you select `Magnetic flux density versus magnetic field strength characteristic with hysteresis` for the **Parameterized by** parameter on the **Main** tab. The default is `0` T.

Initial field strength

The value of magnetic field strength at time zero. This parameter is visible only when you select `Magnetic flux density versus magnetic field strength characteristic with hysteresis` for the **Parameterized by** parameter on the **Main** tab. The default is `0` A/m.

Ports

The block has the following ports:

+

   Positive electrical port

-

   Negative electrical port

Examples

For comparison of nonlinear inductor behavior with different parameterization options, see the Nonlinear Inductor Characteristics example.

The Inductor With Hysteresis example shows how modifying the equation coefficients of the Jiles-Atherton magnetic hysteresis equations affects the resulting B-H curve.

References

[1] Jiles, D. C. and D. L. Atherton. "Theory of ferromagnetic hysteresis." *Journal of Magnetism and Magnetic Materials*. Vol. 61, 1986, pp. 48–60.

[2] Jiles, D. C. and D. L. Atherton. "Ferromagnetic hysteresis." *IEEE Transactions on Magnetics*. Vol. 19, No. 5, 1983, pp. 2183–2184.

See Also
Inductor

Nonlinear Transformer

Model transformer with nonideal core

Library

Passive Devices

Description

The Nonlinear Transformer block represents a transformer with a nonideal core. A core may be nonideal due to its magnetic properties and dimensions. The equivalent circuit topology depends upon which of the two winding leakage parameterization options you select:

- `Combined primary and secondary values`
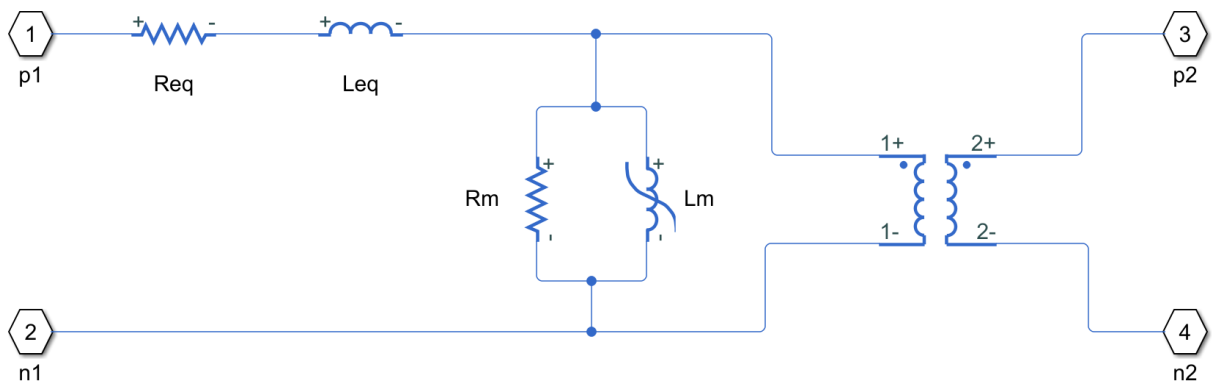
• Separate primary and secondary values
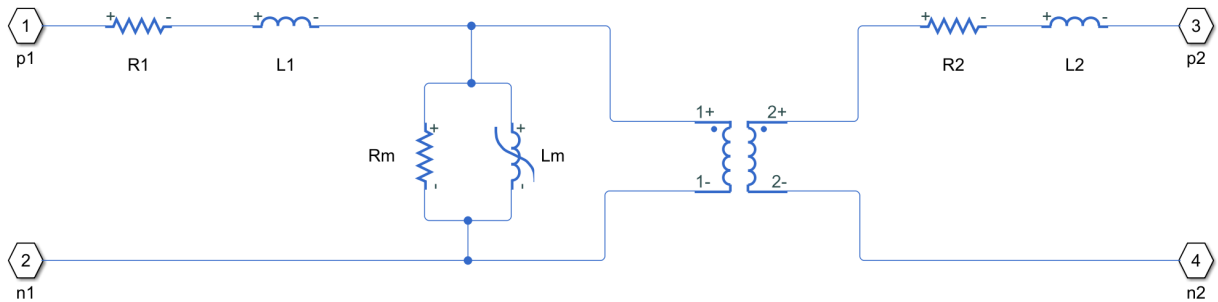
where:

- *Req* is the combined leakage resistance.
- *Leq* is the combined leakage inductance.
- *R1* is the primary leakage resistance.
- *L1* is the primary leakage inductance.
- *R2* is the secondary leakage resistance.
- *L2* is the secondary leakage inductance.
- *Rm* is the magnetization resistance.
- *Lm* is the magnetization inductance.

The block provides the following parameterization options for the nonlinear magnetization inductance:

- Single inductance (linear)
- Single saturation point
- Magnetic flux versus current characteristic
- Magnetic flux density versus magnetic field strength characteristic
- Magnetic flux density versus magnetic field strength characteristic with hysteresis

For more information, see the Nonlinear Inductor block reference page.

Parameters

- "Main Tab" on page 1-425
- "Magnetization Tab" on page 1-426
- "Initial Conditions Tab" on page 1-430
- "Parasitics Tab" on page 1-431

Main Tab

Primary number of turns

The number of turns of wire on the primary winding of the transformer. The default value is `100`.

Secondary number of turns

The number of turns of wire on the secondary winding of the transformer. The default value is `200`.

Winding parameterized by

Select one of the following methods for the winding leakage parameterization:

- `Combined primary and secondary values` — Use the lumped resistance and inductance values representing the combined leakage in the primary and secondary windings. This is the default option.
- `Separate primary and secondary values` — Use separate resistances and inductances to represent leakages in the primary and secondary windings.

Combined leakage resistance

The lumped equivalent resistance *Req*, which represents the combined power loss of the primary and secondary windings. This parameter is visible only when you select `Combined primary and secondary values` for the **Winding parameterized by** parameter. The default value is `0.01` Ω.

Combined leakage inductance

The lumped equivalent inductance *Leq*, which represents the combined magnetic flux loss of the primary and secondary windings. This parameter is visible only when you select `Combined primary and secondary values` for the **Winding parameterized by** parameter. The default value is `1e-4` H.

Primary leakage resistance

The resistance *R1*, which represents the power loss of the primary winding. This parameter is visible only when you select `Separate primary and secondary values` for the **Winding parameterized by** parameter. The default value is `0.01` Ω.

Primary leakage inductance

The inductance *L1*, which represents the magnetic flux loss of the primary winding. This parameter is visible only when you select `Separate primary and secondary values` for the **Winding parameterized by** parameter. The default value is `1e-4` H.

Secondary leakage resistance

The resistance *R2*, which represents the power loss of the secondary winding. This parameter is visible only when you select `Separate primary and secondary values` for the **Winding parameterized by** parameter. The default value is `0.01` Ω.

Secondary leakage inductance

The inductance *L2*, which represents the magnetic flux loss of the secondary winding. This parameter is visible only when you select `Separate primary and secondary values` for the **Winding parameterized by** parameter. The default value is `1e-4` H.

Magnetization Tab

Magnetization resistance

The resistance *Rm*, which represents the magnetic losses in the transformer core. The default value is `100` Ω.

Magnetization inductance parameterized by

Select one of the following methods for the nonlinear magnetization inductance parameterization:

- `Single inductance (linear)` — Provide the unsaturated inductance value.

- `Single saturation point` — Provide the values for the unsaturated and saturated inductances, as well as saturation magnetic flux. This is the default option.

- `Magnetic flux versus current characteristic` — Provide the current vector and the magnetic flux vector, to populate the magnetic flux versus current lookup table.

- `Magnetic flux density versus magnetic field strength characteristic` — Provide the values for effective core length and cross-sectional area, as well as the magnetic field strength vector and the magnetic flux

density vector, to populate the magnetic flux density versus magnetic field strength lookup table.

- `Magnetic flux density versus magnetic field strength characteristic with hysteresis` — In addition to the number of turns and the effective core length and cross-sectional area, provide the values for the initial anhysteretic B-H curve gradient, the magnetic flux density and field strength at a certain point on the B-H curve, as well as the coefficient for the reversible magnetization, bulk coupling coefficient, and inter-domain coupling factor, to define magnetic flux density as a function of both the current value and the history of the field strength.

Unsaturated inductance

The value of inductance used when the magnetization inductance Lm is operating in its linear region. This parameter is visible only when you select `Single inductance (linear)` or `Single saturation point` for the **Magnetization inductance parameterized by** parameter. The default value is `0.04` H.

Saturated inductance

The value of inductance used when the magnetization inductance Lm is operating beyond its saturation point. This parameter is visible only when you select `Single saturation point` for the **Magnetization inductance parameterized by** parameter. The default value is `0.01` H.

Saturation magnetic flux

The value of magnetic flux at which the magnetization inductance Lm saturates. This parameter is visible only when you select `Single saturation point` for the **Magnetization inductance parameterized by** parameter. The default value is `1.6e-4` Wb.

Current, i

The current data used to populate the magnetic flux versus current lookup table. This parameter is visible only when you select `Magnetic flux versus current characteristic` for the **Magnetization inductance parameterized by** parameter. The default value is `[ 0 0.4 0.8 1.2 1.6 2.0 ]` A.

Magnetic flux vector, phi

The magnetic flux data used to populate the magnetic flux versus current lookup table. This parameter is visible only when you select `Magnetic flux versus current characteristic` for the **Magnetization inductance parameterized by** parameter. The default value is `[0 0 0.161 0.25 0.284 0.295 0.299 ].*1e-3` Wb.

Magnetic field strength vector, H

The magnetic field strength data used to populate the magnetic flux density versus magnetic field strength lookup table. This parameter is visible only when you select `Magnetic flux density versus magnetic field strength characteristic` for the **Magnetization inductance parameterized by** parameter. The default value is `[ 0 200 400 600 800 1000 ]` A/m.

Magnetic flux density vector, B

The magnetic flux density data used to populate the magnetic flux density versus magnetic field strength lookup table. This parameter is visible only when you select `Magnetic flux density versus magnetic field strength characteristic` for the **Magnetization inductance parameterized by** parameter. The default value is `[ 0 0.81 1.25 1.42 1.48 1.49 ]` T.

Effective length

The effective core length, that is, the average distance of the magnetic path around the transformer core. This parameter is visible only when you select `Magnetic flux density versus magnetic field strength characteristic` for the **Magnetization inductance parameterized by** parameter. The default value is `0.2` m.

Effective cross-sectional area

The effective core cross-sectional area, that is, the average area of the magnetic path around the transformer core. This parameter is visible only when you select `Magnetic flux density versus magnetic field strength characteristic` for the **Magnetization inductance parameterized by** parameter. The default value is `2e-4` m^2.

Anhysteretic B-H gradient when H is zero

The gradient of the anhysteretic (no hysteresis) B-H curve around zero field strength. Set it to the average gradient of the positive-going and negative-going hysteresis curves. This parameter is visible only when you select `Magnetic flux density versus magnetic field strength characteristic with hysteresis` for the **Magnetization inductance parameterized by** parameter. The default value is `0.005` m*T/A.

Flux density point on anhysteretic B-H curve

Specify a point on the anhysteretic curve by providing its flux density value. Picking a point at high field strength where the positive-going and negative-going hysteresis curves align is the most accurate option. This parameter is visible only when you select `Magnetic flux density versus magnetic field strength`

characteristic with hysteresis for the **Magnetization inductance parameterized by** parameter. The default value is `1.49` T.

Corresponding field strength

The corresponding field strength for the point that you define by the **Flux density point on anhysteretic B-H curve** parameter. This parameter is visible only when you select `Magnetic flux density versus magnetic field strength characteristic with hysteresis` for the **Magnetization inductance parameterized by** parameter. The default value is `1000` A/m.

Coefficient for reversible magnetization, c

The proportion of the magnetization that is reversible. The value should be greater than zero and less than one. This parameter is visible only when you select `Magnetic flux density versus magnetic field strength characteristic with hysteresis` for the **Magnetization inductance parameterized by** parameter. The default value is `0.1`.

Bulk coupling coefficient, K

The Jiles-Atherton parameter that primarily controls the field strength magnitude at which the B-H curve crosses the zero flux density line. This parameter is visible only when you select `Magnetic flux density versus magnetic field strength characteristic with hysteresis` for the **Magnetization inductance parameterized by** parameter. The default value is `200` A/m.

Inter-domain coupling factor, alpha

The Jiles-Atherton parameter that primarily affects the points at which the B-H curves intersect the zero field strength line. Typical values are in the range of 1e-4 to 1e-3. This parameter is visible only when you select `Magnetic flux density versus magnetic field strength characteristic with hysteresis` for the **Magnetization inductance parameterized by** parameter. The default value is `1e-4`.

Interpolation option

The lookup table interpolation option. This parameter is visible only when you select `Magnetic flux versus current characteristic` or `Magnetic flux density versus magnetic field strength characteristic` for the **Magnetization inductance parameterized by** parameter. Select one of the following interpolation methods:

- `Linear` — Select this option to get the best performance.
- `Smooth` — Select this option to produce a continuous curve with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Initial Conditions Tab

Combined leakage inductance initial current

The value of current through the combined leakage inductance *Leq* at time zero. This parameter is visible only when you select `Combined primary and secondary values` for the **Winding parameterized by** parameter on the **Main** tab. The default value is `0` A.

Primary leakage inductance initial current

The value of current through the primary leakage inductance *L1* at time zero. This parameter is visible only when you select `Separate primary and secondary values` for the **Winding parameterized by** parameter on the **Main** tab. The default value is `0` A.

Secondary leakage inductance initial current

The value of current through the secondary leakage inductance *L2* at time zero. This parameter is visible only when you select `Separate primary and secondary values` for the **Winding parameterized by** parameter on the **Main** tab. The default value is `0` A.

Specify magnetization inductance initial state by

Select the appropriate initial state specification option:

- `Current` — Specify the initial state of the magnetization inductance *Lm* by the initial current. This is the default option.
- `Magnetic flux` — Specify the initial state of the magnetization inductance *Lm* by the magnetic flux.

This parameter is not visible when you select `Magnetic flux density versus magnetic field strength characteristic with hysteresis` for the **Magnetization inductance parameterized by** parameter on the **Magnetization** tab.

Magnetization inductance initial current

The initial current value used to calculate the value of magnetic flux within the magnetization inductance *Lm* at time zero. This is the current passing through the magnetization inductance *Lm*. Total magnetization current consists of current passing through the magnetization resistance *Rm* and current passing through the

magnetization inductance *Lm*. This parameter is visible only when you select `Current` for the **Specify magnetization inductance initial state by** parameter. The default value is `0` A.

Magnetization inductance initial magnetic flux

The value of the magnetic flux in the magnetization inductance *Lm* at time zero. This parameter is visible only when you select `Magnetic flux` for the **Specify magnetization inductance initial state by** parameter. The default is `0` Wb.

Magnetization inductance initial magnetic flux density

The value of magnetic flux density at time zero. This parameter is visible only when you select `Magnetic flux density versus magnetic field strength characteristic with hysteresis` for the **Magnetization inductance parameterized by** parameter on the **Magnetization** tab. The default is `0` T.

Magnetization inductance initial field strength

The value of magnetic field strength at time zero. This parameter is visible only when you select `Magnetic flux density versus magnetic field strength characteristic with hysteresis` for the **Magnetization inductance parameterized by** parameter on the **Magnetization** tab. The default is `0` A/m.

Parasitics Tab

Combined leakage inductance parasitic parallel conductance

Use this parameter to represent small parasitic effects in parallel to the combined leakage inductance *Leq*. A small parallel conductance may be required for the simulation of some circuit topologies. This parameter is visible only when you select `Combined primary and secondary values` for the **Winding parameterized by** parameter on the **Main** tab. The default value is `1e-9` 1/Ω.

Primary leakage inductance parasitic parallel conductance

Use this parameter to represent small parasitic effects in parallel to the primary leakage inductance *L1*. A small parallel conductance may be required for the simulation of some circuit topologies. This parameter is visible only when you select `Separate primary and secondary values` for the **Winding parameterized by** parameter on the **Main** tab. The default value is `1e-9` 1/Ω.

Secondary leakage inductance parasitic parallel conductance

Use this parameter to represent small parasitic effects in parallel to the secondary leakage inductance *L2*. A small parallel conductance may be required for the simulation of some circuit topologies. This parameter is visible only when you select

`Separate primary and secondary values` for the **Winding parameterized by** parameter on the **Main** tab. The default value is `1e-9` 1/Ω.

Ports

The block has four electrical conserving ports. Polarity is indicated by the + and - signs. Primary and secondary windings are indicated by numbers 1 and 2, respectively.

Examples

For comparison of nonlinear transformer behavior with different parameterization options, see the Nonlinear Transformer Characteristics example.

See Also

Ideal Transformer | Nonlinear Inductor

NPN Bipolar Transistor

Model NPN bipolar transistor using enhanced Ebers-Moll equations

Library

Semiconductor Devices

Description

The NPN Bipolar Transistor block uses a variant of the Ebers-Moll equations to represent an NPN bipolar transistor. The Ebers-Moll equations are based on two exponential diodes plus two current-controlled current sources. The NPN Bipolar Transistor block provides the following enhancements to that model:

- Early voltage effect
- Optional base, collector, and emitter resistances.
- Optional fixed base-emitter and base-collector capacitances.

The collector and base currents are:

$$
I_C = IS \left[\left(e^{qV_{BE}/(kT_{m1})} - e^{qV_{BC}/(kT_{m1})} \right) \left(1 - \frac{V_{BC}}{V_A} \right) - \frac{1}{\beta_R} \left(e^{qV_{BC}/(kT_{m1})} - 1 \right) \right]
$$

$$
I_B = IS \left[\frac{1}{\beta_F} \left(e^{qV_{BE}/(kT_{m1})} - 1 \right) + \frac{1}{\beta_R} \left(e^{qV_{BC}/(kT_{m1})} - 1 \right) \right]
$$

Where:

- I_B and I_C are base and collector currents, defined as positive into the device.
- *IS* is the saturation current.
- V_{BE} is the base-emitter voltage and V_{BC} is the base-collector voltage.
- β_F is the ideal maximum forward current gain BF
- β_R is the ideal maximum reverse current gain BR
- V_A is the forward Early voltage VAF
- q is the elementary charge on an electron (1.602176e–19 Coulombs).
- k is the Boltzmann constant (1.3806503e–23 J/K).
- T_{m1} is the transistor temperature, as defined by the **Measurement temperature** parameter value.

You can specify the transistor behavior using datasheet parameters that the block uses to calculate the parameters for these equations, or you can specify the equation parameters directly.

If $qV_{BC} / (kT_{m1}) > 40$ or $qV_{BE} / (kT_{m1}) > 40$, the corresponding exponential terms in the equations are replaced with $(qV_{BC} / (kT_{m1}) - 39)e^{40}$ and $(qV_{BE} / (kT_{m1}) - 39)e^{40}$, respectively. This helps prevent numerical issues associated with the steep gradient of the exponential function e^x at large values of x. Similarly, if $qV_{BC} / (kT_{m1}) < -39$ or $qV_{BE} / (kT_{m1}) < -39$ then the corresponding exponential terms in the equations are replaced with $(qV_{BC} / (kT_{m1}) + 40)e^{-39}$ and $(qV_{BE} / (kT_{m1}) + 40)e^{-39}$, respectively.

Optionally, you can specify parasitic fixed capacitances across the base-emitter and base-collector junctions. You also have the option to specify base, collector, and emitter connection resistances.

Modeling Temperature Dependence

The default behavior is that dependence on temperature is not modeled, and the device is simulated at the temperature for which you provide block parameters. You can optionally include modeling the dependence of the transistor static behavior on temperature during simulation. Temperature dependence of the junction capacitances is not modeled, this being a much smaller effect.

When including temperature dependence, the transistor defining equations remain the same. The measurement temperature value, T_{m1}, is replaced with the simulation temperature, T_s. The saturation current, *IS*, and the forward and reverse gains (β_F and β_R) become a function of temperature according to the following equations:

$$IS_{Ts} = IS_{Tm1} \cdot (T_s / T_{m1})^{XTI} \cdot \exp\left(-\frac{EG}{kT_s}(1 - T_s / T_{m1}) \right)$$

$$\beta_{Fs} = \beta_{Fm1}\left(\frac{T_s}{T_{m1}} \right)^{XTB}$$

$$\beta_{Rs} = \beta_{Rm1}\left(\frac{T_s}{T_{m1}} \right)^{XTB}$$

where:

- T_{m1} is the temperature at which the transistor parameters are specified, as defined by the **Measurement temperature** parameter value.

- T_s is the simulation temperature.

- IS_{Tm1} is the saturation current at the measurement temperature.

- IS_{Ts} is the saturation current at the simulation temperature. This is the saturation current value used in the bipolar transistor equations when temperature dependence is modeled.

- β_{Fm1} and β_{Rm1} are the forward and reverse gains at the measurement temperature.

- β_{Fs} and β_{Rs} are the forward and reverse gains at the simulation temperature. These are the values used in the bipolar transistor equations when temperature dependence is modeled.

- *EG* is the energy gap for the semiconductor type measured in Joules. The value for silicon is usually taken to be 1.11 eV, where 1 eV is 1.602e-19 Joules.

- *XTI* is the saturation current temperature exponent.

- *XTB* is the forward and reverse gain temperature coefficient.

- *k* is the Boltzmann constant (1.3806503e–23 J/K).

Appropriate values for *XTI* and *EG* depend on the type of transistor and the semiconductor material used. In practice, the values of *XTI*, *EG*, and *XTB* need tuning to model the exact behavior of a particular transistor. Some manufacturers quote these tuned values in a SPICE Netlist, and you can read off the appropriate values. Otherwise you can determine values for *XTI*, *EG*, and *XTB* by using a datasheet-defined data at a higher temperature T_{m2}. The block provides a datasheet parameterization option for this.

1-435

You can also tune the values of *XTI*, *EG*, and *XTB* yourself, to match lab data for your particular device. You can use Simulink Design Optimization software to help tune the values.

Thermal Port

The block has an optional thermal port, hidden by default. To expose the thermal port, right-click the block in your model, and then from the context menu select **Simscape > Block choices > Show thermal port**. This action displays the thermal port H on the block icon, and adds the **Thermal Port** tab to the block dialog box.

Use the thermal port to simulate the effects of generated heat and device temperature. For more information on using thermal ports and on the **Thermal Port** tab parameters, see "Simulating Thermal Effects in Semiconductors".

Basic Assumptions and Limitations

The NPN Bipolar Transistor model has the following limitations:

- The block does not account for temperature-dependent effects on the junction capacitances.
- You may need to use nonzero ohmic resistance and junction capacitance values to prevent numerical simulation issues, but the simulation may run faster with these values set to zero.

Parameters

- "Main Tab" on page 1-436
- "Ohmic Resistance Tab" on page 1-438
- "Capacitance Tab" on page 1-439
- "Temperature Dependence Tab" on page 1-439

Main Tab

Parameterization

Select one of the following methods for block parameterization:

- `Specify from a datasheet` — Provide parameters that the block converts to equations that describe the transistor. The block calculates the forward Early voltage *VAF* as *Ic/h_oe*, where *Ic* is the **Collector current at which h-parameters are defined** parameter value, and *h_oe* is the **Output admittance h_oe** parameter value [1]. The block sets *BF* to the small-signal **Forward current transfer ratio h_fe** value. The block calculates the saturation current *IS* from the specified **Voltage Vbe** value and the corresponding **Current Ib for voltage Vbe** value when *Ic* is zero. This is the default method.

- `Specify using equation parameters directly` — Provide equation parameters *IS*, *BF*, and *VAF*.

Forward current transfer ratio h_fe

Small-signal current gain. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is `100`.

Output admittance h_oe

Derivative of the collector current with respect to the collector-emitter voltage for a fixed base current. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is `5e-05` 1/Ω.

Collector current at which h-parameters are defined

The h-parameters vary with operating point, and are defined for this value of the collector current. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is `1` mA.

Collector-emitter voltage at which h-parameters are defined

The h-parameters vary with operating point, and are defined for this value of the collector-emitter voltage. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is `5` V.

Voltage Vbe

Base-emitter voltage when the base current is *Ib*. The [*Vbe Ib*] data pair must be quoted for when the transistor is in the normal active region, that is, not in the saturated region. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is `0.55` V.

Current Ib for voltage Vbe

Base current when the base-emitter voltage is *Vbe*. The [*Vbe Ib*] data pair must be quoted for when the transistor is in the normal active region, that is, not in the saturated region. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is `0.5` mA.

1-437

Forward current transfer ratio BF

Ideal maximum forward current gain. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter. The default value is `100`.

Saturation current IS

Transistor saturation current. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter. The default value is `1e-14` A.

Forward Early voltage VAF

In the standard Ebers-Moll equations, the gradient of the *Ic* versus *Vce* curve is zero in the normal active region. The additional forward Early voltage term increases this gradient. The intercept on the *Vce*-axis is equal to –*VAF* when the linear region is extrapolated. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter. The default value is `200` V.

Reverse current transfer ratio BR

Ideal maximum reverse current gain. This value is often not quoted in manufacturer datasheets, because it is not significant when the transistor is biased to operate in the normal active region. When the value is not known and the transistor is not to be operated on the inverse region, use the default value of `1`.

Measurement temperature

Temperature T_{m1} at which *Vbe* and *Ib*, or *IS*, are measured. The default value is 25 °C.

Ohmic Resistance Tab

Collector resistance RC

Resistance at the collector. The default value is `0.01` Ω.

Emitter resistance RE

Resistance at the emitter. The default value is `1e-4` Ω.

Zero bias base resistance RB

Resistance at the base at zero bias. The default value is `1` Ω.

Capacitance Tab

Base-collector junction capacitance

Parasitic capacitance across the base-collector junction. The default value is 5 pF.

Base-emitter junction capacitance

Parasitic capacitance across the base-emitter junction. The default value is 5 pF.

Total forward transit time

Represents the mean time for the minority carriers to cross the base region from the emitter to the collector, and is often denoted by the parameter TF [1]. The default value is 0 µs.

Total reverse transit time

Represents the mean time for the minority carriers to cross the base region from the collector to the emitter, and is often denoted by the parameter TR [1]. The default value is 0µs.

Temperature Dependence Tab

Parameterization

Select one of the following methods for temperature dependence parameterization:

- `None — Simulate at parameter measurement temperature` — Temperature dependence is not modeled, or the model is simulated at the measurement temperature T_{m1} (as specified by the **Measurement temperature** parameter on the **Main** tab). This is the default method.

- `Model temperature dependence` — Provide a value for simulation temperature, to model temperature-dependent effects. You also have to provide a set of additional parameters depending on the block parameterization method. If you parameterize the block from a datasheet, you have to provide values for a second [*Vbe Ib*] data pair and *h_fe* at second measurement temperature. If you parameterize by directly specifying equation parameters, you have to provide the values for *XTI*, *EG*, and *XTB*.

Forward current transfer ratio, h_fe, at second measurement temperature

Small-signal current gain at the second measurement temperature. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter on the **Main** tab. It must be quoted at the same

collector-emitter voltage and collector current as for the **Forward current transfer ratio h_fe** parameter on the **Main** tab. The default value is 125.

Voltage Vbe at second measurement temperature

Base-emitter voltage when the base current is *Ib* and the temperature is set to the second measurement temperature. The [*Vbe Ib*] data pair must be quoted for when the transistor is in the normal active region, that is, not in the saturated region. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter on the **Main** tab. The default value is 0.45 V.

Current Ib for voltage Vbe at second measurement temperature

Base current when the base-emitter voltage is *Vbe* and the temperature is set to the second measurement temperature. The [*Vbe Ib*] data pair must be quoted for when the transistor is in the normal active region, that is, not in the saturated region. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter on the **Main** tab. The default value is 0.5 mA.

Second measurement temperature

Second temperature T_{m2} at which *h_fe,Vbe,* and *Ib* are measured. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter on the **Main** tab. The default value is 125 °C.

Current gain temperature coefficient, XTB

Current gain temperature coefficient value. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter on the **Main** tab. The default value is 0.

Energy gap, EG

Energy gap value. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter on the **Main** tab. The default value is 1.11 eV.

Saturation current temperature exponent, XTI

Saturation current temperature coefficient value. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter on the **Main** tab. The default value is 3.

Device simulation temperature

Temperature T_s at which the device is simulated. The default value is 25 °C.

Ports

The block has the following ports:

B

Electrical conserving port associated with the transistor base terminal

C

Electrical conserving port associated with the transistor collector terminal

E

Electrical conserving port associated with the transistor emitter terminal

Examples

See the NPN Bipolar Transistor Characteristics example.

References

[1] G. Massobrio and P. Antognetti. *Semiconductor Device Modeling with SPICE*. 2nd Edition, McGraw-Hill, 1993.

[2] H. Ahmed and P.J. Spreadbury. *Analogue and digital electronics for engineers*. 2nd Edition, Cambridge University Press, 1984.

See Also

Diode | PNP Bipolar Transistor

Operational Transconductance Amplifier

Model behavioral representation of operational transconductance amplifier

Library

Integrated Circuits

Description

The Operational Transconductance Amplifier block provides a behavioral representation of an operational transconductance amplifier. A transconductance amplifier converts an input voltage into an output current. Applications include variable frequency oscillators, variable gain amplifiers and current-controlled filters. These applications exploit the fact that the transconductance gain is a function of current flowing into the control current pin.

To support faster simulation, the behavioral representation does not model the detailed transistor implementation. Therefore, the model is only valid when operating in the linear region, that is, where the device input resistance, output resistance, and transconductance gain all depend linearly on the control current, and are independent of input signal amplitude. The dynamics are approximated by a first-order lag, based on the value you specify for the block parameter **Bandwidth**.

Control Current

The control current pin C is maintained at the voltage that you specify for the **Minimum output voltage**. In practice, the **Minimum output voltage** equals the negative supply voltage plus the transistor collector-emitter voltage drop. For example, if the **Minimum output voltage** for a supply voltage of +-15V is -14.5, then to achieve a control current of 500µA, a resistor connected between the +15V rail and the control current pin must have a value of (15 - (-14.5)) / 500e-6 = 59kOhm.

Transconductance

The relationship between input voltage, v, and transconductance current, i_{gm}, is:

$$v = v_+ - v_-$$

$$i_{gm} = g_m \cdot v$$

$$g_m = \frac{g_{m0} \cdot i_c}{i_{c0}}$$

where:

- v_+ is the voltage presented at the block + pin.
- v_- is the voltage presented at the block - pin.
- g_m is the transconductance.
- i_c is the control current flowing into the control current pin C.
- i_{c0} is the reference control current, that is, the control current at which transconductance is quoted on the datasheet.
- g_{m0} is the transconductance measured at the reference control current i_{c0}.

Therefore, increasing control current increases the transconductance.

Output Resistance and Determining Output Current

The output resistance, R_{out}, is defined by:

$$i_{gm} + i_o = \frac{v_o}{R_{out}}$$

$$R_{out} = \frac{R_{out0} \cdot i_{c0}}{i_c}$$

where:

- i_{gm} is the transconductance current.
- i_o is the output current, defined as positive if flowing into the transconductance amplifier output pin.

- i_c is the control current flowing into the control current pin C.
- i_{c0} is the reference control current, that is, the control current at which output resistance is quoted on the datasheet.
- R_{out0} is the output resistance measured at the reference control current i_{c0}.

Therefore, increasing control current reduces output resistance.

Input Resistance

The relationship between input voltage, v, across the + and - pins and the current flowing, i, is:

$$\frac{v}{i} = R_{in}$$

$$R_{in} = \frac{R_{in0} \cdot i_{c0}}{i_c}$$

where:

- i_c is the control current flowing into the control current pin C.
- R_{in} is the input resistance for the current control current value, i_c.
- i_{c0} is the reference control current, that is, the control current at which input resistance is quoted on the datasheet.
- R_{in0} is the input resistance measured at the reference control current i_{c0}.

Therefore, increasing control current reduces input resistance.

Limits

Because of the physical construction of an operational transconductance amplifier based on current mirrors, the transconductance current i_{gm} cannot exceed the control current. Hence the value of i_{gm} is limited by:

$$-i_c \leq i_{gm} \leq i_c$$

The output voltage is also limited by the supply voltage:

$$V_{min} \leq v_o \leq V_{max}$$

where $V_{\min}$ is the **Minimum output voltage**, and $V_{\max}$ is the **Maximum output voltage**. Output voltage limiting is implemented by adding a low resistance to the output when the voltage limit is exceeded. The value of this resistance is set by the **Additional output resistance at voltage swing limits** parameter.

The transconductance current is also slew-rate limited, a value for slew rate limiting typically being given on datasheets:

$$-\mu \le \frac{di_{gm}}{dt} \le \mu$$

where μ is the **Maximum current slew rate**.

Parameters

- "Nominal Measurements Tab" on page 1-445
- "Dynamics Tab" on page 1-446
- "Limits Tab" on page 1-446

Nominal Measurements Tab

Transconductance

The transconductance, g_m, when the control current is equal to the **Reference control current**. This is the ratio of the transconductance current, i_{gm}, to the voltage difference, v, across the + and - pins. The default value is 9600 µS.

Input resistance

The input resistance, R_{in}, when the control current is equal to the **Reference control current**. The input resistance is the ratio of the voltage difference, v, across the + and - pins to the current flowing from the + to the - pin. The default value is 25 kOhm.

Output resistance

The output resistance, R_{out}, when the control current is equal to the **Reference control current**. See above for the equation defining output resistance. The default value is 3 MOhm.

Reference control current

The control current at which the **Transconductance**, **Input resistance**, and **Output resistance** are quoted. The default value is 500 µA.

Dynamics Tab

Dynamics

Select one of the following options:

- `No lag` — Do not model the dynamics of the relationship between output current and input voltage. This is the default.
- `Finite bandwidth with slew rate limiting` — Model the dynamics of the relationship between output current and input voltage using a first-order lag. If you select this option, the **Bandwidth**, **Maximum current slew rate**, and **Initial current** parameters appear on the **Dynamics** tab.

Bandwidth

The bandwidth of the first-order lag used to model the dynamics of the relationship between output current and input voltage. The default value is 2 MHz.

Maximum current slew rate

The maximum rate-of-change of transconductance current when there is no feedback around the device. Note that datasheets sometimes quote slew rate as a maximum rate of change of voltage. In this case, the value depends on the particular test circuit. To get an accurate value for **Maximum current slew rate**, reproduce the test circuit in a Simscape Electronics model, and tune the parameter value to match the datasheet value. If the test circuit is open-loop, and the load resistance is quoted, you can obtain an approximate value for the **Maximum current slew rate** by dividing the voltage slew rate by the load resistance. The default value is 2 A/µs.

Initial current

The initial transconductance current (note, not the initial output current). This is the transconductance current sinking to both the internal output resistance, R_{out}, and the output pin. The default value is 0 A.

Limits Tab

Minimum output voltage

The output voltage is limited to be greater than the value of this parameter. The default value is -15 V.

Maximum output voltage

The output voltage is limited to be less than the value of this parameter. The default value is 15 V.

Additional output resistance at voltage swing limits

To limit the output voltage swing, an additional output resistance is applied between output and the power rail when the output voltage exceeds the limit. The value of this resistance should be low compared to the output resistance and circuit load resistance. The default value is 1 Ohm.

Minimum control current for simulation

The control current measured at the control current pin C is limited to be greater than the value of this parameter. This prevents a potential divide-by-zero when calculating input and output resistance values based on the value of the control current. The default value is 0.001 μA.

Ports

The block has the following ports:

+

Positive electrical voltage

-

Negative electrical voltage

C

Control current

OUT

Output current

See Also

Band-Limited Op-Amp | Finite-Gain Op-Amp | Op-Amp

Optocoupler

Model optocoupler as LED, current sensor, and controlled current source

Library

Semiconductor Devices

Description

This block represents an optocoupler using a model that consists of the following components:

- An exponential light-emitting diode in series with a current sensor on the input side
- A controlled current source on the output side

The output-side current flows from the collector junction to the emitter junction. It has a value of $CTR \cdot I_d$, where CTR is the **Current transfer ratio** parameter value and I_d is the diode current.

Use the Optocoupler block to interface two electrical circuits without making a direct electrical connection. A common reason for doing this is that the two circuits work at very different voltage levels.

Note Each electrical circuit must have its own Electrical Reference block.

If the output circuit is a phototransistor, typical values for the **Current transfer ratio** parameter are 0.1 to 0.5. If the output stage consists of a Darlington pair, the parameter value can be much higher than this. The **Current transfer ratio** value also varies with the light-emitting diode current, but this effect is not modeled by the Photodiode block.

Some manufacturers provide a maximum data rate for optocouplers. In practice, the maximum data rate depends on the following factors:

- The capacitance of the photodiode and the type of the driving circuit
- The construction of the phototransistor and its associated capacitance

The Optocoupler block only lets you define the capacitance on the light-emitting diode. You can use the **Junction capacitance** parameter to add your own capacitance across the collector and emitter connections.

The Optocoupler block lets you model temperature dependence of the underlying diode. For details, see the Diode reference page.

Thermal Port

The block has an optional thermal port, hidden by default. To expose the thermal port, right-click the block in your model, and then from the context menu select **Simscape > Block choices > Show thermal port**. This action displays the thermal port H on the block icon, and adds the **Thermal Port** tab to the block dialog box.

Use the thermal port to simulate the effects of generated heat and device temperature. For more information on using thermal ports and on the **Thermal Port** tab parameters, see "Simulating Thermal Effects in Semiconductors".

Variables

Use the **Variables** section of the block interface to set the priority and initial target values for the block variables prior to simulation. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

Basic Assumptions and Limitations

The Optocoupler block has the following limitations:

- The output side is modeled as a controlled current source. As such, it only correctly approximates a bipolar transistor operating in its normal active region. To create a more detailed model, connect the Optocoupler output directly to the base of an NPN Bipolar Transistor block, and set the parameters to maintain a correct overall value for

the current transfer ratio. If you need to connect optocouplers in series, use this approach to avoid the invalid topology of two current sources in series.

- The temperature dependence of the forward current transfer ratio is not modeled. Typically the temperature dependence of this parameter is much less than that of the optical diode I-V characteristic.

- You may need to use nonzero ohmic resistance and junction capacitance values to prevent numerical simulation issues, but the simulation may run faster with these values set to zero.

Parameters

- "Main" on page 1-450
- "Ohmic Resistance" on page 1-451
- "Junction Capacitance" on page 1-451
- "Temperature Dependence" on page 1-452

Main

Current transfer ratio

The output current flowing from the transistor collector to emitter junctions is equal to the product of the current transfer ratio and the current flowing the light-emitting diode. The default value is `0.2`.

Diode parameterization

Select one of the following methods for model parameterization:

- `Use I-V curve data points` — Specify measured data at two points on the diode I-V curve. This is the default method.

- `Use parameters IS and N` — Specify saturation current and emission coefficient.

Currents [I1 I2]

A vector of the current values at the two points on the diode I-V curve that the block uses to calculate *IS* and *N*. This parameter is only visible when you select `Use I-V curve data points` for the **Diode parameterization** parameter. The default value is `[ 0.001 0.015 ]` A.

Voltages [V1 V2]

A vector of the voltage values at the two points on the diode I-V curve that the block uses to calculate *IS* and *N*. This parameter is only visible when you select `Use I-V curve data points` for the **Diode parameterization** parameter. The default value is `[ 0.9 1.05 ]` V.

Saturation current IS

The magnitude of the current that the ideal diode equation approaches asymptotically for very large reverse bias levels. This parameter is only visible when you select `Use parameters IS and N` for the **Diode parameterization** parameter. The default value is `1e-10` A.

Measurement temperature

The temperature at which IS or the I-V curve was measured. The default value is 25 °C.

Emission coefficient N

The diode emission coefficient or ideality factor. This parameter is only visible when you select `Use parameters IS and N` for the **Diode parameterization** parameter. The default value is 2.

Ohmic Resistance

Ohmic resistance RS

The series diode connection resistance. The default value is `0.1` Ω.

Junction Capacitance

Junction capacitance

Select one of the following options for modeling the diode junction capacitance:

- `Fixed or zero junction capacitance` — Model the junction capacitance as a fixed value.

- `Use C-V curve data points` — Specify measured data at three points on the diode C-V curve.

- `Use parameters CJ0, VJ, M & FC` — Specify zero-bias junction capacitance, junction potential, grading coefficient, and forward-bias depletion capacitance coefficient.

Zero-bias junction capacitance CJ0

The value of the capacitance placed in parallel with the exponential diode term. This parameter is only visible when you select `Fixed or zero junction capacitance` or `Use parameters CJ0, VJ, M & FC` for the **Junction capacitance** parameter. The default value is 5 pF.

Junction potential VJ

The junction potential. This parameter is only visible when you select `Use parameters CJ0, VJ, M & FC` for the **Junction capacitance** parameter. The default value is 1 V.

Grading coefficient M

The coefficient that quantifies the grading of the junction. This parameter is only visible when you select `Use parameters CJ0, VJ, M & FC` for the **Junction capacitance** parameter. The default value is `0.5`.

Reverse bias voltages [VR1 VR2 VR3]

A vector of the reverse bias voltage values at the three points on the diode C-V curve that the block uses to calculate *CJ0*, *VJ*, and *M*. This parameter is only visible when you select `Use C-V curve data points` for the **Junction capacitance** parameter. The default value is `[ 0.1 10 100 ]` V.

Corresponding capacitances [C1 C2 C3]

A vector of the capacitance values at the three points on the diode C-V curve that the block uses to calculate *CJ0*, *VJ*, and *M*. This parameter is only visible when you select `Use C-V curve data points` for the **Junction capacitance** parameter. The default value is `[ 3.5 1 0.4 ]` pF.

Capacitance coefficient FC

Fitting coefficient that quantifies the decrease of the depletion capacitance with applied voltage. This parameter is only visible when you select `Use C-V curve data points` or `Use parameters CJ0, VJ, M & FC` for the **Junction capacitance** parameter. The default value is `0.5`.

Temperature Dependence

Parameterization

Select one of the following methods for temperature dependence parameterization:

- `None — Simulate at parameter measurement temperature` — Temperature dependence is not modeled, or the model is simulated at the

measurement temperature T_{m1} (as specified by the **Measurement temperature** parameter on the **Main** tab). This is the default method.

- `Use an I-V data point at second measurement temperature T2` — If you select this option, you specify a second measurement temperature T_{m2}, and the current and voltage values at this temperature. The model uses these values, along with the parameter values at the first measurement temperature T_{m1}, to calculate the energy gap value.

- `Specify saturation current at second measurement temperature T2` — If you select this option, you specify a second measurement temperature T_{m2}, and saturation current value at this temperature. The model uses these values, along with the parameter values at the first measurement temperature T_{m1}, to calculate the energy gap value.

- `Specify the energy gap EG` — Specify the energy gap value directly.

Current I1 at second measurement temperature

Specify the diode current *I1* value when the voltage is *V1* at the second measurement temperature. This parameter is only visible when you select `Use an I-V data point at second measurement temperature T2` for the **Parameterization** parameter. The default value is `0.029` A.

Voltage V1 at second measurement temperature

Specify the diode voltage *V1* value when the current is *I1* at the second measurement temperature. This parameter is only visible when you select `Use an I-V data point at second measurement temperature T2` for the **Parameterization** parameter. The default value is `1.05` V.

Saturation current, IS, at second measurement temperature

Specify the saturation current *IS* value at the second measurement temperature. This parameter is only visible when you select `Specify saturation current at second measurement temperature T2` for the **Parameterization** parameter. The default value is `1.8e-8` A.

Second measurement temperature

Specify the value for the second measurement temperature. This parameter is only visible when you select either `Use an I-V data point at second measurement temperature T2` or `Specify saturation current at second measurement temperature T2` for the **Parameterization** parameter. The default value is `125` °C.

Energy gap parameterization

This parameter is only visible when you select `Specify the energy gap EG` for the **Parameterization** parameter. It lets you select a value for the energy gap from a list of predetermined options, or specify a custom value:

- `Use nominal value for silicon (EG=1.11eV)` — This is the default.
- `Use nominal value for 4H-SiC silicon carbide (EG=3.23eV)`
- `Use nominal value for 6H-SiC silicon carbide (EG=3.00eV)`
- `Use nominal value for germanium (EG=0.67eV)`
- `Use nominal value for gallium arsenide (EG=1.43eV)`
- `Use nominal value for selenium (EG=1.74eV)`
- `Use nominal value for Schottky barrier diodes (EG=0.69eV)`
- `Specify a custom value` — If you select this option, the **Energy gap, EG** parameter appears in the dialog box, to let you specify a custom value for *EG*.

Energy gap, EG

Specify a custom value for the energy gap, *EG*. This parameter is only visible when you select `Specify a custom value` for the **Energy gap parameterization** parameter. The default value is `1.11` eV.

Saturation current temperature exponent parameterization

Select one of the following options to specify the saturation current temperature exponent value:

- `Use nominal value for pn-junction diode (XTI=3)` — This is the default.
- `Use nominal value for Schottky barrier diode (XTI=2)`
- `Specify a custom value` — If you select this option, the **Saturation current temperature exponent, XTI** parameter appears in the dialog box, to let you specify a custom value for *XTI*.

Saturation current temperature exponent, XTI

Specify a custom value for the saturation current temperature exponent, *XTI*. This parameter is only visible when you select `Specify a custom value` for the **Saturation current temperature exponent parameterization** parameter. The default value is `3`.

Device simulation temperature

Specify the value for the temperature T_s, at which the device is to be simulated. The default value is 25 °C.

Ports

The block has the following ports:

+

Electrical conserving port associated with the diode positive terminal

-

Electrical conserving port associated with the diode negative terminal

C

Electrical conserving port associated with the transistor collector terminal

E

Electrical conserving port associated with the transistor emitter terminal

References

[1] G. Massobrio and P. Antognetti. *Semiconductor Device Modeling with SPICE*. 2nd Edition, McGraw-Hill, 1993.

[2] H. Ahmed and P.J. Spreadbury. *Analogue and digital electronics for engineers*. 2nd Edition, Cambridge University Press, 1984.

See Also

Controlled Current Source | Diode | NPN Bipolar Transistor

P-Channel JFET

Model P-Channel JFET

Library
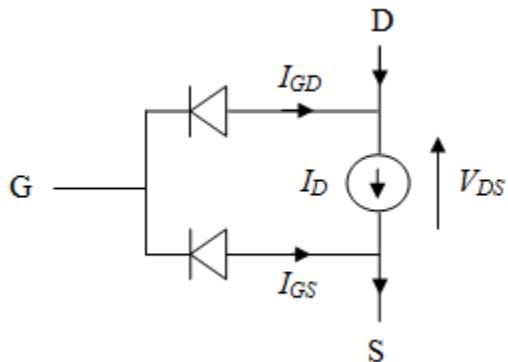
Semiconductor Devices

Description

The P-Channel JFET block uses the Shichman and Hodges equations to represent a P-Channel JFET using a model with the following structure:

G is the transistor gate, D is the transistor drain and S is the transistor source. The drain current, I_D, depends on the region of operation and whether the transistor is operating in normal or inverse mode.

- In normal mode ($-V_{DS} \geq 0$), the block provides the following relationship between the drain current I_D and the drain-source voltage V_{DS}.

Region	Applicable Range of V_{GS} and V_{DS} Values	Corresponding I_D Equation
Off	$-V_{GS} \leq -V_{t0}$	$I_D = 0$
Linear	$0 < -V_{DS} < -V_{GS} + V_{t0}$	$I_D = \beta V_{DS}(2(-V_{GS} + V_{t0}) + V_{DS})(1 - \lambda V_{DS})$
Saturated	$0 < -V_{GS} + V_{t0} \leq -V_{DS}$	$I_D = -\beta (-V_{GS} + V_{t0})^2 (1 - \lambda V_{DS})$

- In inverse mode ($-V_{DS} < 0$), the block provides the following relationship between the drain current I_D and the drain-source voltage V_{DS}.

Region	Applicable Range of V_{GS} and V_{DS} Values	Corresponding I_D Equation
Off	$-V_{GD} \leq -V_{t0}$	$I_D = 0$
Linear	$0 < V_{DS} < - V_{GD} + V_{t0}$	$I_D = \beta V_{DS}(2(-V_{GD} + V_{t0}) - V_{DS})(1 + \lambda V_{DS})$
Saturated	$0 < -V_{GD} + V_{t0} \leq V_{DS}$	$I_D = \beta (-V_{GD} + V_{t0})^2 (1 + \lambda V_{DS})$

In the preceding equations:

- V_{GS} is the gate-source voltage.
- V_{GD} is the gate-drain voltage.
- V_{t0} is the threshold voltage. If you select `Specify using equation parameters directly` for the **Parameterization** parameter, V_{t0} is the **Threshold voltage** parameter value. Otherwise, the block calculates V_{t0} from the datasheet parameters you specify.
- β is the transconductance parameter. If you select `Specify using equation parameters directly` for the **Parameterization** parameter, β is the **Transconductance parameter** parameter value. Otherwise, the block calculates β from the datasheet parameters you specify.
- λ is the channel-length modulation parameter. If you select `Specify using equation parameters directly` for the **Parameterization** parameter, λ is the **Channel-length modulation** parameter value. Otherwise, the block calculates λ from the datasheet parameters you specify.

The currents in each of the diodes satisfy the exponential diode equation

1-457

$$I_{GD} = -IS \cdot \left(e^{-qV_{GD}/kT_{m1}} - 1\right)$$

$$I_{GS} = -IS \cdot \left(e^{-qV_{GS}/kT_{m1}} - 1\right)$$

where:

- *IS* is the saturation current. If you select `Specify using equation parameters directly` for the **Parameterization** parameter, *IS* is the **Saturation current** parameter value. Otherwise, the block calculates *IS* from the datasheet parameters you specify.
- *q* is the elementary charge on an electron (1.602176e–19 Coulombs).
- *k* is the Boltzmann constant (1.3806503e–23 J/K).
- T_{m1} is the measurement temperature. The value comes from the **Measurement temperature** parameter.

The block models gate junction capacitance as a fixed gate-drain capacitance C_{GD} and a fixed gate-source capacitance C_{GS}. If you select `Specify using equation parameters directly` for the **Parameterization** parameter, you specify these values directly using the **Gate-drain junction capacitance** and **Gate-source junction capacitance** parameters. Otherwise, the block derives them from the **Input capacitance Ciss** and **Reverse transfer capacitance Crss** parameter values. The two parameterizations are related as follows:

- C_{GD} = *Crss*
- C_{GS} = *Ciss* – *Crss*

Modeling Temperature Dependence

The default behavior is that dependence on temperature is not modeled, and the device is simulated at the temperature for which you provide block parameters. You can optionally include modeling the dependence of the transistor static behavior on temperature during simulation. Temperature dependence of the junction capacitances is not modeled, this being a much smaller effect.

When including temperature dependence, the transistor defining equations remain the same. The measurement temperature value, T_{m1}, is replaced with the simulation temperature, T_s. The transconductance, β, and the threshold voltage, V_{t0}, become a function of temperature according to the following equations:

$$\beta_{Ts} = \beta_{Tm1} \left(\frac{T_s}{T_{m1}} \right)^{BEX}$$

$$V_{t0s} = V_{t01} + \alpha \, (\, T_s - T_{m1})$$

where:

- T_{m1} is the temperature at which the transistor parameters are specified, as defined by the **Measurement temperature** parameter value.
- T_s is the simulation temperature.
- β_{Tm1} is JFET transconductance at the measurement temperature.
- β_{Ts} is JFET transconductance at the simulation temperature. This is the transconductance value used in the JFET equations when temperature dependence is modeled.
- V_{t01} is the threshold voltage at measurement temperature.
- V_{t0s} is the threshold voltage at simulation temperature. This is the threshold voltage value used in the JFET equations when temperature dependence is modeled.
- *BEX* is the mobility temperature exponent. A typical value of *BEX* is -1.5.
- α is the gate threshold voltage temperature coefficient, dV_{th}/dT.

For most JFETS, you can use the default value of `-1.5` for *BEX*. Some datasheets quote the value for α, but most typically they provide the temperature dependence for the saturated drain current, *I_dss*. Depending on the block parameterization method, you have two ways of specifying α:

- If you parameterize the block from a datasheet, you have to provide *I_dss* at a second measurement temperature. The block then calculates the value for α based on this data.
- If you parameterize by specifying equation parameters, you have to provide the value for α directly.

If you have more data comprising drain current as a function of gate-source voltage for fixed drain-source voltage plotted at more than one temperature, then you can also use Simulink Design Optimization software to help tune the values for α and *BEX*.

In addition, the saturation current term, *IS*, in the gate-drain and gate-source current equations depends on temperature

$$IS_{Ts} = IS_{Tm1} \cdot (T_s / T_{m1})^{XTI} \cdot \exp\left(-\frac{EG}{kT_s}(1 - T_s / T_{m1}) \right)$$

where:

- IS_{Tm1} is the saturation current at the measurement temperature.
- IS_{Ts} is the saturation current at the simulation temperature. This is the saturation current value used in the bipolar transistor equations when temperature dependence is modeled.
- EG is the energy gap.
- k is the Boltzmann constant (1.3806503e–23 J/K).
- XTI is the saturation current temperature exponent.

Similar to α, you have two ways of specifying EG and XTI:

- If you parameterize the block from a datasheet, you have to specify the gate reverse current, I_gss, at a second measurement temperature. The block then calculates the value for EG based on this data and assuming a p-n junction nominal value of 3 for XTI.
- If you parameterize by specifying equation parameters, you have to provide the values for EG and XTI directly. This option gives you most flexibility to match device behavior, for example, if you have a graph of I_gss as a function of temperature. With this data you can use Simulink Design Optimization software to help tune the values for EG and XTI.

Thermal Port

The block has an optional thermal port, hidden by default. To expose the thermal port, right-click the block in your model, and then from the context menu select **Simscape** > **Block choices** > **Show thermal port**. This action displays the thermal port H on the block icon, and adds the **Thermal Port** tab to the block dialog box.

Use the thermal port to simulate the effects of generated heat and device temperature. For more information on using thermal ports and on the **Thermal Port** tab parameters, see "Simulating Thermal Effects in Semiconductors".

Basic Assumptions and Limitations

The model is based on the following assumptions:

- This block does not allow you to specify initial conditions on the junction capacitances. If you select the **Start simulation from steady state** option in the Solver Configuration block, the block solves the initial voltages to be consistent with the calculated steady state. Otherwise, voltages are zero at the start of the simulation.

- You may need to use nonzero ohmic resistance and junction capacitance values to prevent numerical simulation issues, but the simulation may run faster with these values set to zero.

- The block does not account for temperature-dependent effects on the junction capacitances.

- When you specify *I_dss* at a second measurement temperature, it must be quoted for the same working point (that is, the same drain current and gate-source voltage) as for the *I_dss* value on the **Main** tab. Inconsistent values for *I_dss* at the higher temperature will result in unphysical values for α and unrepresentative simulation results.

- You may need to tune the value of *BEX* to replicate the I_D-V_{GS} relationship (if available) for a given device. The value of *BEX* affects whether the I_D-V_{GS} curves for different temperatures cross each other, or not, for the ranges of I_D and V_{GS} considered.

Parameters

- "Main Tab" on page 1-461
- "Ohmic Resistance Tab" on page 1-463
- "Junction Capacitance Tab" on page 1-463
- "Temperature Dependence Tab" on page 1-464

Main Tab

Parameterization

Select one of the following methods for block parameterization:

- `Specify from a datasheet` — Provide parameters that the block converts to equations that describe the transistor. This is the default method.

- `Specify using equation parameters directly` — Provide equation parameters β, *IS*, V_{t0}, and λ.

Gate reverse current, I_gss

The reverse current that flows in the diode when the drain and source are short-circuited and a large positive gate-source voltage is applied. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is 5 nA.

Saturated drain current, I_dss

The current that flows when a large negative drain-source voltage is applied for a specified gate-source voltage. For a depletion-mode device, this gate-source voltage may be zero, in which case *I_dss* may be referred to as the zero-gate voltage drain current. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is -3 mA.

I_dss measurement point, [V_gs V_ds]

A vector of the values of V_{GS} and V_{DS} at which *I_dss* is measured. Normally V_{GS} is zero. V_{DS} should be greater than zero. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is [0 -15] V.

Small-signal parameters, [g_fs g_os]

A vector of the values of *g_fs* and *g_os*. *g_fs* is the forward transfer conductance, that is, the conductance for a fixed drain-source voltage. *g_os* is the output conductance, that is, the conductance for a fixed gate-source voltage. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is [2.5e+3 75] uS.

Small-signal measurement point [V_gs V_ds]

A vector of the values of V_{GS} and V_{DS} at which *g_fs* and *g_os* are measured. V_{DS} should be greater than zero. For depletion-mode devices, V_{GS} is typically zero. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is [0 -15] V.

Transconductance parameter

The derivative of drain current with respect to gate voltage. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter. The default value is `1e-04` A/V^2.

Saturation current

The magnitude of the current that the ideal diode equation approaches asymptotically for very large reverse bias levels. This parameter is only visible when you select

Specify using equation parameters directly for the **Parameterization** parameter. The default value is `1e-14` A.

Threshold voltage

The gate-source voltage above which the transistor produces a nonzero drain current. For an enhancement device, Vt0 should be negative. For a depletion mode device, Vt0 should be positive. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter. The default value is 2 V.

Channel-length modulation

The channel-length modulation. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter. The default value is `0` 1/V.

Measurement temperature

The temperature for which the datasheet parameters are quoted. The default value is 25 °C.

Ohmic Resistance Tab

Source ohmic resistance

The transistor source resistance. The default value is `1e-4` Ω. The value must be greater than or equal to `0`.

Drain ohmic resistance

The transistor drain resistance. The default value is `0.01` Ω. The value must be greater than or equal to `0`.

Junction Capacitance Tab

Parameterization

Select one of the following methods for block parameterization:

- `Specify from a datasheet` — Provide parameters that the block converts to junction capacitance values. This is the default method.

- `Specify using equation parameters directly` — Provide junction capacitance parameters directly.

Input capacitance, Ciss

The gate-source capacitance with the drain shorted to the source. This parameter is only visible when you select `Specify from a datasheet` for the **Model junction capacitance** parameter. The default value is 4.5 pF.

Reverse transfer capacitance, Crss

The drain-gate capacitance with the source connected to ground. This parameter is only visible when you select `Specify from a datasheet` for the **Model junction capacitance** parameter. The default value is 1.5 pF.

Gate-source junction capacitance

The value of the capacitance placed between the gate and the source. This parameter is only visible when you select `Specify using equation parameters directly` for the **Model junction capacitance** parameter. The default value is 3 pF.

Gate-drain junction capacitance

The value of the capacitance placed between the gate and the drain. This parameter is only visible when you select `Specify using equation parameters directly` for the **Model junction capacitance** parameter. The default value is 1.5 pF.

Temperature Dependence Tab

Parameterization

Select one of the following methods for temperature dependence parameterization:

- `None — Simulate at parameter measurement temperature` — Temperature dependence is not modeled. This is the default method.

- `Model temperature dependence` — Model temperature-dependent effects. You also have to provide a set of additional parameters depending on the block parameterization method. If you parameterize the block from a datasheet, you have to provide values for *I_gss* and *I_dss* at second measurement temperature. If you parameterize by directly specifying equation parameters, you have to provide the values for *EG*, *XTI*, and the gate threshold voltage temperature coefficient, dV_{t0}/dT. Regardless of the block parameterization method, you also have to provide values for *BEX* and for the simulation temperature, T_s.

Gate reverse current, I_gss, at second measurement temperature

The value of the gate reverse current, *I_gss*, at the second measurement temperature. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter on the **Main** tab. It must be quoted for the same

working point (drain current and gate-source voltage) as the **Drain-source on resistance, R_DS(on)** parameter on the **Main** tab. The default value is 950 nA.

Saturated drain current, I_dss, at second measurement temperature

The value of the saturated drain current, *I_dss*, at the second measurement temperature, and when the *I_dss* measurement point is the same as defined by the **I_dss measurement point, [V_gs V_ds]** parameter on the **Main** tab. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter on the **Main** tab. The default value is -2.3 mA.

Second measurement temperature

Second temperature T_{m2} at which **Gate reverse current, I_gss, at second measurement temperature** and **Saturated drain current, I_dss, at second measurement temperature** are measured. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter on the **Main** tab. The default value is 125 °C.

Energy gap, EG

Energy gap value. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter on the **Main** tab. The default value is 1.11 eV.

Saturation current temperature exponent, XTI

Saturation current temperature coefficient value. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter on the **Main** tab. The default value is 3.

Gate threshold voltage temperature coefficient, dVt0/dT

The rate of change of gate threshold voltage with temperature. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter on the **Main** tab. The default value is 1 mV/K.

Mobility temperature exponent, BEX

Mobility temperature coefficient value. You can use the default value for most JFETs. See the "Basic Assumptions and Limitations" on page 1-461 section for additional considerations. The default value is -1.5.

Device simulation temperature

Temperature T_s at which the device is simulated. The default value is 25 °C.

Ports

The block has the following ports:

G

Electrical conserving port associated with the transistor gate terminal

D

Electrical conserving port associated with the transistor drain terminal

S

Electrical conserving port associated with the transistor source terminal

References

[1] H. Shichman and D. A. Hodges, *Modeling and simulation of insulated-gate field-effect transistor switching circuits*. IEEE J. Solid State Circuits, SC-3, 1968.

[2] G. Massobrio and P. Antognetti. *Semiconductor Device Modeling with SPICE*. 2nd Edition, McGraw-Hill, 1993. Chapter 2.

See Also

N-Channel JFET

P-Channel LDMOS FET

Model P-Channel LDMOS or VDMOS transistors suitable for high voltage

Library

Semiconductor Devices

Description

The P-Channel LDMOS FET block lets you model LDMOS (or VDMOS) transistors suitable for high voltage. The model is based on surface potential and includes effects due to an extended drain (drift) region:

- Nonlinear capacitive effects associated with the drift region
- Surface scattering and velocity saturation in the drift region
- Velocity saturation and channel-length modulation in the channel region
- Charge conservation inside the model, so you can use the model for charge sensitive simulations
- The intrinsic body diode
- Reverse recovery in the body diode model
- Temperature scaling of physical parameters
- For the thermal variant (see "Thermal Port" on page 1-469), dynamic self-heating

For information on physical background and defining equations, see the N-Channel LDMOS FET block reference page. Both the p-type and n-type versions of the LDMOS model use the same underlying code with appropriate voltage transformations, to account for the different device types.

The charge model is similar to that of the surface-potential-based MOSFET model, with additional expressions to account for the charge in the drift region. The block uses the

derived equations as described in [1], which include both inversion and accumulation in the drift region.

Modeling Body Diode

The block models the body diode as an ideal, exponential diode with both junction and diffusion capacitances:

$$I_{dio} = I_s \left[\exp\left(-\frac{V_{BD}}{n\phi_T} \right) - 1 \right]$$

$$C_j = \frac{C_{j0}}{\sqrt{1 + \dfrac{V_{BD}}{V_{bi}}}}$$

$$C_{diff} = \frac{\tau I_s}{n\phi_T} \exp\left(-\frac{V_{BD}}{n\phi_T} \right)$$

where:

- I_{dio} is the current through the diode.
- I_s is the reverse saturation current.
- V_{BD} is the body-drain voltage.
- n is the ideality factor.
- ϕ_T is the thermal voltage.
- C_j is the junction capacitance of the diode.
- C_{j0} is the zero-bias junction capacitance.
- V_{bi} is the built-in voltage.
- C_{diff} is the diffusion capacitance of the diode.
- τ is the transit time.

The capacitances are defined through an explicit calculation of charges, which are then differentiated to give the capacitive expressions above. The block computes the capacitive diode currents as time derivatives of the relevant charges, similar to the computation in the surface-potential-based MOSFET model.

Modeling Temperature Dependence

The default behavior is that dependence on temperature is not modeled, and the device is simulated at the temperature for which you provide block parameters. To model the dependence on temperature during simulation, select `Model temperature dependence` for the **Parameterization** parameter on the **Temperature Dependence** tab.

The model includes temperature effects on the capacitance characteristics, as well as modeling the dependence of the transistor static behavior on temperature during simulation.

The **Measurement temperature** parameter on the **Main** tab specifies temperature T_{m1} at which the other device parameters have been extracted. The **Temperature Dependence** tab provides the simulation temperature, T_s, and the temperature-scaling coefficients for the other device parameters. For more information, see "Temperature Dependence Tab" on page 1-474.

Thermal Port

The block has an optional thermal port, hidden by default. To expose the thermal port, right-click the block in your model, and then from the context menu select **Simscape > Block choices > Show thermal port**. This action displays the thermal port H on the block icon, and adds the **Thermal Port** tab to the block dialog box.

Use the thermal port to simulate the effects of generated heat and device temperature. For more information on using thermal ports and on the **Thermal Port** tab parameters, see "Simulating Thermal Effects in Semiconductors".

The thermal variant of the block includes dynamic self-heating, that is, lets you simulate the effect of self-heating on the electrical characteristics of the device.

Parameters

- "Main Tab" on page 1-470
- "Ohmic Resistance Tab" on page 1-472
- "Capacitances Tab" on page 1-473
- "Body Diode Tab" on page 1-473

Main Tab

Gain, [channel drift_region]

The gain, β, of the MOSFET regions. The parameter value is a two-element vector, with the first element corresponding to the channel, and the second — to the drift region. This parameter primarily defines the linear region of operation on an I_D–V_{DS} characteristic. The values of both elements must be greater than 0. The default value is `[11.6, 0.01]` A/V^2.

Flatband voltage, [channel drift_region]

The flatband voltage, V_{FB}, defines the gate bias that must be applied in order to achieve the flatband condition at the surface of the silicon. The parameter value is a two-element vector, with the first element corresponding to the channel, and the second — to the drift region. The default value is `[-1.05, -0.1]` V. You can also use this parameter to arbitrarily shift the threshold voltage due to material work function differences, and to trapped interface or oxide charges. In practice, however, it is usually recommended to modify the threshold voltage by using the **Body factor** and **Surface potential at strong inversion** parameters first, and only use this parameter for fine-tuning.

The threshold voltage for the channel region, for a short-circuited source-bulk connection, is approximately

$$-V_T = V_{FB} + 2\phi_B + 2\phi_T + \gamma\sqrt{2\phi_B + 2\phi_T}$$

where $2\phi_B$ is the surface potential at strong inversion and γ is the body factor, both at the channel region.

Body factor, [channel drift_region]

Body factor, γ, in the surface-potential equation. The parameter value is a two-element vector, with the first element corresponding to the channel, and the second — to the drift region. The default value is `[3.4, 2.5]` V$^{1/2}$.

For the channel region, the body factor is

$$\gamma = \frac{\sqrt{2q\varepsilon_{Si}N_A}}{C_{ox}}$$

See the N-Channel MOSFET block reference page for details on this equation. The drift region equation is similar, except that N_A is replaced by the doping density, N_D. The channel-region parameter value primarily impacts the threshold voltage. For the drift region, this parameter primarily affects the charge model, and also has a minor effect on the pinch-off behavior of the bulk current through the drift region.

Surface potential at strong inversion, [channel drift_region]

The $2\phi_B$ term in the surface-potential equation. The parameter value is a two-element vector, with the first element corresponding to the channel, and the second — to the drift region. The default value is [0.95, 0.95] V.

The channel-region parameter value also primarily impacts the threshold voltage. For the drift region, this parameter affects the charge model only.

Velocity saturation factor, [channel drift_region]

Velocity saturation, θ_{sat}, in the drain-current equation. Use this parameter in cases where a good fit to linear operation leads to a saturation current that is too high. By increasing this parameter value, you reduce the saturation current. The parameter value is a two-element vector, with the first element corresponding to the channel, and the second — to the drift region. The default value is [0.0, 0.1] 1/V, which means that velocity saturation in the channel region is off by default.

Drift region surface scattering factor

Surface scattering factor, θ_{surf}, in the drain-current equation. This parameter applies to the drift region only and accounts for scattering in the accumulation layer due to the vertical electric field. The default value is 0 1/V.

Channel-length modulation factor

The factor, α, multiplying the logarithmic term in the $G_{\Delta L}$ equation. See the N-Channel MOSFET block reference page for details on this equation. This parameter describes the onset of channel-length modulation. For device characteristics that exhibit a positive conductance in saturation, increase the parameter value to fit this behavior. This parameter applies to the channel region only. The default value is 0, which means that channel-length modulation is off by default.

Channel-length modulation voltage

The voltage V_p in the $G_{\Delta L}$ equation. See the N-Channel MOSFET block reference page for details on this equation. This parameter controls the drain-voltage at which channel-length modulation starts to become active. This parameter applies to the channel region only. The default value is 50 mV.

Linear-to-saturation transition coefficient

This parameter controls how smoothly the MOSFET transitions from linear into saturation, particularly when velocity saturation is enabled. This parameter can usually be left at its default value, but you can use it to fine-tune the knee of the I_D–V_{DS} characteristic. This parameter applies both to the channel and drift regions. The expected range for this parameter value is between 2 and 8. The default value is 8.

Measurement temperature

Temperature T_{m1} at which the block parameters are measured. If the **Device simulation temperature** parameter on the **Temperature Dependence** tab differs from this value, then device parameters will be scaled from their defined values according to the simulation and reference temperatures. For more information, see "Temperature Dependence Tab" on page 1-474. The default value is 25 °C.

Ohmic Resistance Tab

Source ohmic resistance

The transistor source resistance, that is, the series resistance associated with the source contact. The default value is 1e-4 Ω. The value must be greater than or equal to 0.

Drain ohmic resistance

The transistor drain resistance, that is, the series resistance associated with the drain contact and with the LOCOS part of the drift region, which is not heavily impacted by the applied gate voltage. The default value is 0.07 Ω. The value must be greater than or equal to 0.

Gate ohmic resistance

The transistor gate resistance, that is, the series resistance associated with the gate contact. The default value is 8.4 Ω. The value must be greater than or equal to 0.

Drift region low-bias resistance for gated region

Resistance R_D in the drain-current equation. It represents the resistance of the bulk part of the drift region in the absence of depletion from the top and bottom interfaces. The default value is 0.1 Ω. The value must be greater than or equal to 0.

Drift region depletion layer thickness factor

Parameter λ_D in the drain-current equation. It is the ratio of vertical depths y_1 and y_2 at zero bias, where y_1 represents the space-charge region and y_2 represents the undepleted part of the drift region. See the N-Channel LDMOS FET block reference page for an illustration. The default value is 0.2.

Capacitances Tab

Oxide capacitance

The parallel plate gate-channel and gate-drift-region capacitance. The parameter value is a two-element vector, with the first element corresponding to the channel, and the second — to the drift region. The default value is [1600.0, 1000.0] pF.

Gate-source overlap capacitance

The fixed, linear capacitance associated with the overlap of the gate electrode with the source well. The default value is 15 pF.

Gate-drain overlap capacitance

The fixed, linear capacitance associated with the overlap of the gate electrode with the drain well. The default value is 15 pF.

Body Diode Tab

Reverse saturation current

The current designated by the I_s symbol in the body-diode equations. The default value is 1e-13 A.

Built-in voltage

The built-in voltage of the diode, designated by the V_{bi} symbol in the body-diode equations. The default value is 0.6 V.

Ideality factor

The factor designated by the n symbol in the body-diode equations. The default value is 1.

Zero-bias junction capacitance

The capacitance between the drain and bulk contacts at zero-bias due to the body diode alone. It is designated by the C_{j0} symbol in the body-diode equations. The default value is 1800 pF.

Transit time

The time designated by the τ symbol in the body-diode equations. The default value is 50 ns.

Temperature Dependence Tab

Parameterization

Select one of the following methods for temperature dependence parameterization:

- None — Simulate at parameter measurement temperature — Temperature dependence is not modeled. This is the default method.
- Model temperature dependence — Model temperature-dependent effects. Provide a value for the device simulation temperature, T_s, and the temperature-scaling coefficients for other block parameters.

Device simulation temperature

Temperature T_s at which the device is simulated. The default value is 25 °C.

Gain temperature exponent, [channel drift_region]

The parameter value is a two-element vector, with the first element corresponding to the channel, and the second — to the drift region. Both in the channel and the drift region, the MOSFET gain, β, is assumed to scale exponentially with temperature, $\beta = \beta_{m1}(T_{m1}/T_s)^{\wedge}\eta_\beta$. β_{m1} is the value of the channel or drift region gain, as specified by the **Gain, [channel drift_region]** parameter from the **Main** tab. η_β is the corresponding element of the **Gain temperature exponent, [channel drift_region]** parameter. The default value is [1.3, 1.3].

Flatband voltage temperature coefficient, [channel drift_region]

The parameter value is a two-element vector, with the first element corresponding to the channel, and the second — to the drift region. The flatband voltage, V_{FB}, is assumed to scale linearly with temperature, $V_{FB} = V_{FBm1} + (T_s - T_{m1})S_{T,V_{FB}}$. V_{FBm1} is the value of the channel or drift region flatband voltage, as specified by the **Flatband voltage, [channel drift_region]** parameter from the **Main** tab. $S_{T,V_{FB}}$ is the corresponding element of the **Flatband voltage temperature coefficient, [channel drift_region]** parameter. The default value is [0.0005, 0.0005] V/K.

Surface potential at strong inversion temperature coefficient

The surface potential at strong inversion, $2\phi_B$, is assumed to scale linearly with temperature, $2\phi_B = 2\phi_{Bm1} + (T_s - T_{m1})S_{T,\phi_B}$. $2\phi_{Bm1}$ is the value of the **Surface potential at strong inversion** parameter from the **Main** tab and S_{T,ϕ_B} is the **Surface potential at strong inversion temperature coefficient**. The default value is -8.5e-4 V/K.

Velocity saturation temperature exponent, [channel drift_region]

The parameter value is a two-element vector, with the first element corresponding to the channel, and the second — to the drift region. The velocity saturation, θ_{sat}, is

assumed to scale exponentially with temperature, $\theta_{sat} = \theta_{sat,m1}(T_{m1}/T_s)\hat{}\eta_\theta$. $\theta_{sat,m1}$ is the value of the channel or drift region velocity saturation factor, as specified by the **Velocity saturation factor, [channel drift_region]** parameter from the **Main** tab. η_θ is the corresponding element of the **Velocity saturation temperature exponent, [channel drift_region]** parameter. The default value is [1.04, 1.04].

Ohmic resistance temperature exponent

The series resistances are assumed to correspond to semiconductor resistances. Therefore, they decrease exponentially with increasing temperature. $R_i = R_{i,m1}(T_{m1}/T_s)\hat{}\eta_R$, where i is S, D, or G, for the source, drain, or gate series resistance, respectively. $R_{i,m1}$ is the value of the corresponding series resistance parameter from the **Ohmic Resistance** tab and η_R is the **Ohmic resistance temperature exponent**. The default value is 0.95.

Drift region low-bias resistance temperature exponent for gated portion

Resistance R_D, the low-bias resistance of the bulk part of the drift region, scales similarly to the other series resistances. A separate value of the temperature exponent for this resistance provides an extra degree of freedom. The default value is 0.95.

Body diode reverse saturation current temperature exponent

The reverse saturation current for the body diode is assumed to be proportional to the square of the intrinsic carrier concentration, $n_i = N_C exp(-E_G/2k_BT)$. N_C is the temperature-dependent effective density of states and E_G is the temperature-dependent bandgap for the semiconductor material. To avoid introducing another temperature-scaling parameter, the block neglects the temperature dependence of the bandgap and uses the bandgap of silicon at 300K (1.12eV) for all device types. Therefore, the temperature-scaled reverse saturation current is given by

$$I_s = I_{s,m1}\left(\frac{T_s}{T_{m1}}\right)^{\eta_{Is}} \cdot exp\left(\frac{E_G}{k_B}\cdot\left(\frac{1}{T_{m1}} - \frac{1}{T_s}\right)\right).$$

$I_{s,m1}$ is the value of the **Reverse saturation current** parameter from the **Body Diode** tab, k_B is Boltzmann's constant (8.617x10-5eV/K), and η_{Is} is the **Body diode reverse saturation current temperature exponent**. The default value is 3, because N_C for silicon is roughly proportional to $T^{3/2}$. You can remedy the effect of neglecting the temperature-dependence of the bandgap by a pragmatic choice of η_{Is}.

Ports

The block has the following ports:

G

    Electrical conserving port associated with the transistor gate terminal

D

    Electrical conserving port associated with the transistor drain terminal

S

    Electrical conserving port associated with the transistor source terminal

References

[1] Aarts, A., N. D'Halleweyn, and R. Van Langevelde. "A Surface-Potential-Based High-Voltage Compact LDMOS Transistor Model." *IEEE Transactions on Electron Devices*. 52(5):999 - 1007. June 2005.

[2] Van Langevelde, R., A. J. Scholten, and D. B .M. Klaassen. "Physical Background of MOS Model 11. Level 1101." *Nat.Lab. Unclassified Report 2003/00239*. April 2003.

[3] Oh, S-Y., D. E. Ward, and R. W. Dutton. "Transient analysis of MOS transistors." *IEEE J. Solid State Circuits*. SC-15, pp. 636-643, 1980.

See Also
N-Channel LDMOS FET | N-Channel MOSFET

Topics
Interactive Generation of LDMOS Characteristics

Introduced in R2016b

P-Channel MOSFET

Model P-Channel MOSFET using either Shichman-Hodges equation or surface-potential-based model

Library

Semiconductor Devices

Description

The P-Channel MOSFET block provides two main modeling variants:

- Based on threshold voltage — Uses the Shichman-Hodges equation to represent the device. This modeling approach, based on threshold voltage, has the benefits of simple parameterization and simple current-voltage expressions. However, these models have difficulty in accurately capturing transitions across the threshold voltage and lack some important effects, such as velocity saturation. For details, see "Threshold-Based Model" on page 1-478.

- Based on surface potential — Uses the surface-potential equation to represent the device. This modeling approach provides a greater level of model fidelity than the simple square-law (threshold-voltage-based) models can provide. The trade-off is that there are more parameters that require extraction. For details, see "Surface-Potential-Based Model" on page 1-481.

Together with the thermal port variants (see "Thermal Port" on page 1-485), the block therefore provides you with four choices. To select the desired variant, right-click the block in your model. From the context menu, select **Simscape > Block choices**, and then one of the following options:

- **Threshold-based** — Basic model, which represents the device using the Shichman-Hodges equation (based on threshold voltage) and does not simulate thermal effects. This is the default.
- **Threshold-based with thermal** — Model based on threshold voltage and with exposed thermal port.
- **Surface-potential-based** — Model based on surface potential. This model does not simulate thermal effects.
- **Surface-potential-based with thermal** — Thermal variant of the model based on surface potential.

Threshold-Based Model

The threshold-based variant of the block uses the Shichman and Hodges equations [1] for an insulated-gate field-effect transistor to represent a P-Channel MOSFET.

The drain-source current, I_{DS}, depends on the region of operation:

- In the off region ($-V_{GS} < -V_{th}$) the drain-source current is:

$$I_{DS} = 0$$

- In the linear region ($0 < -V_{DS} < -V_{GS} + V_{th}$) the drain-source current is:

$$I_{DS} = -K \left((V_{GS} - V_{th})V_{DS} - V_{DS}{}^2 / 2 \right)\left(1 + \lambda |V_{DS}|\right)$$

- In the saturated region ($0 < -V_{GS} + V_{th} < -V_{DS}$) the drain-source current is:

$$I_{DS} = -(K / 2)(V_{GS} - V_{th})^2 \left(1 + \lambda |V_{DS}|\right)$$

In the preceding equations:

- K is the transistor gain.
- V_{DS} is the negative drain-source voltage.
- V_{GS} is the gate-source voltage.
- V_{th} is the threshold voltage.
- λ is the channel modulation.

1-478

Charge Model for Threshold-Based Variant

The block models junction capacitances either by fixed capacitance values, or by tabulated values as a function of the drain-source voltage. In either case, you can either directly specify the gate-source and gate-drain junction capacitance values, or let the block derive them from the input and reverse transfer capacitance values. Therefore, the **Parameterization** options for charge model on the **Junction Capacitance** tab are:

- `Specify fixed input, reverse transfer and output capacitance` — Provide fixed parameter values from datasheet and let the block convert the input and reverse transfer capacitance values to junction capacitance values, as described below. This is the default method.

- `Specify fixed gate-source, gate-drain and drain-source capacitance` — Provide fixed values for junction capacitance parameters directly.

- `Specify tabulated input, reverse transfer and output capacitance` — Provide tabulated capacitance and drain-source voltage values based on datasheet plots. The block converts the input and reverse transfer capacitance values to junction capacitance values, as described below.

- `Specify tabulated gate-source, gate-drain and drain-source capacitance` — Provide tabulated values for junction capacitances and drain-source voltage.
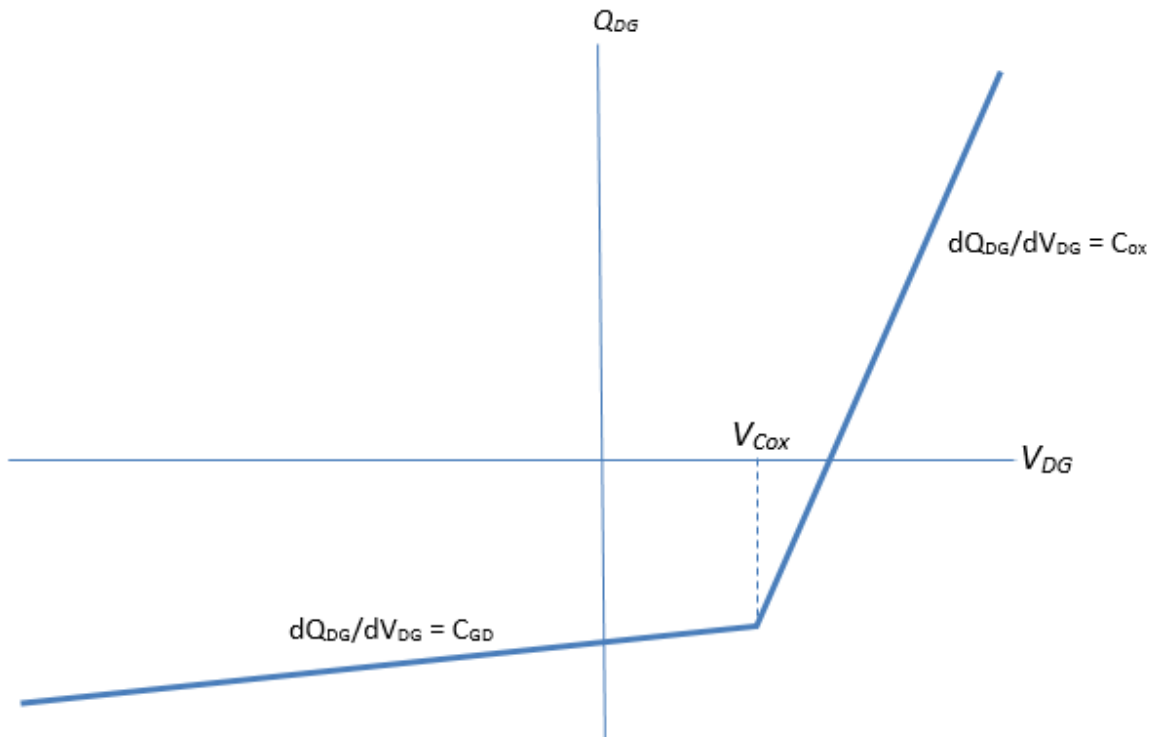
Use one of the tabulated capacitance options (`Specify tabulated input, reverse transfer and output capacitance` or `Specify tabulated gate-source, gate-drain and drain-source capacitance`) when the datasheet provides a plot of junction capacitances as a function of drain-source voltage. Using tabulated capacitance values gives more accurate dynamic characteristics and avoids the need for iterative tuning of parameters to fit the dynamics.

If you use the `Specify fixed gate-source, gate-drain and drain-source capacitance` or `Specify tabulated gate-source, gate-drain and drain-source capacitance` option, the **Junction Capacitance** tab lets you specify the **Gate-drain junction capacitance**, **Gate-source junction capacitance**, and **Drain-source junction capacitance** parameter values (fixed or tabulated) directly. Otherwise, the block derives them from the **Input capacitance, Ciss**, **Reverse transfer capacitance, Crss**, and **Output capacitance, Coss** parameter values. These two parameterization methods are related as follows:

- $C_{GD} = Crss$

- $C_{GS} = Ciss - Crss$
- $C_{DS} = Coss - Crss$

The two fixed capacitance options (`Specify fixed input, reverse transfer and output capacitance` or `Specify fixed gate-source, gate-drain and drain-source capacitance`) let you model gate junction capacitance as a fixed gate-source capacitance C_{GS} and either a fixed or a nonlinear gate-drain capacitance C_{GD}. If you select the `Gate-drain charge function is nonlinear` option for the **Charge-voltage linearity** parameter, then the gate-drain charge relationship is defined by the piecewise-linear function shown in the following figure.

Nonlinear gate-drain charge function

For instructions on how to map a time response to device capacitance values, see the N-Channel IGBT block reference page. However, this mapping is only approximate because the Miller voltage typically varies more from the threshold voltage than in the case for the IGBT.

Note Because this block implementation includes a charge model, you must model the impedance of the circuit driving the gate to obtain representative turn-on and turn-off dynamics. Therefore, if you are simplifying the gate drive circuit by representing it as a controlled voltage source, you must include a suitable series resistor between the voltage source and the gate.
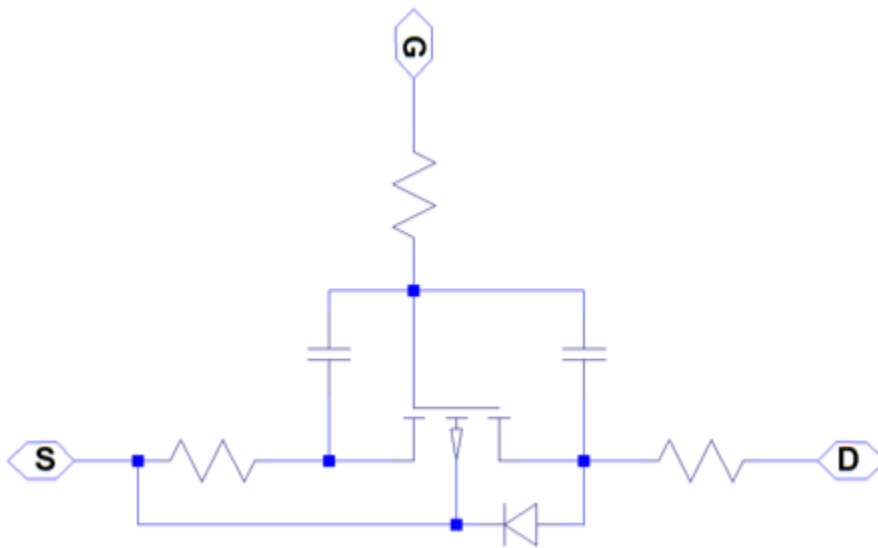

Surface-Potential-Based Model

The surface-potential-based variant of the block provides a greater level of model fidelity than the simple square-law (threshold-voltage-based) model. The surface-potential-based block variant includes the following effects:

- Fully nonlinear capacitance model (including the nonlinear Miller capacitance)
- Charge conservation inside the model, so you can use the model for charge sensitive simulations
- Velocity saturation and channel-length modulation
- The intrinsic body diode
- Reverse recovery in the body diode model
- Temperature scaling of physical parameters
- For the thermal variant, dynamic self-heating (that is, you can simulate the effect of self-heating on the electrical characteristics of the device)

This model is a minimal version of the world-standard PSP model (see http://nsti.org/Nanotech2005/WCM2005/WCM2005-GGildenblat.pdf), including only certain effects from the PSP model to strike a balance between model fidelity and complexity. For details of the physical background to the phenomena included in this model, see [2].

The surface-potential equation is derived similar to the way described on the N-Channel MOSFET block reference page, with all voltages, charges, and currents multiplied by -1.

The overall model consists of an intrinsic MOSFET defined by the surface-potential formulation, a body diode, series resistances, and fixed overlap capacitances, as shown in the schematic.

Modeling Body Diode

The block models the body diode as an ideal, exponential diode with both junction and diffusion capacitances:

$$I_{dio} = I_s \left[\exp\left(-\frac{V_{BD}}{n\phi_T} \right) - 1 \right]$$

$$C_j = \frac{C_{j0}}{\sqrt{1 + \dfrac{V_{BD}}{V_{bi}}}}$$

$$C_{diff} = \frac{\tau I_s}{n\phi_T} \exp\left(-\frac{V_{BD}}{n\phi_T} \right)$$

where:

- I_{dio} is the current through the diode.
- I_s is the reverse saturation current.
- V_{BD} is the body-drain voltage.
- n is the ideality factor.
- ϕ_T is the thermal voltage.
- C_j is the junction capacitance of the diode.
- C_{j0} is the zero-bias junction capacitance.
- V_{bi} is the built-in voltage.
- C_{diff} is the diffusion capacitance of the diode.
- τ is the transit time.

The capacitances are defined through an explicit calculation of charges, which are then differentiated to give the capacitive expressions above. The block computes the capacitive diode currents as time derivatives of the relevant charges, similar to the computation in the surface-potential-based MOSFET model.

Modeling Temperature Dependence

The default behavior is that dependence on temperature is not modeled, and the device is simulated at the temperature for which you provide block parameters. To model the dependence on temperature during simulation, select Model temperature dependence for the **Parameterization** parameter on the **Temperature Dependence** tab.

Threshold-Based Model

For threshold-based variant, you can include modeling the dependence of the transistor static behavior on temperature during simulation. Temperature dependence of the junction capacitances is not modeled, this being a much smaller effect.

When including temperature dependence, the transistor defining equations remain the same. The gain, K, and the threshold voltage, V_{th}, become a function of temperature according to the following equations:

$$K_{Ts} = K_{Tm1}\left(\frac{T_s}{T_{m1}}\right)^{BEX}$$

$$V_{\text{ths}} = V_{\text{th1}} + \alpha \ (\ T_{\text{s}} - T_{\text{m1}})$$

where:

- T_{m1} is the temperature at which the transistor parameters are specified, as defined by the **Measurement temperature** parameter value.
- T_{s} is the simulation temperature.
- K_{Tm1} is the transistor gain at the measurement temperature.
- K_{Ts} is the transistor gain at the simulation temperature. This is the transistor gain value used in the MOSFET equations when temperature dependence is modeled.
- V_{th1} is the threshold voltage at the measurement temperature.
- V_{ths} is the threshold voltage at the simulation temperature. This is the threshold voltage value used in the MOSFET equations when temperature dependence is modeled.
- *BEX* is the mobility temperature exponent. A typical value of *BEX* is -1.5.
- α is the gate threshold voltage temperature coefficient, dV_{th}/dT.

For most MOSFETS, you can use the default value of `-1.5` for *BEX*. Some datasheets quote the value for α, but most typically they provide the temperature dependence for drain-source on resistance, $R_{DS}(on)$. Depending on the block parameterization method, you have two ways of specifying α:

- If you parameterize the block from a datasheet, you have to provide $R_{DS}(on)$ at a second measurement temperature. The block then calculates the value for α based on this data.
- If you parameterize by specifying equation parameters, you have to provide the value for α directly.

If you have more data comprising drain current as a function of gate-source voltage for more than one temperature, then you can also use Simulink Design Optimization software to help tune the values for α and *BEX*.

Surface-Potential-Based Model

The surface-potential-based model includes temperature effects on the capacitance characteristics, as well as modeling the dependence of the transistor static behavior on temperature during simulation.

The **Measurement temperature** parameter on the **Main** tab specifies temperature T_{m1} at which the other device parameters have been extracted. The **Temperature**

Dependence tab provides the simulation temperature, T_s, and the temperature-scaling coefficients for the other device parameters. For more information, see "Temperature Dependence Tab (Surface-Potential-Based Variant)" on page 1-496.

Thermal Port

The block has an optional thermal port, hidden by default. To expose the thermal port, right-click the block in your model, and select the appropriate block variant:

- For a model based on threshold voltage and with exposed thermal port, select **Simscape** > **Block choices** > **Threshold-based with thermal**.

- For a thermal variant of the model based on surface potential, select **Simscape** > **Block choices** > **Surface-potential-based with thermal**.

This action displays the thermal port H on the block icon, and adds the **Thermal Port** tab to the block dialog box.

Use the thermal port to simulate the effects of generated heat and device temperature. For more information on using thermal ports and on the **Thermal Port** tab parameters, see "Simulating Thermal Effects in Semiconductors".

Basic Assumptions and Limitations

When modeling temperature dependence for threshold-based block variant, consider the following:

- The block does not account for temperature-dependent effects on the junction capacitances.

- When you specify $R_{DS}(on)$ at a second measurement temperature, it must be quoted for the same working point (that is, the same drain current and gate-source voltage) as for the other $R_{DS}(on)$ value. Inconsistent values for $R_{DS}(on)$ at the higher temperature will result in unphysical values for α and unrepresentative simulation results. Typically $R_{DS}(on)$ increases by a factor of about 1.5 for a hundred degree increase in temperature.

- You may need to tune the values of *BEX* and threshold voltage, V_{th}, to replicate the I_{DS}–V_{GS} relationship (if available) for a given device. Increasing V_{th} moves the I_{DS}–V_{GS} plots to the right. The value of *BEX* affects whether the I_{DS}–V_{GS} curves for different temperatures cross each other, or not, for the ranges of V_{DS} and V_{GS} considered.

1-485

Therefore, an inappropriate value can result in the different temperature curves appearing to be reordered. Quoting $R_{DS}(on)$ values for higher currents, preferably close to the current at which it will operate in your circuit, will reduce sensitivity to the precise value of *BEX*.

Parameters

- "Main Tab (Threshold-Based Variant)" on page 1-486
- "Main Tab (Surface-Potential-Based Variant)" on page 1-488
- "Ohmic Resistance Tab" on page 1-489
- "Junction Capacitance Tab" on page 1-490
- "Channel Capacitances Tab" on page 1-493
- "Body Diode Tab" on page 1-493
- "Temperature Dependence Tab (Threshold-Based Variant)" on page 1-494
- "Temperature Dependence Tab (Surface-Potential-Based Variant)" on page 1-496

Main Tab (Threshold-Based Variant)

This configuration of the **Main** tab corresponds to the threshold-based block variant, which is the default. If you are using the surface-potential-based variant of the block, see "Main Tab (Surface-Potential-Based Variant)" on page 1-488.

Parameterization

Select one of the following methods for block parameterization:

- Specify from a datasheet — Provide the drain-source on resistance and the corresponding drain current and gate-source voltage. The block calculates the transistor gain for the Shichman and Hodges equations from this information. This is the default method.

- Specify using equation parameters directly — Provide the transistor gain.

Drain-source on resistance, R_DS(on)

The ratio of the drain-source voltage to the drain current for specified values of drain current and gate-source voltage. $R_{DS}(on)$ should have a positive value. This parameter

is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is `0.167` Ω.

Drain current, Ids, for R_DS(on)

The drain current the block uses to calculate the value of the drain-source resistance. I_{DS} should have a negative value. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is `-2.5` A.

Gate-source voltage, Vgs, for R_DS(on)

The gate-source voltage the block uses to calculate the value of the drain-source resistance. V_{GS} should have a negative value. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is `-4.5` V.

Gain, K

Positive constant gain coefficient for the Shichman and Hodges equations. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter. The default value is 2 A/V^2.

Gate-source threshold voltage, Vth

Gate-source threshold voltage V_{th} in the Shichman and Hodges equations. For an enhancement device, V_{th} should be negative. For a depletion mode device, V_{th} should be positive. The default value is `-1.4` V.

Channel modulation, L

The channel-length modulation, usually denoted by the mathematical symbol λ. When in the saturated region, it is minus the rate of change of drain current with drain-source voltage. The effect on drain current is typically small, and the effect is neglected if calculating transistor gain *K* from drain-source on-resistance, $R_{DS}(on)$. A typical value is 0.02, but the effect can be ignored in most circuit simulations. However, in some circuits a small nonzero value may help numerical convergence. The default value is `0` 1/V.

Measurement temperature

Temperature T_{m1} at which **Drain-source on resistance, R_DS(on)** is measured. The default value is `25` °C.

Main Tab (Surface-Potential-Based Variant)

This configuration of the **Main** tab corresponds to the surface-potential-based block variant. If you are using the threshold-based variant of the block, based on the Shichman and Hodges equations, see "Main Tab (Threshold-Based Variant)" on page 1-486.

Gain

The MOSFET gain, β. This parameter primarily defines the linear region of operation on an I_D–V_{DS} characteristic. The value must be greater than 0. The default value is 18 A/V^2.

Flatband voltage

The flatband voltage, V_{FB}, defines the gate bias that must be applied in order to achieve the flatband condition at the surface of the silicon. The default value is -1.1 V. You can also use this parameter to arbitrarily shift the threshold voltage due to material work function differences, and to trapped interface or oxide charges. In practice, however, it is usually recommended to modify the threshold voltage by using the **Body factor** and **Surface potential at strong inversion** parameters first, and only use this parameter for fine-tuning.

Body factor

Body factor, γ, in the surface-potential equation. This parameter primarily impacts the threshold voltage. The default value is 3.5 V$^{1/2}$.

Surface potential at strong inversion

The $2\phi_B$ term in the surface-potential equation. This parameter also primarily impacts the threshold voltage. The default value is 1 V.

Velocity saturation factor

Velocity saturation, θ_{sat}, in the drain-current equation. Use this parameter in cases where a good fit to linear operation leads to a saturation current that is too high. By increasing this parameter value, you reduce the saturation current. For high-voltage devices, it is often the case that a good fit to linear operation leads to a saturation current that is too low. In such a case, either increase both the gain and the drain ohmic resistance or use a P-Channel LDMOS FET block instead. The default value is 0.4 1/V.

Channel-length modulation factor

The factor, α, multiplying the logarithmic term in the $G_{\Delta L}$ equation. This parameter describes the onset of channel-length modulation. For device characteristics that exhibit a positive conductance in saturation, increase the parameter value to fit this

behavior. The default value is 0, which means that channel-length modulation is off by default.

Channel-length modulation voltage

The voltage V_p in the $G_{\Delta L}$ equation. This parameter controls the drain-voltage at which channel-length modulation starts to become active. The default value is 50 mV.

Surface roughness scattering factor

Indicates the strength of the mobility reduction. The mobility is $\mu = \mu_0/G_{mob}$, where μ_0 is the low-field mobility without the effect of surface scattering. The mobility

reduction factor, G_{mob}, is given by $G_{mob} = \sqrt{1 + \left(\theta_{sr} V_{eff} \right)^2}$, where θ_{sr} is the surface roughness scattering factor and V_{eff} is a voltage that is indicative of the effective vertical electric field strength in the channel, E_{eff}. For high vertical electric fields, the mobility is roughly proportional to E_{eff} for holes. The default parameter value is 0 1/V.

Linear-to-saturation transition coefficient

This parameter controls how smoothly the MOSFET transitions from linear into saturation, particularly when velocity saturation is enabled. This parameter can usually be left at its default value, but you can use it to fine-tune the knee of the I_D–V_{DS} characteristic. The expected range for this parameter value is between 2 and 8. The default value is 8.

Measurement temperature

Temperature T_{m1} at which the block parameters are measured. If the **Device simulation temperature** parameter on the **Temperature Dependence** tab differs from this value, then device parameters will be scaled from their defined values according to the simulation and reference temperatures. For more information, see "Temperature Dependence Tab (Surface-Potential-Based Variant)" on page 1-496. The default value is 25 °C.

Ohmic Resistance Tab

Source ohmic resistance

The transistor source resistance, that is, the series resistance associated with the source contact. The value must be greater than or equal to 0. The default value for threshold-based variants is 1e-4 Ω. The default value for surface-potential-based variants is 2e-3 Ω.

Drain ohmic resistance

The transistor drain resistance, that is, the series resistance associated with the drain contact. The value must be greater than or equal to `0`. The default value for threshold-based variants is `0.01` Ω. The default value for surface-potential-based variants is `0.17` Ω.

Gate ohmic resistance

The transistor gate resistance, that is, the series resistance associated with the gate contact. This parameter is visible only for the surface-potential-based block variants. The value must be greater than or equal to `0`. The default value is `8.4` Ω.

Junction Capacitance Tab

This tab is visible only for the threshold-based variant of the block.

Parameterization

Select one of the following methods for capacitance parameterization:

- `Specify fixed input, reverse transfer and output capacitance` — Provide fixed parameter values from datasheet and let the block convert the input, output, and reverse transfer capacitance values to junction capacitance values, as described in "Charge Model for Threshold-Based Variant" on page 1-479. This is the default method.

- `Specify fixed gate-source, gate-drain and drain-source capacitance` — Provide fixed values for junction capacitance parameters directly.

- `Specify tabulated input, reverse transfer and output capacitance` — Provide tabulated capacitance and source-drain voltage values based on datasheet plots. The block converts the input, output, and reverse transfer capacitance values to junction capacitance values, as described in "Charge Model for Threshold-Based Variant" on page 1-479.

- `Specify tabulated gate-source, gate-drain and drain-source capacitance` — Provide tabulated values for junction capacitances and source-drain voltage.

Input capacitance, Ciss

The gate-source capacitance with the drain shorted to the source. This parameter is visible only for the following two values for the **Parameterization** parameter:

- If you select `Specify fixed input, reverse transfer and output capacitance`, the default value is `182 pF`.

- If you select `Specify tabulated input, reverse transfer and output capacitance`, the default value is [225 210 200 185 175 170] pF.

Reverse transfer capacitance, Crss

The drain-gate capacitance with the source connected to ground. This parameter is visible only for the following two values for the **Parameterization** parameter:

- If you select `Specify fixed input, reverse transfer and output capacitance`, the default value is 24 pF.

- If you select `Specify tabulated input, reverse transfer and output capacitance`, the default value is [75 60 50 35 25 20] pF.

Output capacitance, Coss

The drain-source capacitance with the gate and source shorted. This parameter is visible only for the following two values for the **Parameterization** parameter:

- If you select `Specify fixed input, reverse transfer and output capacitance`, the default value is 0 pF.

- If you select `Specify tabulated input, reverse transfer and output capacitance`, the default value is [180 160 125 80 60 45] pF.

Gate-source junction capacitance

The value of the capacitance placed between the gate and the source. This parameter is visible only for the following two values for the **Parameterization** parameter:

- If you select `Specify fixed gate-source, gate-drain and drain-source capacitance`, the default value is 158 pF.

- If you select `Specify tabulated gate-source, gate-drain and drain-source capacitance`, the default value is [150 150 150 150 150 150] pF.

Gate-drain junction capacitance

The value of the capacitance placed between the gate and the drain. This parameter is visible only for the following two values for the **Parameterization** parameter:

- If you select `Specify fixed gate-source, gate-drain and drain-source capacitance`, the default value is 24 pF.

- If you select `Specify tabulated gate-source, gate-drain and drain-source capacitance`, the default value is [75 60 50 35 25 20] pF.

Drain-source junction capacitance

The value of the capacitance placed between the drain and the source. This parameter is visible only for the following two values for the **Parameterization** parameter:

- If you select `Specify fixed gate-source, gate-drain and drain-source capacitance`, the default value is `0` pF.
- If you select `Specify tabulated gate-source, gate-drain and drain-source capacitance`, the default value is `[105 100 75 45 35 25]` pF.

Corresponding source-drain voltages

The source-drain voltages corresponding to the tabulated capacitance values. This parameter is visible only for tabulated capacitance models (`Specify tabulated input, reverse transfer and output capacitance` or `Specify tabulated gate-source, gate-drain and output capacitance`). The default value is `[0.1 0.3 1 3 10 30]` V.

Gate-source voltage, Vgs, for tabulated capacitances

For tabulated capacitance models, this parameter controls the voltage dependence of the **Reverse transfer capacitance, Crss** or the **Gate-drain junction capacitance** parameter (depending on the selected parameterization option). These capacitances are a function of the drain-gate voltage. The block calculates drain-gate voltages by subtracting this gate-source voltage value from the negative of the values specified for the **Corresponding source-drain voltages** parameter ($Vdg = -Vsd - Vgs$). The default value is `0` V. This parameter is visible only for tabulated capacitance models (`Specify tabulated input, reverse transfer and output capacitance` or `Specify tabulated gate-source, gate-drain and output capacitance`).

Charge-voltage linearity

The two fixed capacitance options (`Specify fixed input, reverse transfer and output capacitance` or `Specify fixed gate-source, gate-drain and drain-source capacitance`) let you model gate junction capacitance as a fixed gate-source capacitance C_{GS} and either a fixed or a nonlinear gate-drain capacitance C_{GD}. Select whether the gate-drain capacitance is fixed or nonlinear:

- `Gate-drain capacitance is constant` — The capacitance value is constant and defined according to the selected parameterization option, either directly or derived from a datasheet. This is the default method.
- `Gate-drain charge function is nonlinear` — The gate-drain charge relationship is defined according to the piecewise-nonlinear function described in

"Charge Model for Threshold-Based Variant" on page 1-479. Two additional parameters appear to let you define the gate-drain charge function.

Gate-drain oxide capacitance

The gate-drain capacitance when the drain-gate voltage is less than the **Drain-gate voltage at which oxide capacitance becomes active** parameter value. This parameter is only visible when you select `Gate-drain charge function is nonlinear` for the **Charge-voltage linearity** parameter. The default value is `200` pF.

Drain-gate voltage at which oxide capacitance becomes active

The drain-gate voltage at which the drain-gate capacitance switches between off-state (C_{GD}) and on-state (C_{ox}) capacitance values. This parameter is only visible when you select `Gate-drain charge function is nonlinear` for the **Charge-voltage linearity** parameter. The default value is `0.5` V.

Channel Capacitances Tab

This tab is visible only for the surface-potential-based variant of the block.

Oxide capacitance

The parallel plate gate-channel capacitance. The default value is `1500` pF.

Gate-source overlap capacitance

The fixed, linear capacitance associated with the overlap of the gate electrode with the source well. The default value is `100` pF.

Gate-drain overlap capacitance

The fixed, linear capacitance associated with the overlap of the gate electrode with the drain well. The default value is `14` pF.

Body Diode Tab

Reverse saturation current

The current designated by the I_s symbol in the body-diode equations. The default value for threshold-based variant is `0` A. The default value for surface-potential-based variant is `5.2e-13` A.

Built-in voltage

The built-in voltage of the diode, designated by the V_{bi} symbol in the body-diode equations. Built-in voltage has an impact only on the junction capacitance equation. It does not affect the conduction current. The default value is `0.6` V.

Ideality factor

The factor designated by the *n* symbol in the body-diode equations. The default value is 1.

Zero-bias junction capacitance

The capacitance between the drain and bulk contacts at zero-bias due to the body diode alone. It is designated by the C_{j0} symbol in the body-diode equations. The default value for threshold-based variant is 0 pF. The default value for surface-potential-based variant is 480 pF.

Transit time

The time designated by the τ symbol in the body-diode equations. The default value is 50 ns.

Temperature Dependence Tab (Threshold-Based Variant)

This configuration of the **Temperature Dependence** tab corresponds to the threshold-based block variant, which is the default. If you are using the surface-potential-based variant of the block, see "Temperature Dependence Tab (Surface-Potential-Based Variant)" on page 1-496

Parameterization

Select one of the following methods for temperature dependence parameterization:

- None — Simulate at parameter measurement temperature — Temperature dependence is not modeled. This is the default method.

- Model temperature dependence — Model temperature-dependent effects. Provide a value for simulation temperature, T_s, a value for *BEX*, and a value for the measurement temperature T_{m1} (using the **Measurement temperature** parameter on the **Main** tab). You also have to provide a value for α using one of two methods, depending on the value of the **Parameterization** parameter on the **Main** tab. If you parameterize the block from a datasheet, you have to provide $R_{DS}(on)$ at a second measurement temperature, and the block will calculate α based on that. If you parameterize by specifying equation parameters, you have to provide the value for α directly.

Drain-source on resistance, R_DS(on), at second measurement temperature

The ratio of the drain-source voltage to the drain current for specified values of drain current and gate-source voltage at second measurement temperature. This parameter is only visible when you select Specify from a datasheet for the

Parameterization parameter on the **Main** tab. It must be quoted for the same working point (drain current and gate-source voltage) as the **Drain-source on resistance, R_DS(on)** parameter on the **Main** tab. The default value is `0.25 Ω`.

Second measurement temperature

Second temperature T_{m2} at which **Drain-source on resistance, R_DS(on), at second measurement temperature** is measured. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter on the **Main** tab. The default value is `125 °C`.

Gate threshold voltage temperature coefficient, dVth/dT

The rate of change of gate threshold voltage with temperature. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter on the **Main** tab. The default value is `2 mV/K`.

Mobility temperature exponent, BEX

Mobility temperature coefficient value. You can use the default value for most MOSFETs. See the "Basic Assumptions and Limitations" on page 1-485 section for additional considerations. The default value is `-1.5`.

Body diode reverse saturation current temperature exponent

The reverse saturation current for the body diode is assumed to be proportional to the square of the intrinsic carrier concentration, $n_i = N_C \exp(-E_G/2k_BT)$. N_C is the temperature-dependent effective density of states and E_G is the temperature-dependent bandgap for the semiconductor material. To avoid introducing another temperature-scaling parameter, the block neglects the temperature dependence of the bandgap and uses the bandgap of silicon at 300K (1.12eV) for all device types. Therefore, the temperature-scaled reverse saturation current is given by

$$I_s = I_{s,m1} \left(\frac{T_s}{T_{m1}} \right)^{\eta_{Is}} \cdot \exp\left(\frac{E_G}{k_B} \cdot \left(\frac{1}{T_{m1}} - \frac{1}{T_s} \right) \right).$$

$I_{s,m1}$ is the value of the **Reverse saturation current** parameter from the **Body Diode** tab, k_B is Boltzmann's constant (8.617x10-5eV/K), and η_{Is} is the **Body diode reverse saturation current temperature exponent**. The default value is 3, because N_C for silicon is roughly proportional to $T^{3/2}$. You can remedy the effect of neglecting the temperature-dependence of the bandgap by a pragmatic choice of η_{Is}.

Device simulation temperature

Temperature T_s at which the device is simulated. The default value is 25 °C.

Temperature Dependence Tab (Surface-Potential-Based Variant)

This configuration of the **Temperature Dependence** tab corresponds to the surface-potential-based block variant. If you are using the threshold-based variant of the block, see "Temperature Dependence Tab (Threshold-Based Variant)" on page 1-494

Parameterization

Select one of the following methods for temperature dependence parameterization:

- `None — Simulate at parameter measurement temperature` — Temperature dependence is not modeled. This is the default method.

- `Model temperature dependence` — Model temperature-dependent effects. Provide a value for the device simulation temperature, T_s, and the temperature-scaling coefficients for other block parameters.

Gain temperature exponent

The MOSFET gain, β, is assumed to scale exponentially with temperature, $\beta = \beta_\mathrm{m1}(T_\mathrm{m1}/T_\mathrm{s})^\wedge\eta_\beta$. β_m1 is the value of the **Gain** parameter from the **Main** tab and η_β is the **Gain temperature exponent**. The default value is `1.3`.

Flatband voltage temperature coefficient

The flatband voltage, V_FB, is assumed to scale linearly with temperature, $V_\mathrm{FB} = V_\mathrm{FBm1} + (T_\mathrm{s} - T_\mathrm{m1})S_\mathrm{T,V_{FB}}$. V_FBm1 is the value of the **Flatband voltage** parameter from the **Main** tab and $S_\mathrm{T,V_{FB}}$ is the **Flatband voltage temperature coefficient**. The default value is `5e-4` V/K.

Surface potential at strong inversion temperature coefficient

The surface potential at strong inversion, $2\phi_\mathrm{B}$, is assumed to scale linearly with temperature, $2\phi_\mathrm{B} = 2\phi_\mathrm{Bm1} + (T_\mathrm{s} - T_\mathrm{m1})S_\mathrm{T,\phi_B}$. $2\phi_\mathrm{Bm1}$ is the value of the **Surface potential at strong inversion** parameter from the **Main** tab and $S_\mathrm{T,\phi_B}$ is the **Surface potential at strong inversion temperature coefficient**. The default value is `-8.5e-4` V/K.

Velocity saturation temperature exponent

The velocity saturation, θ_sat, is assumed to scale exponentially with temperature, $\theta_\mathrm{sat} = \theta_\mathrm{sat,m1}(T_\mathrm{m1}/T_\mathrm{s})^\wedge\eta_\theta$. $\theta_\mathrm{sat,m1}$ is the value of the **Velocity saturation factor** parameter from the **Main** tab and η_θ is the **Velocity saturation temperature exponent**. The default value is `1.04`.

Surface roughness scattering temperature exponent

This parameter leads to a temperature-dependent reduction in the MOSFET transconductance at high gate voltage. The surface roughness scattering, θ_{sr}, is assumed to scale exponentially with temperature, $\theta_{sr} = \theta_{sr,m1}(T_{m1}/T_s)\char`^\eta_{sr}$. $\theta_{sr,m1}$ is the value of the **Surface roughness scattering factor** parameter from the **Main** tab and η_{sr} is the **Surface roughness scattering temperature exponent**. The default value is 0.65.

Resistance temperature exponent

The series resistances are assumed to correspond to semiconductor resistances. Therefore, they decrease exponentially with increasing temperature. $R_i = R_{i,m1}(T_{m1}/T_s)\char`^\eta_R$, where i is S, D, or G, for the source, drain, or gate series resistance, respectively. $R_{i,m1}$ is the value of the corresponding series resistance parameter from the **Ohmic Resistance** tab and η_R is the **Resistance temperature exponent**. The default value is 0.95.

Body diode reverse saturation current temperature exponent

The reverse saturation current for the body diode is assumed to be proportional to the square of the intrinsic carrier concentration, $n_i = N_C \exp(-E_G/2k_BT)$. N_C is the temperature-dependent effective density of states and E_G is the temperature-dependent bandgap for the semiconductor material. To avoid introducing another temperature-scaling parameter, the block neglects the temperature dependence of the bandgap and uses the bandgap of silicon at 300K (1.12eV) for all device types. Therefore, the temperature-scaled reverse saturation current is given by

$$I_s = I_{s,m1}\left(\frac{T_s}{T_{m1}}\right)^{\eta_{Is}} \cdot \exp\left(\frac{E_G}{k_B} \cdot \left(\frac{1}{T_{m1}} - \frac{1}{T_s}\right)\right).$$

$I_{s,m1}$ is the value of the **Reverse saturation current** parameter from the **Body Diode** tab, k_B is Boltzmann's constant (8.617x10-5eV/K), and η_{Is} is the **Body diode reverse saturation current temperature exponent**. The default value is 3, because N_C for silicon is roughly proportional to $T^{3/2}$. You can remedy the effect of neglecting the temperature-dependence of the bandgap by a pragmatic choice of η_{Is}.

Device simulation temperature

Temperature T_s at which the device is simulated. The default value is 25 °C.

Ports

The block has the following ports:

G

    Electrical conserving port associated with the transistor gate terminal

D

    Electrical conserving port associated with the transistor drain terminal

S

    Electrical conserving port associated with the transistor source terminal

References

[1] Shichman, H. and D. A. Hodges. "Modeling and simulation of insulated-gate field-effect transistor switching circuits." *IEEE J. Solid State Circuits*. SC-3, 1968.

[2] Van Langevelde, R., A. J. Scholten, and D. B .M. Klaassen. "Physical Background of MOS Model 11. Level 1101." *Nat.Lab. Unclassified Report 2003/00239*. April 2003.

[3] Oh, S-Y., D. E. Ward, and R. W. Dutton. "Transient analysis of MOS transistors." *IEEE J. Solid State Circuits*. SC-15, pp. 636-643, 1980.
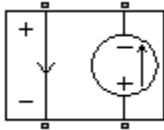
See Also
N-Channel MOSFET

Topics
Interactive Generation of MOSFET Characteristics

PCCCS

Model polynomial current-controlled current source

Library

SPICE-Compatible Components/Sources

Description

The PCCCS (Polynomial Current-Controlled Current Source) block represents a current source whose output current value is a polynomial function of the current through the input ports. The following equations describe the current through the source as a function of time:

- If you specify an *n*-element vector of polynomial coefficients for the **Polynomial coefficients** parameter:

$$I_{out} = p(0) + p(1) * I_{in} + ... + p(n-1) * I_{in}^{n-1} + p(n) * I_{in}^{n}$$

- If you specify a scalar coefficient for the **Polynomial coefficients** parameter:

$$I_{out} = p * I_{in}$$

where:

- I_{in} is the current through the input ports.
- p is the **Polynomial coefficients** parameter value.

The block uses a small conductance internally to prevent numerical simulation issues. The conductance connects the output ports of the device and has a conductance *GMIN*:

- By default, *GMIN* matches the **Minimum conductance GMIN** parameter of the SPICE Environment Parameters block, whose default value is 1e–12.
- To change *GMIN*, add a SPICE Environment Parameters block to your model and set the **Minimum conductance GMIN** parameter to the desired value.

Parameters

Polynomial coefficients

>   The polynomial coefficients that relate the input current to the output current, as described in the preceding section. The default value is [0 1].

Ports

The block has the following ports:

+

>   Positive electrical input voltage.

-

>   Negative electrical input voltage.

N+

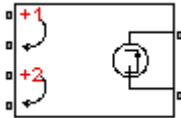>   Positive electrical output voltage.

N-

>   Negative electrical output voltage.

See Also

, PCCCS2PCCVS, PVCCS, and PVCVS

PCCCS2

Model polynomial current-controlled current source with two controlling inputs

Library

SPICE-Compatible Components/Sources

Description

The PCCCS2 (Two-Input Polynomial Current-Controlled Current Source) block represents a current source whose output current value is a polynomial function of the currents through the pairs of controlling input ports. The following equations describes the current through the source as a function of time:

$$I_{out} = p_1 + p_2 * I_{in1} + p_3 * I_{in2} + p_4 * I_{in1}^2 + p_5 I_{in1} * I_{in2} + p_6 * I_{in2}^2 + \ldots$$

where:

- I_{in1} is the current across the first pair of input ports.
- I_{in2} is the current across the second pair of input ports.
- p is the **Polynomial coefficients** parameter value.

The block uses a small conductance internally to prevent numerical simulation issues. The conductance connects the output ports of the device and has a conductance *GMIN*:

- By default, *GMIN* matches the **Minimum conductance GMIN** parameter of the SPICE Environment Parameters block, whose default value is 1e–12.
- To change *GMIN*, add a SPICE Environment Parameters block to your model and set the **Minimum conductance GMIN** parameter to the desired value.

Parameters

Polynomial coefficients

The polynomial coefficients that relate the input current to the output current, as described in the preceding section. The default value is [0 1 1].

Ports

The block has the following ports:

+1

Positive electrical input voltage of first controlling source.

-1

Negative electrical input voltage of first controlling source.

+2

Positive electrical input voltage of second controlling source.

-2

Negative electrical input voltage of second controlling source.

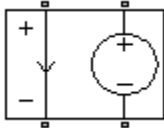N+

Positive electrical output voltage.

N-

Negative electrical output voltage.

See Also

PCCCS, PCCVS2, PVCCS2, and PVCVS2

PCCVS

Model polynomial current-controlled voltage source

Library

SPICE-Compatible Components/Sources

Description

The PCCVS (Polynomial Current-Controlled Voltage Source) block represents a voltage source whose output voltage value is a polynomial function of the current through the input ports. The following equations describe the voltage across the source as a function of time:

- If you specify an *n*-element vector of polynomial coefficients for the **Polynomial coefficients** parameter:

$$V_{out} = p(0) + p(1) * I_{in} + ... + p(n-1) * I_{in}^{n-1} + p(n) * I_{in}^{n}$$

- If you specify a scalar coefficient for the **Polynomial coefficients** parameter:

$$V_{out} = p * I_{in}$$

where:

- I_{in} is the current through the input ports.
- p is the **Polynomial coefficients** parameter value.

Parameters

Polynomial coefficients

The polynomial coefficients that relate the input current to the output voltage, as described in the preceding section. The default value is [0 1].

Ports

The block has the following ports:

+

Positive electrical input voltage.

-

Negative electrical input voltage.
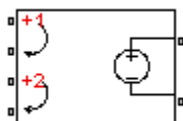
N+

Positive electrical output voltage.

N-

Negative electrical output voltage.

See Also

PCCCS, PCCVS2, PVCCS, and PVCVS

PCCVS2

Model polynomial current-controlled voltage source with two controlling inputs

Library

SPICE-Compatible Components/Sources

Description

The PCCVS2 (Two-Input Polynomial Current-Controlled Voltage Source) block represents a voltage source whose output voltage value is a polynomial function of the currents through the pairs of controlling input ports. The following equations describes the voltage across the source as a function of time:

$$V_{out} = p_1 + p_2 * I_{in1} + p_3 * I_{in2} + p_4 * I_{in1}^2 + p_5 I_{in1} * I_{in2} + p_6 * I_{in2}^2 + \dots$$

where:

- I_{in1} is the current across the first pair of input ports.
- I_{in2} is the current across the second pair of input ports.
- p is the **Polynomial coefficients** parameter value.

The block uses a small conductance internally to prevent numerical simulation issues. The conductance connects the output ports of the device and has a conductance *GMIN*:

- By default, *GMIN* matches the **Minimum conductance GMIN** parameter of the SPICE Environment Parameters block, whose default value is 1e–12.
- To change *GMIN*, add a SPICE Environment Parameters block to your model and set the **Minimum conductance GMIN** parameter to the desired value.

Parameters

Polynomial coefficients

> The polynomial coefficients that relate the input current to the output voltage, as described in the preceding section. The default value is [0 1 1].

Ports

The block has the following ports:

+1

> Positive electrical input voltage of first controlling source.

-1

> Negative electrical input voltage of first controlling source.

+2

> Positive electrical input voltage of second controlling source.

-2

> Negative electrical input voltage of second controlling source.

N+

> Positive electrical output voltage.

N-

> Negative electrical output voltage.

See Also

PCCCS2, PCCVS, PVCCS2, and PVCVS2

Photodiode

Model photodiode as parallel controlled current source and exponential diode

Library

Sensors

Description

The Photodiode block represents a photodiode as a controlled current source and an exponential diode connected in parallel. The controlled current source produces a current I_p that is proportional to the radiant flux density:

$I_\mathrm{p} = DeviceSensitivity \cdot RadiantFluxDensity$

where:

- *DeviceSensitivity* is the ratio of the current produced to the incident radiant flux density.

  - If you select `Specify measured current for given flux density` for the **Sensitivity parameterization** parameter, the block calculates this variable by converting the **Measured current** parameter value to units of amps and dividing it by the **Flux density** parameter values.

  - If you select `Specify current per unit flux density` for the **Sensitivity parameterization** parameter, this variable is defined by the **Device sensitivity** parameter value.

- *RadiantFluxDensity* is the incident radiant flux density.

To model dynamic response time, use the **Junction capacitance** parameter to include the diode junction capacitance in the model.

The exponential diode model provides the following relationship between the diode current I and the diode voltage V:

$$I = IS \cdot \left(e^{\frac{qV}{NkT_{m1}}} - 1 \right)$$

where:

- q is the elementary charge on an electron (1.602176e–19 Coulombs).
- k is the Boltzmann constant (1.3806503e–23 J/K).
- N is the emission coefficient.
- IS is the saturation current, which is equal to the **Dark current** parameter value.
- T_{m1} is the temperature at which the diode parameters are specified, as defined by the **Measurement temperature** parameter value.

When $(qV / NkT_{m1}) > 80$, the block replaces $e^{\frac{qV}{NkT_{m1}}}$ with $(qV / NkT_{m1} - 79)e^{80}$, which matches the gradient of the diode current at $(qV / NkT_{m1}) = 80$ and extrapolates linearly.

When $(qV / NkT_{m1}) < -79$, the block replaces $e^{\frac{qV}{NkT_{m1}}}$ with $(qV / NkT_{m1} + 80)e^{-79}$, which also matches the gradient and extrapolates linearly. Typical electrical circuits do not reach these extreme values. The block provides this linear extrapolation to help convergence when solving for the constraints during simulation.

When you select `Use dark current and N` for the **Diode parameterization** parameter, you specify the diode in terms of the **Dark current** and **Emission coefficient N** parameters. When you select `Use dark current plus a forward bias I-V data point` for the **Diode parameterization** parameter, you specify the **Dark current** parameter and a voltage and current measurement point on the diode I-V curve. The block calculates N from these values as follows:

$$N = V_F / (V_t \log(I_F / IS + 1))$$

where:

- V_F is the **Forward voltage VF** parameter value.

- $V_t = kT_{m1} / q$.
- I_F is the **Current IF at forward voltage VF** parameter value.

The exponential diode model provides the option to include a junction capacitance:

- When you select `Fixed or zero junction capacitance` for the **Junction capacitance** parameter, the capacitance is fixed.
- When you select `Use parameters CJO, VJ, M & FC` for the **Junction capacitance** parameter, the block uses the coefficients CJO, VJ, M, and FC to calculate a junction capacitance that depends on the junction voltage.
- When you select `Use C-V curve data points` for the **Junction capacitance** parameter, the block uses three capacitance values on the C-V capacitance curve to estimate CJO, VJ, and M and uses these values with the specified value of FC to calculate a junction capacitance that depends on the junction voltage. The block calculates CJO, VJ, and M as follows:

  -
  $$CJ0 = C_1((V_{R2} - V_{R1}) / (V_{R2} - V_{R1}(C_2 / C_1)^{-1/M}))^M$$

  -
  $$VJ = -(-V_{R2}(C_1 / C_2)^{-1/M} + V_{R1}) / (1 - (C_1 / C_2)^{-1/M})$$

  -
  $$M = \log(C_3 / C_2) / \log(V_{R2} / V_{R3})$$

  where:

  - V_{R1}, V_{R2}, and V_{R3} are the values in the **Reverse bias voltages [VR1 VR2 VR3]** vector.
  - C_1, C_2, and C_3 are the values in the **Corresponding capacitances [C1 C2 C3]** vector.

It is not possible to estimate FC reliably from tabulated data, so you must specify its value using the **Capacitance coefficient FC** parameter. In the absence of suitable data for this parameter, use a typical value of 0.5.

The reverse bias voltages (defined as positive values) should satisfy $V_{R3} > V_{R2} > V_{R1}$. This means that the capacitances should satisfy $C_1 > C_2 > C_3$ as reverse bias widens the depletion region and hence reduces capacitance. Violating these inequalities results in an error. Voltages V_{R2} and V_{R3} should be well away from the Junction potential VJ. Voltage V_{R1} should be less than the Junction potential VJ, with a typical value for V_{R1} being 0.1 V.

The voltage-dependent junction is defined in terms of the capacitor charge storage Q_j as:

- For $V < FC \cdot VJ$:

$$Q_j = CJ0 \cdot (VJ / (M-1)) \cdot ((1 - V/VJ)^{1-M} - 1)$$

- For $V \geq FC \cdot VJ$:

$$Q_j = CJ0 \cdot F_1 + (CJ0 / F_2) \cdot (F_3 \cdot (V - FC \cdot VJ) + 0.5(M/VJ) \cdot (V^2 - (FC \cdot VJ)^2))$$

where:

- $$F_1 = (VJ / (1-M)) \cdot (1 - (1 - FC)^{1-M})$$

- $$F_2 = (1 - FC)^{1+M})$$

- $$F_3 = 1 - FC \cdot (1 + M)$$

These equations are the same as used in [2], except that the temperature dependence of *VJ* and *FC* is not modeled. This model does not include the diffusion capacitance term that affects performance for high frequency switching applications.

The Photodiode block contains several options for modeling the dependence of the diode current-voltage relationship on the temperature during simulation. Temperature dependence of the junction capacitance is not modeled, this being a much smaller effect. For details, see the Diode reference page.

Thermal Port

The block has an optional thermal port, hidden by default. To expose the thermal port, right-click the block in your model, and then from the context menu select **Simscape > Block choices > Show thermal port**. This action displays the thermal port H on the block icon, and adds the **Thermal Port** tab to the block dialog box.

Use the thermal port to simulate the effects of generated heat and device temperature. For more information on using thermal ports and on the **Thermal Port** tab parameters, see "Simulating Thermal Effects in Semiconductors".

Variables

Use the **Variables** section of the block interface to set the priority and initial target values for the block variables prior to simulation. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

Basic Assumptions and Limitations

The Photodiode block has the following limitations:

- When you select `Use dark current plus a forward bias I-V curve data point` for the **Diode parameterization** parameter, choose a voltage near the diode turn-on voltage. Typically this will be in the range from 0.05 to 1 Volt. Using a value outside of this region may lead to a poor estimate for *N*.

- You may need to use nonzero ohmic resistance and junction capacitance values to prevent numerical simulation issues, but the simulation may run faster with these values set to zero.

Parameters

- "Main" on page 1-511
- "Junction Capacitance" on page 1-513
- "Temperature Dependence" on page 1-514

Main

Sensitivity parameterization

Select one of the following methods for sensitivity parameterization:

- `Specify measured current for given flux density` — Specify the measured current and the corresponding flux density. This is the default method.

- `Specify current per unit flux density` — Specify the device sensitivity directly.

Measured current

The current the block uses to calculate the device sensitivity. This parameter is only visible when you select `Specify measured current for given flux density` for the **Sensitivity parameterization** parameter. The default value is 25 µA.

Flux density

The flux density the block uses to calculate the device sensitivity. This parameter is only visible when you select `Specify measured current for given flux density` for the **Sensitivity parameterization** parameter. The default value is 5 W/m^2.

Device sensitivity

The current per unit flux density. This parameter is only visible when you select `Specify current per unit flux density` for the **Sensitivity parameterization** parameter. The default value is `5e-06` m^2*A/W.

Diode parameterization

Select one of the following methods for diode model parameterization:

- `Use dark current plus a forward bias I-V data point` — Specify the dark current and a point on the diode I-V curve. This is the default method.
- `Use dark current and N` — Specify dark current and emission coefficient.

Current I1

The current at the forward-biased point on the diode I-V curve that the block uses to calculate *IS* and *N*. This parameter is only visible when you select `Use dark current plus a forward bias I-V data point` for the **Diode parameterization** parameter. The default value is `0.1` A.

Voltage V1

The corresponding voltage at the forward-biased point on the diode I-V curve that the block uses to calculate *IS* and *N*. This parameter is only visible when you select and `Use dark current plus a forward bias I-V data point` for the **Diode parameterization** parameter. The default value is `1.3` V.

Dark current

The current through the diode when it is not exposed to light. The default value is `5e-9` A.

Emission coefficient, N

The diode emission coefficient or ideality factor. This parameter is only visible when you select `Use dark current and N` for the **Diode parameterization** parameter. The default value is 3.

Ohmic resistance, RS

The series diode connection resistance. The default value is 0.1 Ω.

Measurement temperature

The temperature at which the I-V curve or dark current was measured. The default value is 25 °C.

Junction Capacitance

Junction capacitance

Select one of the following options for modeling the junction capacitance:

- `Fixed or zero junction capacitance` — Model the junction capacitance as a fixed value.
- `Use C-V curve data points` — Specify measured data at three points on the diode C-V curve.
- `Use parameters CJ0, VJ, M & FC` — Specify zero-bias junction capacitance, junction potential, grading coefficient, and forward-bias depletion capacitance coefficient.

Zero-bias junction capacitance, CJ0

The value of the capacitance placed in parallel with the exponential diode term. This parameter is only visible when you select `Fixed or zero junction capacitance` or `Use parameters CJ0, VJ, M & FC` for the **Junction capacitance** parameter. The default value is 60 pF. When you select `Fixed or zero junction capacitance` for the **Junction capacitance** parameter, a value of zero omits junction capacitance.

Reverse bias voltages [VR1 VR2 VR3]

A vector of the reverse bias voltage values at the three points on the diode C-V curve that the block uses to calculate *CJ0*, *VJ*, and *M*. This parameter is only visible when you select `Use C-V curve data points` for the **Junction capacitance** parameter. The default value is [0.1 10 100] V.

Corresponding capacitances [C1 C2 C3]

A vector of the capacitance values at the three points on the diode C-V curve that the block uses to calculate *CJ0, VJ,* and *M*. This parameter is only visible when you select `Use C-V curve data points` for the **Junction capacitance** parameter. The default value is [45 30 6] pF.

Junction potential, VJ

The junction potential. This parameter is only visible when you select `Use parameters CJ0, VJ, M & FC` for the **Junction capacitance** parameter. The default value is 1 V.

Grading coefficient, M

The grading coefficient. This parameter is only visible when you select `Use parameters CJ0, VJ, M & FC` for the **Junction capacitance** parameter. The default value is 0.5.

Capacitance coefficient, FC

Fitting coefficient that quantifies the decrease of the depletion capacitance with applied voltage. This parameter is only visible when you select `Use C-V curve data points` or `Use parameters CJ0, VJ, M & FC` for the **Junction capacitance** parameter. The default value is 0.5.

Temperature Dependence

Parameterization

Select one of the following methods for temperature dependence parameterization:

- `None — Simulate at parameter measurement temperature` — Temperature dependence is not modeled, or the model is simulated at the measurement temperature T_{m1} (as specified by the **Measurement temperature** parameter on the **Main** tab). This is the default method.

- `Use an I-V data point at second measurement temperature T2` — If you select this option, you specify a second measurement temperature T_{m2}, and the current and voltage values at this temperature. The model uses these values, along with the parameter values at the first measurement temperature T_{m1}, to calculate the energy gap value.

- `Specify saturation current at second measurement temperature T2` — If you select this option, you specify a second measurement temperature T_{m2}, and saturation current value at this temperature. The model uses these values,

along with the parameter values at the first measurement temperature T_{m1}, to calculate the energy gap value.

- `Specify the energy gap EG` — Specify the energy gap value directly.

Current I1 at second measurement temperature

Specify the diode current *I1* value when the voltage is *V1* at the second measurement temperature. This parameter is only visible when you select `Use an I-V data point at second measurement temperature T2` for the **Parameterization** parameter. The default value is `0.07` A.

Voltage V1 at second measurement temperature

Specify the diode voltage *V1* value when the current is *I1* at the second measurement temperature. This parameter is only visible when you select `Use an I-V data point at second measurement temperature T2` for the **Parameterization** parameter. The default value is `1.3` V.

Saturation current, IS, at second measurement temperature

Specify the saturation current *IS* value at the second measurement temperature. This parameter is only visible when you select `Specify saturation current at second measurement temperature T2` for the **Parameterization** parameter. The default value is `2.5e-7` A.

Second measurement temperature

Specify the value for the second measurement temperature. This parameter is only visible when you select either `Use an I-V data point at second measurement temperature T2` or `Specify saturation current at second measurement temperature T2` for the **Parameterization** parameter. The default value is `125` °C.

Energy gap parameterization

This parameter is only visible when you select `Specify the energy gap EG` for the **Parameterization** parameter. It lets you select a value for the energy gap from a list of predetermined options, or specify a custom value:

- `Use nominal value for silicon (EG=1.11eV)` — This is the default.
- `Use nominal value for 4H-SiC silicon carbide (EG=3.23eV)`
- `Use nominal value for 6H-SiC silicon carbide (EG=3.00eV)`
- `Use nominal value for germanium (EG=0.67eV)`
- `Use nominal value for gallium arsenide (EG=1.43eV)`

1-515

- Use nominal value for selenium (EG=1.74eV)
- Use nominal value for Schottky barrier diodes (EG=0.69eV)
- Specify a custom value — If you select this option, the **Energy gap, EG** parameter appears in the dialog box, to let you specify a custom value for *EG*.

Energy gap, EG

Specify a custom value for the energy gap, *EG*. This parameter is only visible when you select Specify a custom value for the **Energy gap parameterization** parameter. The default value is 1.11 eV.

Saturation current temperature exponent parameterization

Select one of the following options to specify the saturation current temperature exponent value:

- Use nominal value for pn-junction diode (XTI=3) — This is the default.
- Use nominal value for Schottky barrier diode (XTI=2)
- Specify a custom value — If you select this option, the **Saturation current temperature exponent, XTI** parameter appears in the dialog box, to let you specify a custom value for *XTI*.

Saturation current temperature exponent, XTI

Specify a custom value for the saturation current temperature exponent, *XTI*. This parameter is only visible when you select Specify a custom value for the **Saturation current temperature exponent parameterization** parameter. The default value is 3.

Device simulation temperature

Specify the value for the temperature T_s, at which the device is to be simulated. The default value is 25 °C.

Ports

The block has the following ports:

D

Physical port representing incident flux

+

    Electrical conserving port associated with the diode positive terminal

-

   Electrical conserving port associated with the diode negative terminal

References

[1] MH. Ahmed and P.J. Spreadbury. *Analogue and digital electronics for engineers*. 2nd Edition, Cambridge University Press, 1984.

[2] G. Massobrio and P. Antognetti. *Semiconductor Device Modeling with SPICE*. 2nd Edition, McGraw-Hill, 1993.

See Also

Diode | Light-Emitting Diode | Optocoupler

Piezo Linear Motor

Model force-speed characteristics of linear piezoelectric traveling wave motor
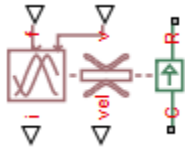
Library

Translational Actuators

Description

The Piezo Linear Motor block represents the force-speed characteristics of a linear piezoelectric traveling wave motor. The block represents the force-speed relationship of the motor at a level that is suitable for system-level modeling. To simulate the motor, the block uses the following models:

- "Mass and Friction Model for Unpowered Motor" on page 1-518
- "Resonant Circuit Model for Powered Motor" on page 1-519

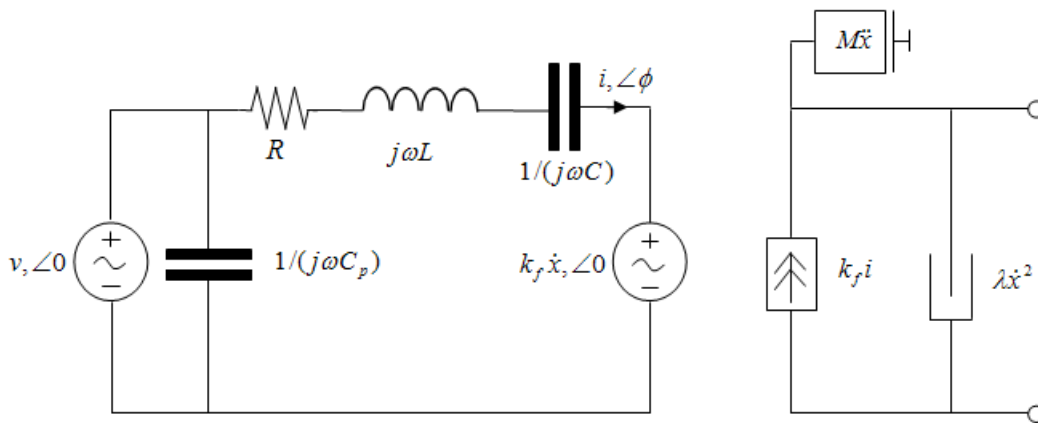### Mass and Friction Model for Unpowered Motor

The motor is unpowered when the physical signal input v is zero. This corresponds to applying zero RMS volts to the motor. In this scenario, the block models the motor using the following elements:

- A mass whose value is the **Plunger mass** parameter value.
- A friction whose characteristics you specify using the parameter values in the **Motor-Off Friction** tab.

The block uses a Simscape Translational Friction block to model the friction component. For detailed information about the friction model, see the Translational Friction block reference page.

Resonant Circuit Model for Powered Motor

When the motor is active, Piezo Linear Motor block represents the motor characteristics using the following equivalent circuit model.

In the preceding figure:

- The AC voltage source represents the block's physical signal input of frequency f and magnitude v.
- The resistor R provides the main electrical and mechanical damping term.
- The inductor L represents the rotor vibration inertia.
- The capacitor C represents the piezo crystal stiffness.
- The capacitor C_p represents the phase capacitance. This is the electrical capacitance associated with each of the two motor phases.
- The force constant k_f relates the RMS current i to the resulting mechanical force.
- The quadratic mechanical damping term, $\lambda \dot{x}^2$, shapes the force-speed curve predominantly at speeds close to maximum RPM. $\dot{x}$ is the linear speed.
- The term $M\dot{x}$ represents the plunger inertia.

At model initialization, the block calculates the model parameters R, L, C, k_t and λ to ensure that the steady-state force-speed curve matches the values for the following user-specified parameters:

- **Rated force**
- **Rated speed**
- **No-load maximum speed**
- **Maximum (stall) force**

These parameter values are defined for the **Rated RMS voltage** and **Motor natural frequency** (or rated frequency) parameter values.

The quadratic mechanical damping term produces a quadratic force-speed curve. Piezoelectric motors force-speed curves can typically be approximated more accurately using a quadratic function than a linear one because the force-speed gradient becomes steeper as the motor approaches the maximum speed.

If the plunger mass M is not specified on the datasheet, you can select a value that provides a good match to the quoted response time. The response time is often defined as the time for the rotor to reach maximum speed when starting from rest, under no-load conditions.

The quality factor that you specify using the **Resonance quality factor** parameter relates to the equivalent circuit model parameters as follows:

$$Q = \frac{1}{R}\sqrt{\frac{L}{C}}$$

This term is not usually provided on a datasheet. You can calculate its value by matching the sensitivity of force to driving frequency.

To reverse the motor direction of operation, make the physical signal input v negative.

Basic Assumptions and Limitations

The block has the following limitations:

- When the motor is powered, the model is valid only between zero and maximum speed, for the following reasons:

- Datasheets do not provide information for operation outside of normal range.
- Piezoelectric motors are not designed to operate in the powered braking and generating regions.

The block behaves as follows outside the valid operating region:

- Below zero speed, the model maintains a constant force with a zero speed value. The zero speed value is the **Maximum (stall) force** parameter value if the RMS input voltage equals the **Rated RMS voltage** parameter value, and the frequency input equals the **Motor natural frequency** parameter value.
- Above maximum speed, the model produces the negative force predicted by the equivalent circuit model, but limits the absolute value of the force to the zero-speed maximum force.

- The force-speed characteristics are most representative when operating the model close to the rated voltage and resonant frequency.

Parameters

- "Electrical Force Tab" on page 1-521
- "Mechanical Tab" on page 1-522
- "Motor-Off Friction Tab" on page 1-522

Electrical Force Tab

Motor natural frequency

Frequency at which the piezoelectric crystal naturally resonates. For most applications, set the input signal at port f to this frequency. To slow down the motor, for example in a closed-loop speed control, use a frequency slightly less than the motor natural frequency. The default value is 92 kHz.

Rated RMS voltage

Voltage at which the motor is designed to operate. The default value is 5.7 V.

Rated force

Force the motor delivers at the rated RMS voltage. The default value is 0.1 N.

Rated speed

Motor speed when the motor drives a load at the rated force. The default value is 50 mm/s.

No-load maximum speed

Motor speed when driving no load and powered at the rated voltage and driving frequency. The default value is 150 mm/s.

Maximum (stall) force

Maximum force the motor delivers when actively driving a load and powered at the rated voltage and frequency. The default value is 0.15 N.

Note The **Holding force** parameter value, the load force the motor holds when stationary, may be greater than the **Maximum (stall) force** parameter value.

Resonance quality factor

Quality factor *Q* that specifies how force varies as a function of driving frequency. Increasing the quality factor results in a much more rapid decrease in force as driving frequency is moved away from the natural frequency. The default value is 100.

Capacitance per phase

Electrical capacitance associated with each of the two motor phases. The default value is 5 nF.

Mechanical Tab

Plunger mass

Mass of the moving part of the motor. The default value is 0.3 g.

Initial rotor speed

Rotor speed at the start of the simulation. The default value is 0 mm/s.

Motor-Off Friction Tab

Holding force

The sum of the Coulomb and the static frictions. It must be greater than or equal to the **Coulomb friction force** parameter value. The default value is 0.3 N.

Coulomb friction force

The friction that opposes rotation with a constant force at any velocity. The default value is 0.15 N.

Viscous friction coefficient

Proportionality coefficient between the friction force and the relative velocity. The parameter value must be greater than or equal to zero. The default value is `1e-05` s*N/mm.

Transition approximation coefficient

The parameter sets the coefficient value that is used to approximate the transition between the static and the Coulomb frictions. For detailed information about the coefficient, c_v, see the Simscape Translational Friction block reference page. The default value is `0.1` s/mm.

Linear region velocity threshold

The parameter sets the small vicinity near zero velocity, within which friction force is considered to be linearly proportional to the relative velocity. MathWorks recommends that you use values between `1e-6` and `1e-4` mm/s. The default value is `0.1` mm/s.

Ports

The block has the following ports:

f

Physical signal input value specifying the motor driving frequency in Hz.

v

Physical signal input magnitude specifying the RMS supply voltage, and sign specifying the direction of rotation. If v is positive, then a positive force acts from port C to port R.

i

Physical signal output value that is the RMS phase current.

vel

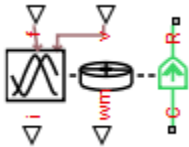Physical signal output value that is the linear speed of the rotor.

C

Mechanical translational conserving port.

R

Mechanical translational conserving port.

Piezo Rotary Motor

Model torque-speed characteristics of rotary piezoelectric traveling wave motor

Library

Rotational Actuators

Description

The Piezo Rotary Motor block represents the torque-speed characteristics of a piezoelectric traveling wave motor. The block represents the torque-speed relationship of the motor at a level that is suitable for system-level modeling. To simulate the motor, the block uses the following models:

- "Inertia and Friction Model for Unpowered Motor" on page 1-524
- "Resonant Circuit Model for Powered Motor" on page 1-525

Inertia and Friction Model for Unpowered Motor
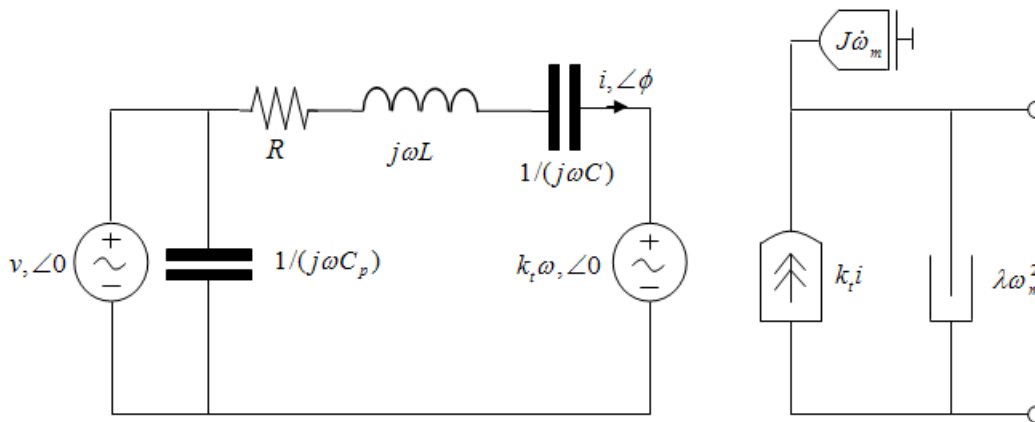
The motor is unpowered when the physical signal input *v* is zero. This corresponds to applying zero RMS volts to the motor. In this scenario, the block models the motor using the following elements:

- An inertia whose value is the **Rotor inertia** parameter value.
- A friction whose characteristics are determined by the parameter values in the **Motor-Off Friction** tab.

The block uses a Simscape Rotational Friction block to model the friction component. For detailed information about the friction model, see the Rotational Friction block reference page.

Resonant Circuit Model for Powered Motor

When the motor is active, Piezo Rotary Motor block represents the motor characteristics using the following equivalent circuit model.

In the preceding figure:

- The AC voltage source represents the block's physical signal input of frequency f and magnitude v.
- The resistor R provides the main electrical and mechanical damping term.
- The inductor L represents the rotor vibration inertia.
- The capacitor C represents the piezo crystal stiffness.
- The capacitor C_p represents the phase capacitance. This is the electrical capacitance associated with each of the two motor phases.
- The torque constant k_t relates the RMS current i to the resulting mechanical torque.
- The quadratic mechanical damping term, $\lambda \omega_m{}^2$, shapes the torque-speed curve predominantly at speeds close to maximum RPM. ω_m is the mechanical rotational speed.
- The term $J\dot{\omega}_m$ represents the rotor inertia.

At model initialization, the block calculates the model parameters R, L, C, k_t and λ to ensure that the steady-state torque-speed curve matches the values of the following user-specified parameter values:

- **Rated torque**
- **Rated rotational speed**
- **No-load maximum rotational speed**
- **Maximum torque**

These parameter values are defined for the **Rated RMS voltage** and **Motor natural frequency** (or rated frequency) parameter values.

The quadratic mechanical damping term produces a quadratic torque-speed curve. Piezoelectric motors torque-speed curves can typically be approximated more accurately using a quadratic function than a linear one because the torque-speed gradient becomes steeper as the motor approaches the maximum speed.

If the rotor inertia J is not specified on the datasheet, you can select a value that provides a good match to the quoted response time. The response time is often defined as the time for the rotor to reach maximum speed when starting from rest, under no-load conditions.

The quality factor that you specify using the **Resonance quality factor** parameter relates to the equivalent circuit model parameters as follows:

$$Q = \frac{1}{R}\sqrt{\frac{L}{C}}$$

This term is not usually provided on a datasheet. You can calculate its value by matching the sensitivity of torque to driving frequency.

To reverse the motor direction of operation, make the physical signal input v negative.

Basic Assumptions and Limitations

The block has the following limitations:

- When the motor is powered, the model is valid only between zero and maximum speed, for the following reasons:

- Datasheets do not provide information for operation outside of normal range.
- Piezoelectric motors are not designed to operate in the powered braking and generating regions.

The block behaves as follows outside the valid operating region:

- Below zero speed, the model maintains a constant torque that is the zero rpm torque value. The zero rpm torque value is the **Maximum torque** parameter value if the RMS input voltage equals the **Rated RMS voltage** parameter value, and the frequency input equals the **Motor natural frequency** parameter value.
- Above maximum speed, the model produces the negative torque predicted by the equivalent circuit model, but limits the absolute value of the torque to the zero-speed maximum torque.
- The torque-speed characteristics are most representative when operating the model close to the rated voltage and resonant frequency.

Parameters

- "Electrical Torque Tab" on page 1-527
- "Mechanical Tab" on page 1-528
- "Motor-Off Friction Tab" on page 1-528

Electrical Torque Tab

Motor natural frequency

Frequency at which the piezoelectric crystal naturally resonates. For most applications, set the input signal at port f to this frequency. To slow down the motor, for example in a closed-loop speed control, use a frequency slightly less than the motor natural frequency. The default value is 40 kHz.

Rated RMS voltage

Voltage at which the motor is designed to operate. The default value is 130 V.

Rated torque

Torque the motor delivers at the rated RMS voltage. The default value is 0.5 N*m.

Rated rotational speed

Motor speed when the motor drives a load at the rated torque. The default value is 100 rpm.

No-load maximum rotational speed

Motor rotational speed when driving no load and powered at the rated voltage and driving frequency. The default value is 160 rpm.

Maximum torque

Maximum torque that the motor delivers when actively driving a load and powered at the rated voltage and frequency. The default value is 1 N*m.

> **Note** The **Holding torque** parameter value, the load torque the motor holds when stationary, may be greater than the **Maximum torque** parameter value.

Resonance quality factor

Quality factor Q that specifies how torque varies as a function of driving frequency. Increasing the quality factor results in a much more rapid decrease in torque as driving frequency is moved away from the natural frequency. The default value is 100.

Capacitance per phase

Electrical capacitance associated with each of the two motor phases. The default value is 5 nF.

Mechanical Tab

Rotor inertia

Rotor resistance to change in motor motion. The default value is 200 g*cm^2.

Initial rotor speed

Rotor speed at the start of the simulation. The default value is 0 rpm.

Motor-Off Friction Tab

Holding torque

The sum of the Coulomb and the static frictions. It must be greater than or equal to the **Coulomb friction torque** parameter value. The default value is 1.5 N*m.

Coulomb friction torque

The friction that opposes rotation with a constant torque at any velocity. The default value is 1 N*m.

Viscous friction coefficient

Proportionality coefficient between the friction torque and the relative angular velocity. The parameter value must be greater than or equal to zero. The default value is `0.001` N*m/(rad*s).

Transition approximation coefficient

The parameter sets the coefficient value that is used to approximate the transition between the static and the Coulomb frictions. For detailed information about the coefficient, c_v, see the Simscape Rotational Friction block reference page. The default value is `10` s/rad.

Linear region velocity threshold

The parameter sets the small vicinity near zero velocity, within which friction torque is considered to be linearly proportional to the relative velocity. MathWorks recommends that you use values in the range between `1e-5` and `1e-3` rad/s. The default value is `1e-04` rad/s.

Ports

The block has the following ports:

f

Physical signal input value specifying the motor driving frequency in Hz.

v

Physical signal input magnitude specifying the RMS supply voltage, and sign specifying the direction of rotation. If v is positive, then a positive torque acts from port C to port R.

i

Physical signal output value that is the RMS phase current.

wm

Physical signal output value that is the rotational speed of the rotor.

C

Mechanical rotational conserving port.

R

Mechanical rotational conserving port.

Piezo Stack

Model electrical and force characteristics of piezoelectric stacked actuator
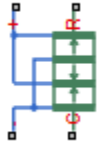
Library

Translational Actuators

Description

The Piezo Stack block represents the electrical and force characteristics of a piezoelectric stacked actuator using the following equations:

$$S = s^E T + d^{'} E$$
$$D = dT + \varepsilon^T E$$

where

- S is the strain tensor.
- T is the stress tensor.
- E is the electric field vector.
- D is the electric displacement vector.
- s^E is the elastic compliance matrix when subjected to a constant electric field.
- d is the piezoelectric constant matrix.
- ε^T is the permittivity measured at a constant stress.

Note The block models one-dimensional lumped parameter behavior, so S, T, E and D are all scalar values.

You can specify the block parameters that determine static force using either datasheet parameters or material properties, as determined by the value of the **Parameterization** parameter on the **Static Force** tab of the block dialog box.

The **Dynamic Forces** tab of the block dialog box lets you include optional effective mass and mechanical damping effects.

- If you specify a nonzero value for the **Effective mass** parameter or a finite value for the **Resonant frequency at constant field** parameter, the block attaches a lumped mass to the mechanical R port. When you specify a finite resonant frequency, the block calculates the effective mass to achieve the correct resonant frequency.

- If you specify a nonzero value for the **Damping** parameter or a finite value for the **Mechanical quality factor** parameter, the block adds a damping term across the R and C mechanical ports. When you specify a mechanical quality factor, Q_m, the block

  calculates the damping from this parameter value as $\sqrt{Mk}/Q_m$, where k is the short-circuit device stiffness, or equivalently the stiffness at constant field.

A positive voltage across the electrical + to – ports creates a positive displacement acting from the mechanical C to R ports.

Basic Assumptions and Limitations

The model does not include hysteresis effects.

Parameters

- "Static Force Tab" on page 1-532
- "Dynamic Forces Tab" on page 1-533
- "Initial Conditions Tab" on page 1-534

Static Force Tab

Parameterization

Select one of the following methods for static force parameterization:

- `Specify from a datasheet` — Provide datasheet parameters that the block converts to static force values. This is the default method.
- `Specify material properties` — Provide material properties that the block converts to static force values.

Stack area

Cross-sectional area of the stack. The default value is `100` mm^2.

Stack length

Stack length when no load and no electrical potential are applied. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is `36` mm.

No-load displacement at V0 volts

Unconstrained displacement of the stack when a voltage of V0 volts is applied. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is `0.038` mm.

Blocking force at V0 volts

Force the stack produces when a voltage of V0 volts is applied and the stack is physically prevented from expanding. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is `3.8e+03` N.

Test voltage V0

Voltage used to determine the no-load displacement and blocking force. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is `120` V.

Capacitance

This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is `13` uF.

Piezo layer thickness

Thickness of each layer in the piezo stack. This parameter is only visible when you select `Specify material properties` for the **Parameterization** parameter. The default value is `0.3` mm.

Number of layers

Number of layers in the piezo stack. This parameter is only visible when you select `Specify material properties` for the **Parameterization** parameter. The default value is `50`.

Piezoelectric charge constant

Mechanical strain per unit electric field applied. This parameter is only visible when you select `Specify material properties` for the **Parameterization** parameter. The default value is `5e-10` m/V.

Dielectric constant

Permittivity or dielectric displacement per unit electric field measured at constant stress. This parameter is only visible when you select `Specify material properties` for the **Parameterization** parameter. The default value is `2.124e-08` F/m.

Elastic compliance

Strain produced in a piezoelectric material per unit of stress applied. This parameter is only visible when you select `Specify material properties` for the **Parameterization** parameter. The default value is `1.9e-11` m^2/N.

Dynamic Forces Tab

Parameterization

Select one of the following methods for dynamic force parameterization:

- `Specify from a datasheet` — Provide datasheet parameters that the block converts to dynamic force values. This is the default method.
- `Specify material properties` — Provide material properties that the block converts to dynamic force values.

Resonant frequency at constant field

Frequency at which the actuator naturally resonates if mechanically perturbed with the electrical ports shorted. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is `Inf` kHz.

Mechanical quality factor

Factor that affects the damping across the R and C mechanical ports. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is `Inf`.

Damping

Translational damping term. This parameter is only visible when you select `Specify material properties` for the **Parameterization** parameter. The default value is `0` N/(m/s).

Effective mass

Mass that approximates the distributed dynamics of the device and causes the stack to resonate at the correct frequency when attached to the mechanical R port. This mass is usually about one third of the actual stack mass. This parameter is only visible when you select `Specify material properties` for the **Parameterization** parameter. The default value is `0` g.

Initial Conditions Tab

Initial stack deflection

Stack deflection at time zero. If you have an external Ideal Translational Motion Sensor block attached across the Piezo Stack block, you must use the same initial deflection parameter for both blocks. The default value is `0` mm.

Initial voltage

Stack voltage at time zero. The default value is `0` V.

Ports

The block has the following ports:

+

Positive electrical port

-

Negative electrical port

C

Mechanical translational conserving port

R

Mechanical translational conserving port

PNP Bipolar Transistor

Model PNP bipolar transistor using enhanced Ebers-Moll equations

Library

Semiconductor Devices

Description

The PNP Bipolar Transistor block uses a variant of the Ebers-Moll equations to represent an PNP bipolar transistor. The Ebers-Moll equations are based on two exponential diodes plus two current-controlled current sources. The PNP Bipolar Transistor block provides the following enhancements to that model:

- Early voltage effect
- Optional base, collector, and emitter resistances.
- Optional fixed base-emitter and base-collector capacitances.

The collector and base currents are [1]:

$$I_C = -IS \left[\left(e^{-qV_{BE}/(kT_{m1})} - e^{-qV_{BC}/(kT_{m1})} \right) \left(1 + \frac{V_{BC}}{V_A} \right) - \frac{1}{\beta_R} \left(e^{-qV_{BC}/(kT_{m1})} - 1 \right) \right]$$

$$I_B = -IS \left[\frac{1}{\beta_F} \left(e^{-qV_{BE}/(kT_{m1})} - 1 \right) + \frac{1}{\beta_R} \left(e^{-qV_{BC}/(kT_{m1})} - 1 \right) \right]$$

Where:

- I_B and I_C are base and collector currents, defined as positive into the device.
- *IS* is the saturation current.
- V_{BE} is the base-emitter voltage and V_{BC} is the base-collector voltage.
- β_F is the ideal maximum current gain BF
- β_R is the ideal maximum current gain BR
- V_A is the forward Early voltage VAF
- *q* is the elementary charge on an electron (1.602176e-19 Coulombs).
- *k* is the Boltzmann constant (1.3806503e-23 J/K).
- T_{m1} is the transistor temperature, as defined by the **Measurement temperature** parameter value.

You can specify the transistor behavior using datasheet parameters that the block uses to calculate the parameters for these equations, or you can specify the equation parameters directly.

If $-qV_{BC} / (kT_{m1}) > 40$ or $-qV_{BE} / (kT_{m1}) > 40$, the corresponding exponential terms in the equations are replaced with $(-qV_{BC} / (kT_{m1}) - 39)e^{40}$ and $(-qV_{BE} / (kT_{m1}) - 39)e^{40}$, respectively. This helps prevent numerical issues associated with the steep gradient of the exponential function e^x at large values of *x*. Similarly, if $-qV_{BC} / (kT_{m1}) < -39$ or $-qV_{BE} / (kT_{m1}) < -39$ then the corresponding exponential terms in the equations are replaced with $(-qV_{BC} / (kT_{m1}) + 40)e^{-39}$ and $(-qV_{BE} / (kT_{m1}) + 40)e^{-39}$, respectively.

Optionally, you can specify parasitic fixed capacitances across the base-emitter and base-collector junctions. You also have the option to specify base, collector, and emitter connection resistances.

Modeling Temperature Dependence

The default behavior is that dependence on temperature is not modeled, and the device is simulated at the temperature for which you provide block parameters. You can optionally include modeling the dependence of the transistor static behavior on temperature during simulation. Temperature dependence of the junction capacitances is not modeled, this being a much smaller effect.

When including temperature dependence, the transistor defining equations remain the same. The measurement temperature value, T_{m1}, is replaced with the simulation temperature, T_s. The saturation current, *IS*, and the forward and reverse gains (β_F and β_R) become a function of temperature according to the following equations:

$$IS_{Ts} = IS_{Tm1} \cdot (T_s / T_{m1})^{XTI} \cdot \exp\left(-\frac{EG}{kT_s}(1 - T_s / T_{m1}) \right)$$

$$\beta_{Fs} = \beta_{Fm1}\left(\frac{T_s}{T_{m1}} \right)^{XTB}$$

$$\beta_{Rs} = \beta_{Rm1}\left(\frac{T_s}{T_{m1}} \right)^{XTB}$$

where:

- T_{m1} is the temperature at which the transistor parameters are specified, as defined by the **Measurement temperature** parameter value.

- T_s is the simulation temperature.

- IS_{Tm1} is the saturation current at the measurement temperature.

- IS_{Ts} is the saturation current at the simulation temperature. This is the saturation current value used in the bipolar transistor equations when temperature dependence is modeled.

- β_{Fm1} and β_{Rm1} are the forward and reverse gains at the measurement temperature.

- β_{Fs} and β_{Rs} are the forward and reverse gains at the simulation temperature. These are the values used in the bipolar transistor equations when temperature dependence is modeled.

- EG is the energy gap for the semiconductor type measured in Joules. The value for silicon is usually taken to be 1.11 eV, where 1 eV is 1.602e-19 Joules.

- XTI is the saturation current temperature exponent.

- XTB is the forward and reverse gain temperature coefficient.

- k is the Boltzmann constant (1.3806503e–23 J/K).

Appropriate values for *XTI* and *EG* depend on the type of transistor and the semiconductor material used. In practice, the values of *XTI*, *EG*, and *XTB* need tuning to model the exact behavior of a particular transistor. Some manufacturers quote these tuned values in a SPICE Netlist, and you can read off the appropriate values. Otherwise you can determine values for *XTI*, *EG*, and *XTB* by using a datasheet-defined data at a higher temperature T_{m2}. The block provides a datasheet parameterization option for this.

You can also tune the values of *XTI*, *EG*, and *XTB* yourself, to match lab data for your particular device. You can use Simulink Design Optimization software to help tune the values.

Thermal Port

The block has an optional thermal port, hidden by default. To expose the thermal port, right-click the block in your model, and then from the context menu select **Simscape > Block choices > Show thermal port**. This action displays the thermal port H on the block icon, and adds the **Thermal Port** tab to the block dialog box.

Use the thermal port to simulate the effects of generated heat and device temperature. For more information on using thermal ports and on the **Thermal Port** tab parameters, see "Simulating Thermal Effects in Semiconductors".

Basic Assumptions and Limitations

The PNP Bipolar Transistor model has the following limitations:

*   The block does not account for temperature-dependent effects on the junction capacitances.
*   You may need to use nonzero ohmic resistance and junction capacitance values to prevent numerical simulation issues, but the simulation may run faster with these values set to zero.

Parameters

*   "Main Tab" on page 1-538
*   "Ohmic Resistance Tab" on page 1-540
*   "Capacitance Tab" on page 1-541
*   "Temperature Dependence Tab" on page 1-541

Main Tab

Parameterization

Select one of the following methods for block parameterization:

- `Specify from a datasheet` — Provide parameters that the block converts to equations that describe the transistor. The block calculates the forward Early voltage *VAF* as *Ic/h_oe*, where *Ic* is the **Collector current at which h-parameters are defined** parameter value, and *h_oe* is the **Output admittance h_oe** parameter value [2]. The block sets *BF* to the small-signal **Forward current transfer ratio h_fe** value. The block calculates the saturation current *IS* from the specified **Voltage Vbe** value and the corresponding **Current Ib for voltage Vbe** value when *Ic* is zero. This is the default method.

- `Specify using equation parameters directly` — Provide equation parameters *IS*, *BF*, and *VAF*.

Forward current transfer ratio h_fe

Small-signal current gain. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is `100`.

Output admittance h_oe

Derivative of the collector current with respect to the collector-emitter voltage for a fixed base current. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is `5e-5` 1/Ω.

Collector current at which h-parameters are defined

The h-parameters vary with operating point, and are defined for this value of the collector current. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is `-1` mA.

Collector-emitter voltage at which h-parameters are defined

The h-parameters vary with operating point, and are defined for this value of the collector-emitter voltage. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is `-5` V.

Voltage Vbe

Base-emitter voltage when the base current is *Ib*. The [*Vbe Ib*] data pair must be quoted for when the transistor is in the normal active region, that is, not in the saturated region. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is `-0.55` V.

Current Ib for voltage Vbe

Base current when the base-emitter voltage is *Vbe*. The [*Vbe Ib*] data pair must be quoted for when the transistor is in the normal active region, that is, not in the saturated region. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter. The default value is `-0.5` mA.

Forward current transfer ratio BF

Ideal maximum forward current gain. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter. The default value is `100`.

Saturation current IS

Transistor saturation current. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter. The default value is `1e-14` A.

Forward Early voltage VAF

In the standard Ebers-Moll equations, the gradient of the *Ic* versus *Vce* curve is zero in the normal active region. The additional forward Early voltage term increases this gradient. The intercept on the *Vce*-axis is equal to –*VAF* when the linear region is extrapolated. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter. The default value is `200` V.

Reverse current transfer ratio BR

Ideal maximum reverse current gain. This value is often not quoted in manufacturer datasheets because it is not significant when the transistor is biased to operate in the normal active region. When the value is not known and the transistor is not to be operated on the inverse region, use the default value of `1`.

Measurement temperature

Temperature T_{m1} at which *Vbe* and *Ib*, or *IS*, are measured. The default value is 25 °C.

Ohmic Resistance Tab

Collector resistance RC

Resistance at the collector. The default value is `0.01` Ω.

Emitter resistance RE

Resistance at the emitter. The default value is `1e-4` Ω.

Zero bias base resistance RB

Resistance at the base at zero bias. The default value is `1` Ω.

Capacitance Tab

Base-collector junction capacitance

Parasitic capacitance across the base-collector junction. The default value is 5 pF.

Base-emitter junction capacitance

Parasitic capacitance across the base-emitter junction. The default value is 5 pF.

Total forward transit time

Represents the mean time for the minority carriers to cross the base region from the emitter to the collector, and is often denoted by the parameter TF [1]. The default value is 0 µs.

Total reverse transit time

Represents the mean time for the minority carriers to cross the base region from the collector to the emitter, and is often denoted by the parameter TR [1]. The default value is 0µs.

Temperature Dependence Tab

Parameterization

Select one of the following methods for temperature dependence parameterization:

- `None — Simulate at parameter measurement temperature` — Temperature dependence is not modeled, or the model is simulated at the measurement temperature T_{m1} (as specified by the **Measurement temperature** parameter on the **Main** tab). This is the default method.

- `Model temperature dependence` — Provide a value for simulation temperature, to model temperature-dependent effects. You also have to provide a set of additional parameters depending on the block parameterization method. If you parameterize the block from a datasheet, you have to provide values for a second [*Vbe Ib*] data pair and *h_fe* at second measurement temperature. If you parameterize by directly specifying equation parameters, you have to provide the values for *XTI*, *EG*, and *XTB*.

Forward current transfer ratio, h_fe, at second measurement temperature

Small-signal current gain at second measurement temperature. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter on the **Main** tab. It must be quoted at the same collector-emitter voltage

and collector current as for the **Forward current transfer ratio h_fe** parameter on the **Main** tab. The default value is 125.

Voltage Vbe at second measurement temperature

Base-emitter voltage when the base current is *Ib* and the temperature is set to the second measurement temperature. The [*Vbe Ib*] data pair must be quoted for when the transistor is in the normal active region, that is, not in the saturated region. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter on the **Main** tab. The default value is -0.45 V.

Current Ib for voltage Vbe at second measurement temperature

Base current when the base-emitter voltage is *Vbe* and the temperature is set to the second measurement temperature. The [*Vbe Ib*] data pair must be quoted for when the transistor is in the normal active region, that is, not in the saturated region. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter on the **Main** tab. The default value is -0.5 mA.

Second measurement temperature

Second temperature T_{m2} at which *h_fe*,*Vbe*, and *Ib* are measured. This parameter is only visible when you select `Specify from a datasheet` for the **Parameterization** parameter on the **Main** tab. The default value is 125 °C.

Current gain temperature coefficient, XTB

Current gain temperature coefficient value. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter on the **Main** tab. The default value is 0.

Energy gap, EG

Energy gap value. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter on the **Main** tab. The default value is 1.11 eV.

Saturation current temperature exponent, XTI

Saturation current temperature coefficient value. This parameter is only visible when you select `Specify using equation parameters directly` for the **Parameterization** parameter on the **Main** tab. The default value is 3.

Device simulation temperature

Temperature T_s at which the device is simulated. The default value is 25 °C.

Ports

The block has the following ports:

B

Electrical conserving port associated with the transistor base terminal

C

Electrical conserving port associated with the transistor collector terminal

E

Electrical conserving port associated with the transistor emitter terminal

Examples

See the PNP Bipolar Transistor Characteristics example.

References

[1] G. Massobrio and P. Antognetti. *Semiconductor Device Modeling with SPICE*. 2nd Edition, McGraw-Hill, 1993.

[2] H. Ahmed and P.J. Spreadbury. *Analogue and digital electronics for engineers*. 2nd Edition, Cambridge University Press, 1984.

See Also

Diode | NPN Bipolar Transistor

Positive Supply Rail

Model ideal positive supply rail

Library

Sources

Description

The Positive Supply Rail block represents an ideal positive supply rail. Use this block instead of the Simscape DC Voltage Source block to define the output voltage relative to the Simscape Electrical Reference block that must appear in each model.

Note Do not attach more than one Positive Supply Rail block to any connected line.

Parameters

Constant voltage

The voltage at the output port relative to the Electrical Reference block ground port. The default value is 1 V.

Ports

The block has the following ports:

+

Positive electrical voltage

See Also

DC Voltage Source | Negative Supply Rail

Potentiometer

Model rotary or linear-travel potentiometer controlled by physical signal

Library

Passive Devices

Description

The Potentiometer block represents a rotary or linear-travel potentiometer, with the wiper position controlled by the input physical signal.

If the potentiometer resistance changes linearly based on wiper position, then the resistance between the wiper position and port L is:

$$R_{WL} = \frac{R_0}{x_{\max} - x_{\min}}(x - x_{\min})$$

where

- R_{WL} is the resistance between the wiper position and port L.
- R_0 is the total resistance between ports L and R.
- x is the wiper position.
- x_{min} is the value of the wiper position when the wiper is at port L.
- x_{max} is the value of the wiper position when the wiper is at port R.

If you specify LOG for the potentiometer resistance **Taper** parameter, then the resistance between the wiper position and port L is:

$$R_{WL} = \begin{cases} A\left(e^{\lambda(x-x_{\min})} - 1\right) & \text{if resistance gradient is higher at R} \\ R_0 - A\left(e^{\lambda(x_{\max}-x)} - 1\right) & \text{if resistance gradient is higher at L} \end{cases}$$

where A and λ are chosen such that R_{WL} at x_{max} is R_0, and R_{WL} at $x = (x_{max} + x_{min}) / 2$ is equal to R_{av}, the resistance when the wiper is centered.

Note Potentiometers widely described as LOG or logarithmic taper are, in fact, exponential taper. That is, the gradient of the resistance between wiper and left-hand port increases as the resistance increases. The Potentiometer block implements this behavior.

For both linear and logarithmic tapers, the resistance between the wiper position and port R is:

$$R_{WR} = R_0 - R_{WL}$$

where

- R_{WR} is the resistance between the wiper position and port R.
- R_0 is the total resistance between ports L and R.
- R_{WL} is the resistance between the wiper position and port L.

Parameters

Total resistance

The resistance between port L and port R when port W is open-circuit. The default value is 1000Ω.

Residual resistance

The lower limit placed on the resistance between the wiper and the two end ports. It must be greater than zero. A typical value is 5e-3 times the total resistance. The default value is 1Ω.

Resistance when centered

This parameter is available only if you select LOG for the **Taper** parameter. If you select Higher at R for the **Resistance gradient** parameter, then **Resistance when**

centered is the resistance between port L and port W when the wiper is centered. Otherwise, if you select `Higher at R` for the **Resistance gradient** parameter, then **Resistance when centered** is the resistance between port R and port W when the wiper is centered. Because the resistance taper is exponential in shape, the value of the **Resistance when centered** parameter must be less than half of the **Total resistance** parameter value. The default value is 200Ω.

PS input for wiper at L

The value of the input physical signal at port x that corresponds to the wiper being located at port L. The default value is 0.

PS input for wiper at R

The value of the input physical signal at port x that corresponds to the wiper being located at port R. The default value is 1.

Taper

Specifies the potentiometer resistance taper behavior: `LIN` (linear) or `LOG` (logarithmic). The default value is `LIN`.

Resistance gradient

Specifies whether the potentiometer resistance varies more rapidly at the left or the right end: `Higher at L` or `Higher at R`. This parameter is available only if you select `LOG` for the **Taper** parameter. The default value is `Higher at R`.

Ports

The block has the following ports:

L

Electrical port representing the left pin

R

Electrical port representing the right pin

W

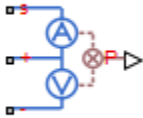Electrical port representing the wiper pin

x

Physical signal input port controlling the wiper position

See Also

Variable Resistor

Power Sensor

Measure instantaneous or cycle-average power

Library

Sensors

Description

The Power Sensor block calculates the power taken by the load connected across the + and - terminals under the assumption that only the load is connected to the + terminal. Refer to the block icon for the arrangement of internal current and voltage sensors.

The sensor can return either instantaneous power, or power averaged over a fixed time period. Use the latter option for periodic current and voltage waveforms such as those associated with PWM control.

The following figure shows how you connect the block to measure power dissipated in a resistor.
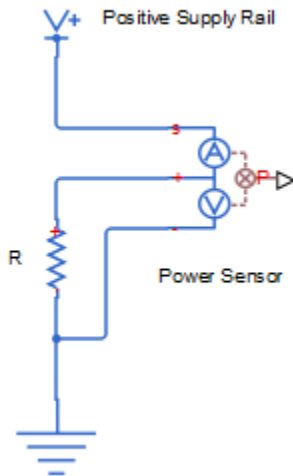
For an alternative workflow using data logging to view component powers, see the Buck Converter example.

Parameters

Measurement type

Select whether you want to measure `Instantaneous power` or `Average power over a specified period`. The default value is `Instantaneous power`.

Averaging period

The fixed period of time for measuring the average power. This parameter is only visible when you select `Average power over a specified period` for the **Measurement type** parameter. The default value is `1e-4` s.

Ports

The block has the following ports:

S

Electrical conserving port connected to the positive supply rail

+

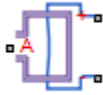   Electrical conserving port connected to the positive terminal of the load

-

    Electrical conserving port connected to the negative terminal of the load

P

    Physical signal port that outputs the measured power

Pressure Transducer

Model generic pressure transducer that turns pressure measurement into voltage

Library

Sensors

Description

The Pressure Transducer block models a generic pressure transducer that turns a pressure measurement into a voltage. The output voltage is linearly proportional to the pressure, and the block outputs zero volts if the pressure is less than zero. An input pressure equal to the **Pressure range** parameter value results in an output voltage equal to the **Full-scale deflection** parameter value. For higher pressures, the output voltage remains at this **Full-scale deflection** value.

You have three choices of operation mode, which let you select between vacuum, atmospheric pressure, or sealed-gauge reference pressure as the reference point for the pressure measurement.

Optionally, if you set the **Dynamics** parameter to `Model transducer bandwidth`, then the dynamics of the sensor are approximated by a first-order lag. The lag is determined by the **Bandwidth** parameter. If you select this option, you must also specify an initial condition for the lag by using the **Measured pressure** variable target.

If running your simulation with a fixed-step solver, or generating code for hardware-in-the-loop testing, MathWorks recommends that you set the **Dynamics** parameter to `No dynamics — Suitable for HIL`, because this avoids the need for a small simulation time step if the sensor bandwidth is high.

Variables

Use the **Variables** section of the block interface to set the priority and initial target values for the block variables prior to simulation. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

The **Measured pressure** variable target specifies the initial output for the sensor.

Parameters

Pressure range

The maximum pressure that the sensor can measure. The default value is `1e6` Pa.

Operation mode

Select one of the following options to define the reference point for the pressure measurement:

- `Absolute` — The pressure measurement is with respect to zero absolute pressure, that is, vacuum. This is the default option.
- `Gauge` — The pressure measurement is with respect to atmospheric pressure. Atmospheric pressure is defined by the Gas Properties block in the Simscape Foundation library.
- `Sealed-Gauge` — The pressure measurement is referenced to an internal sealed chamber. If you select this option, use the **Reference pressure** parameter to specify the reference point for pressure measurement.

Reference pressure

The reference pressure in the internal sealed chamber. This parameter is only visible when you select `Sealed-Gauge` for the **Operation mode** parameter. The default value is `1.01325e5` Pa.

Full-scape deflection

The output voltage when the measured pressure is equal to, or greater than, the **Pressure range** parameter value. The default value is 5 V.

Output resistance

The output resistance of the transducer. The default value is `200` Ω.

Dynamics

Select one of the following options for modeling sensor dynamics:

- `No dynamics — Suitable for HIL` — Do not model sensor dynamics. Use this option when running your simulation fixed step or generating code for hardware-in-the-loop testing, because this avoids the need for a small simulation time step if the sensor bandwidth is high. This is the default option.
- `Model transducer bandwidth` — Model sensor dynamics with a first-order lag approximation, based on the **Bandwidth** parameter value. You can control the initial condition for the lag by specifying the **Measured pressure** variable target.

Bandwidth

Determines the value of the sensor lag. This parameter is only visible when you select `Model transducer bandwidth` for the **Dynamics** parameter. The default value is 5 kHz.

Ports

The block has the following ports:

A

Pneumatic port

+

Positive electrical port

-

Negative electrical port

Proximity Sensor
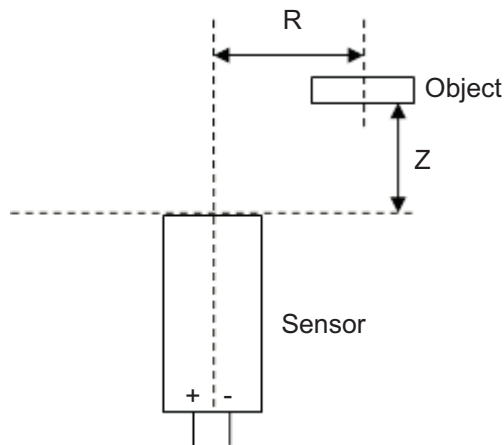
Model simple distance sensor

Library

Sensors

Description

The Proximity Sensor block represents a simple proximity sensor. The sensing distance Z is defined as the distance normal to the sensor surface at which the sensor detects an object for a given radial offset R, as shown in the following figure.

A typical sensing distance curve is shown in the following figure.

Distance Z
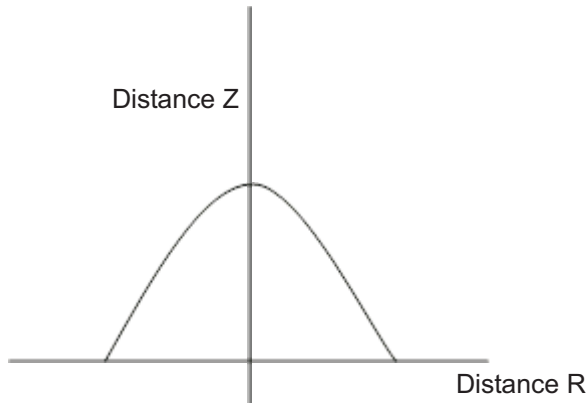
Distance R

The output is modeled by an electrical switch which can either be Normally Open (N.O.) or Normally Closed (N.C.) when no object is detected.

Parameters

Vector of radial offset distances R

Vector of distances from the sensor to the object resolved into a plane tangential to the sensor head. The default value is `[ -25 -20 -15 -10 -5 0 5 10 15 20 25 ]` mm.

Corresponding sensing distances Z

Vector of distances from the sensor to the object resolved with respect to a normal vector at the sensor head. The default value is `[ 0 0 5 8 9.5 10 9.5 8 5 0 0 ]` mm.

Output when not detected

Indicates whether the output is `Normally Open (N.O.)`, meaning the output becomes closed only when the object is detected, or `Normally Closed (N.C.)`, meaning the output becomes open only when the object is detected. The default value is `Normally Open (N.O.)`.

Closed resistance R_closed

The resistance between the + and - ports when the output contacts are closed. The default value is `0.01` Ω.

Open conductance G_open

The conductance between the + and - ports when the output contacts are open. The default value is `1e-08` 1/Ω.

Ports

The block has the following ports:

R

Radial distance to the sensor

Z

Perpendicular distance to the sensor

+

Positive electrical voltage

-

Negative electrical voltage

PS Sensor

Model generic linear sensor

Library

Sensors

Description

The PS Sensor block represents a generic linear sensor. The block converts the physical signal input U into an electrical output Y across the + and - ports. The **Output type** parameter value determines which of the following electrical outputs the block produces:

- Output voltage
- Output current
- Output resistance

Y is related to U as $Y = \max(\min(A \cdot U + B, Y_{max}), Y_{min})$, where Y_{min} and Y_{max} are minimum and maximum limits on the output, respectively.

Parameters

Output type

Indicates whether the sensor output is a `Variable voltage` of Y V, a `Variable current` of Y A, or `Variable resistor` with a value of Y Ω. The default value is `Variable voltage`.

Sensor gain, A

The sensitivity of the output Y with respect to the input U, dY/dU. The default value is 1.

Sensor offset, B

The output when the input U is zero. The output does not exceed the limits Y_{max} and Y_{min}. The default value is `0`.

Maximum output, Ymax

The upper limit on the sensor output. The following table shows the units of this parameter, which depend on the selected value of the **Output type** parameter.

Output type	Units
Variable voltage	V
Variable current	A
Variable resistor	Ω

The default value is 5.

Minimum output, Ymin

The lower limit on the sensor output. The following table shows the units of this parameter, which depend on the selected value of the **Output type** parameter.

Output type	Units
Variable voltage	V
Variable current	A
Variable resistor	Ω

The default value is `0.01`.

If you select `Variable resistance` for the **Output type** parameter, the minimum resistance Y_{min} must be greater than zero.

Ports

The block has the following ports:

U

Physical input signal

+

Positive electrical voltage

-

Negative electrical voltage

See Also

Controlled Current Source | Controlled Voltage Source | Variable Resistor

PTC Thermistor

Model switching type positive temperature coefficient (PTC) thermistor

Library

Sensors

Description

The PTC Thermistor block represents a switching type PTC thermistor. This type of thermistor has a decreasing resistance with temperature increasing up to the Curie temperature. Above the Curie temperature the resistance increases very rapidly with increasing temperature, as shown in the following plot. The region to the right of the Curie temperature is called the Positive Temperature Coefficient (PTC) regime. To represent a non-switching linear PTC thermistor, use the Thermal Resistor block.
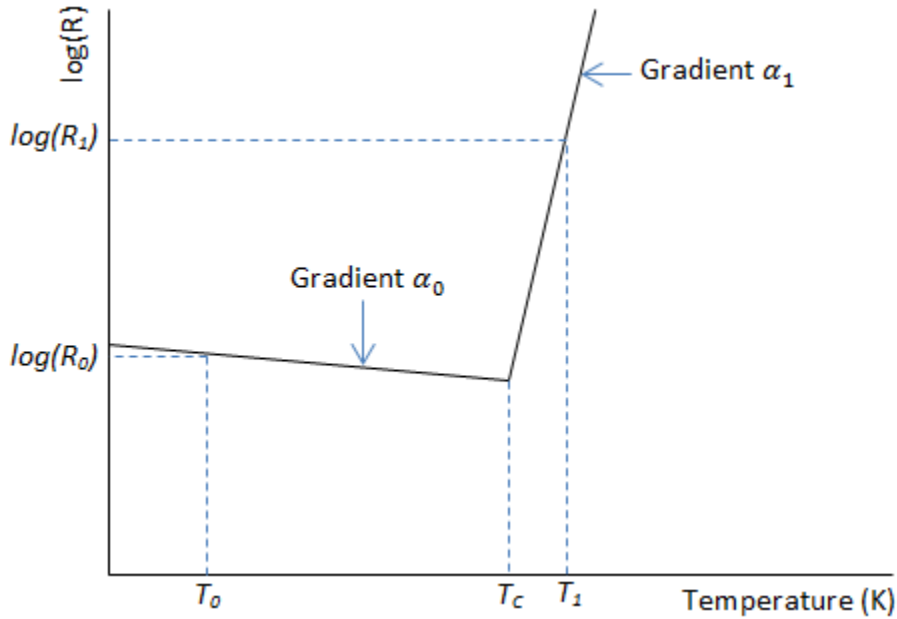
For a switching type PTC thermistor, the resistance R at temperature T is given by

$$R = \begin{cases} R_0 e^{\alpha_0 (T-T_0)} & \text{for } T < T_c \\ R_1 e^{\alpha_1 (T-T_1)} & \text{for } T \geq T_c \end{cases}$$

$$T_c = \frac{\log(R_1) - \log(R_0) + \alpha_0 T_0 - \alpha_1 T_1}{\alpha_0 - \alpha_1}$$

where:

- T_c is the Curie temperature.
- R_0 is the resistance at nominal temperature T_0.
- R_1 is the resistance at reference temperature T_1.
- T_0 is the nominal temperature at which the resistance is quoted, usually room temperature. T_0 is less than the Curie temperature T_c.

- T_1 is the reference temperature, equal or greater than the Curie temperature T_c, which means that at this temperature the PTC regime is in force.
- α_0 is the temperature coefficient at nominal temperature T_0.
- α_1 is the temperature coefficient at reference temperature T_1.

The following equation describes the thermal behavior of the block:

$$Q = K_d t_c \frac{dT}{dt}$$

where:

- Q is the net heat flow into port A.
- K_d is the **Dissipation factor** parameter value.
- t_c is the **Thermal time constant** parameter value.
- dT/dt is the rate of change of the temperature.

Variables

Use the **Variables** section of the block interface to set the priority and initial target values for the block variables prior to simulation. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

Parameters

- "Electrical Tab" on page 1-564
- "Thermal Tab" on page 1-565

Electrical Tab

Nominal resistance R0 at T0

The nominal resistance of the thermistor at the nominal temperature. Many datasheets quote the nominal resistance at 25°C and list it as R25. The default value is 1000 Ω.

Temperature coefficient alpha0 at T0

The temperature coefficient at the nominal temperature. The value must be less than zero. The default value is -0.01 1/K.

Nominal temperature T0

The temperature at which the nominal resistance is measured. The default value is 298.15 K.

Reference resistance R1 at T1

The reference resistance of the thermistor at the reference temperature. The default value is 10000 Ω.

Temperature coefficient alpha1 at T1

The temperature coefficient at the reference temperature. The value must be greater than zero. The default value is 1 1/K.

Reference temperature T1

The temperature at which the reference resistance is measured. This temperature must be in the PTC regime. The default value is 398.15 K.

Thermal Tab

Thermal time constant

The time it takes the sensor temperature to reach 63% of the final temperature change when a step change in ambient temperature occurs. The default value is 5 s.

Dissipation factor

The thermal power required to raise the thermistor temperature by one K. The default value is 7.5e-4 W/K.

Ports

The block has the following ports:

A

Thermal port

+

Positive electrical port

-

Negative electrical port

See Also

Resistor | Thermal Resistor | Thermistor

Pulse Current Source

Model periodic square pulse current source

Library

SPICE-Compatible Components/Sources

Description

The Pulse Current Source block represents a current source whose output current value is a periodic square pulse as a function of time and is independent of the voltage across the terminals of the source. The following equations describe the current through the source as a function of time:

$$I_{out}(0) = I1$$
$$I_{out}(TD) = I1$$
$$I_{out}(TD + TR) = I2$$
$$I_{out}(TD + TR + PW) = I2$$
$$I_{out}(TD + TR + PW + TF) = I1$$
$$I_{out}(TD + PER) = I1$$

where:

- *I1* is the **Initial value, I1** parameter value.
- *I2* is the **Pulse value, I2** parameter value.
- *TD* is the **Pulse delay time, TD** parameter value.

- *TR* is the **Pulse rise time, TR** parameter value.
- *TF* is the **Pulse fall time, TF** parameter value.
- *PW* is the **Pulse width, PW** parameter value.
- *PER* is the **Pulse period, PER** parameter value.

The block determines the values at intermediate time points by linear interpolation.

The specified values for *PW* and *PER* have the following effect on the block output:

- If both *PW* and *PER* are infinite, the block produces a step response at time *TD*.
- If *PER* is infinite and *PW* is finite, the block produces a single pulse of width *PW* and infinite period.
- If *PW* is infinite and *PER* is finite, the block produces a step response with pulses of width *TR* to a value *I1* every *PER* seconds.
- If *PW* > *PER*, the block produces a step response with pulses of width *TR* to a value *I1* every *PER* seconds.

The block uses a small conductance internally to prevent numerical simulation issues. The conductance connects the + and - ports of the device and has a conductance *GMIN*:

- By default, *GMIN* matches the **Minimum conductance GMIN** parameter of the SPICE Environment Parameters block, whose default value is 1e–12.
- To change *GMIN*, add a SPICE Environment Parameters block to your model and set the **Minimum conductance GMIN** parameter to the desired value.

Parameters

Initial value, I1

The value of the output current at time zero. The default value is 0 A.

Pulse value, I2

The value of the output current when the output is high. The default value is 0 A.

Pulse delay time, TD

The time at which the pulse first starts. The default value is 0 s.

Pulse rise time, TR

The time it takes the output current to rise from the **Initial value, I1** value to the **Pulse value, I2** value. The default value is `1e-09` s. The value must be greater than or equal to `0`.

Pulse fall time, TF

The time it takes the output current to fall from the **Pulse value, I2** value to the **Initial value, I1** value. The default value is `1e-09` s. The value must be greater than or equal to `0`.

Pulse width, PW

The time width of the output pulse. The default value is `Inf` s. The value must be greater than `0`.

Pulse period, PER

The period of the output pulse. The default value is `Inf` s. This value means that the block produces a single pulse with an infinite period. The value must be greater than `0`.

Ports

The block has the following ports:

+

Positive electrical voltage.

-

Negative electrical voltage.

See Also

Pulse Voltage Source

Pulse Voltage Source

Model periodic square pulse voltage source

Library

SPICE-Compatible Components/Sources

Description

The Pulse Voltage Source block represents a voltage source whose output voltage value is a periodic square pulse as a function of time and is independent of the current through the source. The following equations describe the output voltage as a function of time:

$$V_{out}(0) = V1$$
$$V_{out}(TD) = V1$$
$$V_{out}(TD + TR) = V2$$
$$V_{out}(TD + TR + PW) = V2$$
$$V_{out}(TD + TR + PW + TF) = V1$$
$$V_{out}(TD + PER) = V1$$

where:

- *V1* is the **Initial value, V1** parameter value.
- *V2* is the **Pulse value, V2** parameter value.
- *TD* is the **Pulse delay time, TD** parameter value.
- *TR* is the **Pulse rise time, TR** parameter value.

- *TF* is the **Pulse fall time, TF** parameter value.
- *PW* is the **Pulse width, PW** parameter value.
- *PER* is the **Pulse period, PER** parameter value.

The block determines the values at intermediate time points by linear interpolation.

The specified values for *PW* and *PER* have the following effect on the block output:

- If both *PW* and *PER* are infinite, the block produces a step response at time *TD*.
- If *PER* is infinite and *PW* is finite, the block produces a single pulse of width *PW* and infinite period.
- If *PW* is infinite and *PER* is finite, the block produces a step response with pulses of width *TR* to a value *V1* every *PER* seconds.
- If *PW* > *PER*, the block produces a step response with pulses of width *TR* to a value *V1* every *PER* seconds.

Parameters

Initial value, V1

 The value of the output voltage at time zero. The default value is `0` V.

Pulse value, V2

 The value of the output voltage when the output is high. The default value is `0` V.

Pulse delay time, TD

 The time at which the pulse first starts. The default value is `0` s.

Pulse rise time, TR

 The time it takes the output voltage to rise from the **Initial Value, V1** value to the **Pulse Value, V2** value. The default value is `1e-09` s. The value must be greater than or equal to `0`.

Pulse fall time, TF

 The time it takes the output voltage to fall from the **Pulse Value, V2** value to the **Initial Value, V1** value. The default value is `1e-09` s. The value must be greater than or equal to `0`.

Pulse width, PW

 The time width of the output pulse. The default value is `Inf` s.

Pulse period, PER

> The period of the output pulse. The default value is `Inf` s. This value means that the block produces a single pulse with an infinite period.

Ports

The block has the following ports:

+

> Positive electrical voltage.

-

> Negative electrical voltage.

See Also

Pulse Current Source

Push-Pull Output

Represent CMOS complementary output stage behaviorally
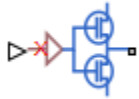
Library

Integrated Circuits

Description

The Push-Pull Output block represents a CMOS complementary output stage behaviorally. To improve simulation speed, the block does not model all the internal individual MOSFET devices that make up the gate. You can use this block to create a representative output current-voltage relationship when defining an integrated circuit model behavior with Physical Signal blocks from the Simscape Foundation library.

You can choose between are two output current-voltage relationships:

- Linear — The block represents the output as a voltage source plus series resistance and parallel capacitance, as shown in the following figure. The value you specify for the **Output resistance** parameter is assigned to the series resistance, and the capacitance values are determined by matching the RC time constant to the **Propagation delay** parameter value.
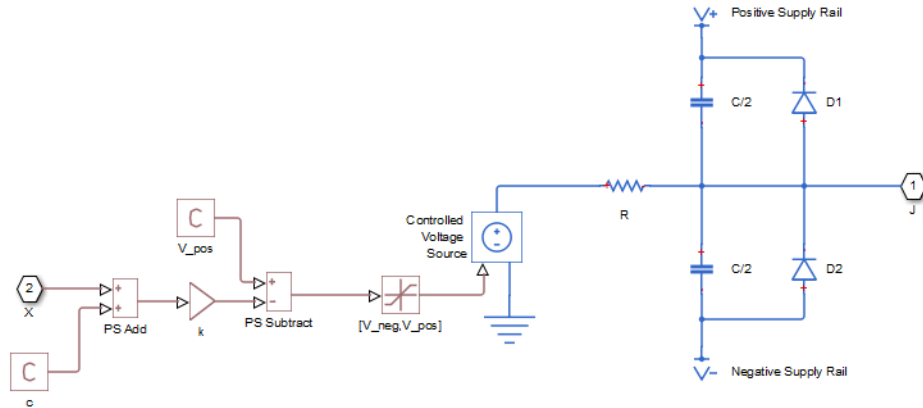
The input to the Controlled Voltage Source block is limited to be between the supply rails, and it is also inverted by subtraction from the supply voltage. The inversion makes it behave like a complementary output stage, with a high gate-source voltage resulting in a low output.

- Quadratic — The output stage is modeled by the two MOSFETs that constitute the complementary pair. The MOSFET parameters are derived from the output resistance values and short-circuit currents that you specify as mask parameters. The gate input demand is lagged to approximate the **Propagation delay** parameter value.
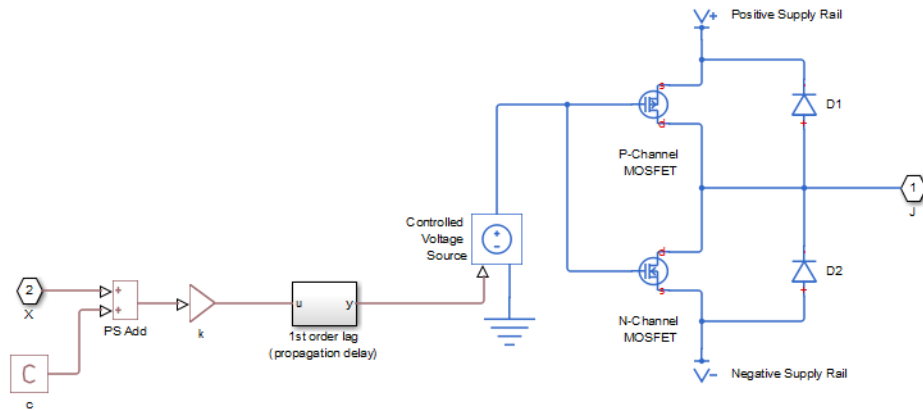
Both `Linear` and `Quadratic` output models add an offset and scale the physical input X so that the gate voltage is given by:

$$Vg = k \cdot (X + c)$$

where

- k is the input signal scaling.
- c is the input signal offset.

The offset and scaling can be used, for example, to match logical values for X (that is, range `[0,1]`) to [V-, V+] at the output pin. For example, if V+ = 10V and V- = 0, then to match the signal logical values to this voltage range, set c = `-1` and k = `-10`.

For both `Linear` and `Quadratic` output models, the protection diodes D1 and D2 act to limit the output voltage range. These diodes are Diode blocks from the Simscape Foundation library, that is, piecewise linear diodes defined by their forward voltage and on resistance. If the voltage across D1 rises above the forward voltage, then the diode starts to conduct, and provided that the on resistance is low, it effectively prevents the output rising above V+ plus the diode forward voltage drop. An equivalent behavior results if the output voltage drops too low.

The output model is very similar to that used for the logic blocks. For a plot of a typical output V-I characteristic when using the `Quadratic` output model, see Selecting the Output Model for Logic Blocks.

Note This block is constructed out of blocks from the Simscape Physical Signals library (such as PS Add, PS Gain, and so on). Currently, the blocks in the Physical Signals library do not support unit propagation and checking. For more information, see "How to Work with Physical Units" (Simscape).

Basic Assumptions and Limitations

The Push-Pull Output block has the following limitations:

- The block does not accurately model dynamic response.
- The `Quadratic` output model does not model any output capacitance effects. Add output capacitance externally to the block if required.

Parameters

- "Input Scaling Tab" on page 1-576
- "Output Characteristics Tab" on page 1-576
- "Supply Voltage Tab" on page 1-578
- "Initial Conditions Tab" on page 1-578

Input Scaling Tab

Input signal scaling, k

The input physical signal X is mapped to the gate voltage by $Vg = k \cdot (X + c)$, where k is the input signal scaling. Use this parameter in conjunction with the **Input signal offset, c** to map the range of X to the voltage range defined by the power supply. The default value is 1 V.

Input signal offset, c

The input physical signal X is mapped to the gate voltage by $Vg = k \cdot (X + c)$, where c is the input signal offset. Use this parameter in conjunction with the **Input signal scaling, k** to map the range of X to the voltage range defined by the power supply. The default value is 0.

Output Characteristics Tab

Output current-voltage relationship

Select the output model, `Linear` or `Quadratic` output model options. If `Linear`, the output voltage drops linearly with output current. If you select `Quadratic`, then the output voltage dependency on output current is defined by the quadratic I-V characteristics of the two output MOSFET devices. The default value is `Linear`.

Output resistance

Defines one over the slope of the output I-V characteristic. This parameter is available when you select the `Linear` option for the **Output current-voltage relationship** parameter. The default value is 25 Ω.

Power rail voltages, [V- V+], used for measurements

Defines the rail voltages for which mask data output resistances and currents are defined. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter. The default value is [0 5] V.

Output resistance values at zero output current and at I_OH when Vg=V-

A row vector [R_OH1 R_OH2] of two resistance values. The first value R_OH1 is the gradient of the output voltage-current relationship when the complementary pair output is HIGH (Vg=V-) and there is no output current. The second value R_OH2 is the gradient of the output voltage-current relationship when the output is HIGH and the output current is I_OH. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter. The default value is [25 250] Ω.

Output current I_OH when output is shorted to V- and Vg=V-

The resulting current when the output is HIGH (Vg=V-), but the load forces the output voltage to the negative supply rail. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter. The default value is 63 mA.

Output resistance values at zero output current and at I_OL when Vg=V+

A row vector [R_OL1 R_OL2] of two resistance values. The first value R_OL1 is the gradient of the output voltage-current relationship when the complementary pair output is LOW (Vg=V+) and there is no output current. The second value R_OL2 is the gradient of the output voltage-current relationship when the output is LOW and the output current is I_OL. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter. The default value is [30 800] Ω.

Output current I_OL when output is shorted to V+ and Vg=V+

The resulting current when the output is LOW (Vg=V+), but the load forces the output voltage to the positive supply voltage. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter. The default value is -45 mA.

Propagation delay

Time it takes for the output to reach 63.2% of its final value following a step change in the input, X. For `Quadratic` output, it is implemented by the lagged gate input demand. The default value is 25 ns.

Protection diode on resistance

The gradient of the voltage-current relationship for the protection diodes when forward biased. The default value is 5 Ω.

Protection diode forward voltage

The voltage above which the protection diode is turned on. The default value is `0.6` V.

Supply Voltage Tab

Negative power rail voltage, V-

Negative power supply voltage applied to the N-channel MOSFET source pin. The default value is 0 V.

Positive power rail voltage, V+

Positive power supply voltage applied to the P-channel MOSFET source pin. The default value is 5 V.

Initial Conditions Tab

Initial output voltage

This parameter is visible when you select the Linear option for the **Output current-voltage relationship** parameter on the **Output Characteristics** tab. The parameter is used to set the voltage on the output capacitors so that the output voltage is initialized to the parameter's value. The default value is 0 V.

Initial input signal

This parameter is visible when you select the Quadratic option for the **Output current-voltage relationship** parameter on the **Output Characteristics** tab. The parameter is used to initialize the propagation delay first-order lag such that there is no transient at time zero. The default value is 0 V.

Ports

The block has one input physical signal port X and one electrical conserving port that outputs the resulting voltage.

PVCCS

Model polynomial voltage-controlled current source
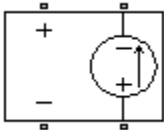
Library

SPICE-Compatible Components/Sources

Description

The PVCCS (Polynomial Voltage-Controlled Current Source) block represents a current source whose output current value is a polynomial function of the voltage across the input ports. The following equations describe the current through the source as a function of time:

- If you specify an *n*-element vector of polynomial coefficients for the **Polynomial coefficients** parameter:

$$I_{out} = p(0) + p(1) * V_{in} + ... + p(n-1) * V_{in}^{n-1} + p(n) * V_{in}^{n}$$

- If you specify a scalar coefficient for the **Polynomial coefficients** parameter:

$$I_{out} = p * V_{in}$$

where:

- V_{in} is the voltage across the input ports.
- *p* is the **Polynomial coefficients** parameter value.

The block uses a small conductance internally to prevent numerical simulation issues. The conductance connects the output ports of the device and has a conductance *GMIN*:

- By default, *GMIN* matches the **Minimum conductance GMIN** parameter of the SPICE Environment Parameters block, whose default value is 1e–12.
- To change *GMIN*, add a SPICE Environment Parameters block to your model and set the **Minimum conductance GMIN** parameter to the desired value.

Parameters

Polynomial coefficients

The polynomial coefficients that relate the input voltage to the output current, as described in the preceding section. The default value is [0 1].

Ports

The block has the following ports:

+

Positive electrical input voltage.

-

Negative electrical input voltage.

N+

Positive electrical output voltage.

N-

Negative electrical output voltage.

See Also

PCCCS, PCCVS, PVCCS2, and PVCVS

PVCCS2

Model polynomial voltage-controlled current source with two controlling inputs
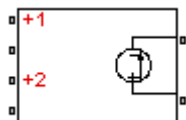
Library

SPICE-Compatible Components/Sources

Description

The PVCCS2 (Two-Input Polynomial Voltage-Controlled Current Source) block represents a current source whose output current value is a polynomial function of the voltages across the pairs of controlling input ports. The following equations describes the current through the source as a function of time:

$$I_{out} = p_1 + p_2 * V_{in1} + p_3 * V_{in2} + p_4 * V_{in1}^2 + p_5 V_{in1} * V_{in2} + p_6 * V_{in2}^2 + \ldots$$

where:

- V_{in1} is the voltage across the first pair of input ports.
- V_{in2} is the voltage across the second pair of input ports.
- p is the **Polynomial coefficients** parameter value.

The block uses a small conductance internally to prevent numerical simulation issues. The conductance connects the output ports of the device and has a conductance *GMIN*:

- By default, *GMIN* matches the **Minimum conductance GMIN** parameter of the SPICE Environment Parameters block, whose default value is 1e–12.
- To change *GMIN*, add a SPICE Environment Parameters block to your model and set the **Minimum conductance GMIN** parameter to the desired value.

1-581

Parameters

Polynomial coefficients

The polynomial coefficients that relate the input voltage to the output current, as described in the preceding section. The default value is [0 1 1].

Ports

The block has the following ports:

+1

Positive electrical input voltage of first controlling source.

-1

Negative electrical input voltage of first controlling source.

+2

Positive electrical input voltage of second controlling source.

-2

Negative electrical input voltage of second controlling source.

N+

Positive electrical output voltage.

N-

Negative electrical output voltage.

See Also

PCCCS2, PCCVS2, PVCCS, and PVCVS2

PVCVS

Model polynomial voltage-controlled voltage source
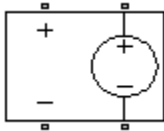
Library

SPICE-Compatible Components/Sources

Description

The PVCVS (Polynomial Voltage-Controlled Voltage Source) block represents a voltage source whose output voltage value is a polynomial function of the voltage across the input ports. The following equations describe the voltage across the source as a function of time:

- If you specify an *n*-element vector of polynomial coefficients for the **Polynomial coefficients** parameter:

$$V_{out} = p(0) + p(1) * V_{in} + ... + p(n-1) * V_{in}^{n-1} + p(n) * V_{in}^{n}$$

- If you specify a scalar coefficient for the **Polynomial coefficients** parameter:

$$V_{out} = p * V_{in}$$

where:

- V_{in} is the voltage across the input ports.
- p is the **Polynomial coefficients** parameter value.

Parameters

Polynomial coefficients

The polynomial coefficients that relate the input voltage to the output voltage, as described in the preceding section. The default value is [0 1].

Ports

The block has the following ports:

+

Positive electrical input voltage.

-

Negative electrical input voltage.

N+

Positive electrical output voltage.

N-

Negative electrical output voltage.

See Also

PCCCS, PCCVS, PVCCS, and PVCVS2

PVCVS2

Model polynomial voltage-controlled voltage source with two controlling inputs
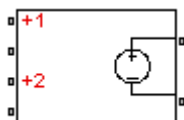
Library

SPICE-Compatible Components/Sources

Description

The PVCVS2 (Two-Input Polynomial Voltage-Controlled Voltage Source) block represents a voltage source whose output voltage value is a polynomial function of the voltages across the pairs of controlling input ports. The following equations describes the voltage across the source as a function of time:

$$V_{out} = p_1 + p_2 * V_{in1} + p_3 * V_{in2} + p_4 * V_{in1}^2 + p_5 V_{in1} * V_{in2} + p_6 * V_{in2}^2 + \ldots$$

where:

- V_{in1} is the voltage across the first pair of input ports.
- V_{in2} is the voltage across the second pair of input ports.
- p is the **Polynomial coefficients** parameter value.

The block uses a small conductance internally to prevent numerical simulation issues. The conductance connects the output ports of the device and has a conductance *GMIN*:

- By default, *GMIN* matches the **Minimum conductance GMIN** parameter of the SPICE Environment Parameters block, whose default value is 1e–12.
- To change *GMIN*, add a SPICE Environment Parameters block to your model and set the **Minimum conductance GMIN** parameter to the desired value.

1-585

Parameters

Polynomial coefficients

The polynomial coefficients that relate the input voltage to the output voltage, as described in the preceding section. The default value is [0 1 1].

Ports

The block has the following ports:

+1

Positive electrical input voltage of first controlling source.

-1

Negative electrical input voltage of first controlling source.

+2

Positive electrical input voltage of second controlling source.

-2

Negative electrical input voltage of second controlling source.

N+

Positive electrical output voltage.

N-

Negative electrical output voltage.

See Also

PCCCS2, PCCVS2, PVCCS2, and PVCVS

PWL Current Source

Model lookup table current source

Library

SPICE-Compatible Components/Sources

Description

The PWL Current Source block represents a current source that you specify in lookup table form using a vector of time values and a vector of the corresponding current values. You must specify at least four time-current value pairs. The block generates a time-dependent current based on these time-current values using the selected interpolation and extrapolation methods. You have a choice of three interpolation methods and two extrapolation methods. The output current is independent of the voltage across the terminals of the source.

The block uses a small conductance internally to prevent numerical simulation issues. The conductance connects the + and - ports of the device and has a conductance *GMIN*:

- By default, *GMIN* matches the **Minimum conductance GMIN** parameter of the SPICE Environment Parameters block, whose default value is 1e–12.

- To change *GMIN*, add a SPICE Environment Parameters block to your model and set the **Minimum conductance GMIN** parameter to the desired value.

Parameters

Time specification

The vector of time values as a tabulated 1-by-n array. The time values vector must be strictly monotonically increasing. The values can be non-uniformly spaced. The default value is `[ 0 1 2 3 4 ]` s.

Current at specified time

The vector of current values as a tabulated 1-by-n array. The current values vector must be the same size as the time values vector. The default value is `[ 0 0 0 0 0 ]` A.

Interpolation method

Select the method the block uses determine the output current values at intermediate time points that are not specified in the preceding vectors:

- `Linear` — Use a linear function. This is the default method.
- `Cubic` — Use the Piecewise Cubic Hermite Interpolation Polynomial (PCHIP). For more information, see [1] and the `pchip` MATLAB function.
- `Spline` — Use the cubic spline interpolation algorithm described in [2].

Extrapolation method

Select the method the block uses determine the output current values at time points that are outside the time range specified in the preceding vectors:

- `Last point value` — Use the last specified current value at the appropriate end of the range. That is, use the last specified current value for all time values greater than the last specified time argument, and the first specified current value for all time values less than the first specified time argument. This is the default method.
- `Last 2 points` — Extrapolate using the linear method (regardless of the interpolation method specified), based on the last two current values at the appropriate end of the range. That is, use the first and second specified current values if the time value is below the specified range, and the two last specified current values if the time value is above the specified range.

Ports

The block has the following ports:

+

    Positive electrical voltage.

-

    Negative electrical voltage.

References

[1] D. Kahaner, Cleve Moler, and Stephen Nash *Numerical Methods and Software* Prentice Hall, 1988.

[2] W.H. Press, B.P. Flannery, S.A. Teulkolsky, and W.T. Wetterling *Numerical Recipes in C: The Art of Scientific Computing* Cambridge University Press, 1992.

See Also

PWL Voltage Source

PWL Voltage Source

Model lookup table voltage source

Library

SPICE-Compatible Components/Sources

Description

The PWL Voltage Source block represents a voltage source that you specify in lookup table form using a vector of time values and a vector of the corresponding voltage values. You must specify at least four time-current value pairs. The block generates a time-dependent voltage based on these time-voltage values using the selected interpolation and extrapolation methods. You have a choice of three interpolation methods and two extrapolation methods. The output voltage is independent of the current through the source.

Parameters

Time specification

The vector of time values as a tabulated 1-by-n array. The time values vector must be strictly monotonically increasing. The values can be non-uniformly spaced. The default value is [0 1 2 3 4] s.

Voltage at specified time

The vector of voltage values as a tabulated 1-by-n array. The voltage values vector must be the same size as the time values vector. The default value is [0 0 0 0 0] V.

Interpolation method

Select the method the block uses determine the output voltage values at intermediate time points that are not specified in the preceding vectors:

- Linear — Use a linear function. This is the default method.
- Cubic — Use the Piecewise Cubic Hermite Interpolation Polynomial (PCHIP). For more information, see [1] and the pchip MATLAB function.
- Spline — Use the cubic spline interpolation algorithm described in [2].

Extrapolation method

Select the method the block uses determine the output voltage values at time points that are outside the time range specified in the preceding vectors:

- Last point value — Use the last specified voltage value at the appropriate end of the range. That is, use the last specified voltage value for all time values greater than the last specified time argument, and the first specified voltage value for all time values less than the first specified time argument. This is the default method.
- Last 2 points — Extrapolate using the linear method (regardless of the interpolation method specified), based on the last two voltage values at the appropriate end of the range. That is, use the first and second specified voltage values if the time value is below the specified range, and the two last specified voltage values if the time value is above the specified range.

Ports

The block has the following ports:

+

Positive electrical voltage.

-

Negative electrical voltage.

References

[1] D. Kahaner, Cleve Moler, and Stephen Nash *Numerical Methods and Software* Prentice Hall, 1988.

[2] W.H. Press, B.P. Flannery, S.A. Teulkolsky, and W.T. Wetterling *Numerical Recipes in C: The Art of Scientific Computing* Cambridge University Press, 1992.

See Also

PWL Current Source

RC Servo

Radio control servomotor with PWM-based angular position tracking and fault modeling
Library: Simscape / Electronics / Actuators & Drivers /
               Rotational Actuators
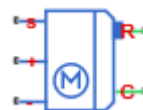
Description

The RC Servo block represents a small DC motor with a gearbox and control circuitry, commonly used in quadcopters, radio-controlled planes and helicopters, and other mechatronic devices. RC servos provide angular position control of the output shaft over a limited angle range. The angle demand is set by the pulse width of a PWM signal applied to port s.

The RC Servo block models the following effects:

- Torque-speed behavior based on DC motor equations
- Position tracking based on the input PWM signal pulse width
- Internal gear reduction ratio, including associated friction losses
- Mechanical end stops, to prevent the output shaft being driven out of range by the load
- Position measurement error
- Fault modeling

The motor equations are the same as those used by the DC Motor block, except that the inductance is not modeled. The RC Servo block determines the equation parameters using the stall torque and no-load speeds, and makes a correction to take account of the backdrive torque.

Faults

The RC Servo block allows you to model several types of faults:

- Fail off — No electrical torque.
- Fail forward — Rotates in a positive direction to hit the upper end stop.
- Fail reverse — Rotates in a negative direction to hit the lower end stop.
- Failed winding — Torque is applied only if the motor rotor lines up with one of the two remaining functioning windings.

The block can trigger fault events:

- At a specific time
- When a current limit is exceeded for longer than a specific time interval

You can enable or disable these trigger mechanisms separately, or use them together if more than one trigger mechanism is required in a simulation. When more than one mechanism is enabled, the first mechanism to trigger the fault takes precedence. In other words, component fails no more than once per simulation.

You can choose whether to issue an assertion when a fault occurs, by using the **Reporting when a fault occurs** parameter. The assertion can take the form of a warning or an error. By default, the block does not issue an assertion.

Variables

Use the **Variables** section of the block interface to set the priority and initial target values for the block variables prior to simulation. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

Basic Assumptions and Limitations

- This block has no optional thermal port.
- If you simulate the model with a fixed-step solver, for example, using a local solver, the step size must be small enough to get the required resolution of the input pulse width. MathWorks recommends that you use this block with variable step solvers for fast desktop simulation.

Ports

Conserving

s — Shaft control
electrical

Electrical conserving port associated with the PWM control signal. The output shaft angle demand is set by the pulse width of the voltage applied to this port.

+ — Positive terminal
electrical

Electrical conserving port associated with the motor positive terminal.

- — Negative terminal
electrical

Electrical conserving port associated with the motor negative terminal.

R — Rotor
mechanical rotational

Mechanical rotational conserving port associated with the rotor.

C — Casing
mechanical rotational

Mechanical rotational conserving port associated with the stator (casing).

Parameters

Electrical Torque

Stall torque — Maximum load torque
2.4 kg*cm (default)

Maximum load torque that the RC servo can move without stalling (stopping).

Time to travel 60 degrees (no load) — Output shaft 60 degrees turn time
0.28 s (default)

The time for the output shaft to turn 60 degrees when the motor is driving no load.

Corresponding nominal voltage — DC supply voltage
4.8 V (default)

The DC supply voltage used when measuring stall torque and time to travel 60 degrees.

Rotational range — Output shaft angular range
[0 180] deg (default)

The output shaft angular range of the RC servo.

Corresponding pulse widths — Input pulse widths range
[550 2330] us (default)

The input pulse widths corresponding to the minimum and maximum output angles, as defined by the **Rotational range** parameter. Pulse widths outside of this range are clipped by the block to stay within this range.

Control

Pulse threshold — Threshold voltage for pulse control
3 V (default)

The input pulse is detected as high when the voltage between the s and - ports is above this level.

Signal input resistance — Impedance
18 kOhm (default)

The electrical impedance measured between the s and - ports.

Angle resolution — Minimum error allowed between the demanded and measured output shaft angle
0.1 deg (default)

When the error between the demanded output shaft angle and measured output shaft angle drops below the angle resolution, the motor is powered off. This parameter models

the hysteresis usually incorporated into an RC servo controller to prevent chatter around a set point.

Angle measurement error — Model error in angle measurement
0 deg (default)

This parameter allows you to model an angle measurement error, such as can happen due to a failing potentiometer angle sensor. For example, if you want to model the motor being powered against one of the hard stops, you can set a suitable angle measurement error to achieve this.

Mechanical

Backdrive torque (unpowered) — Load torque required to backdrive unpowered motor
0.5 kg*cm (default)

Load torque required to backdrive the motor when it is unpowered. The block uses this value to determine the gear friction parameters.

Gear reduction ratio — Approximate reduction ratio from motor shaft to RC servo output shaft
320 (default)

Reduction ratio from the DC motor shaft to the RC servo output shaft. This parameter affects only the impact of rotor inertia on equivalent inertia value at the output shaft. It has no impact on no-load speed. Therefore, the value does not have to be precise.

Rotor inertia — Inertia of DC motor and rotor gearing
10 g*cm^2 (default)

Inertia of the DC motor, plus inertia of the gearing reflected to the rotor (typically small if the gears are plastic).

End-stop angles — Mechanical angle range
[-5 185] deg (default)

Mechanical end stops prevent rotation of the output shaft beyond the specified range. The range specified by the end stop angles must be larger than that specified by the **Rotational range** parameter.

End-stop stiffness — Stiffness of mechanical end stops
1e6 N*m/rad (default)

Stiffness of mechanical end stops.

End-stop damping — Damping of mechanical end stops
0.01 N*m/(rad/s) (default)

Damping of mechanical end stops.

Faults

Enable faults — Select Yes to enable faults modeling
No (default) | Yes

Select Yes to enable faults modeling. The associated parameters in the **Faults** section become visible to let you select the reporting method and specify the trigger mechanism (temporal or behavioral). You can enable these trigger mechanisms separately or use them together.

Reporting when a fault occurs — Choose whether to issue an assertion when a fault occurs
None (default) | Warn | Error

Choose whether to issue an assertion when a fault occurs:

- None — The block does not issue an assertion.
- Warn — The block issues a warning.
- Error — Simulation stops with an error.

Dependencies

Enabled when the **Enable faults** parameter is set to Yes.

Fault type — Select the type of fault
Fail off (default) | Fail forward | Fail reverse | Failed winding

Select the type of fault:

- Fail off — No electrical torque.
- Fail forward — Rotates in a positive direction to hit the upper end stop.

- `Fail reverse` — Rotates in a negative direction to hit the lower end stop.
- `Failed winding` — Torque is applied only if the motor rotor lines up with one of the two remaining functioning windings.

Dependencies

Enabled when the **Enable faults** parameter is set to Yes.

Enable temporal fault trigger — Select Yes to enable time-based fault triggering
No (default) | Yes

Select Yes to enable time-based fault triggering. You can enable the temporal and behavioral trigger mechanisms separately or use them together.

Dependencies

Enabled when the **Enable faults** parameter is set to Yes.

Simulation time for a fault event — Time before entering faulted state
1 s (default)

Set the simulation time at which you want the block to enter the faulted state.

Dependencies

Enabled when the **Enable temporal fault trigger** parameter is set to Yes.

Enable behavioral fault trigger — Select Yes to enable behavioral fault triggering
No (default) | Yes

Select Yes to enable behavioral fault triggering. You can enable the temporal and behavioral trigger mechanisms separately or use them together.

Dependencies

Enabled when the **Enable faults** parameter is set to Yes.

Maximum permissible current — Current threshold to fault transition
1 A (default)

Specify the maximum permissible current value. If the current exceeds this value for longer than the **Time to fail when exceeding maximum permissible current** parameter value, then the block enters the faulted state.

Dependencies

Enabled when the **Enable behavioral fault trigger** parameter is set to `Yes`.

`Time to fail when exceeding current` — Maximum length of time the current exceeds the threshold
1 s (default)

Set the maximum length of time that the current can exceed the maximum permissible value without triggering the fault.

Dependencies

Enabled when the **Enable behavioral fault trigger** parameter is set to `Yes`.

See Also
DC Motor

Introduced in R2017b

Relay

Model switching and associated delay of relay

Library

Passive Devices

Description

The Relay block models a relay controlled by an external physical signal. In the steady state, the relay behaves as follows:

- When the physical signal input PS rises above the **Threshold** parameter, the relay is *energized* (meaning closed). The common port C connects to the normally open port S2.

- When the physical signal input PS falls below the **Threshold** parameter, the relay is *not energized* (meaning open). The common port C connects to the normally closed port S1.

The initial state of the block is set automatically based on the signal value at port PS at the start of simulation (t = 0):

- If PS is greater than the threshold, the common port C connects to the S2 contact (relay is closed).

- If PS is less than or equal to the threshold, the common port C connects to the S1 contact (relay is open).

During switching, the relay behaves as follows:

- When the relay closes, the C to S1 connection breaks open after delay **Time-to-break C-S1 connection**. The C to S2 connection closes after delay **Time-to-make C-S2 connection**.

- When the relay opens, the C to S2 connection breaks open after delay **Time-to-break C-S2 connection**. The C to S1 connection closes after delay **Time-to-make C-S1 connection**.

You can specify break delays that are longer than the close delays to implement a make-before-break behavior.

Basic Assumptions and Limitations

If the PS input changes during the switching process, the block behavior can be inaccurate. The switching delay occurs due to both mechanical inertia and the fact that modeling inertia as a delay requires approximation.

Parameters

Time-to-break C-S1 connection

Time it takes the connection between ports C and S1 to break apart when the relay is energized. The default value is 0 s.

Time-to-make C-S1 connection

Time it takes the connection between ports C and S1 to close when the relay is not energized. The default value is 0 s.

Time-to-break C-S2 connection

Time it takes the connection between ports C and S2 to break apart when the relay is not energized. The default value is 0 s.

Time-to-make C-S2 connection

Time it takes the connection between ports C and S2 to close when the relay is not energized. The default value is 0 s.

Connected resistance R

Resistance across closed relay contacts. The parameter value must be greater than zero. The default value is 0.01 Ω.

Open-circuit conductance G

Conductance across open relay contacts. The parameter value must be greater than zero. The default value is `1e-08` 1/Ω.

Threshold

If the physical signal input rises above this value, the relay is energized. Conversely, if the physical signal input falls below this value, the relay is de-energized. The default value is `0`.

Ports

This block has the following ports:

PS

Physical signal that energizes and de-energizes the relay

C

Common electrical port

S1

Normally-closed electrical port

S2

Normally-open electrical port

See Also

Switch

Reluctance with Hysteresis

Nonlinear reluctance with magnetic hysteresis
Library: Simscape / Electronics / Passive Devices

Description

The Reluctance with Hysteresis block models a nonlinear reluctance with magnetic hysteresis. Use this block to build custom inductances and transformers that exhibit magnetic hysteresis.

The length and area parameters in the **Geometry** section let you define the geometry for the part of the magnetic circuit that you are modeling. The block uses the geometry information to map the magnetic domain Through and Across variables to flux density and field strength, respectively:

$$B = \Phi/A_e$$
$$MMF = l_e \cdot H$$

where:

- MMF is magnetomotive force (mmf) across the component.
- Φ is flux through the component.
- B is flux density.
- H is field strength.
- A_e is the effective cross-sectional area of the section being modeled.
- l_e is the effective length of the section being modeled.

The block then implements the relationship between B and H according to the Jiles-Atherton [1 on page 1-610, 2 on page 1-610] equations. The equation that relates B and H to the magnetization of the core is:

$$B = \mu_0 \left(H + M \right)$$

where:

- μ_0 is the magnetic permeability constant.
- M is magnetization of the core.

The magnetization acts to increase the magnetic flux density, and its value depends on both the current value and the history of the field strength H. The block uses the Jiles-Atherton equations to determine M at any given time.

The figure below shows a typical plot of the resulting relationship between B and H.

In this case, the magnetization starts as zero, and hence the plot starts at $B = H = 0$. As the field strength increases, the plot tends to the positive-going hysteresis curve; then on reversal the rate of change of H, it follows the negative-going hysteresis curve. The difference between positive-going and negative-going curves is due to the dependence of M on the trajectory history. Physically the behavior corresponds to magnetic dipoles in the core aligning as the field strength increases, but not then fully recovering to their original position as field strength decreases.

The starting point for the Jiles-Atherton equation is to split the magnetization effect into two parts, one that is purely a function of effective field strength (H_{eff}) and the other an irreversible part that depends on past history:

$$M = cM_{an} + (1-c)M_{irr}$$

The M_{an} term is called the anhysteretic magnetization because it exhibits no hysteresis. It is described by the following function of the current value of the effective field strength, H_{eff}:

$$M_{an} = M_s \left(\coth\left(\frac{H_{eff}}{\alpha} \right) - \frac{\alpha}{H_{eff}} \right)$$

This function defines a saturation curve with limiting values $\pm M_s$ and point of saturation determined by the value of α, the anhysteretic shape factor. It can be approximately thought of as describing the average of the two hysteretic curves. In the block interface, you provide values for dM_{an} / dH_{eff} when $H_{eff} = 0$ and a point $[H_1, B_1]$ on the anhysteretic B-H curve, and these are used to determine values for α and M_s.

The parameter c is the coefficient for reversible magnetization, and dictates how much of the behavior is defined by M_{an} and how much by the irreversible term M_{irr}. The Jiles-Atherton model defines the irreversible term by a partial derivative with respect to field strength:

$$\frac{dM_{irr}}{dH} = \frac{M_{an} - M_{irr}}{K\delta - \alpha(M_{an} - M_{irr})}$$
$$\delta = \begin{cases} 1 & \text{if } H \geq 0 \\ -1 & \text{if } H < 0 \end{cases}$$

Comparison of this equation with a standard first order differential equation reveals that as increments in field strength, H, are made, the irreversible term M_{irr} attempts to track the reversible term M_{an}, but with a variable tracking gain of $1 / \left(K\delta - \alpha(M_{an} - M_{irr}) \right)$. The tracking error acts to create the hysteresis at the points where δ changes sign. The main parameter that shapes the irreversible characteristic is K, which is called the bulk coupling coefficient. The parameter α is called the inter-domain coupling factor, and is also used to define the effective field strength used when defining the anhysteretic curve:

$$H_{eff} = H + \alpha M$$

The value of α affects the shape of the hysteresis curve, larger values acting to increase

the B-axis intercepts. However, notice that for stability the term $K\delta - \alpha(M_{an} - M_{irr})$ must be positive for $\delta > 0$ and negative for $\delta < 0$. Therefore not all values of α are permissible, a typical maximum value being of the order 1e-3.

Procedure for Finding Approximate Values for Jiles-Atherton Equation Coefficients

You can determine representative parameters for the equation coefficients by using the following procedure:

1 Provide a value for the **Anhysteretic B-H gradient when H is zero** parameter

(dM_{an} / dH_{eff} when $H_{eff} = 0$) plus a data point [H_1, B_1] on the anhysteretic B-H curve. From these values, the block initialization determines values for α and M_s.

2 Set the **Coefficient for reversible magnetization, c** parameter to achieve correct initial B-H gradient when starting a simulation from [H B] = [0 0]. The value of c is approximately the ratio of this initial gradient to the **Anhysteretic B-H gradient when H is zero**. The value of c must be greater than 0 and less than 1.

3 Set the **Bulk coupling coefficient, K** parameter to the approximate magnitude of H when $B = 0$ on the positive-going hysteresis curve.

4 Start with α very small, and gradually increase to tune the value of B when crossing $H = 0$ line. A typical value is in the range of 1e-4 to 1e-3. Values that are too large will cause the gradient of the B-H curve to tend to infinity, which is nonphysical and generates a run-time assertion error.

Sometimes you need to iterate on these four steps to get a good match against a predefined B-H curve.

Variables

Use the **Variables** section of the block interface to set the priority and initial target values for the block variables prior to simulation. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

Ports

Conserving

N — North terminal
magnetic

Magnetic conserving port associated with the block North terminal.

S — South terminal
magnetic

Magnetic conserving port associated with the block South terminal.

Parameters

Geometry

`Effective length` **— Effective length of the section being modeled**
0.032 m (default)

Effective length of the section being modeled, that is, the average distance of the magnetic path.

`Effective cross-sectional area` **— Effective cross-sectional area of the section being modeled**
1.6e-5 m^2 (default)

Effective cross-sectional area of the section being modeled, that is, the average area of the magnetic path.

`Averaging period for power logging` **— Excitation period used for averaging**
0 s (default)

Averaging period for the hysteresis losses calculation. These losses are proportional to the area enclosed by the B-H trajectory. If the block is excited at a known, fixed frequency, you can set this value to the corresponding excitation period to calculate the hysteresis loss. In this case, the block logs the hysteresis loss once per AC cycle to the variable

power_dissipated. If you are using a fixed-step solver, this value must be an integer multiple of the simulation step size.

If the block is not excited at a known, fixed frequency, set this parameter to 0. In this case, the block sets power_dissipated to zero, and you can calculate the actual hysteresis loss by post-processing the logged variable power_instantaneous.

Dependencies

This parameter is visible only when you select Magnetic flux density versus magnetic field strength characteristic with hysteresis for the **Parameterized by** parameter.

B-H Curve

Anhysteretic B-H gradient when H is zero — Gradient of the anhysteretic B-H curve around zero field strength
0.005 m*T/A (default)

The gradient of the anhysteretic (no hysteresis) B-H curve around zero field strength. Set it to the average gradient of the positive-going and negative-going hysteresis curves.

Flux density point on anhysteretic B-H curve — Flux density of the point for field strength measurement
1.49 T (default)

Specify a point on the anhysteretic curve by providing its flux density value. Picking a point at high field strength where the positive-going and negative-going hysteresis curves align is the most accurate option.

Corresponding field strength — Field strength at measurement point
1000 A/m (default)

The corresponding field strength for the point that you define by the **Flux density point on anhysteretic B-H curve** parameter.

Coefficient for reversible magnetization, c — Proportion of magnetization that is reversible
0.1 (default)

The proportion of the magnetization that is reversible. The value must be greater than zero and less than one.

`Bulk coupling coefficient, K` — Jiles-Atherton equations parameter
200 A/m (default)

The Jiles-Atherton parameter that primarily controls the field strength magnitude at which the B-H curve crosses the zero flux density line.

`Inter-domain coupling factor, alpha` — Jiles-Atherton equations parameter
0.0001 (default)

The Jiles-Atherton parameter that primarily affects the points at which the B-H curves intersect the zero field strength line. Typical values are in the range of 1e-4 to 1e-3.

References

[1] Jiles, D. C. and D. L. Atherton. "Theory of ferromagnetic hysteresis." *Journal of Magnetism and Magnetic Materials*. Vol. 61, 1986, pp. 48–60.

[2] Jiles, D. C. and D. L. Atherton. "Ferromagnetic hysteresis." *IEEE Transactions on Magnetics*. Vol. 19, No. 5, 1983, pp. 2183–2184.

See Also
Nonlinear Inductor | Nonlinear Transformer

Introduced in R2017b

Resistor

Resistor model including tolerance, operational limits, fault behavior, and noise
Library: Simscape / Electronics / Passive Devices

Description

The Resistor block represents a linear resistor, while letting you model the following effects:

- "Tolerances" on page 1-612
- "Operating Limits" on page 1-612
- "Faults" on page 1-613
- "Thermal Noise" on page 1-613
- "Thermal Port" on page 1-614

You can turn these modeling options on and off independently of each other. When all the additional options are turned off, the component behavior is identical to the Simscape Foundation library Resistor block.

In its simplest form, the Resistor block models a linear resistor, described with the following equation:

$$i = v/R$$

where:

- i is current.
- v is voltage.
- R is resistance.

If you set the **Noise mode** parameter to `Enabled`, then the defining equations are augmented by a discrete variable i_N to represent thermal noise, as described in "Thermal Noise" on page 1-613.

Tolerances

You can apply tolerances to the nominal value you provide for the **Resistance** parameter. Datasheets typically provide a tolerance percentage for a given resistor type. The table shows how the block applies tolerances and calculates resistance based on the selected **Tolerance application** option.

Option	Resistance Value
`None — use nominal value`	R
`Random tolerance`	Uniform distribution: $R \cdot (1 - tol + 2 \cdot tol \cdot$ `rand`$)$ Gaussian distribution: $R \cdot (1 + tol \cdot$ `randn` $/ nSigma)$
`Apply maximum tolerance value`	$R \cdot (1 + tol)$
`Apply minimum tolerance value`	$R \cdot (1 - tol)$

In the table,

- R is the **Resistance** parameter value, nominal resistance.
- *tol* is fractional tolerance, calculated from the percent-based **Tolerance (%)** parameter.
- *nSigma* is the value you provide for the **Number of standard deviations for quoted tolerance** parameter.
- `rand` and `randn` are standard MATLAB functions for generating uniform and normal distribution random numbers.

Operating Limits

You can specify operating limits in terms of power and maximum working voltage. For the thermal variant of the block (see "Thermal Port" on page 1-614), you can also specify operating limits in terms of temperature.

When an operating limit is exceeded, the block can either generate a warning or stop the simulation with an error. For more information, see the "Operating Limits" on page 1-617 parameters section.

Faults

The Resistor block allows you to model an electrical fault as an instantaneous change in resistance. The block can trigger fault events:

- At a specific time
- When a current limit is exceeded for longer than a specific time interval

You can enable or disable these trigger mechanisms separately, or use them together if more than one trigger mechanism is required in a simulation. When more than one mechanism is enabled, the first mechanism to trigger the fault takes precedence. In other words, the component fails no more than once per simulation.

When the resistor fails, its resistance is changed to the value you specify for the **Faulted zero-voltage resistance** parameter. You can also choose whether to issue an assertion when a fault occurs, by using the **Reporting when a fault occurs** parameter. The assertion can take the form of a warning or an error. By default, the block does not issue an assertion.

Thermal Noise

The Resistor block can generate thermal noise current. If you set the **Noise mode** parameter to `Enabled`, then the defining equations are augmented by a discrete variable i_N to represent thermal noise:

$$i = v/R + i_N$$

If the sampling time is h, then the thermal noise is given by:

$$i_N = \sqrt{2kT/R}\,\frac{N(0,1)}{\sqrt{h}}$$

where:

- k is the Boltzmann constant, 1.3806504e-23 J/K.
- T is the temperature.
- R is the resistance.
- N is a Gaussian random number with zero mean and standard deviation of one.

- *2kT/R* is the double-sided thermal noise power distribution (the single-sided equivalent is *4kT/R*).

The block generates Gaussian noise by using the Random Number source in the Simscape Foundation library. You can control the random number seed by setting the **Repeatability** parameter:

- `Not repeatable` — Every time you simulate your model, the block resets the random seed using the MATLAB random number generator:

  `seed = randi(2^32-1);`

- `Repeatable` — The block automatically generates a seed value and stores it inside the block, to always start the simulation with the same random number. This auto-generated seed value is set when you add a Resistor block from the block library to the model. When you make a new copy of the Resistor block from an existing one in a model, a new seed value is generated. The block sets the value using the MATLAB random number generator command shown above.

- `Specify seed` — If you select this option, the additional **Seed** parameter lets you directly specify the random number seed value.

Thermal Port

The block has an optional thermal port, hidden by default. To expose the thermal port, right-click the block in your model, and then from the context menu select **Simscape > Block choices > Show thermal port**. This action displays the thermal port H on the block icon, and adds the **Thermal** tab and the **Variables** tab to the block dialog box.

Use the **Thermal** tab to specify how the resistance value changes with temperature and to set the thermal mass. Use the **Variables** tab to set the initial temperature target.

With the thermal port exposed, the generated noise uses the temperature at the thermal port when determining the instantaneous noise value. Exposing the thermal port also extends the options on the **Operating Limits** tab as follows:

- The **Power rating** parameter becomes temperature dependent. You define a temperature up to which the full power rating is available, plus a higher temperature for which the power rating is reduced to zero. It is assumed that the power rating decreases linearly with temperature between these two values.

- An additional parameter, **Operating temperature range, [Tmin Tmax]**, lets you define the valid temperature range for block operation.

Variables

Use the **Variables** section of the block interface to set the priority and initial target values for the block variables prior to simulation. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

This section appears only for the blocks with exposed thermal port. The **Temperature** variable lets you specify a high-priority target for the temperature at the start of simulation.

Basic Assumptions and Limitations

Simulating with noise enabled slows down simulation. Choose the sample time (*h*) so that noise is generated only at frequencies of interest, and not higher.

Ports

Conserving

+ — Positive terminal
electrical

Electrical conserving port associated with the resistor positive terminal.

- — Negative terminal
electrical

Electrical conserving port associated with the resistor negative terminal.

H — Resistor thermal mass
thermal

Thermal conserving port that represents the resistor thermal mass.

Dependencies

Enabled for the thermal variant of the block. For more information, see "Thermal Port" on page 1-614.

Parameters

Main

Resistance — Nominal resistance value
1 Ω (default)

The nominal resistance value. Resistance value must be greater than zero.

Tolerance (%) — Resistor tolerance, in percent
5 (default)

The resistor tolerance as defined on the manufacturer datasheet.

Tolerance application — Select how to apply tolerance during simulation
None — use nominal value (default) | Random tolerance | Apply maximum tolerance value | Apply minimum tolerance value

Select how to apply tolerance during simulation:

- None — use nominal value — The block does not apply tolerance, uses the nominal resistance value. This is the default.
- Random tolerance — The block applies random offset to the resistance value, within the tolerance value limit. You can choose Uniform or Gaussian distribution for calculating the random number by using the **Tolerance distribution** parameter.
- Apply maximum tolerance value — The resistance is increased by the specified tolerance percent value.
- Apply minimum tolerance value — The resistance is decreased by the specified tolerance percent value.

Tolerance distribution — Select the distribution type
Uniform (default) | Gaussian

Select the distribution type for random tolerance:

- Uniform — Uniform distribution
- Gaussian — Gaussian distribution

Dependencies

Enabled when the **Tolerance application** parameter is set to Random tolerance.

Number of standard deviations for quoted tolerance — Used for calculating the Gaussian random number
4 (default)

Number of standard deviations for calculating the Gaussian random number.

Dependencies

Enabled when the **Tolerance distribution** parameter is set to Gaussian.

Operating Limits

Enable operating limits — Select Yes to enable reporting when the operational limits are exceeded
No (default) | Yes

Select Yes to enable reporting when the operational limits are exceeded. The associated parameters in the **Operating Limits** section become visible to let you select the reporting method and specify the operating limits in terms of power and maximum working voltage. Parameters that specify operating limits in terms of temperature are visible only for blocks with an exposed thermal port (see "Thermal Port" on page 1-614). The default value is No.

Reporting when operating limits exceeded — Select the reporting method
Warn (default) | Error

Select what happens when an operating limit is exceeded:

- Warn — The block issues a warning.
- Error — Simulation stops with an error.

Dependencies

Enabled when the **Enable operating limits** parameter is set to Yes.

Maximum working voltage — Maximum voltage allowed for normal block operation
100 V (default)

Maximum voltage magnitude allowed for normal block operation.

Dependencies

Enabled when the **Enable operating limits** parameter is set to `Yes`.

`Power rating` — Maximum power allowed for normal block operation
1 W (default)

Maximum power allowed for normal block operation.

If you expose the thermal port of the block, this parameter becomes temperature dependent. The value you specify for the **Power rating** parameter applies up to the temperature specified by the **Temperature below which full power rating is available** parameter value. Then the power rating decreases linearly with temperature, until it becomes 0 at temperature specified by the **Temperature above which power rating is reduced to zero** parameter value.

Dependencies

Enabled when the **Enable operating limits** parameter is set to `Yes`.

`Temperature below which full power rating is available` — Maximum temperature where full power rating still applies
70 °C (default)

Maximum temperature where full power rating, specified by the **Power rating** parameter value, still applies.

Dependencies

Enabled for the thermal variant of the block. For more information, see "Thermal Port" on page 1-614.

`Temperature above which power rating is reduced to zero` — Temperature where power rating becomes 0
155 °C (default)

Temperature where power rating becomes 0. Above this temperature, the simulation always issues an assertion regardless of dissipated power. This parameter value must be higher than **Temperature below which full power rating is available**.

Dependencies

Enabled for the thermal variant of the block. For more information, see "Thermal Port" on page 1-614.

`Operating temperature range, [Tmin Tmax]` — Minimum and maximum temperature values allowed for normal block operation
[-50 150] °C (default)

A row vector of length 2 specifying minimum and maximum temperature values allowed for normal block operation. The first element is the lowest allowable operating temperature, and the second element is the largest allowable operating temperature.

Dependencies

Enabled for the thermal variant of the block. For more information, see "Thermal Port" on page 1-614.

Faults

`Enable faults` — Select Yes to enable faults modeling
No (default) | Yes

Select `Yes` to enable faults modeling. The associated parameters in the **Faults** section become visible to let you select the reporting method and specify the trigger mechanism (temporal or behavioral). You can enable these trigger mechanisms separately or use them together.

`Reporting when a fault occurs` — Choose whether to issue an assertion when a fault occurs
None (default) | Warn | Error

Choose whether to issue an assertion when a fault occurs:

- `None` — The block does not issue an assertion.
- `Warn` — The block issues a warning.
- `Error` — Simulation stops with an error.

Dependencies

Enabled when the **Enable faults** parameter is set to `Yes`.

`Faulted resistance` — Resistance when block is in faulted state
inf Ω (default)

Resistance between the + and – ports when the block is in the faulted state.

Dependencies

Enabled when the **Enable faults** parameter is set to Yes.

Enable temporal fault trigger — Select Yes to enable time-based fault triggering
No (default) | Yes

Select Yes to enable time-based fault triggering. You can enable the temporal and behavioral trigger mechanisms separately or use them together.

Dependencies

Enabled when the **Enable faults** parameter is set to Yes.

Simulation time for fault event — Time before entering faulted state
1 s (default)

Set the simulation time at which you want the block to enter the faulted state.

Dependencies

Enabled when the **Enable temporal fault trigger** parameter is set to Yes.

Enable behavioral fault trigger — Select Yes to enable behavioral fault triggering
No (default) | Yes

Select Yes to enable behavioral fault triggering. You can enable the temporal and behavioral trigger mechanisms separately or use them together.

Dependencies

Enabled when the **Enable faults** parameter is set to Yes.

Maximum permissible current — Current threshold to fault transition
1 A (default)

Specify the maximum permissible current value. If the current exceeds this value for longer than the **Time to fail when exceeding maximum permissible current** parameter value, then the block enters the faulted state.

Dependencies

Enabled when the **Enable behavioral fault trigger** parameter is set to Yes.

Time to fail when exceeding maximum permissible current — Maximum length of time the current exceeds the threshold
1 s (default)

Set the maximum length of time that the current can exceed the maximum permissible value without triggering the fault.

Dependencies

Enabled when the **Enable behavioral fault trigger** parameter is set to Yes.

Noise

Noise mode — Select whether to model thermal noise current
Disabled (default) | Enabled

Select whether to model thermal noise current:

- Disabled — No noise is produced by the resistor.
- Enabled — Resistor generates thermal noise current, and the associated parameters become visible in the **Noise** section.

Sample time — Rate at which the noise source is sampled
1e-3 s (default)

Defines the rate at which the noise source is sampled. Choose it to reflect the frequencies of interest in your model. Making the sample time too small will unnecessarily slow down your simulation.

Dependencies

Enabled when the **Noise mode** parameter is set to Enabled.

Repeatability — Select the noise control option
Not repeatable (default) | Repeatable | Specify seed

Select the noise control option:

- Not repeatable — The random sequence used for noise generation is not repeatable.
- Repeatable — The random sequence used for noise generation is repeatable, with a system-generated seed.

- `Specify seed` — The random sequence used for noise generation is repeatable, and you control the seed by using the **Seed** parameter.

Dependencies

Enabled when the **Noise mode** parameter is set to `Enabled`.

Auto-generated seed used for repeatable option — Auto-generated random number seed
random real number

Random number seed stored inside the block to make the random sequence repeatable. The parameter value is automatically generated using the MATLAB random number generator command. You can modify this parameter value, but it gets overwritten by a new random value if you copy the block to another block in the model. Therefore, if you want to control the seed of the random sequence, use the `Specify seed` option for the **Repeatability** parameter and specify the desired seed value using the **Seed** parameter.

Dependencies

Enabled when the **Repeatability** parameter is set to `Repeatable`.

Seed — Random number seed
0 (default)

Seed used by the noise random number generator.

Dependencies

Enabled when the **Repeatability** parameter is set to `Specify seed`.

Device simulation temperature — Temperature of resistor at the start of the simulation
25 °C (default)

The temperature of the resistor at the start of the simulation.

Dependencies

Enabled when the **Noise mode** parameter is set to `Enabled`.

For blocks with an exposed thermal port, this parameter is disabled. Instead, use the **Variables** tab to set the initial temperature target. For more information, see "Variables" on page 1-615.

Thermal

This tab appears only for blocks with exposed thermal port. For more information, see "Thermal Port" on page 1-614.

Resistance temperature coefficient — Specifies how the resistance value changes with temperature
0.00393 1/K (default)

The coefficient α in the equation that describes resistance as a function of temperature, $R_T = R\,(1 + \alpha(T - T_0))$. The default value is for copper.

Measurement temperature — Temperature corresponding to nominal resistance
25 °C (default)

The temperature T_0, for which the nominal resistance R is specified.

Thermal mass — Thermal mass associated with port H
100 J/K (default)

Thermal mass associated with the thermal port H. It represents the energy required to raise the temperature of the thermal port by one degree.

See Also

Fault | Resistor

Introduced in R2009a

Resolver

Model rotary transformer that measures motor rotation angle
Library: Simscape / Simscape Components / Sensors
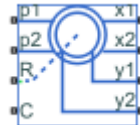
Description

The Resolver block models a generic resolver, which measures the electrical phase angle of a signal through electromagnetic coupling. The resolver consists of a rotary transformer that couples an AC voltage applied to the primary winding to two secondary windings. These secondary windings are physically oriented at 90 degrees to each other. As the rotor angle changes, the relative coupling between the primary and the two secondary windings varies. In the Resolver block model, the first secondary winding is oriented such that peak coupling occurs when the rotor is at zero degrees, and therefore the second secondary winding has minimum coupling when the rotor is at zero degrees.
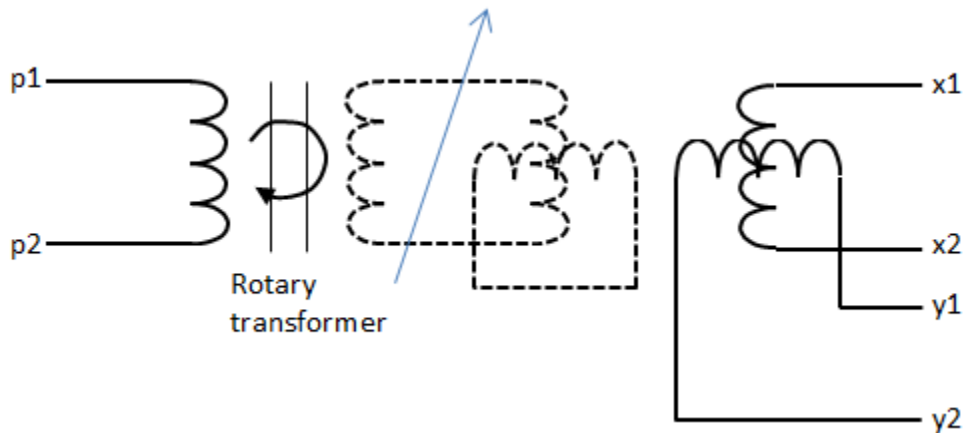
Without loss of generality, it is assumed that the transformer between primary and rotor circuit is ideal with a ratio of 1:1. This results in the rotor current and voltage being equivalent to the primary current and voltage.

You have two options for defining the block equations:

- Omit the dynamics by neglecting the transformer inductive terms. This model is only valid if the sensor is driven by a sine wave because any DC component on the primary side will pass to the output side.
- Include the inductive terms, thereby capturing voltage amplitude loss and phase differences. This model is valid for any input waveform. Within this option, you can either specify the inductances and the peak coupling coefficient directly, or specify the transformation ratio and measured impedances, in which case the block uses these values to determine the inductive terms.

Equations when Omitting Dynamics

The equations are based on the superposition of two ideal transformers, both with coupling coefficients that depend on rotor angle. The two ideal transformers have a common primary winding. See the Simscape Ideal Transformer block reference page for more information on modeling ideal transformers. The equations are:

$$K_x = R \cos(N \Theta)$$

$$K_y = R \sin(N \Theta)$$

$$v_x = K_x v_p$$

$$v_y = K_y v_p$$

$$i_p = - K_x i_x - K_y i_y$$

where:

- v_p and i_p are the rotor (or equivalently primary) voltage and current, respectively.
- v_x and i_x are the first secondary voltage and current, respectively.
- v_y and i_y are the second secondary voltage and current, respectively.
- K_x is the coupling coefficient for the first secondary winding.
- K_y is the coupling coefficient for the second secondary winding.

- *R* is the transformation ratio.
- *N* is the number of pole pairs.
- *Θ* is the rotor angle.

Equations when Including Dynamics

The equations are based on the superposition of two mutual inductors, both with coupling coefficients that depend on rotor angle. The two mutual inductors have a common primary winding. See the Simscape Mutual Inductor block reference page for more information on modeling mutual inductors. The equations are:

$$v_p = R_p i_p + L_p \frac{di_p}{dt} + \sqrt{L_p L_s} \, k \left(\cos(N\theta) \frac{di_x}{dt} + \sin(N\theta) \frac{di_y}{dt} \right)$$

$$v_x = R_s i_x + L_s \frac{di_x}{dt} + \sqrt{L_p L_s} \, k \cos(N\theta) \frac{di_p}{dt}$$

$$v_y = R_s i_y + L_s \frac{di_y}{dt} + \sqrt{L_p L_s} \, k \sin(N\theta) \frac{di_p}{dt}$$

where:

- v_p and i_p are the rotor (or equivalently primary) voltage and current, respectively.
- v_x and i_x are the first secondary voltage and current, respectively.
- v_y and i_y are the second secondary voltage and current, respectively.
- R_p is the rotor (or primary) resistance.
- L_p is the rotor (or primary) inductance.
- R_s is the stator (or secondary) resistance.
- L_s is the stator (or secondary) inductance.
- *N* is the number of pole pairs.
- *k* is the coefficient of coupling.
- *Θ* is the rotor angle.

It is assumed that coupling between the two secondary windings is zero.

Datasheets typically do not quote the coefficient of coupling and inductance parameters, but instead give the transformation ratio R and measured impedances. If you select `Specify transformation ratio and measured impedances` for the **Parameterization** parameter, then the values you provide are used to determine values for the equation coefficients, as defined above.

Variables

Use the **Variables** section of the block interface to set the priority and initial target values for the block variables prior to simulation. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

Assumptions and Limitations

The model is based on the following assumptions:

- The resolver draws no torque between the mechanical rotational ports R and C.
- The transformer between primary and rotor circuit is ideal with a ratio of 1:1.
- The coupling between the two secondary windings is zero.

Ports

Conserving

p1 — Primary winding positive terminal
electrical

Electrical conserving port associated with the positive terminal of the primary winding.

p2 — Primary winding negative terminal
electrical

Electrical conserving port associated with the negative terminal of the primary winding.

R — Resolver rotor
mechanical rotational

Mechanical rotational conserving port connected to the rotor.

C — Resolver case
mechanical rotational

Mechanical rotational conserving port connected to the resolver case.

x1 — Secondary winding *x* positive terminal
electrical

Electrical conserving port associated with the positive terminal of secondary winding *x*.

x2 — Secondary winding *x* negative terminal
electrical

Electrical conserving port associated with the negative terminal of secondary winding *x*.

y1 — Secondary winding *y* positive terminal
electrical

Electrical conserving port associated with the positive terminal of secondary winding *y*.

y2 — Secondary winding *y* negative terminal
electrical

Electrical conserving port associated with the negative terminal of secondary winding *y*.

Parameters

Parameterization — Resolver parameterization
Specify transformation ratio and omit dynamics (default) | Specify transformation ratio and measured impedances | Specify equation parameters directly

Select one of the following methods for block parameterization:

- Specify transformation ratio and omit dynamics — Provide values for transformation ratio, number of pole pairs, and initial rotor angle only. This model neglects the transformer inductive terms, and is only valid if the sensor is driven by a sine wave. The equations are based on the superposition of two ideal transformers,

both with coupling coefficients that depend on rotor angle. For more information, see "Equations when Omitting Dynamics" on page 1-625.

- `Specify transformation ratio and measured impedances` — Provide additional values to determine the transformer inductive terms, to model the voltage amplitude loss and phase differences. This model is valid for any input waveform. The equations are based on the superposition of two mutual inductors, both with coupling coefficients that depend on rotor angle. For more information, see "Equations when Including Dynamics" on page 1-626.

- `Specify equation parameters directly` — Model the dynamics, but provide values for rotor and stator inductances and the peak coefficient of coupling, instead of transformation ratio and measured impedances. For more information, see "Equations when Including Dynamics" on page 1-626. This model is valid for any input waveform.

`Transformation ratio` — **Peak output to input voltage ratio**
`0.5` (default) | positive number

Ratio between the peak output voltage and the peak input voltage assuming negligible secondary voltage drop due to resistance and inductance.

Dependencies

To enable this parameter, set the **Parameterization** parameter to `Specify transformation ratio and omit dynamics` or `Specify transformation ratio and measured impedances`. If you select `Specify transformation ratio and measured impedances` for the **Parameterization** parameter, then the transformation ratio takes the voltage drop due to primary winding resistance into account.

`Rotor resistance` — **Primary resistance**
`70 Ohm` (default) | positive number

Rotor ohmic resistance. This resistance is also referred to as the primary resistance.

Dependencies

To enable this parameter, set the **Parameterization** parameter to `Specify transformation ratio and measured impedances` or `Specify equation parameters directly`.

`Stator resistance` — **Secondary resistance**
`180 Ohm` (default) | positive number

Stator ohmic resistance. This resistance is also referred to as the secondary resistance. It is assumed that both secondaries have the same resistance.

Dependencies

To enable this parameter, set the **Parameterization** parameter to `Specify transformation ratio and measured impedances` or `Specify equation parameters directly`.

Rotor reactance — Primary reactance
`100 Ohm` (default) | positive number

Rotor reactance when the secondary windings are open-circuit. This reactance is also referred to as the primary reactance.

Dependencies

To enable this parameter, set the **Parameterization** parameter to `Specify transformation ratio and measured impedances`.

Stator reactance — Secondary reactance
`300 Ohm` (default) | positive number

Stator reactance when the primary winding is open-circuit. This reactance is also referred to as the secondary reactance.

Dependencies

To enable this parameter, set the **Parameterization** parameter to `Specify transformation ratio and measured impedances`.

Frequency at which reactances and transformation ratio are specified — Sinusoidal source frequency
`10 kHz` (default) | positive number

Frequency of the sinusoidal source used when measuring the reactances.

Dependencies

To enable this parameter, set the **Parameterization** parameter to `Specify transformation ratio and measured impedances`.

Rotor inductance — Primary reactance
`0.0016 H` (default) | positive number

Rotor or primary inductance, L_p.

Dependencies

To enable this parameter, set the **Parameterization** parameter to `Specify equation parameters directly`.

Stator inductance — Secondary reactance
`0.0048` H (default) | positive number

Stator or secondary inductance, L_s.

Dependencies

To enable this parameter, set the **Parameterization** parameter to `Specify equation parameters directly`.

Peak coefficient of coupling — Maximum coupling coefficient
`0.35` (default) | number between zero and one, exclusive

Peak coefficient of coupling between the primary and secondary windings.

Dependencies

To enable this parameter, set the **Parameterization** parameter to `Specify equation parameters directly`.

Number of pole pairs — Rotor pole pairs
`1` (default) | positive number

Number of pole pairs on the rotor.

See Also
Incremental Shaft Encoder

Introduced in R2017b

S-R Latch

Model an S-R Latch behaviorally

Library

Logic

Description

The S-R Latch block is an abstracted behavioral model of a set-reset latch. It does not model the internal individual MOSFET devices (see "Basic Assumptions and Limitations" on page 1-633 for details). Therefore, the block runs quickly during simulation but retains the correct I/O behavior.

If the gate voltage is greater than the threshold voltage V_{TH}, then the input taken is 1 (HIGH). Otherwise, the input is zero (LOW). The gate threshold voltage V_{TH} is halfway between the **Low level input voltage** (V_{IL}) and **High level input voltage** (V_{IH}) parameters.

The block output logic level is either HIGH or LOW, according to the logic levels of the gate inputs and the S-R latch truth table.

S	R	Q
0	0	1
0	1	0
1	0	1
1	1	1

The block models the gate as follows:

- The gate inputs have infinite resistance and finite or zero capacitance.

- The gate output offers a selection of two models: `Linear` and `Quadratic`. For more information, see "Selecting the Output Model for Logic Blocks". Use the **Output current-voltage relationship** parameter to specify the output model.

- You can specify propagation delay for both output models. For `Linear` output, the block sets the value of the gate output capacitor such that the resistor-capacitor time constant equals the **Propagation delay** parameter value. For `Quadratic` output, the gate input demand is lagged to approximate the **Propagation delay** parameter value.

The block output voltage depends on the output model selected:

- For `Linear` model, output high is the **High level output voltage** parameter value, and output low is the **Low level output voltage** parameter value.

- For `Quadratic` model, the output voltage for High and Low states is a function of the output current, as explained in "Quadratic Model Output and Parameters". For zero load current, output high is Vcc (the **Supply voltage** parameter value), and output low is zero volts.

Basic Assumptions and Limitations

The block does not model the internal individual MOSFET devices that make up the gate (except for the final MOSFET pair if you select the `Quadratic` option for the **Output current-voltage relationship** parameter). This limitation has the following implications:

- The behavior of this block is abstracted. In particular, response to input noise and inputs that are around the logic threshold voltage can be inaccurate. Also, dynamic response is approximate.

- The linear drop in output voltage as a function of output current is an approximation to the MOSFET or bipolar output behavior.

- Modeling of the output as a controlled voltage source is representative of a totem-pole or push-pull output stage. To model a device with an open-collector:

  1 Connect the output pin to the base of an NPN Bipolar Transistor or PNP Bipolar Transistor block.

  2 Set the **Output resistance** parameter to a suitable value.

Parameters

- "Inputs Tab" on page 1-634
- "Outputs Tab" on page 1-634
- "Initial Conditions Tab" on page 1-636

Inputs Tab

Low level input voltage

Voltage value less than which the block interprets the input voltage as `LOW`. The default value is 2 V.

High level input voltage

Voltage value greater than which the block interprets the input voltage as `HIGH`. The default value is 3 V.

Average input capacitance

Fixed capacitance that approximates the input capacitance for a MOSFET gate. You can usually find this capacitance value on a manufacturer datasheet. The default value is 5 pF. Setting this value to zero can result in faster simulation times.

Outputs Tab

Output current-voltage relationship

Select the output model, `Linear` or `Quadratic`. The default value is `Linear`.

Low level output voltage

Voltage value at the output when the output logic level is `LOW`. The default value is 0 V. This parameter is available when you select the `Linear` option for the **Output current-voltage relationship** parameter.

High level output voltage

Voltage value at the output when the output logic level is `HIGH`. The default value is 5 V. This parameter is available when you select the `Linear` option for the **Output current-voltage relationship** parameter.

Output resistance

Value of the series output resistor that is used to model the drop in output voltage resulting from the output current. The default value is 25 Ω. You can derive this value

from a datasheet by dividing the high-level output voltage by the maximum low-level output current. This parameter is available when you select the `Linear` option for the **Output current-voltage relationship** parameter.

Supply voltage

Supply voltage value applied to the gate in your circuit. The default value is 5 V. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Measurement voltage

The gate supply voltage for which mask data output resistances and currents are defined. The default value is 5 V. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic HIGH output resistance at zero current and at I_OH

A row vector [*R_OH1 R_OH2*] of two resistance values. The first value *R_OH1* is the gradient of the output voltage-current relationship when the gate is logic HIGH and there is no output current. The second value *R_OH2* is the gradient of the output voltage-current relationship when the gate is logic HIGH and the output current is *I_OH*. The default value is [25 250] Ω. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic HIGH output current I_OH when shorted to ground

The resulting current when the gate is in the logic HIGH state, but the load forces the output voltage to zero. The default value is 63 mA. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic LOW output resistance at zero current and at I_OL

A row vector [*R_OL1 R_OL2*] of two resistance values. The first value *R_OL1* is the gradient of the output voltage-current relationship when the gate is logic LOW and there is no output current. The second value *R_OL2* is the gradient of the output voltage-current relationship when the gate is logic LOW and the output current is *I_OL*. The default value is [30 800] Ω. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic LOW output current I_OL when shorted to Vcc

The resulting current when the gate is in the logic LOW state, but the load forces the output voltage to the supply voltage Vcc. The default value is -45 mA. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Propagation delay

Time it takes for the output to swing from LOW to HIGH or HIGH to LOW after the input logic levels change. The default value is 25 ns.

Protection diode on resistance

The gradient of the voltage-current relationship for the protection diodes when forward biased. The default value is 5 Ω. This parameter is available when you select the Quadratic option for the **Output current-voltage relationship** parameter.

Protection diode forward voltage

The voltage above which the protection diode is turned on. The default value is 0.6 V. This parameter is available when you select the Quadratic option for the **Output current-voltage relationship** parameter.

Initial Conditions Tab

Output initial state

Specify whether the initial output state of the block is High or Low. This parameter is used for both linear and quadratic output states, provided that the **Propagation delay** parameter is greater than zero and the Solver Configuration block does not have the **Start simulation from steady state** option selected. The default value is Low.

Ports

This block has the following ports:

S

Electrical input port corresponding to the set pin

R

Electrical input port corresponding to the reset pin

Q

Electrical output port corresponding to the output pin

Schmitt Trigger

Behavioral model of Schmitt trigger

Library

Logic

Description

The Schmitt Trigger block implements a behavioral model of Schmitt trigger.

The block output logic level is `HIGH` when the input rises above the **High level input voltage** (V_{IH}) value and does not go `LOW` until the input falls below the lower-valued **Low level input voltage** (V_{IL}) value. This logic implements a hysteresis characteristic between input and output.
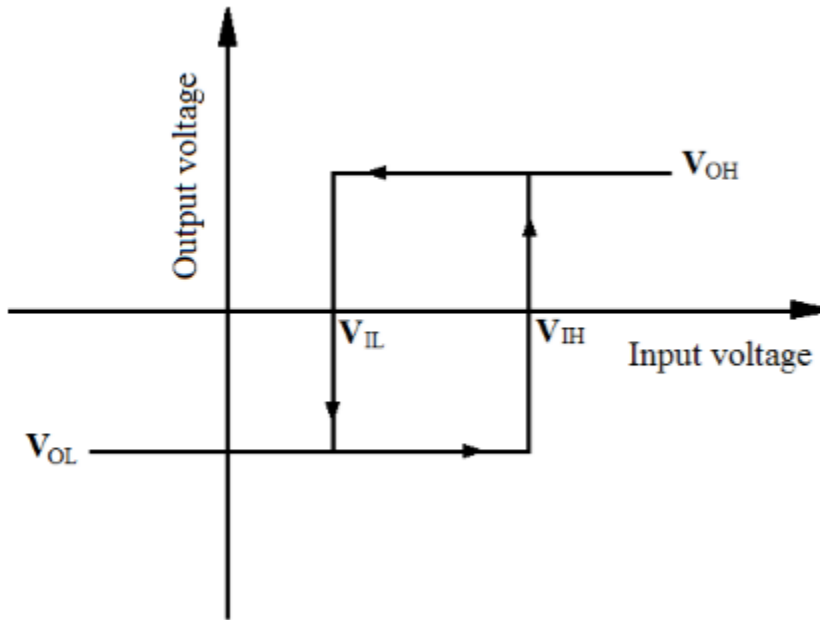
In the graphic, V_{OH} and V_{OL} correspond to the **High level output voltage** and **Low level output voltage** values, respectively.

The next figure shows a sample output of the block with parameters V_{IH} = 2V, V_{IL} = -2V, V_{OH} = 3V, and V_{OL} = -3V.
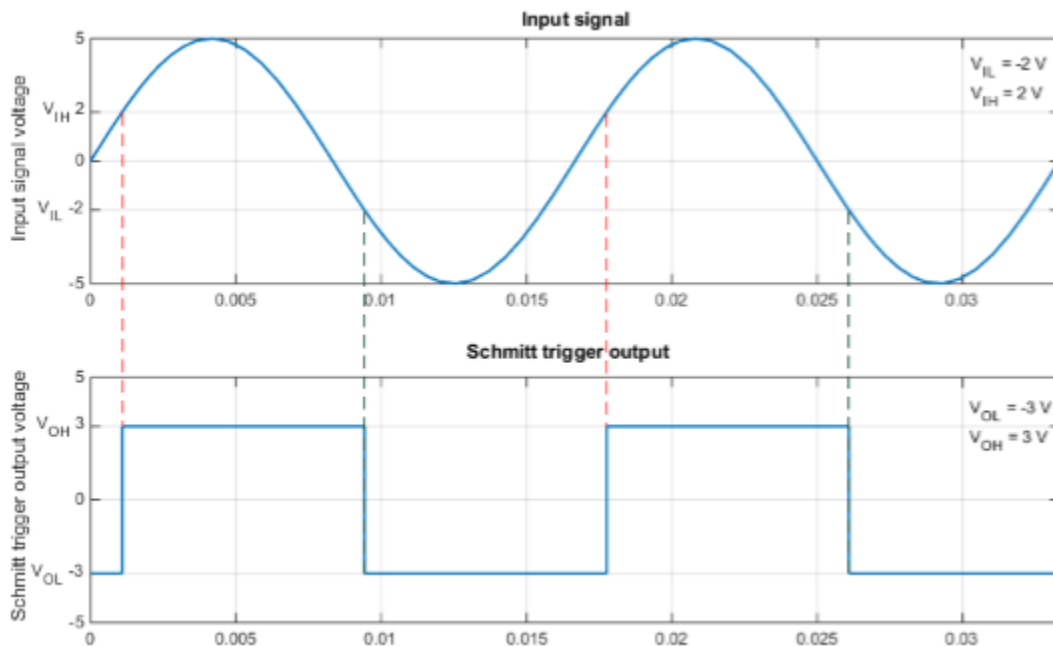
The block determines the logic levels of the gate inputs as follows:

- If the gate voltage is greater than the threshold voltage, the block interprets the input as logic 1.
- Otherwise, the block interprets the input as logic 0.

The threshold voltage is the voltage value at midpoint between the **High level input voltage** parameter value and the **Low level input voltage** parameter value.

Note To improve simulation speed, the block does not model all the internal individual MOSFET devices that make up the gate. See "Basic Assumptions and Limitations" on page 1-640 for details.

The block models the gate as follows:

- The gate inputs have infinite resistance and finite or zero capacitance.

1-639

- The gate output offers a selection of two models: `Linear` and `Quadratic`. For more information, see "Selecting the Output Model for Logic Blocks". Use the **Output current-voltage relationship** parameter to specify the output model.

- You can specify propagation delay for both output models. For `Linear` output, the block sets the value of the gate output capacitor such that the resistor-capacitor time constant equals the **Propagation delay** parameter value. For `Quadratic` output, the gate input demand is lagged to approximate the **Propagation delay** parameter value.

The block output voltage depends on the output model selected:

- For `Linear` model, output high is the **High level output voltage** parameter value, and output low is the **Low level output voltage** parameter value.

- For `Quadratic` model, the output voltage for High and Low states is a function of the output current, as explained in "Quadratic Model Output and Parameters". For zero load current, output high is Vcc (the **Supply voltage** parameter value), and output low is zero volts.

Basic Assumptions and Limitations

The block does not model the internal individual MOSFET devices that make up the gate (except for the final MOSFET pair if you select the `Quadratic` option for the **Output current-voltage relationship** parameter). This limitation has the following implications:

- The block does not accurately model the gate's response to input noise and inputs that are around the logic threshold voltage.

- The block does not accurately model dynamic response.

For circuits that involve a feedback path around a set of logic gates, you might need to set a nonzero propagation delay on one or more gates.

This block is implemented using event equations. This means that you must provide an initial output state that is consistent with the block input at time zero. For example, if you set initial output state HIGH, but the initial input voltage is below the **Low level input voltage**, then the initial output stays HIGH, the state only correcting itself when the input voltage rises above the **High level input voltage** value.

Parameters

- "Inputs Tab" on page 1-641
- "Outputs Tab" on page 1-641
- "Initial Conditions Tab" on page 1-643

Inputs Tab

Low level input voltage

Voltage value below which the block interprets the input voltage as logic `LOW`. The default value is 2 V.

High level input voltage

Voltage value above which the block interprets the input voltage as logic `HIGH`. The default value is 3 V.

Average input capacitance

Fixed capacitance that approximates the input capacitance for a MOSFET gate. The MOSFET capacitance depends on the applied voltage. When you drive this block with another gate, the **Average input capacitance** produces a rise time similar to that of the MOSFET. You can usually find this capacitance value on a manufacturer datasheet. The default value is 5 pF. Setting this value to zero can result in faster simulation times.

Outputs Tab

Output current-voltage relationship

Select the output model, `Linear` or `Quadratic`. The default value is `Linear`.

Low level output voltage

Voltage value at the output when the output logic level is `LOW`. The default value is `0` V. This parameter is available when you select the `Linear` option for the **Output current-voltage relationship** parameter.

High level output voltage

Voltage value at the output when the output logic level is `HIGH`. The default value is `5` V. This parameter is available when you select the `Linear` option for the **Output current-voltage relationship** parameter.

Output resistance

Value of the series output resistor that is used to model the drop in output voltage resulting from the output current. The default value is 25 Ω. You can derive this value from a datasheet by dividing the high-level output voltage by the maximum low-level output current. This parameter is available when you select the `Linear` option for the **Output current-voltage relationship** parameter.

Supply voltage

Supply voltage value applied to the gate in your circuit. The default value is 5 V. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Measurement voltage

The gate supply voltage for which mask data output resistances and currents are defined. The default value is 5 V. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic HIGH output resistance at zero current and at I_OH

A row vector [*R_OH1 R_OH2*] of two resistance values. The first value *R_OH1* is the gradient of the output voltage-current relationship when the gate is logic `HIGH` and there is no output current. The second value *R_OH2* is the gradient of the output voltage-current relationship when the gate is logic `HIGH` and the output current is *I_OH*. The default value is [25 250] Ω. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic HIGH output current I_OH when shorted to ground

The resulting current when the gate is in the logic `HIGH` state, but the load forces the output voltage to zero. The default value is 63 mA. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic LOW output resistance at zero current and at I_OL

A row vector [*R_OL1 R_OL2*] of two resistance values. The first value *R_OL1* is the gradient of the output voltage-current relationship when the gate is logic `LOW` and there is no output current. The second value *R_OL2* is the gradient of the output voltage-current relationship when the gate is logic `LOW` and the output current is *I_OL*. The default value is [30 800] Ω. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Logic LOW output current I_OL when shorted to Vcc

The resulting current when the gate is in the logic `LOW` state, but the load forces the output voltage to the supply voltage Vcc. The default value is `-45` mA. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Propagation delay

Time it takes for the output to swing from `LOW` to `HIGH` or `HIGH` to `LOW` after the input logic levels change. The default value is `25` ns.

Protection diode on resistance

The gradient of the voltage-current relationship for the protection diodes when forward biased. The default value is `5` Ω. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Protection diode forward voltage

The voltage above which the protection diode is turned on. The default value is `0.6` V. This parameter is available when you select the `Quadratic` option for the **Output current-voltage relationship** parameter.

Initial Conditions Tab

Output initial state

Specify whether the initial output state of the block is `High` or `Low`. This parameter is used for both linear and quadratic output states, provided that the **Propagation delay** parameter is greater than zero and the Solver Configuration block does not have the **Start simulation from steady state** option selected. The default value is `Low`.

Ports

The block has the following ports:

A

Electrical input port

J

Electrical output port

Introduced in R2015a

Servomotor

Brushless motor model with closed-loop torque control
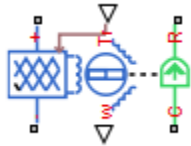
Library

Rotational Actuators

Description

The Servomotor block represents a brushless motor model with closed-loop torque control. This block abstracts the torque-speed behavior of the combined motor and motor driver in order to support system-level simulation where simulation speed is important.

The block permits only the range of torques and speeds that the torque-speed envelope defines. In the default block configuration, you specify this data in the block dialog box as a set of speed data points and corresponding maximum torque values. The following figure shows a typical torque-speed envelope for a servomotor.
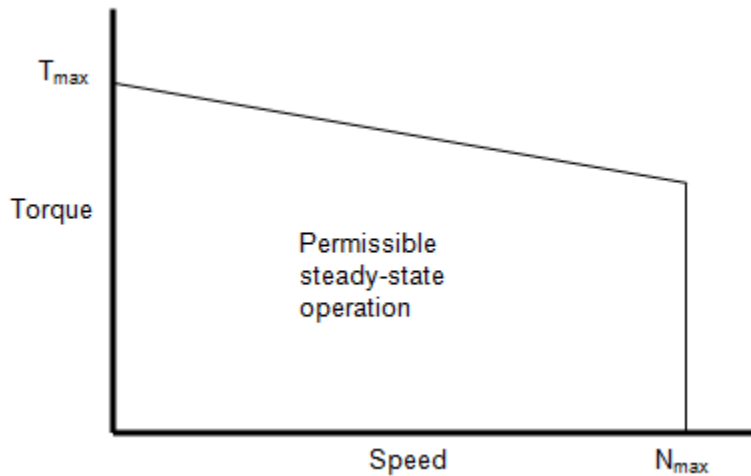
Specify the torque-speed envelope for the positive torque region only, that is, quadrants 1 and 4. If you specify only for positive speeds (quadrant 1 or, equivalently, the motoring region), then the quadrant 4 torque envelope is defined by the block as the mirror image of quadrant 1. The servomotor torque-speed envelope has the same profile when the motor is operating in a reverse direction (quadrants 2 and 3).

Instead of providing tabulated torque-speed data, you can specify a maximum torque and a maximum power. This results in the torque-speed envelope profile shown below. The other three operating quadrants are constrained by this same profile.
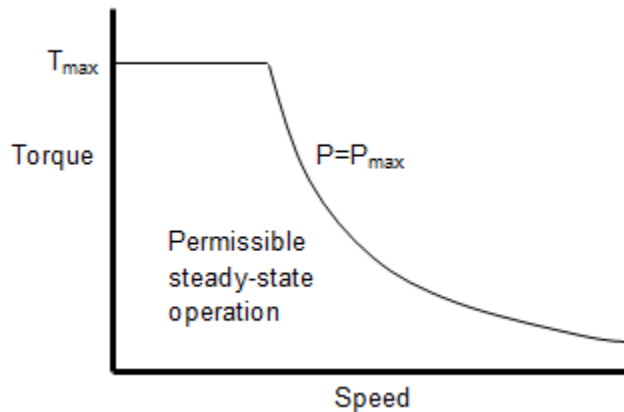
The block produces a positive torque acting from the mechanical C to R ports.

Modeling Electrical Losses

The block allows both simplified and tabulated definition of electrical losses. The default, simplified, behavior is to model the losses as the sum of the following four terms:

- A series resistance between the DC power supply and the motor drive.
- Fixed losses independent of torque and speed, P_0. Use this to account for fixed converter losses.
- A torque-dependent electrical loss $k\tau^2$, where τ is the torque and k is a constant. This represents ohmic losses in the copper windings.
- A speed-dependent electrical loss $k_w\omega^2$, where ω is the speed and k_w is a constant. This represents iron losses due to eddy currents.

Alternatively you can provide tabulated loss values as a function of motor speed and load torque. When using this option, provide data for all of the operating quadrants that your simulation will run in. If you provide partial data (for example, just for the quadrant 1 forward motoring region), then the other quadrants are assumed to repeat the same pattern of losses. This will normally be correct for the reverse motoring region, but may be an approximation for the braking/generating quadrants. The block does no extrapolation of loss values for speed and torque magnitudes that exceed the range of the table.

Finally, you can specify electrical losses by using tabulated efficiency data, instead of a single efficiency measurement or tabulated loss data. When using this option, also provide data for all of the operating quadrants that your simulation will run in. If you provide partial data (for example, just for the quadrant 1 forward motoring region), then the other quadrants are assumed to repeat the same pattern of losses.

The best practice is to provide tabulated loss data as a function of speed and torque, rather than tabulated efficiency data, because:

- Efficiency becomes ill-defined for zero speed or zero torque.
- Using losses, you can also account for fixed losses that are still present for zero speed or torque.

If you use the tabulated efficiencies option:

- The block converts the efficiency values you provide into losses and uses the tabulated losses for simulation.
- Efficiency values you provide for zero speed or zero torque are ignored, and losses are assumed zero when either torque or speed is zero.
- The block uses linear interpolation to determine losses. Provide tabulated data for low speeds and low torques, as required, to get the desired level of accuracy for lower power conditions.
- The block does no extrapolation of loss values for speed and torque magnitudes that exceed the range of the table.

Block Variants

The block provides four modeling variants, accessible by right-clicking the block in your block diagram and then selecting the appropriate option from the context menu, under **Simscape** > **Block choices**:

- **No thermal port** — Basic model that does not simulate faults or thermal effects. This is the default.
- **Show thermal port** — Model with exposed thermal port. This model does not simulate faults.
- **Faultable| No thermal port** — Model with exposed fault control port. This model does not simulate thermal effects.
- **Faultable | Show thermal port** — Model that lets you simulate both faults and thermal effects. Both the thermal port and the fault input port are exposed.

Thermal Ports

The block has an optional thermal port, hidden by default. To expose the thermal port, select one of the block variants that model thermal effects, as described in "Block Variants" on page 1-648. This action displays the thermal port H on the block icon, and adds the **Temperature Dependence** and **Thermal Port** tabs to the block dialog box. These tabs are described further on this reference page.

Use the thermal port to simulate the effects of copper resistance losses that convert electrical power to heat. For more information on using thermal ports in actuator blocks, see "Simulating Thermal Effects in Rotational and Translational Actuators".

Simulating Faults

You can use the physical signal input port F to simulate servomotor failure, as well as connecting and disconnecting the DC supply. You cannot simulate disconnecting the DC supply by simply opening a switch, because there must be a finite voltage on the servomotor terminals, producing the current that balances the electrical and mechanical power.

To expose the fault control port, select one of the faultable block variants, as described in "Block Variants" on page 1-648. This action displays the physical signal input port F on the block icon, and adds the **Faults** tab to the block dialog box. This tabs are is described further on this reference page.

If a signal is connected to port F, then the block operates according to the parameter settings on the **Faults** tab. For example, if **Fault condition** is `Faulted if F >= Fault threshold`, then when the signal at port F rises above the **Fault threshold** value, the servomotor stops operating, zero current is taken from the supply side, and zero current is supplied to the load side.

Basic Assumptions and Limitations

This model is based on the following assumptions:

- The motor driver tracks a torque demand with a time constant *Tc*.
- Motor speed fluctuations due to mechanical load do not affect the motor torque tracking.

Parameters

Electrical Torque Tab

Parameterize by

Select one of the following methods for block parameterization:

- `Tabulated torque-speed envelope` — Provide the vectors of rotational speeds and corresponding maximum torque values. This is the default option.
- `Maximum torque and power` — Define the torque-speed envelope by providing values for maximum permissible torque and motor power.

Vector of rotational speeds

Rotational speeds for permissible steady-state operation. This parameter is visible only if you select `Tabulated torque-speed envelope` for the **Parameterize by** parameter. The default value is `[0 3.75e+03 7.5e+03 8e+03]` rpm. To avoid poor performance due to an infinite slope in the torque-speed curve, specify a vector of rotational speeds that does not contain duplicate consecutive values.

Vector of maximum torque values

Maximum torque values for permissible steady-state operation. This parameter is visible only if you select `Tabulated torque-speed envelope` for the **Parameterize by** parameter. These values correspond to the speeds in the **Vector of rotational speeds** parameter and define the torque-speed envelope for the motor. The default value is `[0.09 0.08 0.07 0]` Nm.

Maximum torque

The maximum permissible motor torque. This parameter is visible only if you select `Maximum torque and power` for the **Parameterize by** parameter. The default value is `0.1` Nm.

Maximum power

The maximum permissible motor power. This parameter is visible only if you select `Maximum torque and power` for the **Parameterize by** parameter. The default value is 30 W.

Torque Control time constant, Tc

Time constant with which the motor driver tracks a torque demand. The default value is 0.02 s.

Electrical Losses Tab

Parameterize losses by

Select one of the following methods for electrical loss parameterization:

- `Single efficiency measurement` — Model the losses as the sum of the four terms, listed in the block description, at a single measurement point. This is the default option.
- `Tabulated loss data` — Determine the losses by two-dimensional table lookup based on the provided tabulated data for motor speeds, load torques, and corresponding losses.
- `Tabulated efficiency data` — Determine the losses by two-dimensional table lookup based on the provided tabulated data for motor speeds, load torques, and corresponding efficiencies. See "Modeling Electrical Losses" on page 1-647 for details.

Motor and driver overall efficiency (percent)

The block defines overall efficiency as

$$\eta = 100 \frac{\tau_0 \omega_0}{\tau_0 \omega_0 + P_0 + k\tau_0^2 + k_w \omega_0^2}$$

where:

- τ_0 represents the **Torque at which efficiency is measured**.
- ω_0 represents the **Speed at which efficiency is measured**.
- P_0 represents the **Fixed losses independent of torque or speed**.
- $k\tau_0^2$ represents the torque-dependent electrical losses.

- $k_w\omega^2$ represents the speed-dependent iron losses.

At initialization, the block solves the efficiency equation for *k*. The block neglects losses associated with the rotor damping. This parameter is visible only if **Parameterize losses by** is set to `Single efficiency measurement`. The default value is `100` %.

Speed at which efficiency is measured

Speed that the block uses to calculate torque-dependent electrical losses. This parameter is visible only if **Parameterize losses by** is set to `Single efficiency measurement`. The default value is `3.75e+03` rpm.

Torque at which efficiency is measured

Torque that the block uses to calculate torque-dependent electrical losses. This parameter is visible only if **Parameterize losses by** is set to `Single efficiency measurement`. The default value is `0.08` Nm.

Iron losses

Iron losses at the speed and torque at which efficiency is defined. This parameter is visible only if **Parameterize losses by** is set to `Single efficiency measurement`. The default value is `0` W.

Fixed losses independent of torque and speed

Fixed electrical loss associated with the driver when the motor current and torque are zero. This parameter is visible only if **Parameterize losses by** is set to `Single efficiency measurement`. The default value is `0` W.

Vector of speeds (w) for tabulated losses

The vector of speed values, to be used for table lookup when calculating losses. This parameter is visible only if **Parameterize losses by** is set to `Tabulated loss data` or `Tabulated efficiency data`. The default value is `[-8000 -4000 0 4000 8000]` rpm.

Vector of torques (T) for tabulated losses

The vector of speed values, to be used for table lookup when calculating losses. This parameter is visible only if **Parameterize losses by** is set to `Tabulated loss data` or `Tabulated efficiency data`. The default value is `[0 0.03 0.06 0.09]` Nm.

Corresponding losses, P(w,T)

Tabulated values for electrical losses as a function of speed and torque, to be used for 2D table lookup. Each value in the matrix specifies the losses for a specific combination of speed and torque. The matrix size must match the dimensions defined by the speed and torque vectors. This parameter is visible only if **Parameterize**

losses by is set to `Tabulated loss data`. The default value is `[1.49 1.67 2.21 3.10; 0.42 0.69 1.14 2.03; 0.06 0.24 0.78 1.68; 0.42 0.69 1.14 2.03; 1.49 1.67 2.21 3.10]` W.

Corresponding efficiency (percent), E(w,T)

Tabulated efficiency values, in percent, as a function of speed and torque, to be used for 2D table lookup. Each value in the matrix specifies the efficiency for a specific combination of speed and torque. The matrix size must match the dimensions defined by the speed and torque vectors. Efficiency values you provide for zero speed or zero torque are ignored, and losses are assumed zero when either torque or speed is zero. The block uses linear interpolation to determine losses. Provide tabulated data for low speeds and low torques, as required, to get the desired level of accuracy for lower power conditions. This parameter is visible only if **Parameterize losses by** is set to `Tabulated efficiency data`. The default value is `[95 95 95 95; 95 95 95 95; 95 95 95 95; 95 95 95 95; 95 95 95 95]`.

Supply series resistance

The equivalent resistance used in series with the DC supply to model electrical losses that are proportional to the driver supply current. The block assumes that the DC supply current is approximately constant under constant load conditions. The default value is `0` Ω.

Faults Tab

This tab appears only for blocks with an exposed fault control port. For more information, see "Simulating Faults" on page 1-649.

Fault condition

Selects whether the fault is triggered by a signal that is high or low:

- `Faulted if F >= Fault threshold` — Servomotor is disabled if the signal at port F rises above the threshold value. This is the default option.
- `Faulted if F <= Fault threshold` — Servomotor is disabled if the signal at port F falls below the threshold value.

Fault threshold

The threshold value used to detect a fault. The default value is `0.5`.

Mechanical Tab

Rotor inertia

Rotor resistance to change in motor motion. The default value is `5e-06` kg*m^2. The value can be zero.

Rotor damping

Rotor damping. The default value is `1e-05` N*m/(rad/s). The value can be zero.

Initial rotor speed

Rotor speed at the start of the simulation. The default value is `0` rpm.

Temperature Dependence Tab

This tab appears only for blocks with an exposed thermal port. For more information, see "Thermal Ports" on page 1-649.

Resistance temperature coefficient

Parameter α in the equation defining resistance as a function of temperature, as described in "Thermal Model for Actuator Blocks". This parameter is visible only if the **Parameterize losses by** parameter on the **Electrical Losses** tab is set to `Single efficiency measurement`. The default value is for copper, and is `0.00393` 1/K.

Measurement temperature

The temperature for which motor parameters are defined. If you parameterize electrical losses by tabulated loss data, then this is the temperature for which the **Corresponding losses, P(w,T)** are given on the **Electrical Losses** tab. The default value is 25 °C.

Second measurement temperature

The temperature for which the **Corresponding losses, P(w,T), at second measurement temperature** are given. This parameter is visible only if the **Parameterize losses by** parameter on the **Electrical Losses** tab is set to `Tabulated loss data`. The default value is 125 °C.

Corresponding losses, P(w,T), at second measurement temperature

Iron losses at the second measurement temperature, corresponding to the speed and torque tabulated values on the **Electrical Losses** tab. This parameter is visible only if the **Parameterize losses by** parameter on the **Electrical Losses** tab is set to `Tabulated loss data`. The default value is `[1.49 1.74 2.49 3.74;0.42 0.67`

1.42 2.67;0.06 0.31 1.06 2.31;0.42 0.67 1.42 2.67;1.49 1.74 2.49 3.74] W.

Thermal Port Tab

This tab appears only for blocks with an exposed thermal port. For more information, see "Thermal Ports" on page 1-649.

Thermal mass

Thermal mass of the electrical winding, defined as the energy required to raise the temperature by one degree. The default value is 100 J/K.

Initial temperature

The temperature of the thermal port at the start of simulation. The default value is 25 °C.

Ports

This block has the following ports:

+

Positive electrical DC supply

-

Negative electrical DC supply

Tr

Reference torque demand

w

Mechanical speed output

C

Mechanical rotational conserving port

R

Mechanical rotational conserving port

See Also

DC Motor | Induction Motor | Shunt Motor | Universal Motor

SFFM Current Source

Model single-frequency FM current source

Library

SPICE-Compatible Components/Sources

Description

The SFFM Current Source block represents a single-frequency current source whose frequency-modulated output current value is independent of the voltage across its terminals. The following equation describes the current through the source as a function of time:

$$I_{out} = IO + IA * \sin\left(\left(2\pi * FC * Time\right) + MI * \sin\left(2\pi * FS * Time\right)\right)$$

where:

- *I0* is the **Current offset, IO** parameter value.
- *IA* is the **Current amplitude, IA** parameter value.
- *FC* is the **Carrier frequency, FC** parameter value.
- *MI* is the **Modulation index, MI** parameter value.
- *FS* is the **Signal frequency, FS** parameter value.

The block uses a small conductance internally to prevent numerical simulation issues. The conductance connects the + and - ports of the device and has a conductance *GMIN*:

- By default, *GMIN* matches the **Minimum conductance GMIN** parameter of the SPICE Environment Parameters block, whose default value is 1e–12.

- To change *GMIN*, add a SPICE Environment Parameters block to your model and set the **Minimum conductance GMIN** parameter to the desired value.

Parameters

Current offset, IO

> The magnitude of the time-independent part of the output current. The default value is 0 A.

Current amplitude, IA

> The magnitude of the sinusoidal part of the output current. The default value is 0 A.

Carrier frequency, FC

> Frequency of the carrier wave. The default value is 0 Hz. The value must be greater than or equal to 0.

Modulation index, MI

> The amount by which the modulated signal varies around its unmodulated level. The default value is 0. The value must be greater than or equal to 0.

Signal frequency, FS

> Frequency of the modulated signal. The default value is 0 Hz. The value must be greater than or equal to 0.

Ports

The block has the following ports:

+

> Positive electrical voltage.

-

> Negative electrical voltage.

See Also

SFFM Voltage Source

SFFM Voltage Source

Model single-frequency FM voltage source

Library

SPICE-Compatible Components/Sources

Description

The SFFM Voltage Source block represents a single-frequency voltage source whose frequency-modulated output voltage value is independent of the current through the source. The following equation describes the output voltage as a function of time:

$$V_{out} = VO + VA * \sin\left(\left(2\pi * FC * Time\right) + MI * \sin\left(2\pi * FS * Time\right)\right)$$

where:

- *V0* is the **Voltage offset, VO** parameter value.
- *VA* is the **Voltage amplitude, VA** parameter value.
- *FC* is the **Carrier frequency, FC** parameter value.
- *MI* is the **Modulation index, MI** parameter value.
- *FS* is the **Signal frequency, FS** parameter value.

Parameters

Voltage offset, VO

   The magnitude of the time-independent part of the output voltage. The default value is 0 V.

Voltage amplitude, VA

The magnitude of the sinusoidal part of the output voltage. The default value is 0 V.

Carrier frequency, FC

Frequency of the carrier wave. The default value is 0 Hz. The value must be greater than or equal to 0.

Modulation index, MI

The amount by which the modulated signal varies around its unmodulated level. The default value is 0. The value must be greater than or equal to 0.

Signal frequency, FS

Frequency of the modulated signal. The default value is 0 Hz. The value must be greater than or equal to 0.

Ports

The block has the following ports:

+

Positive electrical voltage.

-

Negative electrical voltage.

See Also

SFFM Current Source

Shunt Motor

Model electrical and torque characteristics of shunt motor
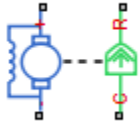
Library

Rotational Actuators

Description

The Shunt Motor block represents the electrical and torque characteristics of a shunt motor using the following equivalent circuit model.
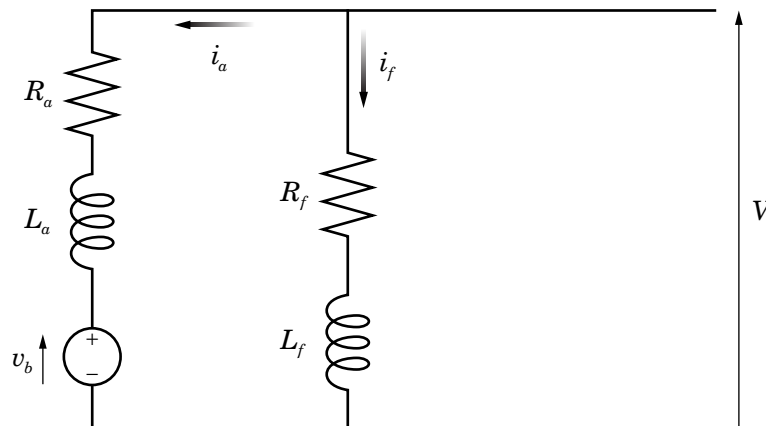
When you set the **Model parameterization** parameter to By equivalent circuit parameters, you specify the equivalent circuit parameters for this model:

- R_a — **Armature resistance**
- L_a — **Armature inductance**
- R_f — **Field winding resistance**
- L_f — **Field winding inductance**

The Shunt Motor block computes the motor torque as follows:

1 The magnetic field in the motor induces the following back emf v_b in the armature:

$$v_b = L_{af} i_f \omega$$

where L_{af} is a constant of proportionality and ω is the angular velocity.

2 The mechanical power is equal to the power reacted by the back emf:

$$P = v_b i_a = L_{af} i_f i_a \omega$$

3 The motor torque is:

$$T = P / \omega = L_{af} i_f i_a$$

The torque-speed characteristic for the Shunt Motor block model is related to the parameters in the preceding figure. When you set the **Model parameterization** parameter to `By rated power, rated speed & no-load speed`, the block solves for the equivalent circuit parameters as follows:

1 For the steady-state torque-speed relationship, L has no effect.

2 Sum the voltages around the loop:

$$V = i_a R_a + L_{af} i_f \omega$$
$$V = i_f R_f$$

3 Solve the preceding equations for i_a and i_f:

$$i_f = \frac{V}{R_f}$$
$$i_a = \frac{V}{R_a} \left(1 - \frac{L_{af} w}{R_f} \right)$$

4 Substitute these values of i_a and i_f into the equation for torque:

$$T = \frac{L_{af}}{R_a R_f}\left(1 - \frac{L_{af}\omega}{R_f}\right)V^2$$

The block uses the rated speed and power to calculate the rated torque. The block uses the rated torque and no-load speed values to get one equation that relates R_a and L_{af}/R_f. It uses the no-load speed at zero torque to get a second equation that relates these two quantities. Then, it solves for R_a and L_{af}/R_f.

The block models motor inertia J and damping B for all values of the **Model parameterization** parameter. The output torque is:

$$T_{load} = \frac{L_{af}}{R_a R_f}\left(1 - \frac{L_{af}\omega}{R_f}\right)V^2 - J\dot{\omega} - B\omega$$

The block produces a positive torque acting from the mechanical C to R ports.

Thermal Ports

The block has two optional thermal ports, one per winding, hidden by default. To expose the thermal ports, right-click the block in your model, and then from the context menu select **Simscape > Block choices > Show thermal port**. This action displays the thermal ports on the block icon, and adds the **Temperature Dependence** and **Thermal Port** tabs to the block dialog box. These tabs are described further on this reference page.

Use the thermal ports to simulate the effects of copper resistance losses that convert electrical power to heat. For more information on using thermal ports in actuator blocks, see "Simulating Thermal Effects in Rotational and Translational Actuators".

Parameters

- "Electrical Torque Tab" on page 1-664
- "Mechanical Tab" on page 1-665

Electrical Torque Tab

Model parameterization

Select one of the following methods for block parameterization:

- By equivalent circuit parameters — Provide electrical parameters for an equivalent circuit model of the motor. This is the default method.
- By rated power, rated speed & no-load speed — Provide power and speed parameters that the block converts to an equivalent circuit model of the motor.

Armature resistance

Resistance of the armature. This parameter is only visible when you select By equivalent circuit parameters for the **Model parameterization** parameter. The default value is 110 Ω.

Field winding resistance

Resistance of the field winding. This parameter is only visible when you select By equivalent circuit parameters for the **Model parameterization** parameter. The default value is 2.5e+03 Ω.

Back-emf constant

The ratio of the voltage generated by the motor to the motor speed. The default value is 5.11 s*V/rad/A.

Armature inductance

Inductance of the armature. If you do not have information about this inductance, set the value of this parameter to a small, nonzero number. The default value is 0.1 H. The value can be zero.

Field winding inductance

Inductance of the field winding. If you do not have information about this inductance, set the value of this parameter to a small, nonzero number. The default value is 0.1 H. The value can be zero.

No-load speed

Speed of the motor when no load is applied. This parameter is only visible when you select By `rated power, rated speed & no-load speed` for the **Model parameterization** parameter. The default value is `4.6e+03` rpm.

Rated speed (at rated load)

Motor speed at the rated load. This parameter is only visible when you select By `rated power, rated speed & no-load speed` for the **Model parameterization** parameter. The default value is `4e+03` rpm.

Rated load (mechanical power)

The mechanical load for which the motor is rated to operate. This parameter is only visible when you select By `rated power, rated speed & no-load speed` for the **Model parameterization** parameter. The default value is `50` W.

Rated DC supply voltage

The voltage at which the motor is rated to operate. This parameter is only visible when you select By `rated power, rated speed & no-load speed` for the **Model parameterization** parameter. The default value is `220` V.

Starting current at rated DC supply voltage

The initial current when starting the motor with the rated DC supply voltage. This parameter is only visible when you select By `rated power, rated speed & no-load speed` for the **Model parameterization** parameter. The default value is `2.09` A.

Mechanical Tab

Rotor inertia

Rotor inertia. The default value is `2e-04` kg*m^2. The value can be zero.

Rotor damping

Rotor damping. The default value is `1e-06` N*m/(rad/s). The value can be zero.

Initial rotor speed

Speed of the rotor at the start of the simulation. The default value is `0` rpm.

Temperature Dependence Tab

This tab appears only for blocks with exposed thermal ports. For more information, see "Thermal Ports" on page 1-663.

Resistance temperature coefficients, [alpha_f alpha_a]

A 1 by 2 row vector defining the coefficient α in the equation relating resistance to temperature, as described in "Thermal Model for Actuator Blocks". The first element corresponds to the field winding, and the second to the armature. The default value is for copper, and is [0.00393 0.00393] 1/K.

Measurement temperature

The temperature for which motor parameters are defined. The default value is 25 °C.

Thermal Port Tab

This tab appears only for blocks with exposed thermal ports. For more information, see "Thermal Ports" on page 1-663.

Thermal masses, [Mf Ma]

A 1 by 2 row vector defining the thermal mass for the field and armature windings. The thermal mass is the energy required to raise the temperature by one degree. The default value is [100 100] J/K.

Initial temperatures, [Tf Ta]

A 1 by 2 row vector defining the temperature of the field and armature thermal ports at the start of simulation. The default value is [25 25] °C.

Ports

The block has the following ports:

+

Positive electrical input.

-

Negative electrical input.

C

Mechanical rotational conserving port.

R

Mechanical rotational conserving port.

Hf

    Field winding thermal port. For more information, see "Thermal Ports" on page 1-663.

Ha

    Armature winding thermal port. For more information, see "Thermal Ports" on page 1-663.

References

[1] Bolton, W. *Mechatronics: Electronic Control Systems in Mechanical and Electrical Engineering*, 3rd edition Pearson Education, 2004.

See Also

DC Motor | Induction Motor | Servomotor | Universal Motor

Sinusoidal Current Source

Model damped sinusoidal current source

Library

SPICE-Compatible Components/Sources

Description

The Sinusoidal Current Source block represents a damped sinusoidal current source whose output current is independent of the voltage across the terminals of the source. The following equations describe the current through the source as a function of time:

$$I_{out}\left(Time \leq TD\right) = IO$$

$$I_{out}\left(Time > TD\right) = IO + IA * e^{-(Time-TD)*DF} * \sin\left(2\pi * FREQ * \left(Time - TD\right)\right)$$

where:

- *I0* is the **Current offset, IO** parameter value.
- *IA* is the **Sinusoidal amplitude, IA** parameter value.
- *FREQ* is the **Sinusoidal frequency, FREQ** parameter value.
- *TD* is the **Time delay, TD** parameter value.
- *DF* is the **Damping factor, DF** parameter value.

The block uses a small conductance internally to prevent numerical simulation issues. The conductance connects the + and - ports of the device and has a conductance *GMIN*:

- By default, *GMIN* matches the **Minimum conductance GMIN** parameter of the SPICE Environment Parameters block, whose default value is 1e–12.

- To change *GMIN*, add a SPICE Environment Parameters block to your model and set the **Minimum conductance GMIN** parameter to the desired value.

Parameters

Current offset, I0

The magnitude of the time-independent part of the output current. The default value is 0 A.

Sinusoidal amplitude, IA

The magnitude of the sinusoidal part of the output current. The default value is 0 A.

Sinusoidal frequency, FREQ

The frequency of the output sine wave. The default value is 1e+06 Hz. The value can be less than 0.

Time delay, TD

The time at which the sine wave first starts. The default value is 0 s. The value can be less than 0.

Damping factor, DF

The amount by which to amplify or reduce the exponential damping term that multiples the sine wave to produce the output current. The default value is 0 1/s. The value must be greater than or equal to 0.

Ports

The block has the following ports:

+

Positive electrical voltage.

-

Negative electrical voltage.

See Also

Sinusoidal Voltage Source

Sinusoidal Voltage Source

Model damped sinusoidal voltage source

Library

SPICE-Compatible Components/Sources

Description

The Sinusoidal Voltage Source block represents a damped sinusoidal voltage source whose output voltage is independent of the current through the source. The following equations describe the output as a function of time:

$$V_{out}\left(Time \leq TD\right) = VO$$

$$V_{out}\left(Time > TD\right) = VO + VA * e^{-(Time-TD)*DF} * \sin\left(2\pi * FREQ * \left(Time - TD\right)\right)$$

where:

- $V0$ is the **Voltage offset, VO** parameter value.
- VA is the **Sinusoidal amplitude, VA** parameter value.
- $FREQ$ is the **Sinusoidal frequency, FREQ** parameter value.
- TD is the **Time delay, TD** parameter value.
- DF is the **Damping factor, DF** parameter value.

Parameters

Voltage offset, V0

The magnitude of the time-independent part of the output voltage. The default value is 0 V.

Sinusoidal amplitude, VA

The magnitude of the sinusoidal part of the output voltage. The default value is 0 V.

Sinusoidal frequency, FREQ

The frequency of the output sine wave. The default value is `1e+06` Hz. The value can be less than 0.

Time delay, TD

The time at which the sine wave first starts. The default value is 0 s. The value can be less than 0.

Damping factor, DF

The amount by which to amplify or reduce the exponential damping term that multiples the sine wave to produce the output voltage. The default value is 0 1/s. The value must be greater than or equal to 0.

Ports

The block has the following ports:

+

Positive electrical voltage.

-

Negative electrical voltage.

See Also

Sinusoidal Current Source

Solar Cell

Solar cell model

Library

Sources

Description

The Solar Cell block represents a solar cell current source.

The solar cell model includes the following components:

- "Solar-Induced Current" on page 1-673
- "Temperature Dependence" on page 1-675
- "Thermal Port" on page 1-677

Solar-Induced Current

The block represents a single solar cell as a resistance R_s that is connected in series with a parallel combination of the following elements:

- Current source
- Two exponential diodes
- Parallel resistor R_p

The following illustration shows the equivalent circuit diagram:
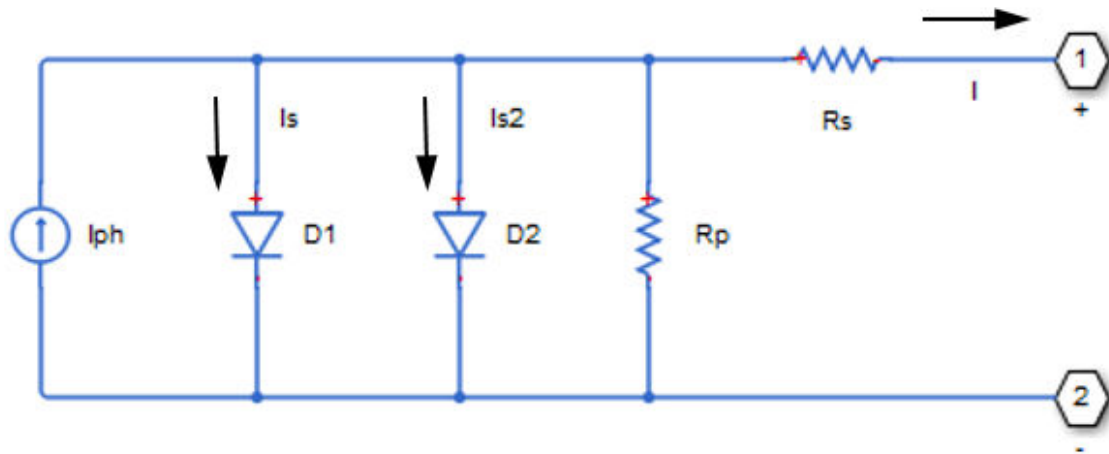
The output current *I* is:

$$I = I_{ph} - I_s * \left(e^{(V+I*R_s)/(N*V_t)} - 1\right) - I_{s2} * (e^{(V+I*R_s)/(N_2*V_t)} - 1) - \left(V + I*R_s\right)/R_p$$

where:

- I_{ph} is the solar-induced current:

$$I_{ph} = I_{ph0} \times \frac{I_r}{I_{r0}}$$

  where:

  - I_r is the irradiance (light intensity) in W/m² falling on the cell.
  - I_{ph0} is the measured solar-generated current for the irradiance I_{r0}.
- I_s is the saturation current of the first diode.
- I_{s2} is the saturation current of the second diode.
- V_t is the thermal voltage, kT/q, where:

  - k is the Boltzmann constant.

- T is the **Device simulation temperature** parameter value.
- q is the elementary charge on an electron.
- N is the quality factor (diode emission coefficient) of the first diode.
- N_2 is the quality factor (diode emission coefficient) of the second diode.
- V is the voltage across the solar cell electrical ports.

The quality factor varies for amorphous cells, and is typically 2 for polycrystalline cells.

The block lets you choose between two models:

- An 8-parameter model where the preceding equation describes the output current
- A 5-parameter model that applies the following simplifying assumptions to the preceding equation:

    - The saturation current of the second diode is zero.
    - The impedance of the parallel resistor is infinite.

If you choose the 5-parameter model, you can parameterize this block in terms of the preceding equivalent circuit model parameters or in terms of the short-circuit current and open-circuit voltage the block uses to derive these parameters.

All models adjust the block resistance and current parameters as a function of temperature.

You can model any number of solar cells connected in series using a single Solar Cell block by setting the parameter **Number of series cells** to a value larger than 1. Internally the block still simulates only the equations for a single solar cell, but scales up the output voltage according to the number of cells. This results in a more efficient simulation than if equations for each cell were simulated individually.

If you want to model N cells in parallel, you can do so for single cells by scaling the parameter values accordingly. That is, multiply short-circuit current, diode saturation current, and solar-generated currents by N, and divide series resistance by N. To connect solar cell blocks in parallel, where each block contains multiple cells in series, make multiple copies of the block and connect accordingly.

Temperature Dependence

Several solar cell parameters depend on temperature. The solar cell temperature is specified by the **Device simulation temperature** parameter value.

The block provides the following relationship between the solar-induced current I_{ph} and the solar cell temperature T:

$$I_{ph}(t) = I_{ph} * \left(1 + TIPH1 * \left(T - T_{meas}\right)\right)$$

where:

- *TIPH1* is the **First order temperature coefficient for Iph, TIPH1** parameter value.
- T_{meas} is the **Measurement temperature** parameter value.

The block provides the following relationship between the saturation current of the first diode I_s and the solar cell temperature T:

$$I_{s1}(T) = I_{s1} * \left(\frac{T}{T_{meas}}\right)^{\left(TXIS1/N\right)} * e^{\left(EG*\left(\frac{T}{T_{meas}}-1\right)\middle/\left(N*V_t\right)\right)}$$

where *TXIS1* is the **Temperature exponent for Is, TXIS1** parameter value.

The block provides the following relationship between the saturation current of the second diode I_{s2} and the solar cell temperature T:

$$I_{s2}(T) = I_{s2} * \left(\frac{T}{T_{meas}}\right)^{\left(TXIS2/N_2\right)} * e^{\left(EG*\left(\frac{T}{T_{meas}}-1\right)\middle/\left(N_2*V_t\right)\right)}$$

where *TXIS2* is the **Temperature exponent for Is2, TXIS2** parameter value.

The block provides the following relationship between the series resistance R_s and the solar cell temperature T:

$$R_s(T) = R_s * \left(\frac{T}{T_{meas}}\right)^{TRS1}$$

where *TRS1* is the **Temperature exponent for Rs, TRS1** parameter value.

The block provides the following relationship between the parallel resistance R_p and the solar cell temperature T:

$$R_p(T) = R_p * \left(\frac{T}{T_{meas}} \right)^{TRP1}$$

where *TRP1* is the **Temperature exponent for Rp, TRP1** parameter value.

Thermal Port

The block has an optional thermal port, hidden by default. To expose the thermal port, right-click the block in your model, and then from the context menu select **Simscape > Block choices > Show thermal port**. This action displays the thermal port H on the block icon, and adds the **Thermal Port** tab to the block dialog box.

The thermal port model, shown in the following illustration, represents just the thermal mass of the device. The thermal mass is directly connected to the component thermal port H. An internal Ideal Heat Flow Source supplies a heat flow to the port and thermal mass. This heat flow represents the internally generated heat.
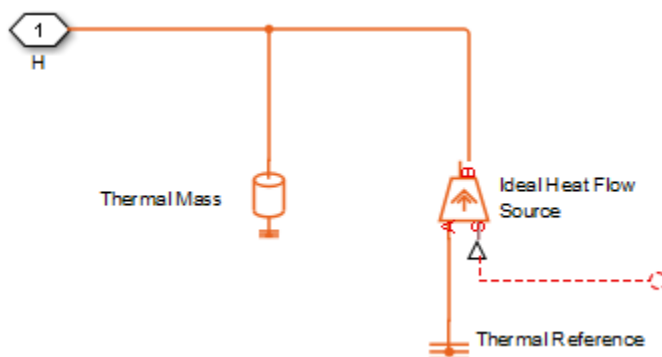
The internally generated heat in the solar cell is calculated according to the equivalent circuit diagram, shown at the beginning of the reference page, in the "Solar-Induced Current" on page 1-673 section. It is the sum of the i^2·R losses for each of the resistors plus the losses in each of the diodes.

The internally generated heat due to electrical losses is a separate heating effect to that of the solar irradiation. To model thermal heating due to solar irradiation, you must account for it separately in your model and add the heat flow to the physical node connected to the solar cell thermal port.

Parameters

- "Cell Characteristics Tab" on page 1-678
- "Configuration Tab" on page 1-680
- "Temperature Dependence Tab" on page 1-680
- "Thermal Port Tab" on page 1-681

Cell Characteristics Tab

Parameterize by

Select one of the following methods for block parameterization:

- By `s/c current and o/c voltage, 5 parameter` — Provide short-circuit current and open-circuit voltage that the block converts to an equivalent circuit model of the solar cell. This is the default option.

- By `equivalent circuit parameters, 5 parameter` — Provide electrical parameters for an equivalent circuit model of the solar cell using the 5-parameter solar cell model that makes the following assumptions:

  - The saturation current of the second diode is zero.
  - The parallel resistor has infinite impedance.

- By `equivalent circuit parameters, 8 parameter` — Provide electrical parameters for an equivalent circuit model of the solar cell using the 8-parameter solar cell model.

Short-circuit current, Isc

The current that flows when you short-circuit the solar cell. This parameter is only visible when you select By `s/c current and o/c voltage, 5 parameter` for the **Parameterize by** parameter. The default value is `7.34` A.

Open-circuit voltage, Voc

The voltage across the solar cell when it is not connected. This parameter is only visible when you select By `s/c current and o/c voltage, 5 parameter` for the **Parameterize by** parameter. The default value is `0.6` V.

Diode saturation current, Is

The asymptotic reverse current of the first diode for increasing reverse bias in the absence of any incident light. This parameter is only visible when you select one of the following settings:

- By equivalent circuit parameters, 5 parameter for the **Parameterize by** parameter
- By equivalent circuit parameters, 8 parameter for the **Parameterize by** parameter

The default value is `1e-06` A.

Diode saturation current, Is2

The asymptotic reverse current of the second diode for increasing reverse bias in the absence of any incident light. This parameter is only visible when you select By equivalent circuit parameters, 8 parameter for the **Parameterize by** parameter. The default value is `0` A.

Solar-generated current, Iph0

The solar-induced current when the irradiance is I_{r0}. This parameter is only visible when you select one of the following settings:

- By equivalent circuit parameters, 5 parameter for the **Parameterize by** parameter
- By equivalent circuit parameters, 8 parameter for the **Parameterize by** parameter

The default value is `7.34` A.

Irradiance used for measurements, Ir0

The irradiance that produces a current of I_{ph0} in the solar cell. The default value is `1000` W/m^2.

Quality factor, N

The emission coefficient of the first diode. The default value is `1.5`.

Quality factor, N2

The emission coefficient of the second diode. This parameter is only visible when you select By equivalent circuit parameters, 8 parameter for the **Parameterize by** parameter. The default value is `2`.

Series resistance, Rs

The internal series resistance. The default value is `0` Ω.

Parallel resistance, Rp

The internal parallel resistance. This parameter is only visible when you select By equivalent circuit parameters, 8 parameter for the **Parameterize by** parameter. The default value is `inf` Ω.

Configuration Tab

Number of series cells

The number of series-connected solar cells modeled by the block. The default value is 1. The value must be greater than 0.

Temperature Dependence Tab

First order temperature coefficient for Iph, TIPH1

The order of the linear increase in the solar-generated current as temperature increases. The default value is 0 1/K. The value must be greater than or equal to 0.

Energy gap, EG

The solar cell activation energy. The default value is 1.11 eV. The value must be greater than or equal to 0.1.

Temperature exponent for Is, TXIS1

The order of the exponential increase in the current from the first diode as temperature increases. The default value is 3. The value must be greater than or equal to 0.

Temperature exponent for Is2, TXIS2

The order of the exponential increase in the current from the second diode as temperature increases. This parameter is only visible when you select By equivalent circuit parameters, 8 parameter for the **Parameterize by** parameter. The default value is 3. The value must be greater than or equal to 0.

Temperature exponent for Rs, TRS1

The order of the exponential increase in the series resistance as temperature increases. The default value is 0. The value must be greater than or equal to 0.

Temperature exponent for Rp, TRP1

The order of the exponential increase in the parallel resistance as temperature increases. This parameter is only visible when you select By equivalent circuit parameters, 8 parameter for the **Parameterize by** parameter. The default value is 0. The value must be greater than or equal to 0.

Measurement temperature

The temperature at which the solar cell parameters were measured. The default value is 25 C. The value must be greater than 0.

Device simulation temperature

The temperature at which the solar cell is simulated. The default value is 25 C. The value must be greater than 0.

Thermal Port Tab

This tab appears only for blocks with exposed thermal ports. For more information, see "Thermal Port" on page 1-677.

Thermal mass

The heat energy required to raise the temperature of the solar cell by one degree. When modeling more than one cell in series, specify the thermal mass for a single cell. This value gets multiplied internally by the number of cells to determine the total thermal mass. The default value is 100 J/K.

Initial temperature

The temperature of the solar cell at the start of simulation. The default value is 25 C.

Ports

The block has the following ports:

Ir

Incident irradiance

+

Positive electrical voltage

-

Negative electrical voltage

References

[1] Gow, J.A. and C.D. Manning. "Development of a Photovoltaic Array Model for Use in Power-Electronics Simulation Studies." *IEEE Proceedings of Electric Power Applications*, Vol. 146, No. 2, 1999, pp. 193–200.

Solenoid

Model electrical characteristics and generated force of solenoid

Library

Translational Actuators

Description

The Solenoid block represents the electrical characteristics and generated force for the solenoid in the following figure:
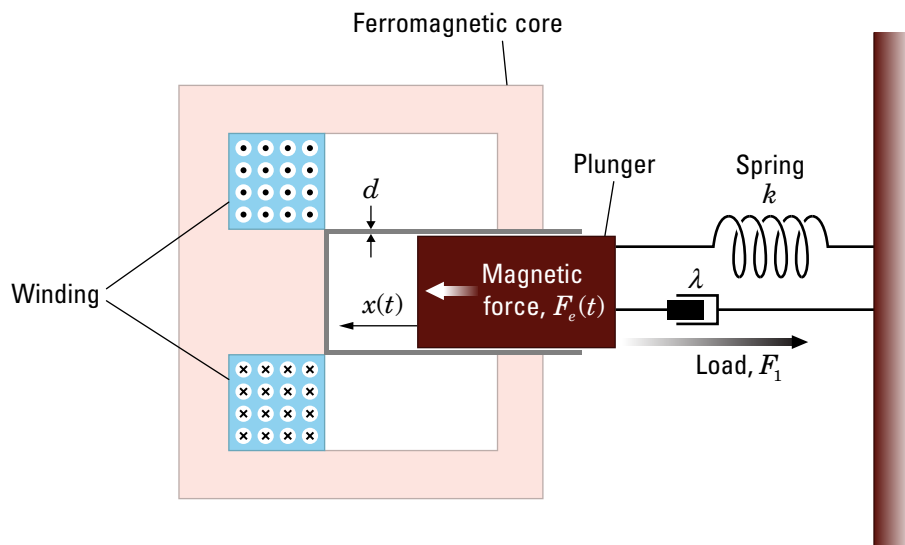
The return spring is optional. To remove the effects of this spring from the model, set the **Spring constant** parameter to `0`.

The equation of motion for the plunger as a function of position, x, is:

$$F_l + m\ddot{x} + \lambda\dot{x} + kx = F_e$$

where F_e is the electromagnetic force, F_l is the load force, λ is the viscous damping term and m is the plunger mass. The electromagnetic force is related to the solenoid current and inductance by:

$$F_e = \frac{1}{2}i^2\frac{\partial L(x)}{\partial x}$$

The inductance, which is derived in [1], can be written as:

$$\frac{\partial L(x)}{\partial x} = \frac{-\beta}{\left(\alpha + \beta x\right)^2}$$

where α and β are constants. Plugging the preceding equation into the equation for electromagnetic force gives the force-stroke relationship of the solenoid for a current i_0:

$$F = \frac{1}{2}i_0{}^2\frac{-\beta}{\left(\alpha + \beta x\right)^2}$$

The Solenoid block solves for α and β by taking the two specified force and stroke measurements and substituting them into the preceding equation. It solves the resulting equations for α and β.

A positive current from the electrical + to - ports creates a negative force (i.e., a pulling force) from the mechanical C to R ports.

Thermal Port

The block has an optional thermal port, hidden by default. To expose the thermal port, right-click the block in your model, and then from the context menu select **Simscape > Block choices > Show thermal port**. This action displays the thermal port H on the

block icon, and adds the **Temperature Dependence** and **Thermal Port** tabs to the block dialog box.

Use the thermal port to simulate the effects of copper resistance losses that convert electrical power to heat. For more information on using thermal ports and on the **Temperature Dependence** and **Thermal Port** tab parameters, see "Simulating Thermal Effects in Rotational and Translational Actuators".

Parameters

- "Magnetic Force Tab" on page 1-684
- "Mechanical Tab" on page 1-685

Magnetic Force Tab

Forces [F1 F2]

A vector of the force values at the two points on the force-stroke curve. The second measurement point must be at a stroke that is greater than that of the first measurement point. When the manufacturer does not provide a force-stroke curve, set F1 to the holding force (when X1 = 0) and F2 to the pull-in force when running the solenoid at the **Rated voltage Vdc** and **Rated current Idc** values. The default value is [7.5 0.75] N.

Stroke [X1 X2]

A vector of the stroke (plunger distance from the fully closed position) values at the two points on the force-stroke curve. The second measurement point must be at a stroke that is greater than that of the first measurement point. To ensure a finite force value, the points must meet the condition

$$\frac{X2}{X1} > \sqrt{\frac{F1}{F2}}$$

The default value is [1 5] mm.

Rated voltage Vdc

The voltage at which the solenoid is rated to operate. This voltage value is used to measure the **Forces [F1 F2]** and **Stroke [X1 X2]** values. The default value is 50 V.

Rated current Idc

The current that flows when the solenoid is supplied with the **Rated voltage Vdc** voltage. The default value is 0.05 A.

Mechanical Tab

Spring constant

Constant representing the stiffness of the spring that acts to retract the plunger when the solenoid is powered off. The force is zero when the plunger is displaced to the **Stroke for zero spring force** parameter value. The default value is 200 N/m. Set the spring constant to zero if there is no spring.

Stroke for zero spring force

The stroke at which the spring provides no force. The default value is 5 mm.

Damping

The term λ in the equation of motion for the plunger as a function of position that linearly damps the plunger motion. The default value is 1 N/(m/s). The value can be zero.

Plunger mass

The weight of the solenoid plunger. The default value is 0.05 kg. The value can be zero.

Maximum stroke

The maximum amount by which the plunger can be displaced. You can use this parameter to model a hard endstop that limits the stroke. The default value is Inf mm, which means no stroke limit.

Initial plunger position

The amount by which the plunger is displaced at the start of the simulation. The default value is 0 m.

Contact stiffness

Stiffness of the plunger contact that models the hard stop at the minimum ($x = 0$) and maximum ($x =$ **Maximum stroke**) plunger positions. The default value is 1e+06 N/m.

Contact damping

Damping of the plunger contact that models the hard stop at the minimum ($x = 0$) and maximum ($x =$ **Maximum stroke**) plunger positions. The default value is 500 N/(m/s).

Ports

The block has the following ports:

+

    Positive electrical input

-

    Negative electrical input

C

    Mechanical translational conserving port

R

    Mechanical translational conserving port

References

[1] S.E. Lyshevski. *Electromechanical Systems, Electric Machines, and Applied Mechatronics*. CRC, 1999.

SPDT Switch

Model single-pole double-throw switch
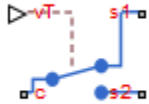
Library

Passive Devices/Switches

Description

The SPDT Switch block models a single-pole double-throw switch:

- When the switch is closed, port c is connected to port s2.
- When the switch is open, port c is connected to port s1.

Closed connections are modeled by a resistor with value equal to the **Closed resistance** parameter value. Open connections are modeled by a resistor with value equal to the reciprocal of the **Open conductance** parameter value.

If the **Threshold width** parameter is set to zero, the switch is closed if the voltage presented at the vT control port exceeds the value of the **Threshold** parameter.

If the **Threshold width** parameter is greater than zero, then switch conductance G varies smoothly between off-state and on-state values:

$$G = \frac{x}{R_{closed}} + (1 - x)G_{open}$$

$$\lambda = \frac{vT - \text{Threshold}}{\text{Threshold width}}$$

$$x = \begin{cases} 0 & \text{for } \lambda \leq 0 \\ 3\lambda^2 - 2\lambda^3 & \text{for } 0 < \lambda < 1 \\ 1 & \text{for } \lambda \geq 1 \end{cases}$$

The block uses the function $3\lambda^2 - 2\lambda^3$ because its derivative is zero for $\lambda = 0$ and $\lambda = 1$.

Defining a small positive **Threshold width** can help solver convergence in some models, particularly if the control port signal vT varies continuously as a function of other network variables. However, defining a nonzero threshold width precludes the solver making use of switched linear optimizations. Therefore, if the rest of your network is switched linear, MathWorks recommends that you set **Threshold width** to zero.

Optionally, you can add a delay between the point at which the voltage at vT passes the threshold and the switch opening or closing. To enable the delay, on the **Dynamics** tab, set the **Model dynamics** parameter to `Model turn-on and turn-off times`.

Parameters

- "Main Tab" on page 1-688
- "Dynamics Tab" on page 1-689

Main Tab

Closed resistance

Resistance between the c and s electrical ports when the switch is closed. The value must be greater than zero. The default value is `0.01` Ω.

Open conductance

Conductance between the c and s electrical ports when the switch is open. The value must be greater than zero. The default value is `1e-6` S.

Threshold

The threshold voltage for the control physical signal input vT above which the switch will turn on. The default value is `0` V.

Threshold width

The minimum increase in the control signal vT above the threshold value that will move the switch from fully open to fully closed. The default value is `0` V.

Dynamics Tab

Model dynamics

Select whether the block models a switching delay:

- `No dynamics` — Do not model the delay. This is the default option.
- `Model turn-on and turn-off times` — Use additional parameters to model a delay between the point at which the voltage at vT passes the threshold and the switch opening or closing.

Turn-on delay

Time between the input voltage exceeding the threshold voltage and the switch closing. This parameter is only visible when you select `Model turn-on and turn-off times` for the **Model dynamics** parameter. The value must be greater than zero. The default value is `1e-3` seconds.

Turn-off delay

Time between the input voltage falling below the threshold voltage and the switch opening. This parameter is only visible when you select `Model turn-on and turn-off times` for the **Model dynamics** parameter. The value must be greater than zero. The default value is `1e-3` seconds.

Initial input value, vT

The value of the physical signal input vT at time zero. This value is used to initialize the delayed control voltage parameter internally. This parameter is only visible when you select `Model turn-on and turn-off times` for the **Model dynamics** parameter. The default value is `0` V.

Ports

This block has the following ports:

vT

Physical signal that opens and closes the switch

c, s1, s2

Electrical conserving ports

See Also

DPDT Switch | DPST Switch | SPST Switch | Switch

SPICE Diode

Model SPICE-compatible diode
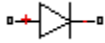
Library

SPICE-Compatible Components/Semiconductor Devices

Description

The SPICE Diode block represents a SPICE-compatible diode.

The SPICE Diode block model includes the following components:

Current-Voltage Model

The block provides the following relationship between the diode current I_d and the diode voltage V_d after adjusting the applicable model parameters for temperature.

Applicable Range of V_d Values	Corresponding I_d Equation
$V_d > 80 * V_t$	$I_d = IS\left(\left(\dfrac{V_d}{V_t} - 79\right)e^{80} - 1\right) + V_d * G\min$
$80 * V_t \geq V_d \geq -3 * V_t$	$I_d = IS * (e^{V_d/V_t} - 1) + V_d * G\min$

Applicable Range of V_d Values	Corresponding I_d Equation
$-3*V_t > V_d \geq -BV$	$I_d = -IS \left(1 + \dfrac{27}{\left(V_d / V_t\right)^3 e^3} \right) + V_d * G\min$
$V_d < -BV$	$I_d = -IBV * \left(e^{(-(BV+V_d)/V_t)} - 1 \right) -$ $IS * \left(1 - \left(\dfrac{3}{e*BV\Big/V_t} \right)^3 \right) + V_d * G\min$

Where:

- *IS* is the **Saturation current, IS** parameter value.
-
  $V_t = N * k * T / q$
- *N* is the **Emission coefficient, N** parameter value.
- *q* is the elementary charge on an electron.
- *k* is the Boltzmann constant.
- *T* is the diode temperature:

  - If you select `Device temperature` for the **Model temperature dependence using** parameter, *T* is the sum of the **Circuit temperature** value plus the **Offset local circuit temperature, TOFFSET** parameter value. The **Circuit temperature** value comes from the SPICE Environment Parameters block, if one exists in the circuit. Otherwise, it comes from the default value for this block.

  - If you select `Fixed temperature` for the **Model temperature dependence using** parameter, *T* is the **Fixed circuit temperature, TFIXED** parameter value.

- *GMIN* is the diode minimum conductance. By default, *GMIN* matches the **Minimum conductance, GMIN** parameter of the SPICE Environment Parameters block, whose default value is 1e–12. To change *GMIN*, add a SPICE Environment Parameters block to your model and set the **Minimum conductance, GMIN** parameter to the desired value.

- *BV* is the **Reverse breakdown voltage, BV** parameter value.

Junction Charge Model

The block provides the following relationship between the diode charge Q_d and the diode voltage V_d after adjusting the applicable model parameters for temperature.

Applicable Range of V_d Values	Corresponding Q_d Equation
$V_d < FC*VJ$	$$Q_d = TT*I_d + CJO*VJ* \frac{1 - \left(1 - \dfrac{V_d}{VJ}\right)^{1-MG}}{1 - MG}$$
$V_d \geq FC*VJ$	$$Q_d = TT*I_d + \\ CJO* \left(F1 + \frac{F3*(V_d - FC*VJ) + \left(\dfrac{MG}{2*VJ}\right)*\left(V_d^2 - (FC*VJ)^2\right)}{F2} \right)$$

Where:

- *FC* is the **Capacitance coefficient, FC** parameter value.
- *VJ* is the **Junction potential, VJ** parameter value.
- *TT* is the **Transit time, TT** parameter value.
- *CJ0* is the **Zero-bias junction capacitance, CJ0** parameter value.
- *MG* is the **Grading coefficient, M** parameter value.

- $$F1 = VJ*(1 - (1 - FC)^{(1-MG)})/(1 - MG)$$

- $$F2 = (1 - FC)^{(1+MG)}$$

- $$F3 = 1 - FC*(1 + MG)$$

Temperature Dependence

Several diode parameters depend on temperature. There are two ways to specify the diode temperature:

- When you select `Device temperature` for the **Model temperature dependence using** parameter, the diode temperature is

$$T = T_C + T_O$$

where:

  - T_C is the **Circuit temperature** parameter value from the SPICE Environment Parameters block. If this block doesn't exist in the circuit, T_C is the default value of this parameter.
  - T_O is the **Offset local circuit temperature, TOFFSET** parameter value.
- When you select `Fixed temperature` for the **Model temperature dependence using** parameter, the diode temperature is the **Fixed circuit temperature, TFIXED** parameter value.

The block provides the following relationship between the saturation current *IS* and the diode temperature *T*:

$$IS(T) = IS * \left(T/T_{meas}\right)^{\frac{XTI}{ND}} * e^{\left(\frac{T}{T_{meas}} - 1\right) * \frac{EG}{V_t}}$$

where:

- *IS* is the **Transport saturation current, IS** parameter value.
- T_{meas} is the **Parameter extraction temperature, TMEAS** parameter value.
- *XTI* is the **Saturation current temperature exponent, XTI** parameter value.
- *ND* is the **Emission coefficient, N** parameter value.
- *EG* is the **Activation energy, EG** parameter value.
- $V_t = kT/q$.

The block provides the following relationship between the junction potential *VJ* and the diode temperature *T*:

$$VJ(T) = VJ * \left(\frac{T}{T_{meas}}\right) - \frac{3 * k * T}{q} * \log\left(\frac{T}{T_{meas}}\right) - \left(\frac{T}{T_{meas}}\right) * EG_{T_{meas}} + EG_T$$

where:

- *VJ* is the **Junction potential, VJ** parameter value.

-
$$EG_{T_{meas}} = 1.16eV - \left(7.02e\text{-}4 * T_{meas}^{\,2}\right) / \left(T_{meas} + 1108\right)$$

-
$$EG_T = 1.16eV - \left(7.02e\text{-}4 * T^{\,2}\right) / \left(T + 1108\right)$$

The block provides the following relationship between the junction capacitance *CJO* and the diode temperature *T*:

$$CJO(T) = CJO * \left[1 + MG * \left(400e - 6 * \left(T - T_{meas}\right) - \frac{VJ(T) - VJ}{VJ}\right)\right]$$

where *CJO* is the **Zero-bias junction capacitance, CJ0** parameter value.

Basic Assumptions and Limitations

The model is based on the following assumptions:

- The SPICE Diode block does not support noise analysis.
- The SPICE Diode block applies initial conditions across junction capacitors and not across the block ports.

Parameters

Main Tab

Device area, AREA

> The diode area. This value multiplies the **Saturation current, IS**, **Zero-bias junction capacitance CJ0**, and **Reverse breakdown current, IBV** parameter values. It divides the **Ohmic resistance, RS** parameter value. The default value is 1 m^2. The value must be greater than 0.

Number of parallel devices, SCALE

> The number of parallel diodes the block represents. This value multiplies the output current and device charges. The default value is 1. The value must be greater than 0.

Saturation current, IS

The magnitude of the current that the ideal diode equation approaches asymptotically for very large reverse bias levels. The default value is `1e-14` A/m^2. The value must be greater than or equal to `0`.

Ohmic resistance, RS

The series diode connection resistance. The default value is `0.01` m^2*Ω. The value must be greater than or equal to `0`.

Emission coefficient, N

The diode emission coefficient or ideality factor. The default value is `1`. The value must be greater than `0`.

Junction Capacitance Tab

Model junction capacitance

Select one of the following options for modeling the junction capacitance:

- `No` — Do not include junction capacitance in the model. This is the default option.
- `Yes` — Specify zero-bias junction capacitance, junction potential, grading coefficient, forward-bias depletion capacitance coefficient, and transit time.

Zero-bias junction capacitance CJ0

The value of the capacitance placed in parallel with the exponential diode term. This parameter is only visible when you select `Yes` for the **Model junction capacitance** parameter. The default value is `0` F/m^2. The value must be greater than or equal to `0`.

Junction potential, VJ

The junction potential. This parameter is only visible when you select `Yes` for the **Model junction capacitance** parameter. The default value is `1` V. The value must be greater than `0.01` V.

Grading coefficient, M

The grading coefficient. This parameter is only visible when you select `Yes` for the **Model junction capacitance** parameter. The default value is `0.5`. The value must be greater than `0` and less than `0.9`.

Capacitance coefficient, FC

The fitting coefficient that quantifies the decrease of the depletion capacitance with applied voltage. This parameter is only visible when you select `Yes` for the **Model**

junction capacitance parameter. The default value is `0.5`. The value must be greater than or equal to `0` and less than `0.95`.

Transit time, TT

The transit time of the minority carriers that cause diffusion capacitance. This parameter is only visible when you select `Yes` for the **Model junction capacitance** parameter. The default value is `0` s. The value must be greater than or equal to `0`.

Specify initial condition

Select one of the following options for specifying an initial condition:

- `No` — Do not specify an initial condition for the model. This is the default option.
- `Yes` — Specify the initial diode voltage.

> **Note** The SPICE Diode block applies the initial diode voltage across the junction capacitors and not across the ports.

Initial voltage, V0

Diode voltage at the start of the simulation. This parameter is only visible when you select `Yes` for the **Model junction capacitance** and `Yes` for the **Specify initial condition** parameter. The default value is `0` V.

> **Note** The block applies the initial condition across the diode junction, so the initial condition is only effective when charge storage is included, i.e. when one or both of the **Zero-bias junction capacitance, CJ0** and **Transit time, TT** parameters are greater than zero.

Reverse Breakdown Tab

Model reverse breakdown

Select one of the following options for modeling the diode reverse breakdown:

- `No` — Don't model reverse breakdown. This is the default option.
- `Yes` — Introduce a second exponential term to the diode I-V relationship, thereby modeling a rapid increase in conductance as the breakdown voltage is exceeded.

Reverse breakdown current, IBV

The diode current that corresponds to the **Reverse breakdown voltage, BV** value. This parameter is only visible when you select `Yes` for the **Model reverse**

breakdown parameter. The default value is `0.001` A/m$_2$. The value must be greater than `0`.

Note The Diode model does not use this parameter at this time.

Reverse breakdown voltage, BV

The voltage below which to model the rapid increase in conductance that occurs at diode breakdown. This parameter is only visible when you select `Yes` for the **Model reverse breakdown** parameter. The default value is `Inf` V. The value must be greater than or equal to `0`.

Temperature Tab

Model temperature dependence using

Select one of the following options for modeling the diode temperature dependence:

- `Device temperature` — Use the device temperature, which is the **Circuit temperature** parameter value (from the SPICE Environment Parameters block, if one exists in the circuit, or the default value for this block otherwise) plus the **Offset local circuit temperature, TOFFSET** parameter value.

- `Fixed temperature` — Use a temperature that is independent of the circuit temperature to model temperature dependence.

Saturation current temperature exponent, XTI

The order of the exponential increase in the saturation current as temperature increases. This parameter is only visible when you select `Device temperature` for the **Model temperature dependence using** parameter. The default value is `3`. The value must be greater than `0`.

Activation energy, EG

The diode activation energy. This parameter is only visible when you select `Device temperature` for the **Model temperature dependence using** parameter. The default value is `1.11` eV. The value must be greater than or equal to `0.1`.

Offset local circuit temperature, TOFFSET

The amount by which the diode temperature differs from the circuit temperature. This parameter is only visible when you select `Device temperature` for the **Model temperature dependence using** parameter. The default value is `0` K.

Parameter extraction temperature, TMEAS

The temperature at which the diode parameters were measured. The default value is `300.15` K. The value must be greater than `0`.

Fixed circuit temperature, TFIXED

The temperature at which to simulate the diode. This parameter is only visible when you select `Fixed temperature` for the **Model temperature dependence using** parameter. The default value is `300.15` K. The value must be greater than `0`.

Ports

The block has the following ports:

+

Positive electrical voltage.

-

Negative electrical voltage.

See Also

Diode

SPICE Environment Parameters

Set parameters that apply to all connected SPICE-compatible blocks

Library

SPICE-Compatible Components/Utilities

Description

The SPICE Environment Parameters block lets you set parameters that apply to all SPICE-compatible blocks in an electrical network:

- Circuit temperature
- Minimum conductance

If your Simulink model does not contain a SPICE Environment Parameters block, all blocks use the default values of these parameters. You must connect every network in the system to a SPICE Environment Parameters block to override the default values.

Note The simple semiconductor models in the Semiconductors sublibrary are not temperature dependent, so the SPICE Environment Parameters block only changes the minimum conductance parameter used by the exponential diode and bipolar transistor models.

Parameters

Circuit temperature

The temperature of the connected SPICE-compatible blocks. The default value is 300.15 K.

Minimum conductance GMIN

The minimum conductance used by some blocks. The default value is `1e-12` 1/Ω.

Ports

The block has the following ports:

OUT

Electrical output.

SPICE NJFET

Model SPICE-compatible N-Channel JFET

Library

SPICE-Compatible Components/Semiconductor Devices

Description

The NJFET block represents a SPICE-compatible N-channel JFET.

The NJFET block model includes the following components:

- "Gate-Source Current-Voltage Model" on page 1-702
- "Gate-Drain Current-Voltage Model" on page 1-703
- "Drain-Source Current-Voltage Model" on page 1-704
- "Junction Charge Model" on page 1-705
- "Temperature Dependence" on page 1-706

Gate-Source Current-Voltage Model

The block provides the following relationship between the gate-source current I_{gs} and the gate-source voltage V_{gs} after adjusting the applicable model parameters for temperature.

Applicable Range of V_{gs} Values	Corresponding I_{gs} Equation
$V_{gs} > 80 * V_t$	$I_{gs} = IS * \left(\left(\dfrac{V_{gs}}{V_t} - 79 \right) e^{80} - 1 \right) + V_{gs} * G \min$

Applicable Range of V_{gs} Values	Corresponding I_{gs} Equation
$80 * V_t \geq V_{gs}$	$I_{gs} = IS * \left(e^{V_{gs}/N_t} - 1 \right) + V_{gs} * G\min$

Where:

- *IS* is the **Saturation current, IS** parameter value.

- $$V_t = ND * k * T \,/\, q$$

- *ND* is the **Emission coefficient, N** parameter value.
- *q* is the elementary charge on an electron.
- *k* is the Boltzmann constant.
- *T* is the diode temperature:

    - If you select `Device temperature` for the **Model temperature dependence using** parameter, *T* is the sum of the **Circuit temperature** value plus the **Offset local circuit temperature, TOFFSET** parameter value. The **Circuit temperature** value comes from the SPICE Environment Parameters block, if one exists in the circuit. Otherwise, it comes from the default value for this block.
    - If you select `Fixed temperature` for the **Model temperature dependence using** parameter, *T* is the **Fixed circuit temperature, TFIXED** parameter value.

- *GMIN* is the diode minimum conductance. By default, *GMIN* matches the **Minimum conductance GMIN** parameter of the SPICE Environment Parameters block, whose default value is 1e-12. To change *GMIN*, add a SPICE Environment Parameters block to your model and set the **Minimum conductance GMIN** parameter to the desired value.

Gate-Drain Current-Voltage Model

The block provides the following relationship between the gate-drain current I_{gd} and the gate-drain voltage V_{gd} after adjusting the applicable model parameters for temperature.

Applicable Range of V_{gd} Values	Corresponding I_{gd} Equation
$V_{gd} > 80 * V_t$	$I_{gd} = IS * \left(\left(\dfrac{V_{gd}}{V_t} - 79 \right) e^{80} - 1 \right) + V_{gd} * G\min$
$80 * V_t \geq V_{gd}$	$I_{gd} = IS * \left(e^{V_{gd}/V_t} - 1 \right) + V_{gd} * G\min$

Drain-Source Current-Voltage Model

The block provides the following relationship between the drain-source current I_{ds} and the

drain-source voltage V_{ds} in normal mode ($V_{ds} \geq 0$) after adjusting the applicable model parameters for temperature.

Applicable Range of V_{gs} and V_{gd} Values	Corresponding I_{ds} Equation
$V_{gs} - V_{to} \leq 0$	$I_{ds} = 0$
$0 < V_{gs} - V_{to} \leq V_{ds}$	$I_{ds} = \beta \left(V_{gs} - V_{to} \right)^2 \left(1 + \lambda V_{ds} \right)$
$0 < V_{ds} < V_{gs} - V_{to}$	$I_{ds} = \beta V_{ds} \left(2 \left(V_{gs} - V_{to} \right) - V_{ds} \right) \left(1 + \lambda V_{ds} \right)$

Where:

- V_{to} is the **Threshold voltage, VTO** parameter value.
- β is the **Transconductance, BETA** parameter value.
- λ is the **Channel modulation, LAMBDA** parameter value.

The block provides the following relationship between the drain-source current I_{ds} and the

drain-source voltage V_{ds} in inverse mode ($V_{ds} < 0$) after adjusting the applicable model parameters for temperature.

Applicable Range of V_{gs} and V_{gd} Values	Corresponding I_{ds} Equation
$V_{gd} - V_{to} \leq 0$	$I_{ds} = 0$
$0 < V_{gd} - V_{to} \leq -V_{ds}$	$I_{ds} = -\beta \left(V_{gd} - V_{to} \right)^2 \left(1 - \lambda V_{ds} \right)$
$0 < -V_{ds} < V_{gs} - V_{to}$	$I_{ds} = \beta V_{ds} \left(2 \left(V_{gd} - V_{to} \right) + V_{ds} \right) \left(1 - \lambda V_{ds} \right)$

Junction Charge Model

The block provides the following relationship between the gate-source charge Q_{gs} and the gate-source voltage V_{gs} after adjusting the applicable model parameters for temperature.

Applicable Range of V_{gs} Values	Corresponding Q_{gs} Equation
$V_{gs} < FC * VJ$	$Q_{gs} = \dfrac{CGS * VJ * \left(1 - \left(1 - \dfrac{V_{gs}}{VJ} \right)^{1-MG} \right)}{1 - MG}$
$V_{gs} \geq FC * VJ$	$Q_{gs} = CGS * \left(F1 + \dfrac{F3 * \left(V_{gs} - FC * VJ \right) + \dfrac{MG * \left(V_{gs}^2 - \left(FC * VJ \right)^2 \right)}{2 * VJ}}{F2} \right)$

Where:

- *FC* is the **Capacitance coefficient FC** parameter value.
- *VJ* is the **Junction potential VJ** parameter value.
- *CGS* is the **Zero-bias GS capacitance, CGS** parameter value.
- *MG* is the **Grading coefficient, M** parameter value.

1-705

- $$F1 = \frac{VJ * \left(1 - \left(1 - FC\right)^{1-MG}\right)}{1 - MG}$$

- $$F2 = \left(1 - FC\right)^{1+MG}$$

- $$F3 = 1 - FC * \left(1 + MG\right)$$

The block provides the following relationship between the gate-drain charge Q_{gd} and the gate-drain voltage V_{gd} after adjusting the applicable model parameters for temperature.

Applicable Range of V_{gd} Values	Corresponding Q_{gd} Equation
$V_{gd} < FC * VJ$	$$Q_{gd} = \frac{CGD * VJ * \left(1 - \left(1 - \frac{V_{gd}}{VJ}\right)^{1-MG}\right)}{1 - MG}$$
$V_{gd} \geq FC * VJ$	$$Q_{gd} = CGD * \left(F1 + \frac{F3 * \left(V_{gd} - FC * VJ\right) + \frac{MG * \left(V_{gd}^2 - \left(FC * VJ\right)^2\right)}{2 * VJ}}{F2}\right)$$

Where:

- CGD is the **Zero-bias GD capacitance, CGD** parameter value.

Temperature Dependence

Several transistor parameters depend on temperature. There are two ways to specify the transistor temperature:

- When you select Device temperature for the **Model temperature dependence using** parameter, the transistor temperature is

$$T = T_C + T_O$$

where:

- T_C is the **Circuit temperature** parameter value from the SPICE Environment Parameters block. If this block doesn't exist in the circuit, T_C is the default value of this parameter.
- T_O is the **Offset local circuit temperature, TOFFSET** parameter value.
- When you select `Fixed temperature` for the **Model temperature dependence using** parameter, the transistor temperature is the **Fixed circuit temperature, TFIXED** parameter value.

The block provides the following relationship between the saturation current *IS* and the transistor temperature *T*:

$$IS(T) = IS * \left(T / T_{meas} \right)^{\frac{XTI}{ND}} * e^{\left(\frac{T}{T_{meas}} - 1 \right) * \frac{EG}{V_t}}$$

where:

- *IS* is the **Saturation current, IS** parameter value.
- T_{meas} is the **Parameter extraction temperature, TMEAS** parameter value.
- *XTI* is the **Saturation current temperature exponent, XTI** parameter value.
- *EG* is the **Energy gap, EG** parameter value.
-
$$V_t = ND * k * T / q$$
- *ND* is the **Emission coefficient, N** parameter value.

The block provides the following relationship between the junction potential *VJ* and the transistor temperature *T*:

$$VJ(T) = VJ * \left(\frac{T}{T_{meas}} \right) - \frac{3 * k * T}{q} * \log \left(\frac{T}{T_{meas}} \right) - \left(\frac{T}{T_{meas}} \right) * EG_{T_{meas}} + EG_T$$

where:

- *VJ* is the **Junction potential VJ** parameter value.

-
$$EG_{T_{meas}} = 1.16eV - \left(7.02e\text{-}4 * T_{meas}^{\,2}\right) / \left(T_{meas} + 1108\right)$$

-
$$EG_T = 1.16eV - \left(7.02e\text{-}4 * T^2\right) / \left(T + 1108\right)$$

The block provides the following relationship between the gate-source junction capacitance *CGS* and the transistor temperature *T*:

$$CGS(T) = CGS * \left[1 + MG * \left(400e - 6 * \left(T - T_{meas}\right) - \frac{VJ(T) - VJ}{VJ}\right)\right]$$

where:

- *CGS* is the **Zero-bias GS capacitance, CGS** parameter value.

The block uses the *CGS(T)* equation to calculate the gate-drain junction capacitance by substituting *CGD* (the **Zero-bias GD capacitance, CGD** parameter value) for *CGS*.

The block provides the following relationship between the forward and reverse beta and the transistor temperature *T*:

$$\beta(T) = \beta * \left(\frac{T}{T_{meas}}\right)$$

where *β* is the **Transconductance, BETA** parameter value.

Basic Assumptions and Limitations

The model is based on the following assumptions:

- The NJFET block does not support noise analysis.
- The NJFET block applies initial conditions across junction capacitors and not across the block ports.

Parameters

Main Tab

Device area, AREA

The transistor area. This value multiplies the **Transconductance, BETA**, **Zero-bias GS capacitance, CGS**, **Zero-bias GD capacitance, CGD**, and **Saturation current, IS** parameter values. It divides the **Source resistance, RS** and **Drain resistance, RD** parameter values. The default value is 1 m^2. The value must be greater than 0.

Number of parallel devices, SCALE

The number of parallel transistors the block represents. This value multiplies the output current and device charges. The default value is 1. The value must be greater than 0.

Threshold voltage, VTO

The gate-source voltage above which the transistor produces a nonzero drain current. The default value is -2 V.

Transconductance, BETA

The derivative of drain current with respect to gate voltage. The default value is $1e-04$ A/m^2/V^2. The value must be greater than or equal to 0.

Channel modulation, LAMBDA

The channel-length modulation. The default value is 0 1/V.

Saturation current, IS

The magnitude of the current that the ideal diode equation approaches asymptotically for very large reverse bias levels. The default value is $1e-14$ A/m^2. The value must be greater than or equal to 0.

Emission coefficient, N

The transistor emission coefficient or ideality factor. The default value is 1. The value must be greater than 0.

Source resistance, RS

The transistor source resistance. The default value is $1e-4$ m^2*Ω. The value must be greater than or equal to 0.

Drain resistance, RD

The transistor drain resistance. The default value is 0.01 m^2*Ω. The value must be greater than or equal to 0.

Junction Capacitance Tab

Model junction capacitance

Select one of the following options for modeling the junction capacitance:

- No — Do not include junction capacitance in the model. This is the default option.
- Yes — Specify zero-bias junction capacitance, junction potential, grading coefficient, forward-bias depletion capacitance coefficient, and transit time.

Zero-bias GS capacitance, CGS

The value of the capacitance placed between the gate and the source. This parameter is only visible when you select Yes for the **Model junction capacitance** parameter. The default value is 0 F/m^2. The value must be greater than or equal to 0.

Zero-bias GD capacitance, CGD

The value of the capacitance placed between the gate and the drain. This parameter is only visible when you select Yes for the **Model junction capacitance** parameter. The default value is 0 F/m^2. The value must be greater than or equal to 0.

Junction potential VJ

The junction potential. This parameter is only visible when you select Yes for the **Model junction capacitance** parameter. The default value is 1 V. The value must be greater than 0.01 V.

Grading coefficient, M

The transistor grading coefficient. The default value is 0.5. The value must be greater than 0 and less than 0.9.

Capacitance coefficient FC

The fitting coefficient that quantifies the decrease of the depletion capacitance with applied voltage. This parameter is only visible when you select Yes for the **Model junction capacitance** parameter. The default value is 0.5. The value must be greater than or equal to 0 and less than or equal to 0.95.

Specify initial condition

Select one of the following options for specifying an initial condition:

- No — Do not specify an initial condition for the model. This is the default option.
- Yes — Specify the initial diode voltage.

> **Note** The NJFET block applies the initial diode voltage across the junction capacitors and not across the ports.

Initial condition voltage ICVDS

Drain-source voltage at the start of the simulation. This parameter is only visible when you select `Yes` for the **Model junction capacitance** and `Yes` for the **Specify initial condition** parameter. The default value is `0` V.

Initial condition voltage ICVGS

Gate-source voltage at the start of the simulation. This parameter is only visible when you select `Yes` for the **Model junction capacitance** and `Yes` for the **Specify initial condition** parameter. The default value is `0` V.

Temperature Tab

Model temperature dependence using

Select one of the following options for modeling the diode temperature dependence:

- `Device temperature` — Use the device temperature, which is the **Circuit temperature** value plus the **Offset local circuit temperature, TOFFSET** value. The **Circuit temperature** value comes from the SPICE Environment Parameters block, if one exists in the circuit. Otherwise, it comes from the default value for this block.

- `Fixed temperature` — Use a temperature that is independent of the circuit temperature to model temperature dependence.

Saturation current temperature exponent, XTI

The order of the exponential increase in the saturation current as temperature increases. The default value is `0`. The value must be greater than or equal to `0`.

Activation energy, EG

The energy gap that affects the increase in the saturation current as temperature increases. The default value is `1.11` eV. The value must be greater than `0.1` eV.

Offset local circuit temperature, TOFFSET

The amount by which the transistor temperature differs from the circuit temperature. This parameter is only visible when you select `Device temperature` for the **Model temperature dependence using** parameter. The default value is `0` K.

Fixed circuit temperature, TFIXED

The temperature at which to simulate the transistor. This parameter is only visible when you select `Fixed temperature` for the **Model temperature dependence using** parameter. The default value is `300.15` K. The value must be greater than `0`.

Parameter extraction temperature, TMEAS

The temperature at which the transistor parameters were measured. The default value is `300.15` K. The value must be greater than `0`.

Ports

The block has the following ports:

`G`

Electrical conserving port associated with the transistor gate terminal.

`D`

Electrical conserving port associated with the transistor drain terminal.

`S`

Electrical conserving port associated with the transistor source terminal.

References

[1] G. Massobrio and P. Antognetti. *Semiconductor Device Modeling with SPICE*. 2nd Edition, McGraw-Hill, 1993. Chapter 3.

See Also

N-Channel JFET, SPICE PJFET

SPICE NMOS

Model SPICE-compatible N-Channel MOSFET

Library

SPICE-Compatible Components/Semiconductor Devices

Description

The NMOS block represents a SPICE-compatible N-channel MOSFET.

The NMOS block model includes the following components:

Resistance Calculations

The following table shows how the NMOS block calculates the transistor drain resistance. The abbreviations in the table represent the values of the following block parameters:

- **Drain resistance, RD**
- **Sheet resistance, RSH**
- **Number of drain squares, NRD**

Drain resistance, RD Parameter	Sheet resistance, RSH Parameter	Drain Resistance
NaN	NaN	0
RD	NaN or *RSH*	*RD*
NaN	*RSH*	*RSH*NRD*

The following table shows how the NMOS block calculates the transistor source resistance. The abbreviations in the table represent the values of the following block parameters:

- **Source resistance, RS**
- **Sheet resistance, RSH**
- **Number of source squares, NRS**

Source resistance, RS Parameter	Sheet resistance, RSH Parameter	Source Resistance
NaN	NaN	0
RS	NaN or *RSH*	*RS*
NaN	*RSH*	*RSH*NRS*

Bulk-Source Diode Model

The block provides the following relationship between the bulk-source current I_{bs} and the bulk-source voltage V_{bs} after adjusting the applicable model parameters for temperature.

Applicable Range of V_{bs} Values	Corresponding I_{gs} Equation
$V_{bs} > 80 * V_{tn}$	$I_{bs} = IS_{bs} * \left(\left(\dfrac{V_{bs}}{V_{tn}} - 79 \right) e^{80} - 1 \right) + V_{bs} * G\min$
$80 V_{tn} \geq V_{bs}$	$I_{bs} = IS_{bs} * \left(e^{V_{bs}/V_{tn}} - 1 \right) + V_{bs} * G\min$

Where:

- IS_{bs} is

  - The product of the **Bulk jct sat current density, JS** parameter value and the **Area of source, AS** parameter value if both these parameter values and the **Area of drain, AD** parameter value are nonzero.

  - The **Bulk saturation current, IS** parameter value, otherwise.

- $V_{tn} = NkT/q$

- q is the elementary charge on an electron, 1.6021918e-19 C.

- N is the **Emission coefficient, N** parameter value.

- k is the Boltzmann constant.

- T is the diode temperature:

  - If you select `Device temperature` for the **Model temperature dependence using** parameter, T is the sum of the **Circuit temperature** value plus the **Offset local circuit temperature, TOFFSET** parameter value. The **Circuit temperature** value comes from the SPICE Environment Parameters block, if one exists in the circuit. Otherwise, it comes from the default value for this block.

  - If you select `Fixed temperature` for the **Model temperature dependence using** parameter, T is the **Fixed circuit temperature, TFIXED** parameter value.

- *GMIN* is the diode minimum conductance. By default, *GMIN* matches the **Minimum conductance GMIN** parameter of the SPICE Environment Parameters block, whose default value is 1e-12. To change *GMIN*, add a SPICE Environment Parameters block to your model and set the **Minimum conductance GMIN** parameter to the desired value.

Bulk-Drain Diode Model

The block provides the following relationship between the bulk-drain current I_{bd} and the bulk-drain voltage V_{bd} after adjusting the applicable model parameters for temperature.

Applicable Range of V_{bs} Values	Corresponding I_{gs} Equation
$V_{bd} > 80 * V_{tn}$	$I_{bd} = IS_{bd} * \left(\left(\frac{V_{bd}}{V_{tn}} - 79 \right) e^{80} - 1 \right) + V_{bd} * G\min$

Applicable Range of V_{bs} Values	Corresponding I_{gs} Equation
$80V_{tn} \geq V_{bd}$	$I_{bd} = IS_{bd} * \left(e^{V_{bd}/V_{tn}} - 1 \right) + V_{bd} * G\min$

Where:

- IS_{bd} is

  - The product of the **Bulk jct sat current density, JS** parameter value and the **Area of drain, AD** parameter value if both these parameter values and the **Area of source, AS** parameter value are nonzero.
  - The **Bulk saturation current, IS** parameter value, otherwise.

Level 1 Drain Current Model

The block provides the following relationship between the drain current I_d and the drain-source voltage V_{ds} in normal mode ($V_{ds} \geq 0$) after adjusting the applicable model parameters for temperature.

Normal Mode

Applicable Range of V_{gs} and V_{ds} Values	Corresponding I_d Equation
$V_{gs} - V_{on} \leq 0$	$I_d = 0$
$0 < V_{gs} - V_{on} \leq V_{ds}$	$I_d = BETA * \left(V_{gs} - V_{on} \right)^2 \dfrac{\left(1 + LAMBDA * V_{ds} \right)}{2}$
$0 < V_{ds} < V_{gs} - V_{on}$	$I_d = BETA * $ $V_{ds} \left(\left(V_{gs} - V_{on} \right) - \dfrac{V_{ds}}{2} \right) \left(1 + LAMBDA * V_{ds} \right)$

Where:

- V_{on} is:

- $$MTYPE * VBI + GAMMA\sqrt{PHI - V_{bs}} \text{ if } V_{bs} \le 0 \, .$$

-

$$MTYPE * VBI + GAMMA\left(\sqrt{PHI} - \frac{V_{bs}}{2\sqrt{PHI}} \right) \text{ if } 0 < V_{bs} \le 2 * PHI \, .$$
  - *MTYPE*VBI* if $V_{bs} > 2*PHI$.

- *MTYPE* is 1.

- *BETA* is *KP*WIDTH/(LENGTH-2*LD)*

- *KP* is:

  - The **Transconductance, KP** parameter value, if this parameter has a numerical value.

  -
    $U0 * 3.9 * \varepsilon_0 / TOX$, if **Transconductance, KP** is NaN and you specify values for both the **Oxide thickness, TOX** and **Substrate doping, NSUB** parameters.

- *WIDTH* is the **Width of channel, WIDTH** parameter value.

- *LENGTH* is the **Length of channel, LENGTH** parameter value.

- *LD* is the **Lateral diffusion, LD** parameter value.

- *VBI* is an built-in voltage value the block uses in calculations. The value is a function of temperature. For a detailed definition, see "Temperature Dependence" on page 1-728.

- *PHI* is:

  - The **Surface potential, PHI** parameter value, if this parameter has a numerical value.

  -
    $2 * kT_{meas} / q * \log(NSUB / n_i)$, if **Surface potential, PHI** is NaN and you specify values for both the **Oxide thickness, TOX** and **Substrate doping, NSUB** parameters.

- *LAMBDA* is the **Channel modulation, LAMBDA** parameter value.

- *GAMMA* is:

  - The **Bulk threshold, GAMMA** parameter value, if this parameter has a numerical value.

- $TOX * \sqrt{2 * 11.7 * \varepsilon_0 * q * NSUB} / (3.9 * \varepsilon_0)$, if **Bulk threshold, GAMMA** is NaN and you specify values for both the **Oxide thickness, TOX** and **Substrate doping, NSUB** parameters.

- ε_0 is the permittivity of free space, 8.854214871e-12 F/m.

- n_i is the carrier concentration of intrinsic silicon, 1.45e10 cm^{-3}.

The block provides the following relationship between the drain current I_d and the drain-source voltage V_{ds} in inverse mode ($V_{ds} < 0$) after adjusting the applicable model parameters for temperature.

Inverse Mode

Applicable Range of V_{gd} and V_{ds} Values	Corresponding I_d Equation
$V_{gd} - V_{on} \leq 0$	$I_d = 0$
$0 < V_{gd} - V_{on} \leq -V_{ds}$	$I_d = -BETA\left(V_{gd} - V_{on}\right)^2 \left(1 - LAMBDA * V_{ds}\right)/2$
$0 < V_{ds} < V_{gd} - V_{on}$	$I_d = BETA * \\ V_{ds}\left(\left(V_{gd} - V_{on}\right) + V_{ds}/2\right)\left(1 - LAMBDA * V_{ds}\right)$

Where:

- V_{on} is:

  - $MTYPE * VBI + GAMMA\sqrt{PHI - V_{bd}}$ if $V_{bd} \leq 0$.

  - $MTYPE * VBI + GAMMA\left(\sqrt{PHI} - \dfrac{V_{bd}}{2\sqrt{PHI}}\right)$ if $0 < V_{bd} \leq 2 * PHI$.
  - $MTYPE*VBI$ if $V_{bd} > 2*PHI$.

Level 3 Drain Current Model

The block provides the following model for drain current I_{ds} in normal mode ($V_{ds} \geq 0$) after adjusting the applicable model parameters for temperature.

$$I_{DS} = I_{DS0} * Scale_{VMAX} * Scale_{LChan} * Scale_{INV}$$

Where:

- I_{DS0} is the Basic Drain Current Model on page 1-719.
- $Scale_{VMAX}$ is the Velocity Saturation Scaling on page 1-722.
- $Scale_{LChan}$ is the Channel Length Modulation Scaling on page 1-722.
- $Scale_{INV}$ is the Weak Inversion Scaling on page 1-724.

The blocks uses the same model for drain current in inverse mode ($V_{ds} < 0$), with the following substitutions:

- V_{bs} – V_{ds} for V_{bs}
- V_{gs} – V_{ds} for V_{ds}
- $-V_{ds}$ for V_{ds}

The block provides the following relationship between the drain current I_{ds} and the drain-source voltage V_{ds}:

$$I_{DS0} = BETA * F_{gate} * \left(V_{GSX} - V_{TH} - \frac{1 + F_B}{2} * V_{DSX} \right) * V_{DSX}$$

- The block calculates BETA as described in "Level 1 Drain Current Model" on page 1-716.
- The block calculates F_{GATE} using the following equation:

$$F_{gate} = \frac{1}{1 + THETA * \left(V_{gsx} - V_{TH} \right)}$$

- *THETA* is the **Vgs dependence on mobility, THETA** parameter value.

- $$V_{gsx} = \max(V_{GS}, V_{on})$$

- If you specify a nonzero value for the **Fast surface state density, NFS** parameter, the block calculates V_{on} using the following equation:

$$V_{on} = V_{TH} + x_n V_T$$

  Otherwise, $V_{on} = V_{TH}$.

- The block calculates x_n using the following equation:

$$x_n = 1 + \frac{q * NFS}{COX} + \frac{\left(GAMMA * F_s * \sqrt{V_{bulk}} + \dfrac{F_n * V_{bulk}}{WIDTH} \right)}{2 * V_{bulk}}$$

- The block calculates V_{bulk} as follows:

  - If $V_{BS} \leq 0$, $V_{bulk} = PHI - V_{BS}$.

  - Otherwise, the block calculates V_{bulk} using the following equation:

$$V_{bulk} = \frac{PHI}{\left(1 + \dfrac{V_{BS}}{2 * PHI} \right)^2}$$

- $V_T = kT/q$

- The block calculates V_{TH} using the equation following equation:

$$V_{TH} = V_{BI} - \frac{8.15e^{-22} * ETA}{COX * \left(LENGTH - 2 * LD \right)^3} * V_{DS} + GAMMA * F_s * \sqrt{V_{bulk}} + F_n * V_{bulk}$$

- For information about how the block calculates V_{BI}, see "Temperature Dependence" on page 1-728.

- *ETA* is the **Vds dependence threshold volt, ETA** parameter value.

- $COX = \varepsilon_{ox}/TOX$, where ε_{ox} is the permittivity of the oxide and *TOX* is the **Oxide thickness, TOX** parameter value.

- If you specify a nonzero value for the **Junction depth, XJ** parameter and a value for the **Substrate doping, NSUB** parameter, the block calculates F_s using the following equations:

$$\alpha = \frac{2\varepsilon_{si}}{qNSUB}$$

$$XD = \sqrt{\alpha}$$

$$wc = .0631353 + .8013292 * \frac{XD * \sqrt{V_{bulk}}}{XJ}$$

$$-.01110777 * \left(\frac{XD * \sqrt{V_{bulk}}}{XJ} \right)^2 + \frac{LD}{XJ}$$

$$F_s = 1 - \left(wc * \sqrt{1 - \left(\frac{XD * \sqrt{V_{bulk}}}{XJ + XD * \sqrt{V_{bulk}}} \right)^2} - \frac{LD}{XJ} \right)$$

where ε_{si} is the permittivity of silicon.

Otherwise, $F_s = 1$.

- The block calculates F_B using the following equation:

$$F_B = \frac{GAMMA * F_s}{4 * \sqrt{V_{bulk}}} + F_n$$

- The block calculates F_n using the following equation:

$$F_n = \frac{DELTA * \pi * \varepsilon_{si}}{2 * COX * WIDTH}$$

- *DELTA* is the **Width effect on threshold, DELTA** parameter value.

- V_{DSX} is the lesser of V_{DS} and the saturation voltage, V_{dsat}.

1-721

- If you specify a positive value for the **Max carrier drift velocity, VMAX** parameter, the block calculates V_{dsat} using the following equation:

$$V_{dsat} = \frac{V_{gsx} - V_{TH}}{1 + F_B} + \frac{(LENGTH - 2*LD)*VMAX}{UO*F_{gate}}$$
$$- \sqrt{\left(\frac{V_{gsx} - V_{TH}}{1 + F_B}\right)^2 + \left(\frac{(LENGTH - 2*LD)*VMAX}{UO*F_{gate}}\right)^2}$$

Otherwise, the block calculates V_{dsat} using the following equation:

$$V_{dsat} = \frac{V_{gsx} - V_{TH}}{1 + F_B}$$

If you specify a positive value for the **Max carrier drift velocity, VMAX** parameter, the block calculates $Scale_{VMAX}$ using the following equation:

$$Scale_{VMAX} = \frac{1}{1 + \dfrac{UO*F_{gate}}{(LENGTH - 2*LD)*VMAX}*V_{DSX}}$$

Otherwise, $Scale_{VMAX}$ = 1.

The block scales the drain current to account for channel length modulation if the block meets all of the following criteria:

- $V_{DS} > V_{sat}$
- The **Max carrier drift velocity, VMAX** parameter value is zero or α is nonzero.

The block scales the drain current using the following equation:

$$Scale_{LChan} = \frac{1}{1 - \dfrac{\Delta l}{(LENGTH - 2*LD)}}$$

The block uses the following procedure to calculate Δl:

1 The block first calculates the intermediate value Δl_0.

- If you specify a positive value for the **Max carrier drift velocity, VMAX** parameter, the block computes the intermediate value g_{dsat} as the greater of 1e-12 and the result of the following equation:

$$I_{DS0} * \left(1 - \frac{1}{1 + Scale_{g_{dsat}} * V_{DSX}} \right) * Scale_{g_{dsat}}$$

where:

$$Scale_{g_{dsat}} = \frac{UO * F_{gate}}{\left(LENGTH - 2 * LD \right) * VMAX}$$

Then, the block uses the following equation to calculate the intermediate value Δl_0:

$$\Delta l_0 = \sqrt{\left(\frac{KA * I_{DS}}{2 * \left(LENGTH - 2 * LD \right) * g_{dsat}} \right)^2 + KA * \left(V_{DS} - V_{dsat} \right)} \\ - \frac{KA * I_{DS}}{2 * \left(LENGTH - 2 * LD \right) * g_{dsat}}$$

where KA is the product of the **Mobility modulation, KAPPA** parameter value and α.

- Otherwise, the block uses the following equation to calculate the intermediate value Δl_0:

$$\Delta l = \sqrt{KA * \left(V_{DS} - V_{dsat} \right)}$$

2 The block checks for punch through and calculates Δl.

- If Δl_0 is greater than ($LENGTH$-2*LD)/2, the block calculates Δl using the following equation:

$$\Delta l = \left(1 - \frac{(LENGTH - 2*LD)}{4*\Delta l_0}\right) * (LENGTH - 2*LD)$$

- Otherwise, $\Delta l = \Delta l_0$.

If V_{GS} is less than V_{on}, the block calculates $Scale_{INV}$ using the following equation:

$$Scale_{INV} = e^{\frac{V_{gs} - V_{on}}{x_n * V_T}}$$

Otherwise, $Scale_{INV} = 1$.

Junction Charge Model

The block models the following junction charges:

- Junction Overlap Charges
- Bulk Junction Charges

The block calculates the following junction overlap charges:

- $Q_{GS} = CGSO*WIDTH*V_{gs}$

  Where:

  - Q_{GS} is the gate-source overlap charge.
  - *CGSO* is the **G-S overlap capacitance, CGSO** parameter value.
  - *WIDTH* is the **Width of channel, WIDTH** parameter value.

- $Q_{GD} = CGDO*WIDTH*V_{gd}$

  Where:

  - Q_{GD} is the gate-drain overlap charge.
  - *CGDO* is the **G-D overlap capacitance, CGDO** parameter value.

- $Q_{GB} = CGBO*(LENGTH-2*LD)*V_{gb}$

  Where:

- Q_{GB} is the gate-bulk overlap charge.
- *CGBO* is the **G-B overlap capacitance, CGBO** parameter value.
- *LENGTH* is the **Length of channel, LENGTH** parameter value.
- *LD* is the **Lateral diffusion, LD** parameter value.

The block provides the following relationship between the bulk-drain bottom junction charge Q_{bottom} and the junction voltage V_{bd} after adjusting the applicable model parameters for temperature.

Applicable Range of V_{bd} Values	Corresponding Q_{bottom} Equation
$V_{bd} < FC * PB$	$$Q_{bottom} = \frac{CBD * PB * \left(1 - \left(1 - \frac{V_{bd}}{PB}\right)^{1-MJ}\right)}{1 - MJ} \quad \text{if } CBD > 0.$$ $$Q_{bottom} = \frac{CJ * AD * PB * \left(1 - \left(1 - \frac{V_{bd}}{PB}\right)^{1-MJ}\right)}{1 - MJ} \quad \text{otherwise.}$$

Applicable Range of V_{bd} Values	Corresponding Q_{bottom} Equation
$V_{bd} \geq FC * PB$	$Q_{bottom} = CBD *$ $$\left(F1 + \frac{F3*\left(V_{bd} - FC*PB\right) + \frac{MJ*\left(V_{bd}^2 - \left(FC*PB\right)^2\right)}{2*PB}}{F2} \right)$$ if $CBD >$ 0. $Q_{bottom} = CJ * AD *$ $$\left(F1 + \frac{F3*\left(V_{bd} - FC*PB\right) + \frac{MJ*\left(V_{bd}^2 - \left(FC*PB\right)^2\right)}{2*PB}}{F2} \right)$$ otherwise.

Where:

- PB is the **Bulk junction potential, PB** parameter value.
- FC is the **Capacitance coefficient FC** parameter value.
- CBD is the **Zero-bias BD capacitance, CBD** parameter value.
- CJ is the **Bottom junction cap per area, CJ** parameter value.
- AD is the **Area of drain, AD** parameter value.
- MJ is the **Bottom grading coefficient, MJ** parameter value.
- $$F1 = \frac{PB*\left(1 - \left(1 - FC\right)^{1-MJ}\right)}{1 - MJ}$$
- $$F2 = \left(1 - FC\right)^{1+MJ}$$

- $$F3 = 1 - FC * (1 + MJ)$$

The block uses the equations in the preceding table to calculate the bulk-source bottom junction charge, with the following substitutions:

- V_{bs} replaces V_{bd}.
- *AS* (the **Area of source, AS** parameter value) replaces *AD*.
- *CBS* (the **Zero-bias BS capacitance, CBS** parameter value) replaces *CBD*.

The block provides the following relationship between the bulk-drain sidewall junction charge $Q_{sidewall}$ and the junction voltage V_{bd} after adjusting the applicable model parameters for temperature.

Applicable Range of V_{bd} Values	Corresponding $Q_{sidewall}$ Equation
$V_{bd} < FC * PB$	$$Q_{sidewall} = \frac{CJSW * PD * PB * \left(1 - \left(1 - \frac{V_{bd}}{PB}\right)^{1-MGSW}\right)}{1 - MGSW}$$
$V_{bd} \geq FC * PB$	$$Q_{sidewall} = CJSW * PD * \left(F1 + \frac{F3 * (V_{bd} - FC * PB) + \frac{MGSW * (V_{bd}^2 - (FC * PB)^2)}{2 * PB}}{F2} \right)$$

Where:

- *CJSW* is the **Side jct cap/area of jct perimeter, CJSW** parameter value.
- *PD* is the **Perimeter of drain, AD** parameter value.
- *MGSW* is the **Side grading coefficient, MJSW** parameter value.

-

$$F1 = \frac{PB*\left(1-\left(1-FC\right)^{1-MJSW}\right)}{1-MJSW}$$

-

$$F2 = \left(1-FC\right)^{1+MJSW}$$

-

$$F3 = 1-FC*\left(1+MJSW\right)$$

The block uses the equations in the preceding table to calculate the bulk-source sidewall junction charge and the sidewall junction voltage, with the following substitutions:

- V_{bs} replaces V_{bd}.

- *PS* (the **Perimeter of source, PS** parameter value) replaces *PD*.

Temperature Dependence

Several transistor parameters depend on temperature. There are two ways to specify the transistor temperature:

- When you select `Device temperature` for the **Model temperature dependence using** parameter, the transistor temperature is

  $$T = T_C + T_O$$

  where:

  - T_C is the **Circuit temperature** parameter value from the SPICE Environment Parameters block. If this block doesn't exist in the circuit, T_C is the default value of this parameter.
  - T_O is the **Offset local circuit temperature, TOFFSET** parameter value.

- When you select `Fixed temperature` for the **Model temperature dependence using** parameter, the transistor temperature is the **Fixed circuit temperature, TFIXED** parameter value.

The block provides the following relationship between the transconductance *KP* and the transistor temperature *T*:

$$KP(T) = \frac{KP}{\left(T \middle/ T_{meas} \right)^{3/2}}$$

where:

- *KP* is the **Transconductance, KP** parameter value.
- T_{meas} is the **Parameter extraction temperature, TMEAS** parameter value.

The block provides the following relationship between the surface potential *PHI* and the transistor temperature *T*:

$$PHI(T) = \frac{T}{T_{meas}} \left(PHI + \frac{kT_{meas}}{q} \left(\log\left(\frac{T_{meas}}{300.15} \right)^3 + \frac{q}{k}\left(\frac{1.115}{300.15} - \frac{EG_{T_{meas}}}{T_{meas}} \right) \right) \right)$$
$$- \frac{kT}{q} \left(\log\left(\frac{T}{300.15} \right)^3 + \frac{q}{k}\left(\frac{1.115}{300.15} - \frac{EG_T}{T} \right) \right)$$

where:

-
$$EG_{T_{meas}} = 1.16eV - \left(7.02e - 4 * T_{meas}^{\;2} \right) / \left(T_{meas} + 1108 \right)$$
-
$$EG_T = 1.16eV - \left(7.02e - 4 * T^2 \right) / \left(T + 1108 \right)$$

The block provides the following relationship between the built-in voltage *VBI* and the transistor temperature *T*:

$$VBI(T) = VTO + MTYPE * \left(\frac{PHI(T) - PHI}{2} - GAMMA\sqrt{PHI} \right)$$
$$+ \frac{EG_{T_{meas}} - EG_T}{2}$$

where:

- *VTO* is:

- The **Threshold voltage, VTO** parameter value, if this parameter has a numerical value.

- $$\Phi - 3.25 + EG_{T_{meas}} / 2 + MTYPE * PHI / 2 - NSS * q * TOX / \left(3.9 * \varepsilon_0\right)$$

  $+ MTYPE * (GAMMA * \sqrt{PHI} + PHI)$, if **Threshold voltage, VTO** is NaN and you specify values for both the **Oxide thickness, TOX** and **Substrate doping, NSUB** parameters.

- Φ is:

  - 3.2, if TPG (the **Gate type?, TPG** parameter value) is 0.

  - $$3.25 + EG_{T_{meas}} / 2 - MTYPE * TPG * EG_{T_{meas}} / 2$$, otherwise.

- $GAMMA$ is:

  - The **Bulk threshold, GAMMA** parameter value, if this parameter has a numerical value.

  - $$TOX * \sqrt{2 * 11.7 * \varepsilon_0 * q * NSUB} / \left(3.9 * \varepsilon_0\right)$$, if **Bulk threshold, GAMMA** is NaN and you specify values for both the **Oxide thickness, TOX** and **Substrate doping, NSUB** parameters.

The block provides the following relationship between the bulk saturation current *IS* and the transistor temperature *T*:

$$IS(T) = IS * e^{\frac{-qEG_T}{ND*kT} + \frac{qEG_{T_{meas}}}{ND*kT_{meas}}}$$

where:

- *ND* is the **Emission coefficient, N** parameter value.
- *IS* is the **Bulk saturation current, IS** parameter value.

The block provides the following relationship between the bulk junction saturation current *JS* and the transistor temperature *T*:

$$JS(T) = JS * e^{\frac{-qEG_T}{ND*kT} + \frac{qEG_{T_{meas}}}{ND*kT_{meas}}}$$

where:

- *JS* is the **Bulk jct sat current density, JS** parameter value.

The block provides the following relationship between the bulk junction potential *PB* and the transistor temperature *T*:

$$PB(T) = \frac{PB + \frac{kT_{meas}}{q}\left(\log\left(\frac{T_{meas}}{300.15}\right)^3 + \frac{q}{k}\left(\frac{1.115}{300.15} - \frac{EG_{T_{meas}}}{T}\right)\right)}{T_{meas}/T}$$
$$- \frac{kT}{q}\left(\log\left(\frac{T}{300.15}\right)^3 + \frac{q}{k}\left(\frac{1.115}{300.15} - \frac{EG_T}{T}\right)\right)$$

where:

- *PB* is the **Bulk junction potential, PB** parameter value.

The block provides the following relationship between the bulk-drain junction capacitance *CBD* and the transistor temperature *T*:

$$CBD(T) = CBD\frac{pbo + MJ * \left(4 * 10^4 * (T - 300.15) * pbo - (PB(T) - pbo)\right)}{pbo + MJ * \left(4 * 10^4 * (T_{meas} - 300.15) * pbo - (PB - pbo)\right)}$$

where:

- *CBD* is the **Zero-bias BD capacitance, CBD** parameter value.
- *MJ* is the **Bottom grading coefficient, MJ** parameter value.
-

$$pbo = \frac{PB + \frac{kT_{meas}}{q}\left(\log\left(\frac{T_{meas}}{300.15}\right)^3 + \frac{q}{k}\left(\frac{1.115}{300.15} - \frac{EG_{T_{meas}}}{T}\right)\right)}{T_{meas}/300.15}$$

The block uses the *CBD(T)* equation to calculate:

- The bulk-source junction capacitance by substituting *CBS* (the **Zero-bias BS capacitance, CBS** parameter value) for *CBD*.

- The bottom junction capacitance by substituting *CJ* (the **Bottom junction cap per area, CJ** parameter value) for *CBD*.

The block provides the following relationship between the sidewall junction capacitance *CJSW* and the transistor temperature *T*:

$$CJSW(T) = CJSW \frac{pbo + MJSW * \left(4 * 10^4 * (T - 300.15) * pbo - (PB(T) - pbo)\right)}{pbo + MJSW * \left(4 * 10^4 * (T_{meas} - 300.15) * pbo - (PB - pbo)\right)}$$

where:

- *MJSW* is the **Side grading coefficient, MJSW** parameter value.

Basic Assumptions and Limitations

The model is based on the following assumptions:

- The NMOS block does not support noise analysis.
- The NMOS block applies initial conditions across junction capacitors and not across the block ports.

Parameters

Model Selection Tab

MOS model

Select one of the following MOSFET model options:

- `Level 1 MOS` — Use the "Level 1 Drain Current Model" on page 1-716. This is the default option.
- `Level 3 MOS` — Use the "Level 3 Drain Current Model" on page 1-719.

Dimensions Tab

Device area factor, AREA

The transistor area. This value multiplies the following parameter values:

- **Transconductance, KP**
- **Bulk saturation current, IS**
- **Bulk jct sat current density, JS**
- **Zero-bias BD capacitance, CBD**
- **Zero-bias BS capacitance, CBS**
- **G-S overlap capacitance, CGSO**
- **G-D overlap capacitance, CGDO**
- **G-B overlap capacitance, CGBO**
- **Bottom junction cap per area CJ**
- **Side jct cap/area of jct perimeter CJSW**

It divides the following parameter values:

- **Drain resistance, RD**
- **Source resistance, RS**
- **Sheet resistance, RSH**

The default value is `1`. The value must be greater than `0`.

Number of parallel devices, SCALE

The number of parallel MOS instances for this device. This parameter multiplies the output current and device charge. The default value is `1`. The value must be greater than `0`.

Length of channel, LENGTH

Length of the channel between the source and drain. The default value is `1e-04` m.

Width of channel, WIDTH

Width of the channel between the source and drain. The default value is `1e-04` m.

Area of drain, AD

Area of the transistor drain diffusion. The default value is `0` m^2. The value must be greater than or equal to `0`.

Area of source, AS

Area of the transistor source diffusion. The default value is 0 m^2. The value must be greater than or equal to 0.

Perimeter of drain, PD

Perimeter of the transistor drain diffusion. The default value is 0 m.

Perimeter of source, PS

Perimeter of the transistor source diffusion. The default value is 0 m.

Resistors Tab

Drain resistance, RD

The transistor drain ohmic resistance. The default value is 0.01 Ω. If you set this parameter to NaN Ω, this value means the parameter is unspecified, so the block calculates the drain resistance as described in "Resistance Calculations" on page 1-713. The value must be equal to *0* or greater than or equal to *Rmin*. *Rmin* is a built-in model constant whose value is 1e-12.

Source resistance, RS

The transistor source ohmic resistance. The default value is 1e-4 Ω. If you set this parameter to NaN Ω, this value means the parameter is unspecified, so the block calculates the drain resistance as described in "Resistance Calculations" on page 1-713. The value must be equal to 0 or greater than or equal to *Rmin*. *Rmin* is a built-in model constant whose value is 1e-12.

Sheet resistance, RSH

Resistance per square of the transistor source and drain. The default value is NaN Ω. This value means the parameter is unspecified. The block only uses this parameter value if you do not specify one or both of the **Drain resistance, RD** and **Source resistance, RS** parameter values, as described in "Resistance Calculations" on page 1-713. The value must be greater than or equal to 0.

Number of drain squares, NRD

Number of squares of resistance that make up the transistor drain diffusion. The default value is 1 . The value must be greater than or equal to 0. The block only uses this parameter value if you do not specify one or both of the **Drain resistance, RD** and **Source resistance, RS** parameter values, as described in "Resistance Calculations" on page 1-713.

Number of source squares, NRS

Number of squares of resistance that make up the transistor source diffusion. The default value is `1` . The value must be greater than or equal to `0`. The block only uses this parameter value if you do not specify one or both of the **Drain resistance, RD** and **Source resistance, RS** parameter values, as described in "Resistance Calculations" on page 1-713.

DC Currents Tab

Threshold voltage, VTO

The gate-source voltage above which the transistor produces a nonzero drain current. The default value is `0` V. If you assign this parameter a value of `NaN`, the block calculates the value from the specified values of the **Oxide thickness, TOX** and **Substrate doping, NSUB** parameters. For more information about this calculation, see "Temperature Dependence" on page 1-728.

Transconductance, KP

The derivative of drain current with respect to gate voltage. The default value is `2e-05` A/V^2. The value must be greater than or equal to `0`. If you assign this parameter a value of `NaN`, the block calculates the value from the specified values of the **Oxide thickness, TOX** and **Substrate doping, NSUB** parameters. For more information about this calculation, see "Level 1 Drain Current Model" on page 1-716 or "Level 3 Drain Current Model" on page 1-719 as appropriate for the selected value of the **MOS model** parameter.

Bulk threshold, GAMMA

Body effect parameter, which relates the threshold voltage, VTH, to the body bias, VBS, as described in "Level 1 Drain Current Model" on page 1-716 and "Level 3 Drain

Current Model" on page 1-719. The default value is $0 \ \sqrt{V}$. The value must be greater than or equal to `0`. If you assign this parameter a value of `NaN`, the block calculates the value from the specified values of the **Oxide thickness, TOX** and **Substrate doping, NSUB** parameters. For more information about this calculation, see "Level 1 Drain Current Model" on page 1-716 or "Level 3 Drain Current Model" on page 1-719 as appropriate for the selected value of the **MOS model** parameter.

Surface potential, PHI

Twice the voltage at which the surface electron concentration becomes equal to the intrinsic concentration and the device transitions between depletion and inversion conditions. The default value is `0.6` V. The value must be greater than or equal to `0`. If you assign this parameter a value of `NaN`, the block calculates the value from the

specified values of the **Oxide thickness, TOX** and **Substrate doping, NSUB** parameters. For more information about this calculation, see "Level 1 Drain Current Model" on page 1-716 or "Level 3 Drain Current Model" on page 1-719 as appropriate for the selected value of the **MOS model** parameter.

Channel modulation, LAMBDA

The channel-length modulation. This parameter is only visible when you select `Level 1 MOS` for the **MOS model** parameter. The default value is `0` 1/V.

Bulk saturation current, IS

The magnitude of the current that the junction approaches asymptotically for very large reverse bias levels. The default value is `1e-14` A. The value must be greater than or equal to `0`.

Bulk jct sat current density, JS

The magnitude of the current per unit area that the junction approaches asymptotically for very large reverse bias levels. The default value is `0` A/m^2. The value must be greater than or equal to `0`.

Emission coefficient, N

The transistor emission coefficient or ideality factor. The default value is `1`. The value must be greater than `0`.

Width effect on threshold, DELTA

The factor that controls the effect of transistor width on threshold voltage. This parameter is only visible when you select `Level 3 MOS` for the **MOS model** parameter. The default value is `0`.

Max carrier drift velocity, VMAX

The maximum drift velocity of the carriers. This parameter is only visible when you select `Level 3 MOS` for the **MOS model** parameter. The default value is `0` m/s.

Fast surface state density, NFS

The fast surface state density adjusts the drain current for the mobility reduction caused by the gate voltage. This parameter is only visible when you select `Level 3 MOS` for the **MOS model** parameter. The default value is `0` 1/cm^2.

Vds dependence threshold volt, ETA

The coefficient that controls how the threshold voltage depends on the drain-source voltage in the drain current calculation. This parameter is only visible when you select `Level 3 MOS` for the **MOS model** parameter. The default value is `0`.

Vgs dependence on mobility, THETA

The coefficient that controls how the mobility affects the gate voltage in the drain current calculation. This parameter is only visible when you select `Level 3 MOS` for the **MOS model** parameter. The default value is `0` 1/V.

Mobility modulation, KAPPA

The coefficient that controls how the mobility affects the channel length in the drain current calculation. This parameter is only visible when you select `Level 3 MOS` for the **MOS model** parameter. The default value is `0.2`.

C-V Tab

Model junction capacitance

Select one of the following options for modeling the junction capacitance:

- `No` — Do not include junction capacitance in the model. This is the default option.
- `Yes` — Specify zero-bias junction capacitance, junction potential, grading coefficient, forward-bias depletion and capacitance coefficient.

G-S overlap capacitance, CGSO

Gate-source capacitance due to the diffusion that occurs when the device operates in depletion mode. This parameter is only visible when you select `Yes` for the **Model junction capacitance** parameter. The default value is `0` F/m. The value must be equal to `0` or greater than or equal to *Cmin*. *Cmin* is a built-in model constant whose value is `1e-18`.

G-D overlap capacitance, CGDO

Gate-drain capacitance due to the diffusion that occurs when the device operates in depletion mode. This parameter is only visible when you select `Yes` for the **Model junction capacitance** parameter. The default value is `0` F/m. The value must be equal to `0` or greater than or equal to *Cmin*. *Cmin* is a built-in model constant whose value is `1e-18`.

G-B overlap capacitance, CGBO

Gate-base capacitance due to the diffusion that occurs when the device operates in depletion mode. This parameter is only visible when you select `Yes` for the **Model junction capacitance** parameter. The default value is `0` F/m. The value must be equal to `0` or greater than or equal to *Cmin*. *Cmin* is a built-in model constant whose value is `1e-18`.

Zero-bias BD capacitance, CBD

The value of the capacitance placed between the base and the drain. This parameter is only visible when you select `Yes` for the **Model junction capacitance** parameter. The default value is `0` F. The value must be equal to `0` or greater than or equal to *Cmin*. *Cmin* is a built-in model constant whose value is `1e-18`.

Zero-bias BS capacitance, CBS

The value of the capacitance placed between the base and the source. This parameter is only visible when you select `Yes` for the **Model junction capacitance** parameter. The default value is `0` F. The value must be equal to `0` or greater than or equal to *Cmin*. *Cmin* is a built-in model constant whose value is `1e-18`.

Bottom junction cap per area CJ

Zero-bias bulk junction bottom capacitance per junction area. This parameter is only visible when you select `Yes` for the **Model junction capacitance** parameter. The default value is `0` F/m^2. The value must be equal to `0` or greater than or equal to *Cmin*. *Cmin* is a built-in model constant whose value is `1e-18`.

Bottom grading coefficient, MJ

The transistor bottom grading coefficient. This parameter is only visible when you select `Yes` for the **Model junction capacitance** parameter. The default value is `0.5`. The value must be equal to `0` or less than *MGmax*. *MGmax* is a built-in model constant whose value is `0.9`.

Side jct cap/area of jct perimeter CJSW

Zero-bias bulk junction sidewall capacitance per junction perimeter. This parameter is only visible when you select `Yes` for the **Model junction capacitance** parameter. The default value is `0` F/m. The value must be equal to `0` or greater than or equal to *Cmin*. *Cmin* is a built-in model constant whose value is `1e-18`.

Side grading coefficient, MJSW

The transistor sidewall grading coefficient. This parameter is only visible when you select `Yes` for the **Model junction capacitance** parameter. The default value is `0.5`. The value must be equal to `0` or less than *MGmax*. *MGmax* is a built-in model constant whose value is `0.9`.

Bulk junction potential, PB

The potential across the bulk junction. This parameter is only visible when you select `Yes` for the **Model junction capacitance** parameter. The default value is `0.8` V. The value must be equal to `0` or greater than or equal to *VJmin*. *VJmin* is a built-in model constant whose value is `0.01`.

Capacitance coefficient FC

The fitting coefficient that quantifies the decrease of the depletion capacitance with applied voltage. This parameter is only visible when you select `Yes` for the **Model junction capacitance** parameter. The default value is `0.5`. The value must be equal to `0` or less than or equal to *FCmax*. *FCmax* is a built-in model constant whose value is `0.95`.

Specify initial condition

Select one of the following options for specifying an initial condition:

- `No` — Do not specify an initial condition for the model. This is the default option.
- `Yes` — Specify the initial diode voltage.

Note The NMOS block applies the initial diode voltage across the junction capacitors and not across the ports.

Initial condition voltage ICVDS

Drain-source voltage at the start of the simulation. This parameter is only visible when you select `Yes` for the **Model junction capacitance** and `Yes` for the **Specify initial condition** parameter. The default value is `0` V.

Initial condition voltage ICVGS

Gate-source voltage at the start of the simulation. This parameter is only visible when you select `Yes` for the **Model junction capacitance** and `Yes` for the **Specify initial condition** parameter. The default value is `0` V.

Initial condition voltage ICVBS

Bulk-source voltage at the start of the simulation. This parameter is only visible when you select `Yes` for the **Model junction capacitance** and `Yes` for the **Specify initial condition** parameter. The default value is `0` V.

Process Tab

Oxide thickness, TOX

Thickness of the gate oxide. The default value is `NaN` m. The value must be greater than or equal to `0`.

Note When you select `Level 3 MOS` for the **MOS model** parameter, the block uses a value of `1e-7` rather than `NaN` by default.

Lateral diffusion, LD

Length of lateral diffusion. The default value is 0 m.

Substrate doping, NSUB

Substrate doping. The default value is NaN $1/cm^3$. The value must be greater than or equal to 1.45e10 (the carrier concentration of intrinsic silicon).

Surface state density, NSS

Substrate doping. The default value is 0 $1/cm^2$.

Surface mobility, U0

Zero-bias surface mobility coefficient. The default value is 600 $cm^2/V/s$.

Junction depth, XJ

Junction depth. This parameter is only visible when you select Level 3 MOS for the **MOS model** parameter. The default value is 0 m.

Gate type?,TPG

Select one of the following MOSFET gate materials (as compared to the substrate):

- Opposite of substrate — The gate material is the opposite of the substrate. This means that TPG = 1 in the device equations. This is the default option.
- Same as substrate — The gate material is the same as the substrate. This means that TPG = –1 in the device equations.
- Aluminum — The gate material is aluminum. This means that TPG = 0 in the device equations.

Temperature Tab

Model temperature dependence using

Select one of the following options for modeling the diode temperature dependence:

- Device temperature — Use the device temperature, which is the **Circuit temperature** value plus the **Offset local circuit temperature, TOFFSET** value. The **Circuit temperature** value comes from the SPICE Environment Parameters block, if one exists in the circuit. Otherwise, it comes from the default value for this block.
- Fixed temperature — Use a temperature that is independent of the circuit temperature to model temperature dependence.

Offset local circuit temperature, TOFFSET

The amount by which the transistor temperature differs from the circuit temperature. This parameter is only visible when you select `Device temperature` for the **Model temperature dependence using** parameter. The default value is `0` K.

Fixed circuit temperature, TFIXED

The temperature at which to simulate the transistor. This parameter is only visible when you select `Fixed temperature` for the **Model temperature dependence using** parameter. The default value is `300.15` K. The value must be greater than `0`.

Parameter extraction temperature, TMEAS

The temperature at which the transistor parameters were measured. The default value is `300.15` K. The value must be greater than `0`.

Ports

The block has the following ports:

G

Electrical conserving port associated with the transistor gate terminal.

D

Electrical conserving port associated with the transistor drain terminal.

S

Electrical conserving port associated with the transistor source terminal.

B

Electrical conserving port associated with the transistor bulk terminal.

References

[1] G. Massobrio and P. Antognetti. *Semiconductor Device Modeling with SPICE*. 2nd Edition, McGraw-Hill, 1993. Chapter 3.

See Also

SPICE PMOS

SPICE NPN

Model Gummel-Poon NPN Transistor

Library

SPICE-Compatible Components/Semiconductor Devices

Description

The NPN block represents a SPICE-compatible four-terminal Gummel-Poon NPN transistor. The substrate port is connected to the transistor body using a capacitor, so these devices are equivalent to a three-terminal transistor when you connect the substrate port to any other port and use the default value of zero for the **C-S junction capacitance, CJS** parameter.

The NPN block model includes the following components:

- "Current-Voltage and Base Charge Model" on page 1-743
- "Base Resistance Model" on page 1-746
- "Transit Charge Modulation Model" on page 1-747
- "Junction Charge Model" on page 1-748
- "Temperature Dependence" on page 1-750

Current-Voltage and Base Charge Model

The current-voltage relationships and base charge relationships for the transistor are calculated adjusting the applicable model parameters for temperature as described in the following sections:

- "Base-Emitter and Base-Collector Junction Currents" on page 1-744

The base-emitter junction current is calculated using the following equations:

- When $V_{BE} > 80 * V_{TF}$:

$$I_{bef} = IS * \left(\left(\frac{V_{BE}}{V_{TF}} - 79 \right) * e^{80} - 1 \right) + G_{\min} * V_{BE}$$

$$I_{bee} = ISE * \left((V_{BE} - 80 * V_{TF} + V_{TE}) * \frac{e^{(80 * V_{TF}/V_{TE})}}{V_{TE}} - 1 \right)$$

- When $V_{BE} \leq 80 * V_{TF}$

$$I_{bef} = IS * \left(e^{(V_{BE}/V_{TF})} - 1 \right) + G_{\min} * V_{BE}$$

$$I_{bee} = ISE * \left(e^{(V_{BE}/V_{TE})} - 1 \right)$$

The base-collector junction current is calculated using the following equations:

- When $V_{BC} > 80 * V_{TR}$:

$$I_{bcr} = IS * \left(\left(\frac{V_{BC}}{V_{TR}} - 79 \right) * e^{80} - 1 \right) + G_{\min} * V_{BC}$$

$$I_{bcc} = ISC * \left((V_{BC} - 80 * V_{TR} + V_{TC}) * \frac{e^{(80 * V_{TR}/V_{TC})}}{V_{TC}} - 1 \right)$$

- When $V_{BC} \leq 80 * V_{TR}$

$$I_{bcr} = IS * \left(e^{(V_{BC}/V_{TR})} - 1 \right) + G_{min} * V_{BC}$$

$$I_{bcc} = ISC * \left(e^{(V_{BC}/V_{TC})} - 1 \right)$$

In the preceding equations:

- V_{BE} is the base-emitter voltage and V_{BC} is the base-collector voltage.

-

  $V_{TE} = NE * k * T / q, V_{TC} = NC * k * T / q, V_{TF} = NF * k * T / q$, and
  $V_{TR} = NR * k * T / q.$

- ISC and ISE are the **B-C leakage current, ISC** and **B-E leakage current, ISE** parameter values, respectively.

- NE, NC, NF, and NR are the **B-E emission coefficient, NE**, **B-C emission coefficient, NC**, **Forward emission coefficient, NF** and **Reverse emission coefficient, NR** parameter values, respectively.

- q is the elementary charge on an electron.

- k is the Boltzmann constant.

- T is the transistor temperature:

  - If you select `Device temperature` for the **Model temperature dependence using** parameter, T is the sum of the **Circuit temperature** value plus the **Offset local circuit temperature, TOFFSET** parameter value. The **Circuit temperature** value comes from the SPICE Environment Parameters block, if one exists in the circuit. Otherwise, it comes from the default value for this block.

  - If you select `Fixed temperature` for the **Model temperature dependence using** parameter, T is the **Fixed circuit temperature, TFIXED** parameter value.

- G_{min} is the minimum conductance. By default, G_{min} matches the **Minimum conductance GMIN** parameter of the SPICE Environment Parameters block, whose default value is 1e-12. To change G_{min}, add a SPICE Environment Parameters block to your model and set the **Minimum conductance GMIN** parameter to the desired value.

The terminal currents, I_B and I_C are the base and collector currents, defined as positive into the device. They are calculated as:

$$I_B = -\left(\frac{I_{ebf}}{BF} + I_{ebe} + \frac{I_{cbr}}{BR} + I_{cbc} \right)$$

$$I_C = -\left(\frac{I_{ebf} - I_{cbr}}{q_b} - \frac{I_{cbr}}{BR} - I_{cbc} \right)$$

where *BF* and *BR* are the **Forward beta, BF** and **Reverse beta, BR** parameter values, respectively.

The base charge, q_b, is calculated using the following equations:

$$q_b = \frac{q_1}{2} \left(1 + \sqrt{0.5 * \left(\sqrt{(1 + 4 * q_2 - eps)^2 + eps^2} + 1 + 4 * q_2 - eps \right) + eps} \right)$$

$$q_1 = \left(1 - \frac{V_{BC}}{VAF} - \frac{V_{BE}}{VAR} \right)^{-1}$$

$$q_2 = \frac{I_{bef}}{IKF} + \frac{I_{bcr}}{IKR}$$

where

- *VAF* and *VAR* are the **Forward Early voltage, VAF** and **Reverse Early voltage, VAR** parameters, respectively.
- *IKF* and *IKR* are the **Forward knee current, IKF** and **Reverse knee current, IKR** parameter values, respectively.
- *eps* is 1e-4.

Base Resistance Model

The block models base resistance in one of two ways:

- If you use the default value of infinity for the **Half base resistance cur, IRB** parameter, the NPN block calculates the base resistance r_{bb} as

$$r_{bb} = RBM + \frac{RB \text{-} RBM}{q_b}$$

where:

- *RBM* is the **Minimum base resistance, RBM** parameter value.
- *RB* is the **Zero-bias base resistance, RB** parameter value.

- If you specify a finite value for the **Half base resistance cur, IRB** parameter, the NPN block calculates the base resistance r_{bb} as

$$r_{bb} = RBM + 3*(RB \text{-} RBM)*\left(\frac{\tan z \text{ -} z}{z*\tan^2 z} \right)$$

where

$$z = \frac{\sqrt{1 + 144 I_B / \left(\pi^2 IRB \right)} - 1}{\left(24 / \pi^2 \right) \sqrt{\left(I_B / IRB \right)}}$$

Transit Charge Modulation Model

If you specify nonzero values for the **Coefficient of TF, XTF** parameter, the block models transit charge modulation by scaling the **Forward transit time, TF** parameter value as follows:

$$TF_{\text{mod}} = \frac{TF*\left[1 + XTF*e^{V_{BC}/(1.44V_{TF})} \left(\frac{I_{BE}}{I_{BE} + ITF} \right)^2 \right]}{q_b}$$

where *ITF* is the **Coefficient of TF, ITF** parameter value.

Junction Charge Model

The block lets you model junction charge. The base-collector charge Q_{bc} and the base-emitter charge Q_{be} depend on an intermediate value, Q_{dep} as follows, after adjusting the applicable model parameters for temperature:

- For the internal base-emitter junctions:

$$Q_{be} = TF_{\text{mod}} * I_{be} + Q_{dep}$$

- For the internal base-collector junctions:

$$Q_{bc} = TR * I_{bc} + XCJC * Q_{dep}$$

- For the external base-collector junctions:

$$Q_{b_{ext}c} = (1 - XCJC) * Q_{dep}$$

Q_{dep} depends on the junction voltage, V_{jct} (V_{BE} for the base-emitter junction and V_{BC} for the base-collector junction) as follows.

Applicable Range of V_{jct} Values	Corresponding Q_{dep} Equation
$V_{jct} < FC * VJ$	$Q_{dep} = C_{jct} * VJ * \dfrac{1 - \left(1 - V_{jct} / VJ\right)^{(1-MJ)}}{1 - MJ}$
$V_{jct} \geq FC * VJ$	$Q_{dep} = C_{jct} * \left[F1 + \dfrac{F3 * \left(V_{jct} - FC * VJ\right) + \dfrac{MJ * \left[V_{jct}^{2} - \left(FC * VJ\right)^{2}\right]}{2 * VJ}}{F2} \right]$

Where:

- *FC* is the **Capacitance coefficient FC** parameter value.

- *VJ* is:

  - The **B-E built-in potential, VJE** parameter value for the base-emitter junction.
  - The **B-C built-in potential, VJC** parameter value for the base-collector junction.

- *MJ* is:

  - The **B-E exponential factor, MJE** parameter value for the base-emitter junction.
  - The **B-C exponential factor, MJC** parameter value for the base-collector junction.

- C_{jct} is:

  - The **B-E depletion capacitance, CJE** parameter value for the base-emitter junction.
  - The **B-C depletion capacitance, CJC** parameter value for the base-collector junction.

- $$F1 = VJ * \left(1 - \left(1 - FC\right)^{(1-MJ)}\right) \Big/ \left(1 - MJ\right)$$

- $$F2 = \left(1 - FC\right)^{(1+MJ)}$$

- $$F3 = 1 - FC * \left(1 + MJ\right)$$

The collector-substrate charge Q_{cs} depends on the collector-substrate voltage V_{cs} as follows, after adjusting the applicable model parameters for temperature.

Applicable Range of V_{cs} Values	Corresponding Q_{cs} Equation
$V_{cs} < 0$	$Q_{cs} = CJS * VJS * \left(\dfrac{1 - \left(1 - V_{cs} / VJS\right)^{(1-MJS)}}{1 - MJS} \right)$
$V_{cs} \geq 0$	$Q_{cs} = CJS * \left(1 + MJS * V_{cs} / (2 * VJS)\right) * V_{cs}$

where:

- *CJS* is the **C-S junction capacitance, CJS** parameter value.

- *VJS* is the **Substrate built-in potential, VJS** parameter value.
- *MJS* is the **Substrate exponential factor, MJS** parameter value.

Temperature Dependence

Several transistor parameters depend on temperature. There are two ways to specify the transistor temperature:

- When you select `Device temperature` for the **Model temperature dependence using** parameter, the transistor temperature is

$$T = T_C + T_O$$

  where:

  - T_C is the **Circuit temperature** parameter value from the SPICE Environment Parameters block. If this block doesn't exist in the circuit, T_C is the default value of this parameter.
  - T_O is the **Offset local circuit temperature, TOFFSET** parameter value.

- When you select `Fixed temperature` for the **Model temperature dependence using** parameter, the transistor temperature is the **Fixed circuit temperature, TFIXED** parameter value.

The block provides the following relationship between the saturation current *IS* and the transistor temperature *T*:

$$IS(T) = IS * \left(T/T_{meas}\right)^{XTI} * e^{\left(\frac{T}{T_{meas}} - 1\right) * \frac{EG}{V_t}}$$

where:

- *IS* is the **Transport saturation current, IS** parameter value.
- T_{meas} is the **Parameter extraction temperature, TMEAS** parameter value.
- *XTI* is the **Temperature exponent for IS, XTI** parameter value.
- *EG* is the **Energy gap, EG** parameter value.
- $V_t = kT/q$.

The block provides the following relationship between the base-emitter junction potential *VJE* and the transistor temperature *T*:

$$VJE(T) = VJE * \left(\frac{T}{T_{meas}} \right) - \frac{3 * k * T}{q} * \log \left(\frac{T}{T_{meas}} \right) - \left(\frac{T}{T_{meas}} \right) * EG_{T_{meas}} + EG_T$$

where:

- *VJE* is the **B-E built-in potential, VJE** parameter value.

- $$EG_{T_{meas}} = 1.16eV - \left(7.02e \text{-} 4 * T_{meas}^2 \right) / \left(T_{meas} + 1108 \right)$$

- $$EG_T = 1.16eV - \left(7.02e \text{-} 4 * T^2 \right) / \left(T + 1108 \right)$$

The block uses the *VJE(T)* equation to calculate the base-collector junction potential by substituting *VJC* (the **B-C built-in potential, VJC** parameter value) for *VJE*.

The block provides the following relationship between the base-emitter junction capacitance *CJE* and the transistor temperature *T*:

$$CJE(T) = CJE * \left[1 + MJE * \left(400e - 6 * \left(T - T_{meas} \right) - \frac{VJE(T) - VJE}{VJE} \right) \right]$$

where:

- *CJE* is the **B-E depletion capacitance, CJE** parameter value.
- *MJE* is the **B-E exponential factor, MJE** parameter value.

The block uses the *CJE(T)* equation to calculate the base-collector junction capacitance by substituting *CJC* (the **B-C depletion capacitance, CJC** parameter value) for *CJE* and *MJC* (the **B-C exponential factor, MJC** parameter value) for *MJE*.

The block provides the following relationship between the forward and reverse beta and the transistor temperature *T*:

$$\beta(T) = \beta * \left(\frac{T}{T_{meas}} \right)^{XTB}$$

where:

- β is the **Forward beta, BF** or **Reverse beta, BR** parameter value.
- *XTB* is the **Beta temperature exponent, XTB** parameter value.

The block provides the following relationship between the base-emitter leakage current *ISE* and the transistor temperature *T*:

$$ISE(T) = ISE * \left(\frac{T}{T_{meas}} \right)^{-XTB} * \left(\frac{IS(T)}{IS} \right)^{1/NE}$$

where:

- *ISE* is the **B-E leakage current, ISE** parameter value.
- *NE* is the **B-E emission coefficient, NE** parameter value.

The block uses this equation to calculate the base-collector leakage current by substituting *ISC* (the **B-C leakage current, ISC** parameter value) for *ISE* and *NC* (the **B-C emission coefficient, NC** parameter value) for *NE*.

Basic Assumptions and Limitations

The model is based on the following assumptions:

- The NPN block does not support noise analysis.
- The NPN block applies initial conditions across junction capacitors and not across the block ports.

Parameters

Main Tab

Device area, AREA

> The transistor area. This value multiplies the following parameter values:

- **Transport saturation current, IS**
- **Forward knee current, IKF**
- **B-E leakage current, ISE**
- **Reverse knee current, IKR**
- **B-C leakage current, ISC**
- **Half base resistance cur, IRB**
- **B-E depletion capacitance, CJE**
- **Coefficient of TF, ITF**
- **B-C depletion capacitance, CJC**
- **C-S junction capacitance, CJS**

It divides the following parameter values:

- **Zero-bias base resistance, RB**
- **Minimum base resistance, RBM**
- **Emitter resistance, RE**
- **Collector resistance, RC**

The default value is 1 m^2. The value must be greater than 0.

Number of parallel devices, SCALE

The number of parallel transistors the block represents. This value multiplies the output current and device charges. The default value is 1. The value must be greater than 0.

Forward Gain Tab

Transport saturation current, IS

The magnitude of the current at which the transistor saturates. The default value is $1e\text{-}16$ A/m^2. The value must be greater than or equal to 0.

Forward beta, BF

The ideal maximum reverse beta. The default value is 100. The value must be greater than 0.

Forward emission coefficient, NF

The reverse emission coefficient or ideality factor. The default value is 1. The value must be greater than 0.

B-E leakage current, ISE

The base-emitter leakage current. The default value is `0` A/m^2. The value must be greater than or equal to `0`.

B-E emission coefficient, NE

The base-collector emission coefficient or ideality factor. The default value is `1.5`. The value must be greater than `0`.

Forward knee current, IKF

The current value at which forward-beta high-current roll-off occurs. The default value is `Inf` A/m^2. The value must be greater than or equal to `0`.

Forward Early voltage, VAF

The forward Early voltage. The default value is `Inf` V. The value must be greater than or equal to `0`.

Reverse Gain Tab

Reverse beta, BR

The ideal maximum reverse beta. The default value is `1`. The value must be greater than `0`.

Reverse emission coefficient, NR

The reverse emission coefficient or ideality factor. The default value is `1`. The value must be greater than `0`.

B-C leakage current, ISC

The base-collector leakage current. The default value is `0` A/m^2. The value must be greater than or equal to `0`.

B-C emission coefficient, NC

The base-collector emission coefficient or ideality factor. The default value is `2`. The value must be greater than `0`.

Reverse knee current, IKR

The current value at which reverse-beta high-current roll-off occurs. The default value is `Inf` A/m^2. The value must be greater than or equal to `0`.

Reverse Early voltage, VAR

The reverse Early voltage. The default value is `Inf` V. The value must be greater than or equal to `0`.

Resistors Tab

Emitter resistance, RE

The resistance of the emitter. The default value is `1e-4` m^2*Ω. The value must be greater than or equal to `0`.

Collector resistance, RC

The resistance of the collector. The default value is `0.01` m^2*Ω. The value must be greater than or equal to `0`.

Zero-bias base resistance, RB

The resistance of the base. The default value is `1` m^2*Ω. The value must be greater than or equal to `0`.

Minimum base resistance, RBM

The minimum resistance of the base. The default value is `0` m^2*Ω. The value must be less than or equal to the **Zero-bias base resistance, RB** parameter value.

Half base resistance cur, IRB

The base current at which the base resistance has dropped to half of its zero-bias value. The default value is `Inf` A/m^2. The value must be greater than or equal to `0`. Use the default value of `Inf` if you do not want to model the change in base resistance as a function of base current.

Capacitance Tab

Model junction capacitance

Select one of the following options for modeling the junction capacitance:

- `No` — Do not include junction capacitance in the model. This is the default option.
- `B-E Capacitance` — Model the junction capacitance across the base-emitter junction.
- `B-C Capacitance` — Model the junction capacitance across the base-collector junction.
- `C-S Capacitance` — Model the junction capacitance across the collector-substrate junction.

Note To include junction capacitance in the model:

1 Select `B-E Capacitance` and specify the base-emitter junction capacitance parameters.

2 Select `B-C Capacitance` and specify the base-collector junction capacitance parameters.

3 Select `C-S Capacitance` and specify the collector-substrate junction capacitance parameters.

You can specify or change any of the common parameters when you select any of the preceding options for the **Model junction capacitance** parameter.

B-E depletion capacitance, CJE

The depletion capacitance across the base-emitter junction. This parameter is only visible when you select `B-E Capacitance` for the **Model junction capacitance** parameter. The default value is `0` F/m^2. The value must be greater than or equal to `0`.

B-E built-in potential, VJE

The base-emitter junction potential. This parameter is only visible when you select `B-E Capacitance` for the **Model junction capacitance** parameter. The default value is `0.75` V. The value must be greater than or equal to `0.01` V.

B-E exponential factor, MJE

The grading coefficient for the base-emitter junction. This parameter is only visible when you select `B-E Capacitance` for the **Model junction capacitance** parameter. The default value is `0.33`. The value must be greater than or equal to `0` and less than or equal to `0.9`.

Forward transit time, TF

The transit time of the minority carriers that cause diffusion capacitance when the base-emitter junction is forward-biased. This parameter is only visible when you select `B-E Capacitance` for the **Model junction capacitance** parameter. The default value is `0`. The value must be greater than or equal to `0`.

Coefficient of TF, XTF

The coefficient for the base-emitter and base-collector bias dependence of the transit time, which produces a charge across the base-emitter junction. This parameter is only visible when you select `B-E Capacitance` for the **Model junction capacitance** parameter. The default value is `0`. The value must be greater than or equal to `0`. Use the default value of `0` if you do not want to model the effect of base-emitter bias on transit time.

VBC dependence of TF, VTF

The coefficient for the base-emitter bias dependence of the transit time. This parameter is only visible when you select `B-E Capacitance` for the **Model junction capacitance** parameter. The default value is `Inf` V. The value must be greater than or equal to `0`.

Coefficient of TF, ITF

The coefficient for the dependence of the transit time on collector current. This parameter is only visible when you select `B-E Capacitance` for the **Model junction capacitance** parameter. The default value is `0` A/m^2. The value must be greater than or equal to `0`. Use the default value of `0` if you do not want to model the effect of collector current on transit time.

B-C depletion capacitance, CJC

The depletion capacitance across the base-collector junction. This parameter is only visible when you select `B-C Capacitance` for the **Model junction capacitance** parameter. The default value is `0` F/m^2. The value must be greater than `0`.

B-C built-in potential, VJC

The base-collector junction potential. This parameter is only visible when you select `B-C Capacitance` for the **Model junction capacitance** parameter. The default value is `0.75` V. The value must be greater than or equal to `0.01` V.

B-C exponential factor, MJC

The grading coefficient for the base-collector junction. This parameter is only visible when you select `B-C Capacitance` for the **Model junction capacitance** parameter. The default value is `0.33`. The value must be greater than or equal to `0` and less than or equal to `0.9`.

B-C capacitance fraction, XCJC

The fraction of the base-collector depletion capacitance that is connected between the internal base and the internal collector. The rest of the base-collector depletion capacitance is connected between the external base and the internal collector. This parameter is only visible when you select `B-C Capacitance` for the **Model junction capacitance** parameter. The default value is `0`. The value must be greater than or equal to `0` and less than or equal to `1`.

Reverse transit time, TR

The transit time of the minority carriers that cause diffusion capacitance when the base-collector junction is reverse-biased. This parameter is only visible when you select `B-C Capacitance` for the **Model junction capacitance** parameter. The default value is `0` s. The value must be greater than or equal to `0`.

Capacitance coefficient FC

The fitting coefficient that quantifies the decrease of the depletion capacitance with applied voltage. This parameter is only visible when you select `B-E Capacitance` or `B-C Capacitance` for the **Model junction capacitance** parameter. The default value is `0.5`. The value must be greater than or equal to `0` and less than or equal to `0.95`.

Specify initial condition

Select one of the following options for specifying an initial condition:

- `No` — Do not specify an initial condition for the model. This is the default option.
- `Yes` — Specify the initial transistor conditions.

Note The NPN block applies the initial transistor voltages across the junction capacitors and not across the ports.

This parameter is only visible when you select `B-E Capacitance` or `B-C Capacitance` for the **Model junction capacitance** parameter.

Initial condition voltage ICVBE

Base-emitter voltage at the start of the simulation. This parameter is only visible when you select `B-E Capacitance` or `B-C Capacitance` for the **Model junction capacitance** and `Yes` for the **Specify initial condition** parameter. The default value is `0` V.

Initial condition voltage ICVCE

Base-collector voltage at the start of the simulation. This parameter is only visible when you select `B-E Capacitance` or `B-C Capacitance` for the **Model junction capacitance** and `Yes` for the **Specify initial condition** parameter. The default value is `0` V.

C-S junction capacitance, CJS

The collector-substrate junction capacitance. This parameter is only visible when you select `C-S Capacitance` for the **Model junction capacitance** parameter. The default value is `0` F/m^2. The value must be greater than or equal to `0`.

Substrate built-in potential, VJS

The potential of the substrate. This parameter is only visible when you select `C-S Capacitance` for the **Model junction capacitance** parameter. The default value is `0.75` V.

Substrate exponential factor, MJS

The grading coefficient for the collector-substrate junction. This parameter is only visible when you select `C-S Capacitance` for the **Model junction capacitance** parameter. The default value is `0`. The value must be greater than or equal to `0` and less than or equal to `0.9`.

Temperature Tab

Model temperature dependence using

Select one of the following options for modeling the transistor temperature dependence:

- `Device temperature` — Use the device temperature, which is the **Circuit temperature** value plus the **Offset local circuit temperature, TOFFSET** value. The **Circuit temperature** value comes from the SPICE Environment Parameters block, if one exists in the circuit. Otherwise, it comes from the default value for this block.
- `Fixed temperature` — Use a temperature that is independent of the circuit temperature to model temperature dependence.

Beta temperature exponent, XTB

The forward and reverse beta temperature exponent that models base current temperature dependence. This parameter is only visible when you select `Device temperature` for the **Model temperature dependence using** parameter. The default value is `0`. The value must be greater than or equal to `0`.

Energy gap, EG

The energy gap that affects the increase in the saturation current as temperature increases. This parameter is only visible when you select `Device temperature` for the **Model temperature dependence using** parameter. The default value is `1.11` eV. The value must be greater than or equal to `0.1`.

Temperature exponent for IS, XTI

The order of the exponential increase in the saturation current as temperature increases. This parameter is only visible when you select `Device temperature` for the **Model temperature dependence using** parameter. The default value is `3`. The value must be greater than or equal to `0`.

Offset local circuit temperature, TOFFSET

The amount by which the transistor temperature differs from the circuit temperature. This parameter is only visible when you select `Device temperature` for the **Model temperature dependence using** parameter. The default value is `0` K.

Parameter extraction temperature, TMEAS

The temperature at which the transistor parameters were measured. The default value is `300.15` K. The value must be greater than `0`.

Fixed circuit temperature, TFIXED

The temperature at which to simulate the transistor. This parameter is only visible when you select `Fixed temperature` for the **Model temperature dependence using** parameter. The default value is `300.15` K. The value must be greater than `0`.

Ports

The block has the following ports:

B

Electrical conserving port associated with the transistor base terminal.

C

Electrical conserving port associated with the transistor collector terminal.

E

Electrical conserving port associated with the transistor emitter terminal.

S

Electrical conserving port associated with the transistor substrate terminal.

References

[1] G. Massobrio and P. Antognetti. *Semiconductor Device Modeling with SPICE*. 2nd Edition, McGraw-Hill, 1993. Chapter 2.

See Also

NPN Bipolar Transistor

SPICE PJFET

Model SPICE-compatible P-Channel JFET

Library

SPICE-Compatible Components/Semiconductor Devices

Description

The PJFET block represents a SPICE-compatible P-channel JFET.

The PJFET block model includes the following components:

Source-Gate Current-Voltage Model

The block provides the following relationship between the source-gate current I_{sg} and the source-gate voltage V_{sg} after adjusting the applicable model parameters for temperature.

Applicable Range of V_{sg} Values	Corresponding I_{sg} Equation
$V_{sg} > 80 * V_t$	$I_{sg} = IS * \left(\left(\dfrac{V_{sg}}{V_t} - 79 \right) e^{80} - 1 \right) + V_{sg} * G\min$

Applicable Range of V_{sg} Values	Corresponding I_{sg} Equation
$80*V_t \geq V_{sg}$	$I_{sg} = IS*\left(e^{V_{sg}/V_t} - 1\right) + V_{sg}*G\min$

Where:

- *IS* is the **Saturation current, IS** parameter value.

- $$V_t = ND*k*T / q$$

- *ND* is the **Emission coefficient, N** parameter value.

- *q* is the elementary charge on an electron.

- *k* is the Boltzmann constant.

- *T* is the diode temperature:

  - If you select `Device temperature` for the **Model temperature dependence using** parameter, *T* is the sum of the **Circuit temperature** value plus the **Offset local circuit temperature, TOFFSET** parameter value. The **Circuit temperature** value comes from the SPICE Environment Parameters block, if one exists in the circuit. Otherwise, it comes from the default value for this block.

  - If you select `Fixed temperature` for the **Model temperature dependence using** parameter, *T* is the **Fixed circuit temperature, TFIXED** parameter value.

- *GMIN* is the diode minimum conductance. By default, *GMIN* matches the **Minimum conductance GMIN** parameter of the SPICE Environment Parameters block, whose default value is 1e-12. To change *GMIN*, add a SPICE Environment Parameters block to your model and set the **Minimum conductance GMIN** parameter to the desired value.

Drain-Gate Current-Voltage Model

The block provides the following relationship between the drain-gate current I_{dg} and the drain-gate voltage V_{dg} after adjusting the applicable model parameters for temperature.

Applicable Range of V_{dg} Values	Corresponding I_{dg} Equation
$V_{dg} > 80*V_t$	$I_{dg} = IS * \left(\left(\dfrac{V_{dg}}{V_t} - 79 \right) e^{80} - 1 \right) + V_{dg} * G\min$
$80*V_t \geq V_{dg}$	$I_{dg} = IS * \left(e^{V_{dg}/V_t} - 1 \right) + V_{dg} * G\min$

Source-Drain Current-Voltage Model

The block provides the following relationship between the source-drain current I_{sd} and the source-drain voltage V_{sd} in normal mode ($V_{sd} \geq 0$) after adjusting the applicable model parameters for temperature.

Applicable Range of V_{sg} and V_{dg} Values	Corresponding I_{sd} Equation
$V_{sg} - V_{to} \leq 0$	$I_{sd} = 0$
$0 < V_{sg} - V_{to} \leq V_{sd}$	$I_{sd} = -\beta * \left(V_{sg} - V_{to} \right)^2 * \left(1 + \lambda * V_{sd} \right)$
$0 < V_{sd} < V_{sg} - V_{to}$	$I_{sd} = \beta * V_{sd} * \left(2 * \left(V_{sg} - V_{to} \right) - V_{sd} \right) * \left(1 + \lambda * V_{sd} \right)$

Where:

- V_{to} is the **Threshold voltage, VTO** parameter value.
- β is the **Transconductance, BETA** parameter value.
- λ is the **Channel modulation, LAMBDA** parameter value.

The block provides the following relationship between the source-drain current I_{sd} and the source-drain voltage V_{sd} in inverse mode ($V_{sd} < 0$) after adjusting the applicable model parameters for temperature.

Applicable Range of V_{sg} and V_{dg} Values	Corresponding I_{sd} Equation
$V_{dg} - V_{to} \leq 0$	$I_{sd} = 0$
$0 < V_{dg} - V_{to} \leq -V_{sd}$	$I_{sd} = \beta * \left(V_{dg} - V_{to} \right)^2 * \left(1 - \lambda * V_{sd} \right)$
$0 < -V_{sd} < V_{dg} - V_{to}$	$I_{sd} = \beta * V_{sd} * \left(2 * \left(V_{dg} - V_{to} \right) + V_{sd} \right) * \left(1 - \lambda * V_{sd} \right)$

Junction Charge Model

The block provides the following relationship between the source-gate charge Q_{sg} and the source-gate voltage V_{sg} after adjusting the applicable model parameters for temperature.

Applicable Range of V_{sg} Values	Corresponding Q_{sg} Equation
$V_{sg} < FC * VJ$	$Q_{sg} = \dfrac{CGS * VJ * \left(1 - \left(1 - \dfrac{V_{sg}}{VJ} \right)^{1-MG} \right)}{1 - MG}$
$V_{sg} \geq FC * VJ$	$Q_{sg} = CGS * \left(F1 + \dfrac{F3 * \left(V_{sg} - FC * VJ \right) + \dfrac{MG * \left(V_{sg}^2 - \left(FC * VJ \right)^2 \right)}{2 * VJ}}{F2} \right)$

Where:

- *FC* is the **Capacitance coefficient FC** parameter value.
- *VJ* is the **Junction potential VJ** parameter value.
- *CGS* is the **Zero-bias GS capacitance, CGS** parameter value.
- *MG* is the **Grading coefficient, M** parameter value.

- $$F1 = \frac{VJ * \left(1 - \left(1 - FC\right)^{1-MG}\right)}{1 - MG}$$

- $$F2 = \left(1 - FC\right)^{1+MG}$$

- $$F3 = 1 - FC * \left(1 + MG\right)$$

The block provides the following relationship between the drain-gate charge Q_{dg} and the drain-gate voltage V_{dg} after adjusting the applicable model parameters for temperature.

Applicable Range of V_{dg} Values	Corresponding Q_{dg} Equation
$V_{dg} < FC * VJ$	$$Q_{dg} = \frac{CGD * VJ * \left(1 - \left(1 - \frac{V_{dg}}{VJ}\right)^{1-MG}\right)}{1 - MG}$$
$V_{dg} \geq FC * VJ$	$$Q_{dg} = CGD * \left(F1 + \frac{F3 * \left(V_{dg} - FC * VJ\right) + \frac{MG * \left(V_{dg}^{2} - \left(FC * VJ\right)^{2}\right)}{2 * VJ}}{F2} \right)$$

Where:

- CGD is the **Zero-bias GD capacitance, CGD** parameter value.

Temperature Dependence

Several transistor parameters depend on temperature. There are two ways to specify the transistor temperature:

- When you select Device temperature for the **Model temperature dependence using** parameter, the transistor temperature is

$$T = T_C + T_O$$

where:

- T_C is the **Circuit temperature** parameter value from the SPICE Environment Parameters block. If this block doesn't exist in the circuit, T_C is the default value of this parameter.
- T_O is the **Offset local circuit temperature, TOFFSET** parameter value.
- When you select `Fixed temperature` for the **Model temperature dependence using** parameter, the transistor temperature is the **Fixed circuit temperature, TFIXED** parameter value.

The block provides the following relationship between the saturation current *IS* and the transistor temperature *T*:

$$IS(T) = IS * \left(T/T_{meas}\right)^{\frac{XTI}{ND}} * e^{\left(\frac{T}{T_{meas}}-1\right)*\frac{EG}{V_t}}$$

where:

- *IS* is the **Saturation current, IS** parameter value.
- T_{meas} is the **Parameter extraction temperature, TMEAS** parameter value.
- *XTI* is the **Saturation current temperature exponent, XTI** parameter value.
- *EG* is the **Energy gap, EG** parameter value.
-
  $$V_t = ND * k * T / q$$
- *ND* is the **Emission coefficient, N** parameter value.

The block provides the following relationship between the junction potential *VJ* and the transistor temperature *T*:

$$VJ(T) = VJ * \left(\frac{T}{T_{meas}}\right) - \frac{3 * k * T}{q} * \log\left(\frac{T}{T_{meas}}\right) - \left(\frac{T}{T_{meas}}\right) * EG_{T_{meas}} + EG_T$$

where:

- *VJ* is the **Junction potential VJ** parameter value.

1-767

- $$EG_{T_{meas}} = 1.16eV - \left(7.02e\text{-}4 * T_{meas}^{\;2}\right) / \left(T_{meas} + 1108\right)$$

- $$EG_{T} = 1.16eV - \left(7.02e\text{-}4 * T^{2}\right) / \left(T + 1108\right)$$

The block provides the following relationship between the gate-source junction capacitance *CGS* and the transistor temperature *T*:

$$CGS(T) = CGS * \left[1 + MG * \left(400e - 6 * \left(T - T_{meas}\right) - \frac{VJ(T) - VJ}{VJ}\right)\right]$$

where:

- *CGS* is the **Zero-bias GS capacitance, CGS** parameter value.

The block uses the *CGS(T)* equation to calculate the gate-drain junction capacitance by substituting *CGD* (the **Zero-bias GD capacitance, CGD** parameter value) for *CGS*.

The block provides the following relationship between the forward and reverse beta and the transistor temperature *T*:

$$\beta(T) = \beta * \left(\frac{T}{T_{meas}}\right)$$

where β is the **Transconductance, BETA** parameter value.

Basic Assumptions and Limitations

The model is based on the following assumptions:

- The PJFET block does not support noise analysis.
- The PJFET block applies initial conditions across junction capacitors and not across the block ports.

Parameters

Main Tab

Device area, AREA

The transistor area. This value multiplies the **Transconductance, BETA**, **Zero-bias GS capacitance, CGS**, **Zero-bias GD capacitance, CGD**, and **Saturation current, IS** parameter values. It divides the **Source resistance, RS** and **Drain resistance, RD** parameter values. The default value is 1 m^2. The value must be greater than 0.

Number of parallel devices, SCALE

The number of parallel transistors the block represents. This value multiplies the output current and device charges. The default value is 1. The value must be greater than 0.

Threshold voltage, VTO

The gate-source voltage above which the transistor produces a nonzero drain current. The default value is -2 V.

Transconductance, BETA

The derivative of drain current with respect to gate voltage. The default value is $1e-04$ A/m^2/V^2. The value must be greater than or equal to 0.

Channel modulation, LAMBDA

The channel-length modulation. The default value is 0 1/V.

Saturation current, IS

The magnitude of the current that the ideal diode equation approaches asymptotically for very large reverse bias levels. The default value is $1e-14$ A/m^2. The value must be greater than or equal to 0.

Emission coefficient, N

The transistor emission coefficient or ideality factor. The default value is 1. The value must be greater than 0.

Source resistance, RS

The transistor source resistance. The default value is $1e-4$ m^2*Ω. The value must be greater than or equal to 0.

Drain resistance, RD

The transistor drain resistance. The default value is 0.01 m^2*Ω. The value must be greater than or equal to 0.

Junction Capacitance Tab

Model junction capacitance

Select one of the following options for modeling the junction capacitance:

- No — Do not include junction capacitance in the model. This is the default option.
- Yes — Specify zero-bias junction capacitance, junction potential, grading coefficient, forward-bias depletion capacitance coefficient, and transit time.

Zero-bias GS capacitance, CGS

The value of the capacitance placed between the gate and the source. This parameter is only visible when you select Yes for the **Model junction capacitance** parameter. The default value is 0 F/m^2. The value must be greater than or equal to 0.

Zero-bias GD capacitance, CGD

The value of the capacitance placed between the gate and the drain. This parameter is only visible when you select Yes for the **Model junction capacitance** parameter. The default value is 0 F/m^2. The value must be greater than or equal to 0.

Junction potential VJ

The junction potential. This parameter is only visible when you select Yes for the **Model junction capacitance** parameter. The default value is 1 V. The value must be greater than 0.01 V.

Grading coefficient, M

The transistor grading coefficient. The default value is 0.5. The value must be greater than 0 and less than 0.9.

Capacitance coefficient FC

The fitting coefficient that quantifies the decrease of the depletion capacitance with applied voltage. This parameter is only visible when you select Yes for the **Model junction capacitance** parameter. The default value is 0.5. The value must be greater than or equal to 0 and less than or equal to 0.95.

Specify initial condition

Select one of the following options for specifying an initial condition:

- No — Do not specify an initial condition for the model. This is the default option.
- Yes — Specify the initial diode voltage.

> **Note** The PJFET block applies the initial diode voltage across the junction capacitors and not across the ports.

Initial condition voltage ICVDS

Drain-source voltage at the start of the simulation. This parameter is only visible when you select `Yes` for the **Model junction capacitance** and `Yes` for the **Specify initial condition** parameter. The default value is `0` V.

Initial condition voltage ICVGS

Gate-source voltage at the start of the simulation. This parameter is only visible when you select `Yes` for the **Model junction capacitance** and `Yes` for the **Specify initial condition** parameter. The default value is `0` V.

Temperature Tab

Model temperature dependence using

Select one of the following options for modeling the diode temperature dependence:

- `Device temperature` — Use the device temperature, which is the **Circuit temperature** value plus the **Offset local circuit temperature, TOFFSET** value. The **Circuit temperature** value comes from the SPICE Environment Parameters block, if one exists in the circuit. Otherwise, it comes from the default value for this block.

- `Fixed temperature` — Use a temperature that is independent of the circuit temperature to model temperature dependence.

Saturation current temperature exponent, XTI

The order of the exponential increase in the saturation current as temperature increases. The default value is `0`. The value must be greater than or equal to `0`.

Activation energy, EG

The energy gap that affects the increase in the saturation current as temperature increases. The default value is `1.11` eV. The value must be greater than `0.1` eVi.

Offset local circuit temperature, TOFFSET

The amount by which the transistor temperature differs from the circuit temperature. This parameter is only visible when you select `Device temperature` for the **Model temperature dependence using** parameter. The default value is `0` K.

Fixed circuit temperature, TFIXED

The temperature at which to simulate the transistor. This parameter is only visible when you select `Fixed temperature` for the **Model temperature dependence using** parameter. The default value is `300.15` K. The value must be greater than `0`.

Parameter extraction temperature, TMEAS

The temperature at which the transistor parameters were measured. The default value is `300.15` K. The value must be greater than `0`.

Ports

The block has the following ports:

G

Electrical conserving port associated with the transistor gate terminal.

D

Electrical conserving port associated with the transistor drain terminal.

S

Electrical conserving port associated with the transistor source terminal.

References

[1] G. Massobrio and P. Antognetti. *Semiconductor Device Modeling with SPICE*. 2nd Edition, McGraw-Hill, 1993. Chapter 3.

See Also

P-Channel JFET, SPICE NJFET

SPICE PMOS

Model SPICE-compatible P-Channel MOSFET

Library

SPICE-Compatible Components/Semiconductor Devices

Description

The PMOS block represents a SPICE-compatible P-channel MOSFET.

The PMOS block model includes the following components:

- "Resistance Calculations" on page 1-773
- "Bulk-Source Diode Model" on page 1-774
- "Bulk-Drain Diode Model" on page 1-775
- "Level 1 Drain Current Model" on page 1-776
- "Level 3 Drain Current Model" on page 1-779
- "Junction Charge Model" on page 1-784
- "Temperature Dependence" on page 1-788

Resistance Calculations

The following table shows how the PMOS block calculates the transistor drain resistance. The abbreviations in the table represent the values of the following block parameters:

- **Drain resistance, RD**
- **Sheet resistance, RSH**
- **Number of drain squares, NRD**

Drain resistance, RD Parameter	Sheet resistance, RSH Parameter	Drain Resistance
NaN	NaN	0
RD	NaN or *RSH*	*RD*
NaN	*RSH*	*RSH*NRD*

The following table shows how the PMOS block calculates the transistor source resistance. The abbreviations in the table represent the values of the following block parameters:

- **Source resistance, RS**
- **Sheet resistance, RSH**
- **Number of source squares, NRS**

Source resistance, RS Parameter	Sheet resistance, RSH Parameter	Source Resistance
NaN	NaN	0
RS	NaN or *RSH*	*RS*
NaN	*RSH*	*RSH*NRS*

Bulk-Source Diode Model

The block provides the following relationship between the bulk-source current I_{sb} and the bulk-source voltage V_{sb} after adjusting the applicable model parameters for temperature.

Applicable Range of V_{sb} Values	Corresponding I_{gs} Equation
$V_{sb} > 80 * V_{tn}$	$I_{sb} = IS_{sb} * \left(\left(\frac{V_{sb}}{V_{tn}} - 79 \right) e^{80} - 1 \right) + V_{sb} * G \min$
$80 V_{tn} \geq V_{sb}$	$I_{sb} = IS_{sb} * \left(e^{V_{sb}/V_{tn}} - 1 \right) + V_{sb} * G \min$

Where:

- IS_{sb} is

  - The product of the **Bulk jct sat current density, JS** parameter value and the **Area of source, AS** parameter value if both these parameter values and the **Area of drain, AD** parameter value are nonzero.

  - The **Bulk saturation current, IS** parameter value, otherwise.

- $V_{tn}=NkT/q$

- q is the elementary charge on an electron, 1.6021918e-19 C.

- N is the **Emission coefficient, N** parameter value.

- k is the Boltzmann constant.

- T is the diode temperature:

  - If you select `Device temperature` for the **Model temperature dependence using** parameter, T is the sum of the **Circuit temperature** value plus the **Offset local circuit temperature, TOFFSET** parameter value. The **Circuit temperature** value comes from the SPICE Environment Parameters block, if one exists in the circuit. Otherwise, it comes from the default value for this block.

  - If you select `Fixed temperature` for the **Model temperature dependence using** parameter, T is the **Fixed circuit temperature, TFIXED** parameter value.

- *GMIN* is the diode minimum conductance. By default, *GMIN* matches the **Minimum conductance GMIN** parameter of the SPICE Environment Parameters block, whose default value is 1e-12. To change *GMIN*, add a SPICE Environment Parameters block to your model and set the **Minimum conductance GMIN** parameter to the desired value.

Bulk-Drain Diode Model

The block provides the following relationship between the bulk-drain current I_{db} and the bulk-drain voltage V_{db} after adjusting the applicable model parameters for temperature.

Applicable Range of V_{db} Values	Corresponding I_{gs} Equation
$V_{db} > 80 * V_{tn}$	$I_{db} = IS_{db} * \left(\left(\frac{V_{db}}{V_{tn}} - 79 \right) e^{80} - 1 \right) + V_{db} * G\min$

Applicable Range of V_{db} Values	Corresponding I_{gs} Equation
$80V_{tn} \geq V_{db}$	$I_{db} = IS_{db} * \left(e^{V_{db}/V_{tn}} - 1 \right) + V_{db} * G\min$

Where:

- IS_{db} is

  - The product of the **Bulk jct sat current density, JS** parameter value and the **Area of drain, AD** parameter value if both these parameter values and the **Area of source, AS** parameter value are nonzero.
  - The **Bulk saturation current, IS** parameter value, otherwise.

Level 1 Drain Current Model

The block provides the following relationship between the drain current I_{sd} and the drain-source voltage V_{sd} in normal mode ($V_{sd} \geq 0$) after adjusting the applicable model parameters for temperature.

Normal Mode

Applicable Range of V_{sg} and V_{sd} Values	Corresponding I_{sd} Equation
$V_{sg} - V_{on} \leq 0$	$I_{sd} = 0$
$0 < V_{sg} - V_{on} \leq V_{sd}$	$I_{sd} = BETA * \left(V_{sg} - V_{on} \right)^2 \dfrac{(1 + LAMBDA * V_{sd})}{2}$
$0 < V_{sd} < V_{sg} - V_{on}$	$I_{sd} = BETA *$ $V_{sd} \left(\left(V_{sg} - V_{on} \right) - \dfrac{V_{sd}}{2} \right)(1 + LAMBDA * V_{sd})$

Where:

- V_{on} is:

- $$MTYPE * VBI + GAMMA\sqrt{PHI - V_{sb}} \text{ if } V_{sb} \leq 0 .$$

-
$$MTYPE * VBI + GAMMA\left(\sqrt{PHI} - \frac{V_{sb}}{2\sqrt{PHI}} \right) \text{ if } 0 < V_{sb} \leq 2 * PHI .$$
  - $MTYPE*VBI$ if $V_{sb} > 2*PHI$.
- *MTYPE* is –1.
- *BETA* is *KP*WIDTH/(LENGTH-2*LD)*
- *KP* is:

  - The **Transconductance, KP** parameter value, if this parameter has a numerical value.

  -
    $U0*3.9*\varepsilon_0 / TOX$, if **Transconductance, KP** is NaN and you specify values for both the **Oxide thickness, TOX** and **Substrate doping, NSUB** parameters.
- *WIDTH* is the **Width of channel, WIDTH** parameter value.
- *LENGTH* is the **Length of channel, LENGTH** parameter value.
- *LD* is the **Lateral diffusion, LD** parameter value.
- *VBI* is an built-in voltage value the block uses in calculations. The value is a function of temperature. For a detailed definition, see "Temperature Dependence" on page 1-728.
- *PHI* is:

  - The **Surface potential, PHI** parameter value, if this parameter has a numerical value.

  -
    $2*kT_{meas} / q*\log(NSUB / n_i)$, if **Surface potential, PHI** is NaN and you specify values for both the **Oxide thickness, TOX** and **Substrate doping, NSUB** parameters.
- *LAMBDA* is the **Channel modulation, LAMBDA** parameter value.
- *GAMMA* is:

  - The **Bulk threshold, GAMMA** parameter value, if this parameter has a numerical value.

  -
    $TOX * \sqrt{2*11.7*\varepsilon_0*q*NSUB} / (3.9*\varepsilon_0)$, if **Bulk threshold, GAMMA** is NaN and you specify values for both the **Oxide thickness, TOX** and **Substrate doping, NSUB** parameters.

- ε_0 is the permittivity of free space, 8.854214871e-12 F/m.
- n_i is the carrier concentration of intrinsic silicon, 1.45e10 cm^{-3}.

The block provides the following relationship between the drain current I_{sd} and the drain-source voltage V_{sd} in inverse mode ($V_{sd} < 0$) after adjusting the applicable model parameters for temperature.

Inverse Mode

Applicable Range of V_{dg} and V_{sd} Values	Corresponding I_{sd} Equation
$V_{dg} - V_{on} \leq 0$	$I_{sd} = 0$
$0 < V_{dg} - V_{on} \leq -V_{sd}$	$I_{sd} = -BETA \left(V_{dg} - V_{on}\right)^2 \left(1 - LAMBDA * V_{sd}\right)/2$
$0 < V_{sd} < V_{dg} - V_{on}$	$I_{sd} = BETA *$ $V_{sd}\left(\left(V_{dg} - V_{on}\right) + V_{sd}/2\right)\left(1 - LAMBDA * V_{sd}\right)$

Where:

- V_{on} is:

  - $$MTYPE * VBI + GAMMA\sqrt{PHI - V_{db}} \text{ if}$$
    $$V_{db} \leq 0$$
    .

  - $$MTYPE * VBI + GAMMA\left(\sqrt{PHI} - \frac{V_{db}}{2\sqrt{PHI}}\right) \text{ if}$$
    $$0 < V_{db} \leq 2 * PHI$$
    .

- $MTYPE*VBI$ if $V_{db} > 2*PHI$.

Level 3 Drain Current Model

The block provides the following model for drain current I_{sd} in normal mode ($V_{sd} \geq 0$) after adjusting the applicable model parameters for temperature.

$$I_{SD} = I_{SD0} * Scale_{VMAX} * Scale_{LChan} * Scale_{INV}$$

Where:

- I_{SD0} is the Basic Drain Current Model on page 1-779.
- $Scale_{VMAX}$ is the Velocity Saturation Scaling on page 1-782.
- $Scale_{LChan}$ is the Channel Length Modulation Scaling on page 1-782.
- $Scale_{INV}$ is the Weak Inversion Scaling on page 1-784.

The blocks uses the same model for drain current in inverse mode ($V_{sd} < 0$), with the following substitutions:

- $V_{sb} - V_{sd}$ for V_{sb}
- $V_{sg} - V_{sd}$ for V_{sd}
- $-V_{sd}$ for V_{sd}

The block provides the following relationship between the drain current I_{sd} and the drain-source voltage V_{ds}:

$$I_{SD0} = BETA * F_{gate} * \left(V_{SGX} - V_{TH} - \frac{1+F_B}{2} * V_{SDX} \right) * V_{SDX}$$

- The block calculates BETA as described in "Level 1 Drain Current Model" on page 1-776.
- The block calculates F_{GATE} using the following equation:

$$F_{gate} = \frac{1}{1 + THETA * \left(V_{sgx} - V_{TH} \right)}$$

- *THETA* is the **Vgs dependence on mobility, THETA** parameter value.

-
$$V_{sgx} = \max(V_{SG}, V_{on})$$

- If you specify a nonzero value for the **Fast surface state density, NFS** parameter, the block calculates V_{on} using the following equation:

$$V_{on} = V_{TH} + x_n V_T$$

Otherwise, $V_{on} = V_{TH}$.

- The block calculates x_n using the following equation:

$$x_n = 1 + \frac{q * NFS}{COX} + \frac{\left(GAMMA * F_s * \sqrt{V_{bulk}} + \frac{F_n * V_{bulk}}{WIDTH} \right)}{2 * V_{bulk}}$$

- The block calculates V_{bulk} as follows:

  - If $V_{SB} \leq 0$, $V_{bulk} = PHI - V_{BS}$.

  - Otherwise, the block calculates V_{bulk} using the following equation:

$$V_{bulk} = \frac{PHI}{\left(1 + \dfrac{V_{SB}}{2 * PHI} \right)^2}$$

- $V_T = kT/q$

- The block calculates V_{TH} using the equation following equation:

$$V_{TH} = V_{BI} - \frac{8.15e^{-22} * ETA}{COX * \left(LENGTH - 2 * LD \right)^3} * V_{SD} \\ + GAMMA * F_s * \sqrt{V_{bulk}} + F_n * V_{bulk}$$

- For information about how the block calculates V_{BI}, see "Temperature Dependence" on page 1-788.

- *ETA* is the **Vds dependence threshold volt, ETA** parameter value.

- $COX = \varepsilon_{ox}/TOX$, where ε_{ox} is the permittivity of the oxide and *TOX* is the **Oxide thickness, TOX** parameter value.
- If you specify a nonzero value for the **Junction depth, XJ** parameter and a value for the **Substrate doping, NSUB** parameter, the block calculates F_s using the following equations:

$$\alpha = \frac{2\varepsilon_{si}}{qNSUB}$$

$$XD = \sqrt{\alpha}$$

$$wc = .0631353 + .8013292 * \frac{XD * \sqrt{V_{bulk}}}{XJ}$$

$$-.01110777 * \left(\frac{XD * \sqrt{V_{bulk}}}{XJ} \right)^2 + \frac{LD}{XJ}$$

$$F_s = 1 - \left(wc * \sqrt{1 - \left(\frac{XD * \sqrt{V_{bulk}}}{XJ + XD * \sqrt{V_{bulk}}} \right)^2} - \frac{LD}{XJ} \right)$$

where ε_{si} is the permittivity of silicon.

Otherwise, $F_s = 1$.

- The block calculates F_B using the following equation:

$$F_B = \frac{GAMMA * F_s}{4 * \sqrt{V_{bulk}}} + F_n$$

- The block calculates F_n using the following equation:

$$F_n = \frac{DELTA * \pi * \varepsilon_{si}}{2 * COX * WIDTH}$$

- *DELTA* is the **Width effect on threshold, DELTA** parameter value.
- V_{SDX} is the lesser of V_{SD} and the saturation voltage, V_{dsat}.

1-781

- If you specify a positive value for the **Max carrier drift velocity, VMAX** parameter, the block calculates V_{dsat} using the following equation:

$$V_{dsat} = \frac{V_{sgx} - V_{TH}}{1 + F_B} + \frac{(LENGTH - 2*LD)*VMAX}{UO*F_{gate}}$$
$$- \sqrt{\left(\frac{V_{sgx} - V_{TH}}{1 + F_B}\right)^2 + \left(\frac{(LENGTH - 2*LD)*VMAX}{UO*F_{gate}}\right)^2}$$

Otherwise, the block calculates V_{dsat} using the following equation:

$$V_{dsat} = \frac{V_{sgx} - V_{TH}}{1 + F_B}$$

If you specify a positive value for the **Max carrier drift velocity, VMAX** parameter, the block calculates $Scale_{VMAX}$ using the following equation:

$$Scale_{VMAX} = \frac{1}{1 + \dfrac{UO*F_{gate}}{(LENGTH - 2*LD)*VMAX}*V_{SDX}}$$

Otherwise, $Scale_{VMAX} = 1$.

The block scales the drain current to account for channel length modulation if the block meets all of the following criteria:

- $V_{SD} > V_{sat}$
- The **Max carrier drift velocity, VMAX** parameter value is zero or α is nonzero.

The block scales the drain current using the following equation:

$$Scale_{LChan} = \frac{1}{1 - \dfrac{\Delta l}{(LENGTH - 2*LD)}}$$

The block uses the following procedure to calculate Δl:

1 The block first calculates the intermediate value Δl_0.

   • If you specify a positive value for the **Max carrier drift velocity, VMAX** parameter, the block computes the intermediate value g_{dsat} as the greater of 1e-12 and the result of the following equation:

   $$I_{SD0}*\left(1-\frac{1}{1+Scale_{g_{dsat}}*V_{SDX}}\right)*Scale_{g_{dsat}}$$

   where:

   $$Scale_{g_{dsat}}=\frac{UO*F_{gate}}{\left(LENGTH-2*LD\right)*VMAX}$$

   Then, the block uses the following equation to calculate the intermediate value Δl_0:

   $$\Delta l_0=\sqrt{\left(\frac{KA*I_{SD}}{2*\left(LENGTH-2*LD\right)*g_{dsat}}\right)^2+KA*\left(V_{SD}-V_{dsat}\right)}$$
   $$-\frac{KA*I_{SD}}{2*\left(LENGTH-2*LD\right)*g_{dsat}}$$

   where KA is the product of the **Mobility modulation, KAPPA** parameter value and α.

   • Otherwise, the block uses the following equation to calculate the intermediate value Δl_0:

   $$\Delta l=\sqrt{KA*\left(V_{SD}-V_{dsat}\right)}$$

2 The block checks for punch through and calculates Δl.

   • If Δl_0 is greater than $(LENGTH\text{-}2*LD)/2$, the block calculates Δl using the following equation:

$$\Delta l = \left(1 - \frac{\left(LENGTH - 2*LD \right)}{4*\Delta l_0} \right) * \left(LENGTH - 2*LD \right)$$

- Otherwise, $\Delta l = \Delta l_0$.

If V_{SG} is less than V_{on}, the block calculates $Scale_{INV}$ using the following equation:

$$Scale_{INV} = e^{\frac{V_{sg} - V_{on}}{x_n * V_T}}$$

Otherwise, $Scale_{INV} = 1$.

Junction Charge Model

The block models the following junction charges:

- Junction Overlap Charges
- Bulk Junction Charges

The block calculates the following junction overlap charges:

- $Q_{SG} = CGSO*WIDTH*V_{sg}$

  Where:

  - Q_{SG} is the gate-source overlap charge.
  - $CGSO$ is the **G-S overlap capacitance, CGSO** parameter value.
  - $WIDTH$ is the **Width of channel, WIDTH** parameter value.
- $Q_{DG} = CGDO*WIDTH*V_{dg}$

  Where:

  - Q_{DG} is the gate-drain overlap charge.
  - $CGDO$ is the **G-D overlap capacitance, CGDO** parameter value.
- $Q_{BG} = CGBO*(LENGTH-2*LD)*V_{bg}$

  Where:

- Q_{BG} is the gate-bulk overlap charge.
- *CGBO* is the **G-B overlap capacitance, CGBO** parameter value.
- *LENGTH* is the **Length of channel, LENGTH** parameter value.
- *LD* is the **Lateral diffusion, LD** parameter value.

The block provides the following relationship between the bulk-drain bottom junction charge Q_{bottom} and the junction voltage V_{db} after adjusting the applicable model parameters for temperature.

Applicable Range of V_{db} Values	Corresponding Q_{bottom} Equation
$V_{db} < FC * PB$	$$Q_{bottom} = \frac{CBD * PB * \left(1 - \left(1 - \frac{V_{db}}{PB}\right)^{1-MJ}\right)}{1 - MJ} \quad \text{if } CBD > 0.$$ $$Q_{bottom} = \frac{CJ * AD * PB * \left(1 - \left(1 - \frac{V_{db}}{PB}\right)^{1-MJ}\right)}{1 - MJ} \quad \text{otherwise.}$$

Applicable Range of V_{db} Values	Corresponding Q_{bottom} Equation
$V_{db} \geq FC * PB$	$Q_{bottom} = CBD *$ $$\left(F1 + \frac{F3*(V_{db} - FC*PB) + \frac{MJ*(V_{db}^2 - (FC*PB)^2)}{2*PB}}{F2} \right)$$ if $CBD >$ 0. $Q_{bottom} = CJ * AD *$ $$\left(F1 + \frac{F3*(V_{db} - FC*PB) + \frac{MJ*(V_{db}^2 - (FC*PB)^2)}{2*PB}}{F2} \right)$$ otherwise.

Where:

- PB is the **Bulk junction potential, PB** parameter value.
- FC is the **Capacitance coefficient FC** parameter value.
- CBD is the **Zero-bias BD capacitance, CBD** parameter value.
- CJ is the **Bottom junction cap per area, CJ** parameter value.
- AD is the **Area of drain, AD** parameter value.
- MJ is the **Bottom grading coefficient, MJ** parameter value.

- $$F1 = \frac{PB*\left(1 - (1 - FC)^{1-MJ}\right)}{1 - MJ}$$

- $$F2 = (1 - FC)^{1+MJ}$$

- $$F3 = 1 - FC*(1+MJ)$$

The block uses the equations in the preceding table to calculate the bulk-source bottom junction charge, with the following substitutions:

- V_{sb} replaces V_{db}.
- *AS* (the **Area of source, AS** parameter value) replaces *AD*.
- *CBS* (the **Zero-bias BS capacitance, CBS** parameter value) replaces *CBD*.

The block provides the following relationship between the bulk-drain sidewall junction charge $Q_{sidewall}$ and the junction voltage V_{db} after adjusting the applicable model parameters for temperature.

Applicable Range of V_{db} Values	Corresponding $Q_{sidewall}$ Equation
$V_{db} < FC*PB$	$$Q_{sidewall} = \frac{CJSW*PD*PB*\left(1-\left(1-\frac{V_{db}}{PB}\right)^{1-MJSW}\right)}{1-MJSW}$$
$V_{db} \geq FC*PB$	$$Q_{sidewall} = CJSW*PD*\left(F1 + \frac{F3*(V_{db}-FC*PB) + \frac{MJSW*\left(V_{db}^2 - (FC*PB)^2\right)}{2*PB}}{F2}\right)$$

Where:

- *CJSW* is the **Side jct cap/area of jct perimeter, CJSW** parameter value.
- *PD* is the **Perimeter of drain, AD** parameter value.
- *MJSW* is the **Side grading coefficient, MJSW** parameter value.

-
$$F1 = \frac{PB * \left(1 - \left(1 - FC\right)^{1-MJSW}\right)}{1 - MJSW}$$

-
$$F2 = \left(1 - FC\right)^{1+MJSW}$$

-
$$F3 = 1 - FC * \left(1 + MJSW\right)$$

The block uses the equations in the preceding table to calculate the bulk-source sidewall junction charge and the sidewall junction voltage, with the following substitutions:

- V_{sb} replaces V_{db}.

- *PS* (the **Perimeter of source, PS** parameter value) replaces *PD*.

Temperature Dependence

Several transistor parameters depend on temperature. There are two ways to specify the transistor temperature:

- When you select `Device temperature` for the **Model temperature dependence using** parameter, the transistor temperature is

$$T = T_C + T_O$$

 where:

  - T_C is the **Circuit temperature** parameter value from the SPICE Environment Parameters block. If this block doesn't exist in the circuit, T_C is the default value of this parameter.

  - T_O is the **Offset local circuit temperature, TOFFSET** parameter value.

- When you select `Fixed temperature` for the **Model temperature dependence using** parameter, the transistor temperature is the **Fixed circuit temperature, TFIXED** parameter value.

The block provides the following relationship between the transconductance *KP* and the transistor temperature *T*:

$$KP(T) = \frac{KP}{\left(T\Big/T_{meas}\right)^{3/2}}$$

where:

- *KP* is the **Transconductance, KP** parameter value.
- T_{meas} is the **Parameter extraction temperature, TMEAS** parameter value.

The block provides the following relationship between the surface potential *PHI* and the transistor temperature *T*:

$$\begin{aligned}
PHI(T) = \frac{T}{T_{meas}} &\left(PHI + \frac{kT_{meas}}{q} \left(\log\left(\frac{T_{meas}}{300.15}\right)^3 + \frac{q}{k}\left(\frac{1.115}{300.15} - \frac{EG_{T_{meas}}}{T_{meas}}\right)\right)\right) \\
&- \frac{kT}{q}\left(\log\left(\frac{T}{300.15}\right)^3 + \frac{q}{k}\left(\frac{1.115}{300.15} - \frac{EG_T}{T}\right)\right)
\end{aligned}$$

where:

-
$$EG_{T_{meas}} = 1.16eV - \left(7.02e\text{-}4 * T_{meas}^{2}\right)/\left(T_{meas}+1108\right)$$

-
$$EG_T = 1.16eV - \left(7.02e\text{-}4 * T^{2}\right)/\left(T+1108\right)$$

The block provides the following relationship between the built-in voltage *VBI* and the transistor temperature *T*:

$$\begin{aligned}
VBI(T) = VTO + MTYPE * &\left(\frac{PHI(T) - PHI}{2} - GAMMA\sqrt{PHI}\right) \\
&+ \frac{EG_{T_{meas}} - EG_T}{2}
\end{aligned}$$

where:

- *VTO* is:

- The **Threshold voltage, VTO** parameter value, if this parameter has a numerical value.

- $$\Phi - 3.25 + EG_{T_{meas}} \big/ 2 + MTYPE * PHI / 2 - NSS * q * TOX / (3.9 * \varepsilon_0)$$

  $+ MTYPE * (GAMMA * \sqrt{PHI} + PHI)$, if **Threshold voltage, VTO** is NaN and you specify values for both the **Oxide thickness, TOX** and **Substrate doping, NSUB** parameters.

- Φ is:

  - 3.2, if TPG (the **Gate type?, TPG** parameter value) is 0.

  - $$3.25 + EG_{T_{meas}} \big/ 2 - MTYPE * TPG * EG_{T_{meas}} \big/ 2$$, otherwise.

- $GAMMA$ is:

  - The **Bulk threshold, GAMMA** parameter value, if this parameter has a numerical value.

  - $$TOX * \sqrt{2 * 11.7 * \varepsilon_0 * q * NSUB} \big/ (3.9 * \varepsilon_0)$$, if **Bulk threshold, GAMMA** is NaN and you specify values for both the **Oxide thickness, TOX** and **Substrate doping, NSUB** parameters.

The block provides the following relationship between the bulk saturation current IS and the transistor temperature T:

$$IS(T) = IS * e^{\frac{-qEG_T}{ND*kT} + \frac{qEG_{T_{meas}}}{ND*kT_{meas}}}$$

where:

- ND is the **Emission coefficient, N** parameter value.
- IS is the **Bulk saturation current, IS** parameter value.

The block provides the following relationship between the bulk junction saturation current density JS and the transistor temperature T:

$$JS(T) = JS * e^{\frac{-qEG_T}{ND*kT} + \frac{qEG_{T_{meas}}}{ND*kT_{meas}}}$$

where:

- *JS* is the **Bulk jct sat current density, JS** parameter value.

The block provides the following relationship between the bulk junction potential *PB* and the transistor temperature *T*:

$$PB(T) = \frac{PB + \frac{kT_{meas}}{q}\left(\log\left(\frac{T_{meas}}{300.15}\right)^3 + \frac{q}{k}\left(\frac{1.115}{300.15} - \frac{EG_{T_{meas}}}{T}\right)\right)}{T_{meas}\Big/T}$$

$$-\frac{kT}{q}\left(\log\left(\frac{T}{300.15}\right)^3 + \frac{q}{k}\left(\frac{1.115}{300.15} - \frac{EG_T}{T}\right)\right)$$

where:

- *PB* is the **Bulk junction potential, PB** parameter value.

The block provides the following relationship between the bulk-drain junction capacitance *CBD* and the transistor temperature *T*:

$$CBD(T) = CBD\frac{pbo + MJ * \left(4 * 10^4 * (T - 300.15) * pbo - (PB(T) - pbo)\right)}{pbo + MJ * \left(4 * 10^4 * (T_{meas} - 300.15) * pbo - (PB - pbo)\right)}$$

where:

- *CBD* is the **Zero-bias BD capacitance, CBD** parameter value.
- *MJ* is the **Bottom grading coefficient, MJ** parameter value.
-

$$pbo = \frac{PB + \frac{kT_{meas}}{q}\left(\log\left(\frac{T_{meas}}{300.15}\right)^3 + \frac{q}{k}\left(\frac{1.115}{300.15} - \frac{EG_{T_{meas}}}{T}\right)\right)}{T_{meas}\Big/300.15}$$

The block uses the *CBD(T)* equation to calculate:

- The bulk-source junction capacitance by substituting *CBS* (the **Zero-bias BS capacitance, CBS** parameter value) for *CBD*.

- The bottom junction capacitance by substituting *CJ* (the **Bottom junction cap per area, CJ** parameter value) for *CBD*.

The block provides the following relationship between the sidewall junction capacitance *CJSW* and the transistor temperature *T*:

$$CJSW(T) = CJSW \frac{pbo + MJSW * \left(4*10^4 * (T - 300.15) * pbo - (PB(T) - pbo)\right)}{pbo + MJSW * \left(4*10^4 * (T_{meas} - 300.15) * pbo - (PB - pbo)\right)}$$

where:

- *MJSW* is the **Side grading coefficient, MJSW** parameter value.

Basic Assumptions and Limitations

The model is based on the following assumptions:

- The PMOS block does not support noise analysis.
- The PMOS block applies initial conditions across junction capacitors and not across the block ports.

Parameters

Model Selection Tab

MOS model

Select one of the following MOSFET model options:

- `Level 1 MOS` — Use the "Level 1 Drain Current Model" on page 1-776. This is the default option.
- `Level 3 MOS` — Use the "Level 3 Drain Current Model" on page 1-779.

Dimensions Tab

Device area factor, AREA

The transistor area. This value multiplies the following parameter values:

- **Transconductance, KP**
- **Bulk saturation current, IS**
- **Bulk jct sat current density, JS**
- **Zero-bias BD capacitance, CBD**
- **Zero-bias BS capacitance, CBS**
- **G-S overlap capacitance, CGSO**
- **G-D overlap capacitance, CGDO**
- **G-B overlap capacitance, CGBO**
- **Bottom junction cap per area CJ**
- **Side jct cap/area of jct perimeter CJSW**

It divides the following parameter values:

- **Drain resistance, RD**
- **Source resistance, RS**
- **Sheet resistance, RSH**

The default value is 1. The value must be greater than 0.

Number of parallel devices, SCALE

The number of parallel MOS instances for this device. This parameter multiplies the output current and device charge. The default value is 1. The value must be greater than 0.

Length of channel, LENGTH

Length of the channel between the source and drain. The default value is 1e-04 m.

Width of channel, WIDTH

Width of the channel between the source and drain. The default value is 1e-04 m.

Area of drain, AD

Area of the transistor drain diffusion. The default value is 0 m^2. The value must be greater than or equal to 0.

Area of source, AS

Area of the transistor source diffusion. The default value is 0 m^2. The value must be greater than or equal to 0.

Perimeter of drain, PD

Perimeter of the transistor drain diffusion. The default value is 0 m.

Perimeter of source, PS

Perimeter of the transistor source diffusion. The default value is 0 m.

Resistors Tab

Drain resistance, RD

The transistor drain ohmic resistance. The default value is 0.01 Ω. If you set this parameter to NaN Ω, this value means the parameter is unspecified, so the block calculates the drain resistance as described in "Resistance Calculations" on page 1-773. The value must be equal to *0* or greater than or equal to *Rmin*. *Rmin* is a built-in model constant whose value is 1e-12.

Source resistance, RS

The transistor source ohmic resistance. The default value is 1e-4 Ω. If you set this parameter to NaN Ω, this value means the parameter is unspecified, so the block calculates the drain resistance as described in "Resistance Calculations" on page 1-773. The value must be equal to 0 or greater than or equal to *Rmin*. *Rmin* is a built-in model constant whose value is 1e-12.

Sheet resistance, RSH

Resistance per square of the transistor source and drain. The default value is Nan Ω. This value means the parameter is unspecified. The block only uses this parameter value if you do not specify one or both of the **Drain resistance, RD** and **Source resistance, RS** parameter values, as described in "Resistance Calculations" on page 1-773. The value must be greater than or equal to 0.

Number of drain squares, NRD

Number of squares of resistance that make up the transistor drain diffusion. The default value is 1 . The value must be greater than or equal to 0. The block only uses this parameter value if you do not specify one or both of the **Drain resistance, RD** and **Source resistance, RS** parameter values, as described in "Resistance Calculations" on page 1-773.

Number of source squares, NRS

Number of squares of resistance that make up the transistor source diffusion. The default value is 1 . The value must be greater than or equal to 0. The block only uses this parameter value if you do not specify one or both of the **Drain resistance, RD** and **Source resistance, RS** parameter values, as described in "Resistance Calculations" on page 1-773.

DC Currents Tab

Threshold voltage, VTO

The gate-source voltage above which the transistor produces a nonzero drain current. The default value is 0 V. If you assign this parameter a value of NaN, the block calculates the value from the specified values of the **Oxide thickness, TOX** and **Substrate doping, NSUB** parameters. For more information about this calculation, see "Temperature Dependence" on page 1-728.

Transconductance, KP

The derivative of drain current with respect to gate voltage. The default value is 2e-05 A/V^2. The value must be greater than or equal to 0. If you assign this parameter a value of NaN, the block calculates the value from the specified values of the **Oxide thickness, TOX** and **Substrate doping, NSUB** parameters. For more information about this calculation, see "Level 1 Drain Current Model" on page 1-716 or "Level 3 Drain Current Model" on page 1-779 as appropriate for the selected value of the **MOS model** parameter.

Bulk threshold, GAMMA

Body effect parameter, which relates the threshold voltage, VTH, to the body bias, VBS, as described in "Level 1 Drain Current Model" on page 1-716. The default value

is 0 $\sqrt{V}$. The value must be greater than or equal to 0. If you assign this parameter a value of NaN, the block calculates the value from the specified values of the **Oxide thickness, TOX** and **Substrate doping, NSUB** parameters. For more information about this calculation, see "Level 1 Drain Current Model" on page 1-716 or "Level 3 Drain Current Model" on page 1-779 as appropriate for the selected value of the **MOS model** parameter.

Surface potential, PHI

Twice the voltage at which the surface electron concentration becomes equal to the intrinsic concentration and the device transitions between depletion and inversion conditions. The default value is 0.6 V. The value must be greater than or equal to 0. If you assign this parameter a value of NaN, the block calculates the value from the

specified values of the **Oxide thickness, TOX** and **Substrate doping, NSUB** parameters. For more information about this calculation, see "Level 1 Drain Current Model" on page 1-716 or "Level 3 Drain Current Model" on page 1-779 as appropriate for the selected value of the **MOS model** parameter.

Channel modulation, LAMBDA

The channel-length modulation. This parameter is only visible when you select `Level 1 MOS` for the **MOS model** parameter. The default value is `0` 1/V.

Bulk saturation current, IS

The magnitude of the current that the junction approaches asymptotically for very large reverse bias levels. The default value is `1e-14` A. The value must be greater than or equal to `0`.

Bulk jct sat current density, JS

The magnitude of the current per unit area that the junction approaches asymptotically for very large reverse bias levels. The default value is `0` A/m^2. The value must be greater than or equal to `0`.

Emission coefficient, N

The transistor emission coefficient or ideality factor. The default value is `1`. The value must be greater than `0`.

Width effect on threshold, DELTA

The factor that controls the effect of transistor width on threshold voltage. This parameter is only visible when you select `Level 3 MOS` for the **MOS model** parameter. The default value is `0`.

Max carrier drift velocity, VMAX

The maximum drift velocity of the carriers. This parameter is only visible when you select `Level 3 MOS` for the **MOS model** parameter. The default value is `0` m/s.

Fast surface state density, NFS

The fast surface state density adjusts the drain current for the mobility reduction caused by the gate voltage. This parameter is only visible when you select `Level 3 MOS` for the **MOS model** parameter. The default value is `0` 1/cm^2.

Vds dependence threshold volt, ETA

The coefficient that controls how the threshold voltage depends on the drain-source voltage in the drain current calculation. This parameter is only visible when you select `Level 3 MOS` for the **MOS model** parameter. The default value is `0`.

Vgs dependence on mobility, THETA

The coefficient that controls how the mobility affects the gate voltage in the drain current calculation. This parameter is only visible when you select `Level 3 MOS` for the **MOS model** parameter. The default value is `0` 1/V.

Mobility modulation, KAPPA

The coefficient that controls how the mobility affects the channel length in the drain current calculation. This parameter is only visible when you select `Level 3 MOS` for the **MOS model** parameter. The default value is `0.2`.

C-V Tab

Model junction capacitance

Select one of the following options for modeling the junction capacitance:

- `No` — Do not include junction capacitance in the model. This is the default option.
- `Yes` — Specify zero-bias junction capacitance, junction potential, grading coefficient, forward-bias depletion and capacitance coefficient.

G-S overlap capacitance, CGSO

Gate-source capacitance due to the diffusion that occurs when the device operates in depletion mode. This parameter is only visible when you select `Yes` for the **Model junction capacitance** parameter. The default value is `0` F/m. The value must be equal to `0` or greater than or equal to *Cmin*. *Cmin* is a built-in model constant whose value is `1e-18`.

G-D overlap capacitance, CGDO

Gate-drain capacitance due to the diffusion that occurs when the device operates in depletion mode. This parameter is only visible when you select `Yes` for the **Model junction capacitance** parameter. The default value is `0` F/m. The value must be equal to `0` or greater than or equal to *Cmin*. *Cmin* is a built-in model constant whose value is `1e-18`.

G-B overlap capacitance, CGBO

Gate-base capacitance due to the diffusion that occurs when the device operates in depletion mode. This parameter is only visible when you select `Yes` for the **Model junction capacitance** parameter. The default value is `0` F/m. The value must be equal to `0` or greater than or equal to *Cmin*. *Cmin* is a built-in model constant whose value is `1e-18`.

Zero-bias BD capacitance, CBD

The value of the capacitance placed between the base and the drain. This parameter is only visible when you select `Yes` for the **Model junction capacitance** parameter. The default value is `0` F. The value must be equal to `0` or greater than or equal to *Cmin*. *Cmin* is a built-in model constant whose value is `1e-18`.

Zero-bias BS capacitance, CBS

The value of the capacitance placed between the base and the source. This parameter is only visible when you select `Yes` for the **Model junction capacitance** parameter. The default value is `0` F. The value must be equal to `0` or greater than or equal to *Cmin*. *Cmin* is a built-in model constant whose value is `1e-18`.

Bottom junction cap per area CJ

Zero-bias bulk junction bottom capacitance per junction area. This parameter is only visible when you select `Yes` for the **Model junction capacitance** parameter. The default value is `0` F/m^2. The value must be equal to `0` or greater than or equal to *Cmin*. *Cmin* is a built-in model constant whose value is `1e-18`.

Bottom grading coefficient, MJ

The transistor bottom grading coefficient. This parameter is only visible when you select `Yes` for the **Model junction capacitance** parameter. The default value is `0.5`. The value must be equal to `0` or less than *MGmax*. *MGmax* is a built-in model constant whose value is `0.9`.

Side jct cap/area of jct perimeter CJSW

Zero-bias bulk junction sidewall capacitance per junction perimeter. This parameter is only visible when you select `Yes` for the **Model junction capacitance** parameter. The default value is `0` F/m. The value must be equal to `0` or greater than or equal to *Cmin*. *Cmin* is a built-in model constant whose value is `1e-18`.

Side grading coefficient, MJSW

The transistor sidewall grading coefficient. This parameter is only visible when you select `Yes` for the **Model junction capacitance** parameter. The default value is `0.5`. The value must be equal to `0` or less than *MGmax*. *MGmax* is a built-in model constant whose value is `0.9`.

Bulk junction potential, PB

The potential across the bulk junction. This parameter is only visible when you select `Yes` for the **Model junction capacitance** parameter. The default value is `0.8` V. The value must be equal to `0` or greater than or equal to *VJmin*. *VJmin* is a built-in model constant whose value is `0.01`.

Capacitance coefficient FC

The fitting coefficient that quantifies the decrease of the depletion capacitance with applied voltage. This parameter is only visible when you select `Yes` for the **Model junction capacitance** parameter. The default value is `0.5`. The value must be equal to `0` or less than or equal to *FCmax*. *FCmax* is a built-in model constant whose value is `0.95`.

Specify initial condition

Select one of the following options for specifying an initial condition:

- `No` — Do not specify an initial condition for the model. This is the default option.
- `Yes` — Specify the initial diode voltage.

Note The PMOS block applies the initial diode voltage across the junction capacitors and not across the ports.

Initial condition voltage ICVDS

Drain-source voltage at the start of the simulation. This parameter is only visible when you select `Yes` for the **Model junction capacitance** and `Yes` for the **Specify initial condition** parameter. The default value is `0` V.

Initial condition voltage ICVGS

Gate-source voltage at the start of the simulation. This parameter is only visible when you select `Yes` for the **Model junction capacitance** and `Yes` for the **Specify initial condition** parameter. The default value is `0` V.

Initial condition voltage ICVBS

Bulk-source voltage at the start of the simulation. This parameter is only visible when you select `Yes` for the **Model junction capacitance** and `Yes` for the **Specify initial condition** parameter. The default value is `0` V.

Process Tab

Oxide thickness, TOX

Thickness of the gate oxide. The default value is `NaN` m. The value must be greater than or equal to `0`.

Note When you select `Level 3 MOS` for the **MOS model** parameter, the block uses a value of `1e-7` rather than `NaN` by default.

Lateral diffusion, LD

Length of lateral diffusion. The default value is 0 m.

Substrate doping, NSUB

Substrate doping. The default value is NaN $1/cm^3$. The value must be greater than or equal to 1.45e10 (the carrier concentration of intrinsic silicon).

Surface state density, NSS

Substrate doping. The default value is 0 $1/cm^2$.

Surface mobility, U0

Zero-bias surface mobility coefficient. The default value is 600 $cm^2/V/s$.

Junction depth, XJ

Junction depth. This parameter is only visible when you select Level 3 MOS for the **MOS model** parameter. The default value is 0 m.

Gate type?,TPG

Select one of the following MOSFET gate materials (as compared to the substrate):

- Opposite of substrate — The gate material is the opposite of the substrate. This means that TPG = 1 in the device equations. This is the default option.
- Same as substrate — The gate material is the same as the substrate. This means that TPG = –1 in the device equations.

- Aluminum — The gate material is aluminum. This means that TPG = 0 in the device equations.

Temperature Tab

Model temperature dependence using

Select one of the following options for modeling the diode temperature dependence:

- Device temperature — Use the device temperature, which is the **Circuit temperature** value plus the **Offset local circuit temperature, TOFFSET** value. The **Circuit temperature** value comes from the SPICE Environment Parameters block, if one exists in the circuit. Otherwise, it comes from the default value for this block.
- Fixed temperature — Use a temperature that is independent of the circuit temperature to model temperature dependence.

Offset local circuit temperature, TOFFSET

The amount by which the transistor temperature differs from the circuit temperature. This parameter is only visible when you select `Device temperature` for the **Model temperature dependence using** parameter. The default value is `0` K.

Fixed circuit temperature, TFIXED

The temperature at which to simulate the transistor. This parameter is only visible when you select `Fixed temperature` for the **Model temperature dependence using** parameter. The default value is `300.15` K. The value must be greater than `0`.

Parameter extraction temperature, TMEAS

The temperature at which the transistor parameters were measured. The default value is `300.15` K. The value must be greater than `0`.

Ports

The block has the following ports:

G

Electrical conserving port associated with the transistor gate terminal.

D

Electrical conserving port associated with the transistor drain terminal.

S

Electrical conserving port associated with the transistor source terminal.

B

Electrical conserving port associated with the transistor bulk terminal.

References

[1] G. Massobrio and P. Antognetti. *Semiconductor Device Modeling with SPICE*. 2nd Edition, McGraw-Hill, 1993. Chapter 3.

See Also

SPICE NMOS

SPICE PNP

Model Gummel-Poon PNP Transistor

Library

SPICE-Compatible Components/Semiconductor Devices

Description

The PNP block represents a SPICE-compatible four-terminal Gummel-Poon PNP transistor. The substrate port is connected to the transistor body using a capacitor, so these devices are equivalent to a three-terminal transistor when you connect the substrate port to any other port and use the default value of zero for the **C-S junction capacitance, CJS** parameter.

The PNP block model includes the following components:

- "Current-Voltage and Base Charge Model" on page 1-803
- "Base Resistance Model" on page 1-806
- "Transit Charge Modulation Model" on page 1-807
- "Junction Charge Model" on page 1-808
- "Temperature Dependence" on page 1-810

Current-Voltage and Base Charge Model

The current-voltage relationships and base charge relationships for the transistor are calculated after adjusting the applicable model parameters for temperature as described in the following sections:

- "Emitter-Base and Collector-Base Junction Currents" on page 1-804

The base-emitter junction current is calculated using the following equations:

- When $V_{EB} > 80 * V_{TF}$:

$$I_{ebf} = IS * \left(\left(\frac{V_{EB}}{V_{TF}} - 79 \right) * e^{80} - 1 \right) + G_{\min} * V_{EB}$$

$$I_{ebe} = ISE * \left((V_{EB} - 80 * V_{TF} + V_{TE}) * \frac{e^{(80 * V_{TF} / V_{TE})}}{V_{TE}} - 1 \right)$$

- When $V_{EB} \leq 80 * V_{TF}$

$$I_{ebf} = IS * \left(e^{(V_{EB} / V_{TF})} - 1 \right) + G_{\min} * V_{EB}$$

$$I_{ebe} = ISE * \left(e^{(V_{EB} / V_{TE})} - 1 \right)$$

The base-collector junction current is calculated using the following equations:

- When $V_{CB} > 80 * V_{TR}$:

$$I_{cbr} = IS * \left(\left(\frac{V_{CB}}{V_{TR}} - 79 \right) * e^{80} - 1 \right) + G_{\min} * V_{CB}$$

$$I_{cbc} = ISC * \left((V_{CB} - 80 * V_{TR} + V_{TC}) * \frac{e^{(80 * V_{TR} / V_{TC})}}{V_{TC}} - 1 \right)$$

- When $V_{CB} \leq 80 * V_{TR}$

$$I_{cbr} = IS * \left(e^{(V_{CB}/V_{TR})} - 1 \right) + G_{min} * V_{CB}$$

$$I_{cbc} = ISC * \left(e^{(V_{CB}/V_{TC})} - 1 \right)$$

In the preceding equations:

- V_{EB} is the emitter-base voltage and V_{CB} is the collector-base voltage.

-

  $V_{TE} = NE * k * T / q, V_{TC} = NC * k * T / q, V_{TF} = NF * k * T / q$, and

  $V_{TR} = NR * k * T / q.$
- ISC and ISE are the **B-C leakage current, ISC** and **B-E leakage current, ISE** parameter values, respectively.

- NE, NC, NF, and NR are the **B-E emission coefficient, NE**, **B-C emission coefficient, NC**, **Forward emission coefficient, NF** and **Reverse emission coefficient, NR** parameter values, respectively.

- q is the elementary charge on an electron.

- k is the Boltzmann constant.

- T is the transistor temperature:

    - If you select `Device temperature` for the **Model temperature dependence using** parameter, T is the sum of the **Circuit temperature** value plus the **Offset local circuit temperature, TOFFSET** parameter value. The **Circuit temperature** value comes from the SPICE Environment Parameters block, if one exists in the circuit. Otherwise, it comes from the default value for this block.

    - If you select `Fixed temperature` for the **Model temperature dependence using** parameter, T is the **Fixed circuit temperature, TFIXED** parameter value.

- G_{min} is the minimum conductance. By default, G_{min} matches the **Minimum conductance GMIN** parameter of the SPICE Environment Parameters block, whose default value is 1e-12. To change G_{min}, add a SPICE Environment Parameters block to your model and set the **Minimum conductance GMIN** parameter to the desired value.

The terminal currents, I_B and I_C are the base and collector currents, defined as positive into the device. They are calculated as:

$$I_B = -\left(\frac{I_{ebf}}{BF} + I_{ebe} + \frac{I_{cbr}}{BR} + I_{cbc} \right)$$

$$I_C = -\left(\frac{I_{ebf} - I_{cbr}}{q_b} - \frac{I_{cbr}}{BR} - I_{cbc} \right)$$

where *BF* and *BR* are the **Forward beta, BF** and **Reverse beta, BR** parameter values, respectively.

The base charge, q_b, is calculated using the following equations:

$$q_b = \frac{q_1}{2}\left(1 + \sqrt{0.5*\left(\sqrt{(1+4*q_2 \ - \ eps)^2 \ + \ eps^2} \ + \ 1 + 4*q_2 - eps\right) + eps}\right)$$

$$q_1 = \left(1 - \frac{V_{CB}}{VAF} - \frac{V_{EB}}{VAR}\right)^{-1}$$

$$q_2 = \frac{I_{ebf}}{IKF} + \frac{I_{cbr}}{IKR}$$

where

- *VAF* and *VAR* are the **Forward Early voltage, VAF** and **Reverse Early voltage, VAR** parameters, respectively.
- *IKF* and *IKR* are the **Forward knee current, IKF** and **Reverse knee current, IKR** parameter values, respectively.
- *eps* is 1e-4.

Base Resistance Model

The block models base resistance in one of two ways:

- If you use the default value of infinity for the **Half base resistance cur, IRB** parameter, the PNP block calculates the base resistance r_{bb} as

$$r_{bb} \ = \ RBM + \frac{RB - RBM}{q_b}$$

where:

- *RBM* is the **Minimum base resistance, RBM** parameter value.
- *RB* is the **Zero-bias base resistance, RB** parameter value.

- If you specify a finite value for the **Half base resistance cur, IRB** parameter, the PNP block calculates the base resistance r_{bb} as

$$r_{bb} \ = \ RBM + 3*(RB - RBM)*\left(\frac{\tan \ z \ - z}{z * \tan^2 z} \right)$$

where:

$$z = \frac{\sqrt{1 + 144 I_B / (\pi^2 IRB)} - 1}{(24/\pi^2)\sqrt{(I_B / IRB)}}$$

Transit Charge Modulation Model

If you specify nonzero values for the **Coefficient of TF, XTF** parameter, the block models transit charge modulation by scaling the **Forward transit time, TF** parameter value as follows:

$$TF_{\text{mod}} = \frac{TF*\left[1 + XTF * e^{V_{CB}/(1.44 V_{TF})} \left(\frac{I_{EB}}{I_{EB} + ITF} \right)^2 \right]}{q_b}$$

where *ITF* is the **Coefficient of TF, ITF** parameter value.

Junction Charge Model

The PNP block lets you model junction charge. The collector-base charge Q_{cb} and the emitter-base charge Q_{eb} depend on an intermediate value, Q_{dep} as follows, after adjusting the applicable model parameters for temperature:

- For the internal base-emitter junctions:

$$Q_{eb} = TF_{mod} * I_{eb} + Q_{dep}$$

- For the internal base-collector junctions:

$$Q_{cb} = TR * I_{cb} + XCJC * Q_{dep}$$

- For the external base-collector junctions:

$$Q_{cb_{ext}} = (1 - XCJC) * Q_{dep}$$

Q_{dep} depends on the junction voltage, V_{jct} (V_{EB} for the emitter-base junction and V_{CB} for the collector-base junction) as follows.

Applicable Range of V_{jct} Values	Corresponding Q_{dep} Equation
$V_{jct} < FC * VJ$	$$Q_{dep} = C_{jct} * VJ * \frac{1 - \left(1 - V_{jct} / VJ\right)^{(1-MJ)}}{1 - MJ}$$
$V_{jct} \geq FC * VJ$	$$Q_{dep} = C_{jct} * \left[F1 + \frac{F3 * \left(V_{jct} - FC * VJ\right) + \frac{MJ * \left[V_{jct}^2 - \left(FC * VJ\right)^2\right]}{2 * VJ}}{F2} \right]$$

Where:

- *FC* is the **Capacitance coefficient FC** parameter value.

- *VJ* is:

  - The **B-E built-in potential, VJE** parameter value for the emitter-base junction.
  - The **B-C built-in potential, VJC** parameter value for the collector-base junction.

- *MJ* is:

  - The **B-E exponential factor, MJE** parameter value for the emitter-base junction.
  - The **B-C exponential factor, MJC** parameter value for the collector-base junction.

- C_{jct} is:

  - The **B-E depletion capacitance, CJE** parameter value for the emitter-base junction.
  - The **B-C depletion capacitance, CJC** parameter value for the collector-base junction.

-
$$F1 = VJ * \left(1 - \left(1 - FC\right)^{(1-MJ)}\right) \Big/ \left(1 - MJ\right)$$

-
$$F2 = \left(1 - FC\right)^{(1+MJ)}$$

-
$$F3 = 1 - FC * \left(1 + MJ\right)$$

The collector-substrate charge Q_{sc} depends on the collector-substrate voltage V_{sc} as follows, after adjusting the applicable model parameters for temperature.

Applicable Range of V_{sc} Values	Corresponding Q_{sc} Equation
$V_{sc} < 0$	$Q_{sc} = CJS * VJS * \left(\dfrac{1 - \left(1 - V_{sc} / VJS\right)^{(1-MJS)}}{1 - MJS} \right)$
$V_{sc} \geq 0$	$Q_{sc} = CJS * \left(1 + MJS * V_{sc} / (2 * VJS)\right) * V_{sc}$

where:

- *CJS* is the **C-S junction capacitance, CJS** parameter value.

- *VJS* is the **Substrate built-in potential, VJS** parameter value.
- *MJS* is the **Substrate exponential factor, MJS** parameter value.

Temperature Dependence

Several transistor parameters depend on temperature. There are two ways to specify the transistor temperature:

- When you select `Device temperature` for the **Model temperature dependence using** parameter, the transistor temperature is

  $$T = T_C + T_O$$

  where:

  - T_C is the **Circuit temperature** parameter value from the SPICE Environment Parameters block. If this block doesn't exist in the circuit, T_C is the default value of this parameter.
  - T_O is the **Offset local circuit temperature, TOFFSET** parameter value.

- When you select `Fixed temperature` for the **Model temperature dependence using** parameter, the transistor temperature is the **Fixed circuit temperature, TFIXED** parameter value.

The block provides the following relationship between the saturation current *IS* and the transistor temperature *T*:

$$IS(T) = IS * (T/T_{meas})^{XTI} * e^{\left(\frac{T}{T_{meas}} - 1\right) * \frac{EG}{V_t}}$$

where:

- *IS* is the **Transport saturation current, IS** parameter value.
- T_{meas} is the **Parameter extraction temperature, TMEAS** parameter value.
- *XTI* is the **Temperature exponent for IS, XTI** parameter value.
- *EG* is the **Energy gap, EG** parameter value.
- $V_t = kT/q$.

The block provides the following relationship between the base-emitter junction potential *VJE* and the transistor temperature *T*:

$$VJE(T) = VJE * \left(\frac{T}{T_{meas}} \right) - \frac{3 * k * T}{q} * \log \left(\frac{T}{T_{meas}} \right) - \left(\frac{T}{T_{meas}} \right) * EG_{T_{meas}} + EG_T$$

where:

- *VJE* is the **B-E built-in potential, VJE** parameter value.

-
$$EG_{T_{meas}} = 1.16eV - \left(7.02e - 4 * T_{meas}^2 \right) / \left(T_{meas} + 1108 \right)$$

-
$$EG_T = 1.16eV - \left(7.02e - 4 * T^2 \right) / \left(T + 1108 \right)$$

The block uses the *VJE(T)* equation to calculate the base-collector junction potential by substituting *VJC* (the **B-C built-in potential, VJC** parameter value) for *VJE*.

The block provides the following relationship between the base-emitter junction capacitance *CJE* and the transistor temperature *T*:

$$CJE(T) = CJE * \left[1 + MJE * \left(400e - 6 * \left(T - T_{meas} \right) - \frac{VJE(T) - VJE}{VJE} \right) \right]$$

where:

- *CJE* is the **B-E depletion capacitance, CJE** parameter value.
- *MJE* is the **B-E exponential factor, MJE** parameter value.

The block uses this equation to calculate the base-collector junction capacitance by substituting *CJC* (the **B-C depletion capacitance, CJC** parameter value) for *CJE* and *MJC* (the **B-C exponential factor, MJC** parameter value) for *MJE*.

The block provides the following relationship between the forward and reverse beta and the transistor temperature *T*:

$$\beta(T) = \beta * \left(\frac{T}{T_{meas}} \right)^{XTB}$$

where:

- β is the **Forward beta, BF** or **Reverse beta, BR** parameter value.
- *XTB* is the **Beta temperature exponent, XTB** parameter value.

The block provides the following relationship between the base-emitter leakage current *ISE* and the transistor temperature *T*:

$$ISE(T) = ISE * \left(\frac{T}{T_{meas}} \right)^{-XTB} * \left(\frac{IS(T)}{IS} \right)^{1/NE}$$

where:

- *ISE* is the **B-E leakage current, ISE** parameter value.
- *NE* is the **B-E emission coefficient, NE** parameter value.

The block uses this equation to calculate the base-collector leakage current by substituting *ISC* (the **B-C leakage current, ISC** parameter value) for *ISE* and *NC* (the **B-C emission coefficient, NC** parameter value) for *NE*.

Basic Assumptions and Limitations

The model is based on the following assumptions:

- The PNP block does not support noise analysis.
- The PNP block applies initial conditions across junction capacitors and not across the block ports.

Parameters

Main Tab

Device area, AREA

The transistor area. This value multiplies the following parameter values:

- **Transport saturation current, IS**
- **Forward knee current, IKF**
- **B-E leakage current, ISE**
- **Reverse knee current, IKR**
- **B-C leakage current, ISC**
- **Half base resistance cur, IRB**
- **B-E depletion capacitance, CJE**
- **Coefficient of TF, ITF**
- **B-C depletion capacitance, CJC**
- **C-S junction capacitance, CJS**

It divides the following parameter values:

- **Zero-bias base resistance, RB**
- **Minimum base resistance, RBM**
- **Emitter resistance, RE**
- **Collector resistance, RC**

The default value is 1 m^2. The value must be greater than 0.

Number of parallel devices, SCALE

The number of parallel transistors the block represents. This value multiplies the output current and device charges. The default value is 1. The value must be greater than 0.

Forward Gain Tab

Transport saturation current, IS

The magnitude of the current at which the transistor saturates. The default value is $1e\text{-}16 \text{ A/m}^2$. The value must be greater than or equal to 0.

Forward beta, BF

The ideal maximum reverse beta. The default value is 100. The value must be greater than 0.

Forward emission coefficient, NF

The reverse emission coefficient or ideality factor. The default value is 1. The value must be greater than 0.

B-E leakage current, ISE

The base-emitter leakage current. The default value is `0` A/m^2. The value must be greater than or equal to `0`.

B-E emission coefficient, NE

The base-collector emission coefficient or ideality factor. The default value is `1.5`. The value must be greater than `0`.

Forward knee current, IKF

The current value at which forward-beta high-current roll-off occurs. The default value is `0` A/m^2. The value must be greater than or equal to `0`. For this parameter, the software interprets a value of `0` as infinity.

Forward Early voltage, VAF

The forward Early voltage. The default value is `0` V. The value must be greater than or equal to `0`. For this parameter, the software interprets a value of `0` as infinity.

Reverse Gain Tab

Reverse beta, BR

The ideal maximum reverse beta. The default value is `1`. The value must be greater than `0`.

Reverse emission coefficient, NR

The reverse emission coefficient or ideality factor. The default value is `1`. The value must be greater than `0`.

B-C leakage current, ISC

The base-collector leakage current. The default value is `0` A/m^2. The value must be greater than or equal to `0`.

B-C emission coefficient, NC

The base-collector emission coefficient or ideality factor. The default value is `2`. The value must be greater than `0`.

Reverse knee current, IKR

The current value at which reverse-beta high-current roll-off occurs. The default value is `0` A/m^2. The value must be greater than or equal to `0`. For this parameter, the software interprets a value of `0` as infinity.

Reverse Early voltage, VAR

The reverse Early voltage. The default value is `0` V. The value must be greater than or equal to `0`. For this parameter, the software interprets a value of `0` as infinity.

Resistors Tab

Emitter resistance, RE

The resistance of the emitter. The default value is `1e-4` m^2*Ω. The value must be greater than or equal to `0`.

Collector resistance, RC

The resistance of the collector. The default value is `0.01` m^2*Ω. The value must be greater than or equal to `0`.

Zero-bias base resistance, RB

The resistance of the base. The default value is `1` m^2*Ω. The value must be greater than or equal to `0`.

Minimum base resistance, RBM

The minimum resistance of the base. The default value is `0` m^2*Ω. The value must be less than or equal to the **Zero-bias base resistance, RB** parameter value.

Half base resistance cur, IRB

The base current at which the base resistance has dropped to half of its zero-bias value. The default value is `Inf` A/m^2. The value must be greater than or equal to `0`. Use the default value of `Inf` if you do not want to model the change in base resistance as a function of base current.

Capacitance Tab

Model junction capacitance

Select one of the following options for modeling the junction capacitance:

- `No` — Do not include junction capacitance in the model. This is the default option.
- `B-E Capacitance` — Model the junction capacitance across the base-emitter junction.
- `B-C Capacitance` — Model the junction capacitance across the base-collector junction.
- `C-S Capacitance` — Model the junction capacitance across the collector-substrate junction.

Note To include junction capacitance in the model:

1 Select `B-E Capacitance` and specify the base-emitter junction capacitance parameters.

2 Select `B-C Capacitance` and specify the base-collector junction capacitance parameters.

3 Select `C-S Capacitance` and specify the collector-substrate junction capacitance parameters.

You can specify or change any of the common parameters when you select any of the preceding options for the **Model junction capacitance** parameter.

B-E depletion capacitance, CJE

The depletion capacitance across the base-emitter junction. This parameter is only visible when you select `B-E Capacitance` for the **Model junction capacitance** parameter. The default value is `0` F/m². The value must be greater than or equal to `0`.

B-E built-in potential, VJE

The base-emitter junction potential. This parameter is only visible when you select `B-E Capacitance` for the **Model junction capacitance** parameter. The default value is `0.75` V. The value must be greater than or equal to `0.01` V.

B-E exponential factor, MJE

The grading coefficient for the base-emitter junction. This parameter is only visible when you select `B-E Capacitance` for the **Model junction capacitance** parameter. The default value is `0.33`. The value must be greater than or equal to `0` and less than or equal to `0.9`.

Forward transit time, TF

The transit time of the minority carriers that cause diffusion capacitance when the base-emitter junction is forward-biased. This parameter is only visible when you select `B-E Capacitance` for the **Model junction capacitance** parameter. The default value is `0`. The value must be greater than or equal to `0`.

Coefficient of TF, XTF

The coefficient for the base-emitter and base-collector bias dependence of the transit time, which produces a charge across the base-emitter junction. This parameter is only visible when you select `B-E Capacitance` for the **Model junction capacitance** parameter. The default value is `0`. The value must be greater than or equal to `0`. Use the default value of `0` if you do not want to model the effect of base-emitter bias on transit time.

VBC dependence of TF, VTF

The coefficient for the base-emitter bias dependence of the transit time. This parameter is only visible when you select `B-E Capacitance` for the **Model junction capacitance** parameter. The default value is `0` V. The value must be greater than or equal to `0`. For this parameter, the software interprets a value of `0` as infinity.

Coefficient of TF, ITF

The coefficient for the dependence of the transit time on collector current. This parameter is only visible when you select `B-E Capacitance` for the **Model junction capacitance** parameter. The default value is `0` A/m^2. The value must be greater than or equal to `0`. Use the default value of `0` if you do not want to model the effect of collector current on transit time.

B-C depletion capacitance, CJC

The depletion capacitance across the base-collector junction. This parameter is only visible when you select `B-C Capacitance` for the **Model junction capacitance** parameter. The default value is `0` F/m^2. The value must be greater than `0`.

B-C built-in potential, VJC

The base-collector junction potential. This parameter is only visible when you select `B-C Capacitance` for the **Model junction capacitance** parameter. The default value is `0.75` V. The value must be greater than or equal to `0.01` V.

B-C exponential factor, MJC

The grading coefficient for the base-collector junction. This parameter is only visible when you select `B-C Capacitance` for the **Model junction capacitance** parameter. The default value is `0.33`. The value must be greater than or equal to `0` and less than or equal to `0.9`.

B-C capacitance fraction, XCJC

The fraction of the base-collector depletion capacitance that is connected between the internal base and the internal collector. The rest of the base-collector depletion capacitance is connected between the external base and the internal collector. This parameter is only visible when you select `B-C Capacitance` for the **Model junction capacitance** parameter. The default value is `0`. The value must be greater than or equal to `0` and less than or equal to `1`.

Reverse transit time, TR

The transit time of the minority carriers that cause diffusion capacitance when the base-collector junction is reverse-biased. This parameter is only visible when you select `B-C Capacitance` for the **Model junction capacitance** parameter. The default value is `0` s. The value must be greater than or equal to `0`.

Capacitance coefficient FC

The fitting coefficient that quantifies the decrease of the depletion capacitance with applied voltage. This parameter is only visible when you select `B-E Capacitance` or `B-C Capacitance` for the **Model junction capacitance** parameter. The default value is `0.5`. The value must be greater than or equal to `0` and less than or equal to `0.95`.

Specify initial condition

Select one of the following options for specifying an initial condition:

- `No` — Do not specify an initial condition for the model. This is the default option.
- `Yes` — Specify the initial transistor conditions.

> **Note** The PNP block applies the initial transistor voltages across the junction capacitors and not across the ports.

This parameter is only visible when you select `B-E Capacitance` or `B-C Capacitance` for the **Model junction capacitance** parameter.

Initial condition voltage ICVBE

Base-emitter voltage at the start of the simulation. This parameter is only visible when you select `B-E Capacitance` or `B-C Capacitance` for the **Model junction capacitance** and `Yes` for the **Specify initial condition** parameter. The default value is `0` V.

Initial condition voltage ICVCE

Base-collector voltage at the start of the simulation. This parameter is only visible when you select `B-E Capacitance` or `B-C Capacitance` for the **Model junction capacitance** and `Yes` for the **Specify initial condition** parameter. The default value is `0` V.

C-S junction capacitance, CJS

The collector-substrate junction capacitance. This parameter is only visible when you select `C-S Capacitance` for the **Model junction capacitance** parameter. The default value is `0` F/m^2. The value must be greater than or equal to `0`.

Substrate built-in potential, VJS

The potential of the substrate. This parameter is only visible when you select `C-S Capacitance` for the **Model junction capacitance** parameter. The default value is `0.75` V.

Substrate exponential factor, MJS

> The grading coefficient for the collector-substrate junction. This parameter is only visible when you select `C-S Capacitance` for the **Model junction capacitance** parameter. The default value is `0`. The value must be greater than or equal to `0` and less than or equal to `0.9`.

Temperature Tab

Model temperature dependence using

> Select one of the following options for modeling the transistor temperature dependence:
>
> - `Device temperature` — Use the device temperature, which is the **Circuit temperature** value plus the **Offset local circuit temperature, TOFFSET** value. The **Circuit temperature** value comes from the SPICE Environment Parameters block, if one exists in the circuit. Otherwise, it comes from the default value for this block.
> - `Fixed temperature` — Use a temperature that is independent of the circuit temperature to model temperature dependence.

Beta temperature exponent, XTB

> The forward and reverse beta temperature exponent that models base current temperature dependence. This parameter is only visible when you select `Device temperature` for the **Model temperature dependence using** parameter. The default value is `0`. The value must be greater than or equal to `0`.

Energy gap, EG

> The energy gap that affects the increase in the saturation current as temperature increases. This parameter is only visible when you select `Device temperature` for the **Model temperature dependence using** parameter. The default value is `1.11` eV. The value must be greater than or equal to `0.1`.

Temperature exponent for IS, XTI

> The order of the exponential increase in the saturation current as temperature increases. This parameter is only visible when you select `Device temperature` for the **Model temperature dependence using** parameter. The default value is `3`. The value must be greater than or equal to `0`.

Offset local circuit temperature, TOFFSET

The amount by which the transistor temperature differs from the circuit temperature. This parameter is only visible when you select `Device temperature` for the **Model temperature dependence using** parameter. The default value is `0` K.

Parameter extraction temperature, TMEAS

The temperature at which the transistor parameters were measured. The default value is `300.15` K. The value must be greater than `0`.

Fixed circuit temperature, TFIXED

The temperature at which to simulate the transistor. This parameter is only visible when you select `Fixed temperature` for the **Model temperature dependence using** parameter. The default value is `300.15` K. The value must be greater than `0`.

Ports

The block has the following ports:

B

Electrical conserving port associated with the transistor base terminal.

C

Electrical conserving port associated with the transistor collector terminal.

E

Electrical conserving port associated with the transistor emitter terminal.

S

Electrical conserving port associated with the transistor substrate terminal.

References

[1] G. Massobrio and P. Antognetti. *Semiconductor Device Modeling with SPICE*. 2nd Edition, McGraw-Hill, 1993. Chapter 2.

See Also

Generic Linear Actuator

SPICE Resistor

Model SPICE-compatible resistor

Library

SPICE-Compatible Components/Passive Devices

Description

The SPICE Resistor block represents a SPICE-compatible resistor. You can specify the resistance in one of the following ways:

- As a resistance value
- As process information that the block uses to calculate a resistance value

The block models temperature dependence. There are two ways to specify the resistor temperature:

- When you select `Device temperature` for the **Model temperature dependence using** parameter, the resistor temperature is

  $$T = T_C + T_O$$

  where:

  - T_C is the **Circuit temperature** parameter value from the SPICE Environment Parameters block. If this block doesn't exist in the circuit, T_C is the default value of this parameter.
  - T_O is the **Offset local circuit temperature, TOFFSET** parameter value.

- When you select `Fixed temperature` for the **Model temperature dependence using** parameter, the resistor temperature is the **Fixed circuit temperature, TFIXED** parameter value.

The block adjusts the specified or calculated resistance value for temperature using the following equation:

$R = R_0(1 + TC1(T - T_{nom}) + TC2(T - T_{nom})^2)$

Where

- R_0 is the specified or calculated resistance value.
- *TC1* is the **First order temperature coefficient, TC1** parameter value.
- *TC2* is the **Second order temperature coefficient, TC2** parameter value.
- T_{nom} is the **Parameter extraction temperature, TMEAS** parameter value.

Parameters

Resistance Tab

Device scale factor, SCALE

> The number of parallel resistors that the block represents. This value multiplies the output current. The default value is 1.

Resistor parameterization

> Select one of the following options for specifying the resistor value:
>
> - Use specified resistance — Provide the resistance value directly. This option is the default.
> - Calculate from process information — Provide process parameters that the block uses to calculate the resistance value.
>
> When you select this option, the block calculates the resistance using the following equation:
>
> $$R = RSH * \frac{(LENGTH - NARROW)}{(WIDTH - NARROW)}$$
>
> where:
>
> - *RSH* is the **Sheet resistance, RSH** parameter value.

1-823

- *LENGTH* is the **Resistor length, LENGTH** parameter value.
- *WIDTH* is the **Resistor width, WIDTH** parameter value.
- *NARROW* is the **Etch narrowing, NARROW** parameter value.

Resistance, R

Resistance value. This parameter is only visible when you select `Use specified resistance` for the **Resistor parameterization** parameter. The default value is 0 Ω.

Sheet resistance, RSH

Resistance per square of the resistor. This parameter is only visible when you select `Calculate from process information` for the **Resistor parameterization** parameter. The default value is 0 Ω.

Resistor length, LENGTH

Length dimension of the resistor. This parameter is only visible when you select `Calculate from process information` for the **Resistor parameterization** parameter. The default value is `1e-06` m.

Resistor width, WIDTH

Width dimension of the resistor. This parameter is only visible when you select `Calculate from process information` for the **Resistor parameterization** parameter. The default value is `1e-06` m.

Etch narrowing, NARROW

Amount by which the resistor length and width are reduced due to side etching. This parameter is only visible when you select `Calculate from process information` for the **Resistor parameterization** parameter. The default value is 0 m.

Temperature Tab

First order temperature coefficient, TC1

Coefficient for the linear term in the equation that the block uses to adjust the specified or calculated resistance value for temperature. The default value is 0 1/K.

Second order temperature coefficient, TC2

Coefficient for the quadratic term in the equation the block uses to adjust the specified or calculated resistance value for temperature. The default value is $0 \ 1/K^2$.

Model temperature dependence using

Select one of the following options for modeling the resistor temperature dependence:

- `Device temperature` — Use the device temperature, which is the **Circuit temperature** parameter value (from the SPICE Environment Parameters block, if one exists in the circuit, or the default value for this block otherwise) plus the **Offset local circuit temperature, TOFFSET** parameter value.

- `Fixed temperature` — Use a temperature that is independent of the circuit temperature to model temperature dependence.

Offset local circuit temperature, TOFFSET

The amount by which the resistor temperature differs from the circuit temperature. This parameter is only visible when you select `Device temperature` for the **Model temperature dependence using** parameter. The default value is `0` K.

Parameter extraction temperature, TMEAS

The temperature at which the resistor parameters were measured. The default value is `300.15` K. The value must be greater than `0`.

Fixed circuit temperature, TFIXED

The temperature at which to simulate the resistor. This parameter is only visible when you select `Fixed temperature` for the **Model temperature dependence using** parameter. The default value is `300.15` K. The value must be greater than `0`.

Ports

The block has the following ports:

+

Positive electrical voltage.

-

Negative electrical voltage.

See Also

Diode

SPST Switch

Model single-pole single-throw switch
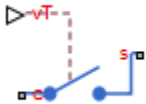
Library

Passive Devices/Switches

Description

The SPST Switch block models a single-pole single-throw switch:

- When the switch is closed, port c is connected to port s through a resistance with value equal to the **Closed resistance** parameter value.
- When the switch is open, port c is connected to port s through a resistance with value equal to the reciprocal of the **Open conductance** parameter value.

If the **Threshold width** parameter is set to zero, the switch is closed if the voltage presented at the vT control port exceeds the value of the **Threshold** parameter.

If the **Threshold width** parameter is greater than zero, then switch conductance G varies smoothly between off-state and on-state values:

$$G = \frac{x}{R_{closed}} + (1-x)G_{open}$$

$$\lambda = \frac{vT - \text{Threshold}}{\text{Threshold width}}$$

$$x = \begin{cases} 0 & \text{for } \lambda \leq 0 \\ 3\lambda^2 - 2\lambda^3 & \text{for } 0 < \lambda < 1 \\ 1 & \text{for } \lambda \geq 1 \end{cases}$$

The block uses the function $3\lambda^2 - 2\lambda^3$ because its derivative is zero for $\lambda = 0$ and $\lambda = 1$.

Defining a small positive **Threshold width** can help solver convergence in some models, particularly if the control port signal vT varies continuously as a function of other network variables. However, defining a nonzero threshold width precludes the solver making use of switched linear optimizations. Therefore, if the rest of your network is switched linear, MathWorks recommends that you set **Threshold width** to zero.

Optionally, you can add a delay between the point at which the voltage at vT passes the threshold and the switch opening or closing. To enable the delay, on the **Dynamics** tab, set the **Model dynamics** parameter to `Model turn-on and turn-off times`.

Parameters

- "Main Tab" on page 1-827
- "Dynamics Tab" on page 1-828

Main Tab

Closed resistance

Resistance between the c and s electrical ports when the switch is closed. The value must be greater than zero. The default value is `0.01` Ω.

Open conductance

Conductance between the c and s electrical ports when the switch is open. The value must be greater than zero. The default value is `1e-6` S.

Threshold

The threshold voltage for the control physical signal input vT above which the switch will turn on. The default value is `0` V.

Threshold width

The minimum increase in the control signal vT above the threshold value that will move the switch from fully open to fully closed. The default value is `0` V.

Dynamics Tab

Model dynamics

Select whether the block models a switching delay:

- `No dynamics` — Do not model the delay. This is the default option.
- `Model turn-on and turn-off times` — Use additional parameters to model a delay between the point at which the voltage at vT passes the threshold and the switch opening or closing.

Turn-on delay

Time between the input voltage exceeding the threshold voltage and the switch closing. This parameter is only visible when you select `Model turn-on and turn-off times` for the **Model dynamics** parameter. The value must be greater than zero. The default value is `1e-3` seconds.

Turn-off delay

Time between the input voltage falling below the threshold voltage and the switch opening. This parameter is only visible when you select `Model turn-on and turn-off times` for the **Model dynamics** parameter. The value must be greater than zero. The default value is `1e-3` seconds.

Initial input value, vT

The value of the physical signal input vT at time zero. This value is used to initialize the delayed control voltage parameter internally. This parameter is only visible when you select `Model turn-on and turn-off times` for the **Model dynamics** parameter. The default value is `0` V.

Ports

This block has the following ports:

vT

Physical signal that opens and closes the switch

c, s

Electrical conserving ports

See Also

DPDT Switch | DPST Switch | SPDT Switch | Switch

Stepper Motor

Model stepper motor
Library: Simscape / Electronics / Actuators & Drivers /
                        Rotational Actuators

Description

The Stepper Motor block represents a stepper motor. It uses the input pulse trains, A and B, to control the mechanical output according to the following equations:

$$e_A = -K_m \omega \sin(N_r \theta)$$

$$e_B = K_m \omega \cos(N_r \theta)$$

$$\frac{di_A}{dt} = \left(v_A - R i_A - e_A \right) / L$$

$$\frac{di_B}{dt} = \left(v_B - R i_B - e_B \right) / L$$

$$J \frac{d\omega}{dt} + B\omega = T_e$$

$$T_e = -K_m \left(i_A - \frac{e_A}{R_m} \right) \sin \left(N_r \theta \right) + K_m \left(i_B - \frac{e_B}{R_m} \right) \cos \left(N_r \theta \right) - T_d \sin \left(4 N_r \theta \right)$$

$$\frac{d\theta}{dt} = \omega$$

where:

- e_A and e_B are the back electromotive forces (emfs) induced in the A and B phase windings, respectively.

- i_A and i_B are the A and B phase winding currents.

- v_A and v_B are the A and B phase winding voltages.

- K_m is the motor torque constant.

- N_r is the number of teeth on each of the two rotor poles. The **Full step size** parameter is $(\pi/2)/N_r$.

- R is the winding resistance.

- L is the winding inductance.

- R_m is the magnetizing resistance.

- B is the rotational damping.

- J is the inertia.

- ω is the rotor speed.

- Θ is the rotor angle.

- T_d is the detent torque amplitude.

If the initial rotor is zero or some multiple of $(\pi/2)/N_r$, the rotor is aligned with the phase winding of pulse A. This happens when there is a positive current flowing from the **A+** to the **A-** ports and there is no current flowing from the **B+** to the **B-** ports.

Use the Stepper Motor Driver block to create the pulse trains for the Stepper Motor block.

The Stepper Motor block produces a positive torque acting from the mechanical **C** to **R** ports when the phase of pulse A leads the phase of pulse B.

Averaged Mode

If you set the **Simulation mode** parameter to `Averaged`, both for a Stepper Motor block and for the Stepper Motor Driver block that controls it, then the individual steps are not simulated. This can be a good way to speed up simulation. In Averaged mode, under nonslipping conditions, the motor and driver are represented by a second-order linear system that tracks the specified step rate. The demanded step rate is determined directly from voltage across **A+** and **A-**. So, for example, a voltage of +10 V across the **A+** and **A-** terminals is interpreted as a step rate demand of 10 steps per second. See the Stepper

Motor Driver reference page for more information on how to connect the Stepper Motor Driver to your step angle controller.

Averaged mode includes a slip estimator to predict whether the stepper motor would have slipped if running in Stepping simulation mode. Slip is predicted if the motor torque exceeds the **Vector of maximum torque values** parameter value for longer than one step period, the step period being determined from the current step rate demand. Upon detecting slip, the simulation will proceed or stop with an error, according to the **Action on slipping** parameter value. If you choose the action that lets the simulation continue, note that simulation results may be incorrect. When slipping occurs, the torque generated by the motor is not generally the maximum available torque; the maximum torque is only achieved if the stepper controller detects slip and adjusts the step rate command accordingly.

The dynamics of the equivalent second-order system are determined from the values that you specify for the **Approximate total load inertia** and **Maximum step rate command** parameters. It is important that you set as accurate values as possible for these parameters, so that the step rate command is tracked, and the block does not generate false slipping warnings or errors.

If you run the motor in Averaged mode with the optional thermal ports exposed (see "Thermal Ports and Effects" on page 1-832), then heat is added to the thermal ports, assuming that the windings are always powered even when the step rate command is zero. The block makes adjustments for half stepping and for reduced torque (and winding currents) at higher speeds. For these adjustments to be correct, the **Vector of maximum torque** parameter values must be correct. For half stepping, at zero speed the heat generated by the block is the average of that generated when stopped at a half step and at a full step.

To validate Averaged mode model configurations where you predict slip to occur, compare results with the same simulation performed in stepping mode.

Thermal Ports and Effects

The block has three optional thermal ports, one for each of the two windings and one for the rotor. These ports are hidden by default. To expose the thermal ports, right-click the block in your model, and then from the context menu select **Simscape > Block choices > Show thermal port**. This action displays the thermal ports on the block icon, and adds the **Temperature Dependence** and **Thermal Port** tabs to the block dialog box. These tabs are described further on this reference page.

Use the thermal ports to simulate the effects of copper resistance and iron losses that convert electrical power to heat. For more information on using thermal ports in actuator blocks, see "Simulating Thermal Effects in Rotational and Translational Actuators".

Assumptions and Limitations

The model is based on the following assumptions:

- This model neglects magnetic saturation effects and any magnetic coupling between phases.

- When you select the **Start simulation from steady state** check box in the SimscapeSolver Configuration block, this block will not initialize an **Initial rotor angle** value between –π and π.

- To use Averaged mode, the Stepper Motor block must be directly connected to a Stepper Motor Driver block also running in Averaged mode.

- The Averaged mode is an approximation, and exact step tracking compared to the Stepping mode should not be expected.

- Slip detection in Averaged mode is approximate, and depends on a good estimate for load inertia and maximum step rate. Incorrect values may result in false slip detection.

- When simulating slip in Averaged mode, it is assumed that the stepper motor controller adjusts the step rate command so as to achieve maximum possible torque.

Ports

Conserving

A+ — A-phase positive terminal
electrical

Electrical conserving port associated with the A-phase positive terminal.

A- — A-phase negative terminal
electrical

Electrical conserving port associated with the A-phase negative terminal.

B+ — B-phase positive terminal
electrical

Electrical conserving port associated with the B-phase positive terminal.

B- — B-phase negative terminal
electrical

Electrical conserving port associated with the B-phase negative terminal.

C — Machine case
mechanical rotational

Mechanical rotational conserving port connected to the motor case.

R — Machine rotor
mechanical rotational

Mechanical rotational conserving port connected to the rotor.

HA — Winding A thermal mass
thermal

Thermal conserving port associated with the thermal mass of winding A. For more information, see "Thermal Ports and Effects" on page 1-832.

HB — Winding B thermal mass
thermal

Thermal conserving port associated with the thermal mass of winding B. For more information, see "Thermal Ports and Effects" on page 1-832.

HR — Rotor thermal mass
thermal

Thermal conserving port associated with the thermal mass of the rotor. For more information, see "Thermal Ports and Effects" on page 1-832.

Parameters

Electrical Torque

`Simulation mode` — Simulation mode
`Stepping` (default) | `Averaged`

Use `Averaged` only if the block is connected directly to a Stepper Motor Driver block also running in Averaged mode.

`Vector of rotational speeds` — Rotor speeds
`[0, 1000, 3000]` rpm (default) | positive vector

Vector of rotational speeds at which to define maximum torque values, for slip prediction.

Dependencies

To enable this parameter, set **Simulation mode** to `Averaged`.

`Vector of maximum torque values` — Maximum torques
`[2, 2, 1]` N*m (default) | positive vector

Vector of maximum torque values, to be used for slip prediction with the **Vector of rotational speeds** parameter. These values are often given on a datasheet, and correspond to the supply voltage and stepping type (half step or full step) specified in the driver.

Dependencies

To enable this parameter, set **Simulation mode** to `Averaged`.

`Action on slipping` — Slip response
`none` (default) | `warn` | `error`

Select the action for the block to perform during simulation upon detecting slip:

- `none` — Continue simulation, limiting the load torque according to the **Vector of maximum torque values**.
- `warn` — Continue simulation, limiting the load torque according to the **Vector of maximum torque values**, and generate a warning that the rotor is slipping.
- `error` — Stop the simulation and generate an error message that the rotor is slipping.

If you choose an action that lets the simulation continue, simulation results might be incorrect. When slipping occurs, the motor does not always generate maximum torque. The maximum torque is only achieved if the stepper controller detects slip and adjusts the step rate command accordingly.

Dependencies

To enable this parameter, set **Simulation mode** to `Averaged`.

Approximate total load inertia — Total load inertia
`1e-4 kg*m^2` (default) | positive number

Approximate total load inertia, including the rotor inertia. This value is used to help predict slip due to rapid acceleration demands.

Dependencies

To enable this parameter, set **Simulation mode** to `Averaged`.

Maximum step rate command — Maximum command frequency
`10 Hz` (default) | positive number

Maximum commanded step rate of the system. It is used to determine a suitable bandwidth for the second order system approximation to the stepper motor and driver.

Dependencies

To enable this parameter, set **Simulation mode** to `Averaged`.

Phase winding resistance — Phase resistance
`0.55 Ohm` (default) | positive number

Resistance of the A and B phase windings.

Dependencies

To enable this parameter, set **Simulation mode** to `Stepping`.

Phase winding inductance — Phase inductance
`1.5e-3 Ohm` (default) | positive number

Inductance of the A and B phase windings.

Dependencies

To enable this parameter, set **Simulation mode** to `Stepping`.

`Motor torque constant` — Torque constant
`0.19` N*m/A (default) | positive number

Motor torque constant, K_m.

Dependencies

To enable this parameter, set **Simulation mode** to `Stepping`.

`Detent torque` — Torque variation amplitude
`0` N*m (default) | positive number

Amplitude of the sinusoidal torque variation observed when rotating the shaft of the unpowered motor.

Dependencies

To enable this parameter, set **Simulation mode** to `Stepping`.

`Magnetizing resistance` — Phase magnetizing resistance
`inf` (default) | strictly positive number

Total magnetizing resistance seen from each of the phase windings. The value must be greater than zero. The default value is `Inf`, which implies that there are no iron losses.

Dependencies

To enable this parameter, set **Simulation mode** to `Stepping`.

`Full step size` — Step size
`1.8` deg (default) | positive number

Step size when changing the polarity of either the A or B phase current.

Mechanical

`Rotor inertia` — Rotational inertia
`4.5e-5` kg*m^2 (default) | zero or positive number

Conservative force resisting rotor acceleration.

Rotor damping — Rotational damping
8.0e-4 N*m/(rad/s) (default) | zero or positive number

Dissipative force resisting rotor speed.

Initial rotor speed — Initial speed
0 rpm (default) | real number

Speed of the rotor at the start of the simulation.

Initial rotor angle — Initial angle
0 rad (default) | real number

Angle of the rotor at the start of the simulation.

Temperature Dependence

This set of parameters appears only for blocks with exposed thermal ports. For more information, see "Thermal Ports and Effects" on page 1-832.

Resistance temperature coefficients, [alpha_A alpha_B] — Temperature coefficients
[.00393, .00393] 1/K (default) | positive vector

Two-element row vector defining the coefficient α in the equation relating resistance to temperature, as described in "Thermal Model for Actuator Blocks". The first element corresponds to winding A, and the second to winding B. The default value is for copper.

Measurement temperature — Reference temperature
25 degC (default) | real number

Temperature for which motor parameters are defined.

Thermal Port

This set of parameters appears only for blocks with exposed thermal ports. For more information, see "Thermal Ports and Effects" on page 1-832.

Winding thermal masses, [M_A M_B] — Winding thermal masses
[100, 100] J/K (default) | positive vector

Two-element row vector defining the thermal mass for the A and B windings. The thermal mass is the energy required to raise the temperature by one degree.

Winding initial temperatures, [T_A T_B] — Initial winding temperatures
[25, 25] degC (default) | real number

Two-element row vector defining the temperature of the A and B thermal ports at the start of simulation.

Rotor thermal mass — Rotor thermal mass
50 J/K (default) | positive number

Thermal mass of the rotor, that is, the energy required to raise the temperature of the rotor by one degree.

Rotor initial temperature — Initial rotor temperature
25 degC (default) | real number

Temperature of the rotor at the start of simulation.

Percentage of magnetizing resistance associated with the rotor — Magnetizing resistance rotor percentage
90 (default) | positive number between 0 and 100

Percentage of the magnetizing resistance associated with the magnetic path through the rotor. It determines how much of the iron loss heating is attributed to the rotor thermal port, **HR**, and winding thermal ports, **HA** and **HB**.

References

[1] M. Bodson, J. N. Chiasson, R. T. Novotnak and R. B. Rekowski. "High-Performance Nonlinear Feedback Control of a Permanent Magnet Stepper Motor." IEEE Transactions on Control Systems Technology, Vol. 1, No. 1, March 1993.

[2] P. P. Acarnley. *Stepping Motors: A Guide to Modern Theory and Practice*. New York: Peregrinus, 1982.

[3] S.E. Lyshevski. *Electromechanical Systems, Electric Machines, and Applied Mechatronics*. CRC, 1999.

See Also

Stepper Motor Driver | Unipolar Stepper Motor

Introduced in R2008a

Stepper Motor Driver

Driver for stepper motor
Library: Simscape / Electronics / Actuators & Drivers / Drivers

Description

The Stepper Motor Driver block represents a driver for a stepper motor. It creates the pulse trains, A and B, required to control the motor. This block initiates a step each time the voltage at the ENA port rises above the **Enable threshold voltage** parameter value.

If the voltage at the REV port is less than or equal to the **Reverse threshold voltage** parameter value, pulse A leads pulse B by 90 degrees. If the voltage at the REV port is greater than the **Reverse threshold voltage** value, pulse B leads pulse A by 90 degrees and the motor direction is reversed.

At time zero, pulse A is positive and pulse B is negative.

If you set the **Stepping mode** parameter to `Half stepping`, the Stepper Motor Driver block can produce the output waveforms required for half stepping. In this mode, there is an intermediate state between the full steps, in which just one of the A or the B half-windings is powered. As a result, the step size is half of the stepper motor's full step size. At half steps, windings that are not powered are short-circuited. This approximates the effect of a freewheeling diode connected across the windings.

Averaged Mode

If you set the **Simulation mode** parameter to `Averaged`, both for a Stepper Motor Driver block and for the Stepper Motor block connected to it, then the individual steps are not simulated. This can be a good way to speed up simulation. The Averaged mode assumes that the external controller provides a step rate demand. This step rate demand is determined from the voltage applied between the ENA and REF ports on the Stepper Motor Driver block, by multiplying this voltage by the value of the **Step rate sensitivity**

parameter. The rotation direction is set by the REF port in the same way as for the Stepping mode.

Averaged mode needs to communicate the step rate demand and also output voltage amplitude information to the Stepper Motor block. To do this, the step rate demand is applied as an equivalent voltage across the A+ and A- ports. Similarly the output voltage amplitude information is conveyed by applying a steady-state voltage across the B+ and B- ports with value equal to the **Output voltage amplitude** parameter.

Assumptions and Limitations

The model is based on these assumptions:

- To use Averaged mode, the Stepper Motor Driver block must be directly connected to a Stepper Motor block also running in Averaged mode.
- When changing from Stepping to Averaged mode and back, you will need to modify your upstream blocks that provide the input voltages to the Stepper Motor Driver. One way to achieve this easily is to use Simulink variant subsystems.

Ports

Conserving

A+ — A-phase positive terminal
electrical

Electrical conserving port associated with the A-phase positive terminal.

A- — A-phase negative terminal
electrical

Electrical conserving port associated with the A-phase negative terminal.

B+ — B-phase positive terminal
electrical

Electrical conserving port associated with the B-phase positive terminal.

B- — B-phase negative terminal
electrical

Electrical conserving port associated with the B-phase negative terminal.

ENA — Triggering input step voltage
electrical

Electrical conserving port associated with the step trigger input.

REF — Input floating reference voltage
electrical

Electrical conserving port associated with the floating reference voltage.

REV — Input voltage that controls motor direction
electrical

Electrical conserving port associated with the motor direction input.

Parameters

Simulation mode — Simulation mode
Stepping (default) | Averaged

Use Averaged only if the block is connected directly to a Stepper Motor block also
running in Averaged mode.

Step rate sensitivity — Step rate demand sensitivity
10 Hz/V (default) | positive number

This parameter converts the voltage presented across the ENA and REF ports into a step
rate demand.

Dependencies

To enable this parameter, set **Simulation mode** to Averaged.

Enable threshold voltage — Step voltage threshold
2.5 V (default) | positive number

When the voltage at the ENA port rises above this threshold, the Stepper Motor Driver block initiates a step.

Dependencies

To enable this parameter, set **Simulation mode** to Stepping.

Reverse threshold voltage — Reversal voltage threshold
2.5 V (default) | positive number

When the voltage at the REV port rises above this threshold, pulse B leads pulse A by 90 degrees, and the motor direction is reversed.

Output voltage amplitude — Pulse output amplitude
10 V (default) | positive number

Amplitude of the output pulse trains.

Stepping mode — Step size
Full stepping (default) | Half stepping

Select Full stepping or Half stepping.

See Also
Controlled PWM Voltage | Stepper Motor

Introduced in R2008a

Strain Gauge

Model deformation sensor

Library

Sensors

Description

The Strain Gauge block represents a sensor that generates a change in resistance as a function of strain using the following equation:

$$\frac{\Delta R}{R} = K\varepsilon$$

where:

- $\Delta R/R$ is the fractional change in resistance.
- ε is the strain at port E.
- K is the **Gauge factor** parameter value.

Parameters

Gauge resistance

   The unstressed gauge resistance. The default value is 100 Ω.

Gauge factor

   The ratio K of the fractional change in resistance to the fractional change in length. The default value is 2.

Ports

The block has the following ports:

E

    Strain input

+

    Positive electrical port

-

    Negative electrical port

Switched Reluctance Motor

Model a switched reluctance motor

Library: Simscape / Simscape Electronics / Actuators & Drivers / Rotational Actuators

Description

The Switched Reluctance Motor implements a three-phase switched reluctance motor (SRM). The stator has three pole pairs, carrying the three motor windings, and the rotor has several nonmagnetic poles. The motor produces torque by energizing a stator pole pair, inducing a force on the closest rotor poles and pulling them toward alignment.

Choose this motor in your application to take advantage of these properties:

- Low cost
- Relatively safe failing currents
- Robustness to high temperature operation
- High torque-to-inertia ratio

Use this block to model an SRM using easily measurable or estimable parameters. To model an SRM using FEM data, see "Switched Reluctance Motor Parameterized with FEM Data".

Equations

The block calculates the voltage, *v*, across each of the windings by:

$$v = iR + L(\theta)\frac{di}{dt} + \frac{\partial \lambda}{\partial \theta}\omega_{mech},$$

where:

- *i* is the winding current.
- λ is the flux linkage.
- θ is the rotor angular position.
- ω_{mech} is the rotor speed.
- *R* is the winding resistance. Set this value using the **Magnetizing resistance** parameter.
- *L* is the inductance of the winding, given by the derivative of flux with respect to current:

$$L(\theta) = \frac{\partial \lambda(\theta)}{\partial i}.$$

The overall torque produced by the motor is:

$$\tau = \int_0^i \frac{\partial \lambda}{\partial \theta} di.$$

Because of the significant effect of saturation on flux linkage, the block approximates it for the above two equations using this relationship:

$$\lambda = \phi_{sat} \left(1 - e^{-L_0(\theta)i/\phi_{sat}} \right).$$

Here:

- Φ_{sat} is the maximum saturated flux linkage.
- L_0 is the unsaturated inductance.

The effects of saturation are more prominent as the product of current and unsaturated inductance approach the saturated flux linkage value. Specify this value using the **Saturated flux linkage** parameter.

Differentiating the flux equation then gives the winding inductance:

$$L(\theta) = L_0(\theta)e^{-L_0(\theta)i/\phi_{sat}}$$

The unsaturated inductance varies between a minimum and maximum value. The minimum value occurs when a rotor pole is directly between two stator poles. The

maximum occurs when the rotor pole is aligned with a stator pole. In between these two points, the block approximates the unsaturated inductance linearly as a function of rotor angle. This figure shows the unsaturated inductance as a rotor pole passes over a stator pole.
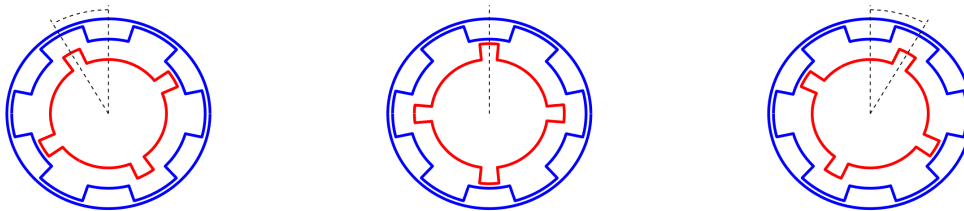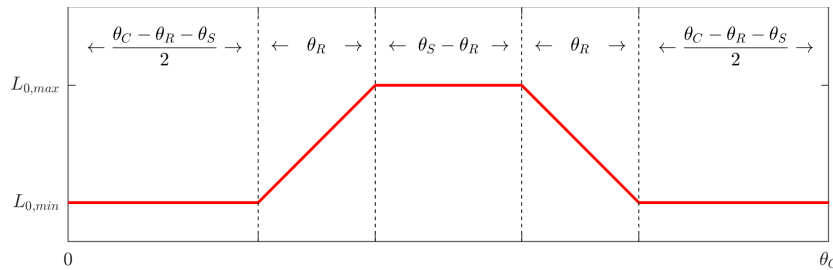
In the figure:

- θ_R corresponds to the angle subtended by the rotor pole. Set this using the **Angle subtended by each rotor pole** parameter.
- θ_S corresponds to the angle subtended by the stator pole.Set this using the **Angle subtended by each stator pole** parameter.
- θ_C corresponds to the angle subtended by this full cycle, determined by *2π/2n* where *n* is the number of stator pole pairs.

Numerical Smoothing

In practice, magnetic edge effects prevent the inductance from taking a trapezoidal shape as a rotor pole passes over a stator pole. To model these effects, and to avoid gradient discontinuities that hinder solver convergence, the block smooths the gradient $\partial L_0/\partial\theta$ at inflection points. To change the angle over which this smoothing is applied, use the **Angle over which flux gradient changes are smoothed** parameter.

Thermal Ports

The block has four optional thermal ports, one for each of the three windings and one for the rotor. These ports are hidden by default. To expose the ports, right-click the block and select **Simscape** > **Block choices** > **Show thermal port**. Thermal ports display on the block icon, and the **Thermal** tab appears in the block dialog box. These tabs are described further on this reference page.

Use the thermal ports to simulate the effects of copper resistance and iron losses that convert electrical power to heat. For more information on using thermal ports in actuator blocks, see "Simulating Thermal Effects in Rotational and Translational Actuators".

Ports

Conserving

a1 — Positive *a*-terminal
electrical

Electrical conserving port associated with the positive terminal of stator winding *a*.

a2 — Negative *a*-terminal
electrical

Electrical conserving port associated with the negative terminal of stator winding *a*.

b1 — Positive *b*-terminal
electrical

Electrical conserving port associated with the positive terminal of stator winding *b*.

b2 — Negative *b*-terminal
electrical

Electrical conserving port associated with the negative terminal of stator winding *b*.

c1 — Positive *c*-terminal
electrical

Electrical conserving port associated with the positive terminal of stator winding *c*.

c2 — Negative *c*-terminal
electrical

Electrical conserving port associated with the negative terminal of stator winding *c*.

R — Rotor
mechanical

Mechanical rotational conserving port associated with the rotor.

C — Casing
mechanical

Mechanical rotational conserving port associated with the stator or casing.

HA — *a*-terminal thermal port
thermal

Thermal conserving port associated with stator winding *a*.

Dependencies

To enable this port, use the **Show thermal port** variant, as described in "Thermal Ports" on page 1-850.

HB — *b*-terminal thermal port
thermal

Thermal conserving port associated with stator winding *b*.

Dependencies

To enable this port, use the **Show thermal port** variant, as described in "Thermal Ports" on page 1-850.

HC — *c*-terminal thermal port
thermal

Thermal conserving port associated with stator winding *c*.

Dependencies

To enable this port, use the **Show thermal port** variant, as described in "Thermal Ports" on page 1-850.

HR — Rotor thermal port
thermal

Thermal conserving port associated with the rotor.

Dependencies

To enable this port, use the **Show thermal port** variant, as described in "Thermal Ports" on page 1-850.

Parameters

Main

Number of rotor poles — Rotor pole
4 (default) | positive integer

Number of poles of the rotor.

Stator resistance per phase — Resistance
3 Ohm (default) | positive number

Per-phase resistance of each of the stator windings.

Magnetizing resistance — Magnetic losses
inf Ohm (default) | real number

The total magnetizing resistance for each of the phase windings.The default value of inf indicates that there are no iron losses.

Aligned inductance — Inductance
0.005 H (default) | positive number

Maximum inductance, achieved when a rotor and stator pole are aligned. The value of this parameter must be greater than the value of the **Unaligned inductance** parameter.

Unaligned inductance — Inductance
7e-4 H (default) | positive number

Maximum inductance, achieved when a rotor pole is midway between two stator poles. The value of this parameter must be less than the value of the **Aligned inductance** parameter.

Saturated flux linkage — Flux linkage
`inf Wb` (default) | positive number

Saturated flux linkage per phase.

Angle subtended by each stator pole — Stator tooth angle
`45 deg` (default) | positive number

Angle spanned by each stator tooth. This value must be greater than or equal to the value of **Angle subtended by each rotor pole**.

Angle subtended by each rotor pole — Rotor tooth angle
`42 deg` (default) | positive number

Angle spanned by each rotor tooth.

Angle over which flux gradient changes are smoothed — Inflection point smoothing
`0.5 deg` (default) | positive number

Angle over which sharp edges in trapezoidal inductance curve are smoothed. This value must be smaller than the value of **Angle subtended by each rotor pole**.

Mechanical

Rotor inertia — Inertia
`0.1 kg*m^2` (default) | zero or positive number

Moment of inertia of the rotor.

Rotor damping — Damping
`1e-3 N*m/(rad/s)` (default) | positive number

Damping of the rotor.

Thermal

This tab appears only for blocks with exposed thermal ports. For more information, see "Thermal Ports" on page 1-850.

Resistance temperature coefficient — Temperature coefficient
3.93e-3 1/K (default) | positive number

Coefficient α in the equation relating resistance to temperature for all three windings, as described in "Thermal Model for Actuator Blocks". The default value, 3.93e-3 1/K, is for copper.

Measurement temperature — Rated temperature
25 degC (default) | real number

The temperature for which motor parameters are quoted.

Winding thermal mass — Winding thermal mass
100 J/K (default) | positive number

The thermal mass value for the *a*-, *b*-, and *c*-windings. The thermal mass is the energy required to raise the temperature by one degree.

Rotor thermal mass — Rotor thermal mass
50 J/K (default) | positive number

The thermal mass of the rotor. The thermal mass is the energy required to raise the temperature of the rotor by one degree.

Percentage of magnetizing resistance associated with the rotor — Iron loss heating distribution
50 (default) | positive number

The percentage of the magnetizing resistance associated with the magnetic path through the rotor. This parameter determines how much of the iron loss heating is attributed to:

• The rotor thermal port **HR**

• The three stator thermal ports **HA**, **HB**, and **HC**

See Also

Introduced in R2018a

Thermistor

Model NTC thermistor using B-parameter equation

Library

Sensors

Description

The Thermistor block represents an NTC thermistor using the B-parameter equation. The resistance at temperature T is

$$R = R_0 e^{B(1/T - 1/T_0)}$$

where:

- R_0 is the nominal resistance at the reference temperature T_0.
- B is the characteristic temperature constant.

The following equation describes the thermal behavior of the block:

$$Q = K_d t_c \frac{dT}{dt}$$

where:

- Q is the net heat flow into port A.
- K_d is the **Dissipation factor** parameter value.

- t_c is the **Thermal time constant** parameter value.
- *dT/dt* is the rate of change of the temperature.

To model the thermistor in free space:

1 Connect the thermistor to the B port of a Simscape Convective Heat Transfer block.

2 Connect the A port of the Convective Heat Transfer block to a Simscape Ideal Temperature Source block whose temperature is set to the ambient temperature.

3 Set the **Area** parameter of the Convective Heat Transfer block to an approximate area A_{nom}.

4 Set the **Heat transfer coefficient** parameter of the Convective Heat Transfer block to K_d/A_{nom}.

Variables

Use the **Variables** section of the block interface to set the priority and initial target values for the block variables prior to simulation. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

Parameters

- "Electrical Tab" on page 1-857
- "Thermal Tab" on page 1-858

Electrical Tab

Nominal resistance R0 at reference temperature T0

The nominal resistance of the thermistor at the reference temperature. Many datasheets quote the nominal resistance at 25°C and list it as R25. The default value is 1000 Ω.

Characteristic temperature constant B

The coefficient *B* in the equation that describes resistance as a function of temperature. The default value is 3500 K.

Reference temperature T0

The temperature at which the nominal resistance was measured. The default value is 298.15 K.

Thermal Tab

Thermal time constant

The time it takes the sensor temperature to reach 63% of the final temperature change when a step change in ambient temperature occurs. The default value is 5 s.

Dissipation factor

The thermal power required to raise the thermistor temperature by one K. The default value is `0.75e-4` W/K.

Ports

The block has the following ports:

A

Thermal port

+

Positive electrical port

-

Negative electrical port

See Also

PTC Thermistor | Resistor | Thermal Resistor

Thermocouple

Model sensor that converts thermal potential difference into electrical potential difference

Library

Sensors

Description

The Thermocouple block represents a thermocouple using the standard polynomial parameterization defined in the NIST ITS-90 Thermocouple Database [1].

For thermocouples Type B, E, J, K (t<=0 degC), N, R, S or T, the voltage E across the device in mV is

$$E(mV) = c_0 + c_1 t + ... + c_n t^n$$

where:

- c_i is the ith element of the **Coefficients [c0 c1 ... cn]** parameter value.
- t is the temperature difference in degrees Celsius between the temperature at the thermal port A and the **Reference temperature** parameter value.

Note The equation for voltage across the device as a function of temperature difference is defined in mV. The units of the voltage across the actual device is V.

For thermocouples Type K (t>=0 degC), the equation contains an additional exponential term:

$$E(mV) = c_0 + c_1 t + \ldots + c_n t^n + a_0 e^{a_1 (t - a_2)^2}$$

where a_0, a_1, and a_2 are additional coefficients, required only by the Type K thermocouple, defined by the **Coefficients [a0 a1 a2]** parameter value.

The following equation describes the thermal behavior of the block:

$$Q = K_d t_c \frac{dT}{dt}$$

where:

- T is the temperature at port A.
- Q is the net heat flow into port A.
- K_d is the **Dissipation factor** parameter value.
- t_c is the **Thermal time constant** parameter value.
- dT/dt is the rate of change of the temperature.

To model the thermocouple in free space:

1   Connect the thermocouple to the B port of a Simscape Convective Heat Transfer block.
2   Connect the A port of the Convective Heat Transfer block to a Simscape Ideal Temperature Source block whose temperature is set to the ambient temperature.
3   Set the **Area** parameter of the Convective Heat Transfer block to an approximate area A_{nom}.
4   Set the **Heat transfer coefficient** parameter of the Convective Heat Transfer block to K_d/A_{nom}.

Basic Assumptions and Limitations

The model is based on the following assumptions:

- The high-order polynomials this block uses are very sensitive to the number of significant figures used for computation. Use all available significant figures when specifying the **Coefficients [c0 c1 ... cn]** parameter.

- **Coefficients [c0 c1 ... cn]** are defined for use over a specified temperature range.
- The maximum supported value for n in the **Coefficients [c0 c1 ... cn]** parameter is 14, that is, the vector cannot have more than 15 elements.

Parameters

- "Electrical Tab" on page 1-861
- "Thermal Tab" on page 1-862

Electrical Tab

Thermocouple model

Select one of the modeling options:

- `Type B, E, J, K (t<=0 degC), N, R, S or T` — This option is equivalent to the block functionality in previous releases.
- `Type K (t>=0 degC)` — This option adds an exponential term to the block equations when the temperature difference is greater than 0 degrees Celsius.

Coefficients [c0 c1 ... cn]

The vector of coefficients c in the equation that describes voltage as a function of temperature. The maximum length of the vector is 15 elements. The default value is `[ 0 0 0 0 0 0 0 0 0 ]`.

Note You can download parameters for standard thermocouple types from the NIST database [1]. For information on how to do this, see the Simulink Approximating Nonlinear Relationships: Type S Thermocouple example.

Coefficients [a0 a1 a2]

The vector of additional coefficients a_0, a_1, and a_2, required only by the Type K thermocouple. This parameter is only visible when you select `Type K (t>=0 degC)` for the **Thermocouple model** parameter. The default value is `[ 0 0 0 ]`.

Thermal Tab

Reference temperature

The temperature the block subtracts from the temperature at the thermal port in calculating the voltage across the device. The default value is `0` °C.

Thermal time constant

The time it takes the thermocouple temperature to reach 63% of the final temperature change when a step change in ambient temperature occurs. The default value is `1` s.

Dissipation factor

The thermal power required to raise the thermocouple temperature by one K. The default value is `0.001` W/K.

Initial temperature

The temperature of the thermocouple at the start of the simulation. The default value is `25` °C.

Ports

The block has the following ports:

A

Thermocouple thermal port

+

Positive electrical port

-

Negative electrical port

References

[1] NIST ITS-90 Thermocouple Database http://srdata.nist.gov/its90/main

See Also

Thermal Resistor

Three-Winding Mutual Inductor

Model three coupled inductors
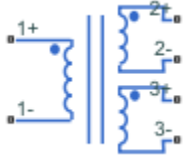
Library

Passive Devices

Description

The Three-Winding Mutual Inductor block represents a set of three coupled inductors or windings. The voltage across the three windings is

$$V_1 = L_1 \frac{dI_1}{dt} + M_{12} \frac{dI_2}{dt} + M_{13} \frac{dI_3}{dt}$$

$$V_2 = M_{12} \frac{dI_1}{dt} + L_2 \frac{dI_2}{dt} + M_{23} \frac{dI_3}{dt}$$

$$V_3 = M_{13} \frac{dI_1}{dt} + M_{23} \frac{dI_2}{dt} + L_3 \frac{dI_3}{dt}$$

where:

- V_i is voltage across the ith winding.
- I_i is current through the ith winding.
- L_i is self inductance of the ith winding.
-
  M_{ij} is mutual inductance of the ith and jth windings, $M_{ij} = K_{ij} \sqrt{L_i L_j}$.

In the preceding equations, currents are positive when flowing into the positive node of their respective inductor terminals.

When you run a simulation that includes this block, the software checks the specified parameter values to ensure that the resulting device is passive. If it is not, the software issues an error.

Parameters

Inductance L1

The self inductance of the first winding. The default value is `0.001` H.

Inductance L2

The self inductance of the second winding. The default value is `0.001` H.

Inductance L3

The self inductance of the third winding. The default value is `0.001` H.

Coefficient of coupling, K12

The coefficient that defines the mutual inductance between the first and second windings. The default value is `0.9`. The absolute value must be between 0 and 1, exclusive.

Coefficient of coupling, K13

The coefficient that defines the mutual inductance between the first and third windings. The default value is `0.9`. The absolute value must be between 0 and 1, exclusive.

Coefficient of coupling, K23

The coefficient that defines the mutual inductance between the second and third windings. The default value is `0.9`. The absolute value must be between 0 and 1, exclusive.

Specify initial condition

Select one of the following options for specifying an initial condition:

- `No` — Do not specify an initial condition for the model. This is the default option.
- `Yes` — Specify the initial inductor currents.

Initial current port 1, IC1

The current flowing through the first winding at the start of the simulation. This parameter is only visible when you select `Yes` for the **Specify initial condition** parameter. The default value is `0` A.

Initial current port 2, IC2

The current flowing through the second winding at the start of the simulation. This parameter is only visible when you select `Yes` for the **Specify initial condition** parameter. The default value is `0` A.

Initial current port 3, IC3

The current flowing through the third winding at the start of the simulation. This parameter is only visible when you select `Yes` for the **Specify initial condition** parameter. The default value is `0` A.

Ports

The block has the following ports:

1+

Positive electrical voltage of the first mutual inductor

1-

Negative electrical voltage of the first mutual inductor

2+

Positive electrical voltage of the second mutual inductor

2-

Negative electrical voltage of the second mutual inductor

3+

Positive electrical voltage of the third mutual inductor

3-

Negative electrical voltage of the third mutual inductor

Thyristor

Model thyristor using NPN and PNP transistors

Library

Semiconductor Devices

Description

The Thyristor block provides two ways of modeling a thyristor:

- As an equivalent circuit based on NPN and PNP bipolar transistors
- By a lookup table approximation to the on-state I-V (current-voltage) curve

Representation by Equivalent Circuit

The equivalent circuit contains a pair of NPN and PNP bipolar transistors, as shown in the following illustration.
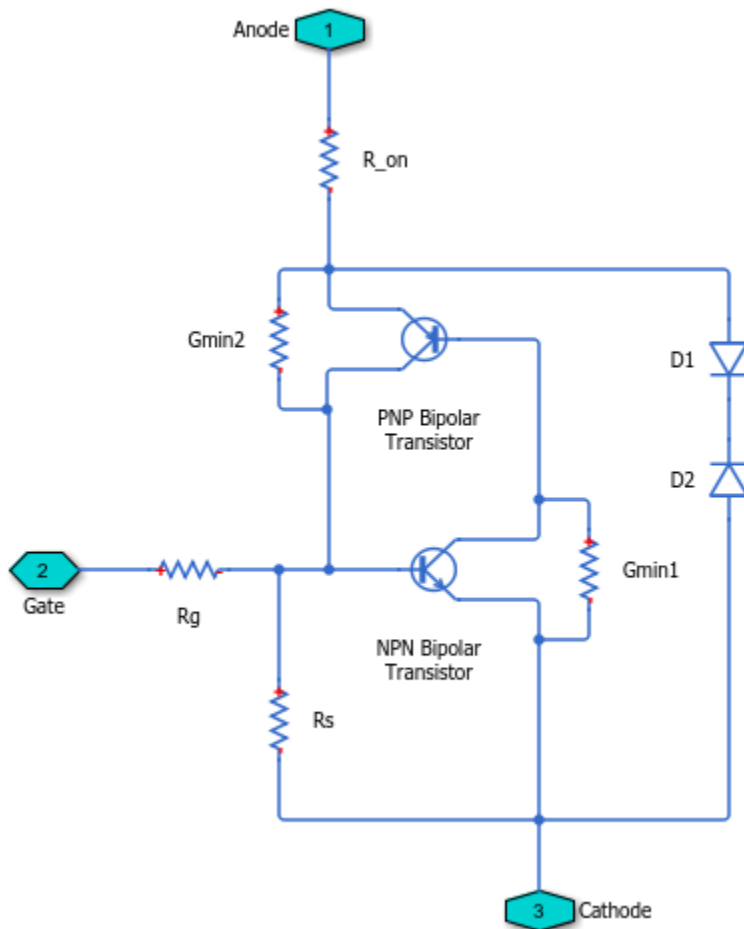
The P-N-P-N structure of a thyristor is matched by the P-N-P and N-P-N structures of the bipolar transistors, the base of each device being connected to the collector of the other device. Ensuring that this circuit behaves like a thyristor is primarily picking suitable parameter values of the NPN and PNP devices, plus external resistors. For example, for the circuit to latch into the on-state, once triggered by a suitable gate current, the total gain of the two transistors must be greater than one. This model structure replicates the behavior of a thyristor in typical application circuits, while at the same time presenting a minimum number of equations to the solver, to improve simulation speed.

> **Note** It is extremely important that you parameterize the thyristor component correctly before using it in your model. To help you do this, there are two test harnesses in the Simscape Electronics examples, Thyristor Static Behavior Validation and Thyristor Dynamic Behavior Validation. Follow the help text for these two examples, plus a datasheet for your device, to re-parameterize the thyristor component so that it replicates the required behavior. You can then copy the parameterized component into your model. Take care to model the gate drive circuitry correctly, including circuit series resistance. Connecting a controlled voltage source directly to the thyristor gate gives nonphysical results because it clamps the gate to the cathode voltage when the gate demand is zero.

The model captures the following thyristor behaviors:

- Off-state currents, I_{DRM} and I_{RRM}. These are typically quoted for the maximum off-state voltages V_{DRM} and V_{RRM}. It is assumed, as is the case for most thyristors, that $I_{DRM} = I_{RRM}$ and $V_{DRM} = V_{RRM}$.

- The gate trigger voltage is equal to the **Gate trigger voltage, v_GT** parameter value when the gate current is equal to the **Gate trigger current, i_GT** parameter value.

- The thyristor latches on when the gate current is equal to the **Gate trigger current, i_GT**. The thyristor does not latch on until the gate current reaches this value. To ensure this is the case, you must set the **Internal shunt resistor, Rs** parameter correctly. If the resistance is too high, then the gate triggers before the gate current reaches i_{GT}. If the resistance is too small, then the gate does not trigger.

  You can determine the value of the internal shunt resistor Rs by running the simulation. To see how this can be done, refer to the Thyristor Static Behavior Validation example. Alternatively, if you are using the thyristor in a circuit where there is an external resistor R_{GK} connected from gate to cathode, then the effect of Rs is usually very small, and it can be set to `inf`.

- With the thyristor in the on-state, if the gate current is removed, the thyristor stays in the on-state, provided that the load current is higher than the latching current. You do not specify the latching current directly because its value is primarily determined by other block parameters.

  However, the latching current can be influenced by the **Product of NPN and PNP forward current gains** parameter on the **Advanced** tab. Reducing the gain increases the latching current.

- The on-state voltage is equal to the **On-state voltage, V_T** parameter value when the load current is equal to the **On-state current, I_T** parameter value. This is ensured

by the R_on resistance value, which takes into account the voltage drop seen across the PNP and NPN devices.

- Triggering by rate of rise of off-state voltage. A rapid change in anode-cathode voltage induces a current in the base-collector capacitance terms. If this current is large enough, it triggers the thyristor into the on-state. The thyristor initialization routine calculates a suitable value for the base-collector capacitance, so that when the rate of change of voltage is equal to the **Critical rate of rise of off-state voltage, dV/dt** parameter value, the thyristor triggers on. This calculation is based on the approximation that the required current is v_{GT} / R_{GK} where R_{GK} is the gate-cathode resistance value used when quoting the critical *dV/dt* value.

- A nonzero gate-controlled turn-on time. This is primarily influenced by the **NPN transistor forward transit time, TF**. You either specify this parameter directly, or calculate an approximate value for TF from the turn-on time.

- A nonzero commuted turn-off time. This is primarily influenced by the **PNP transistor forward transit time, TF**. You can either specify this parameter directly, or set it to be equal to the forward transit time for the NPN transistor.

Resistors Gmin1 and Gmin2 improve numerical robustness at large forward and reverse voltages. Their values influence the off-state currents by no more than 1% at the maximum off-state forward and reverse voltages.

Note Because this block implementation includes a charge model, you must model the impedance of the circuit driving the gate to obtain representative turn-on and turn-off dynamics. Therefore, if you are simplifying the gate drive circuit by representing it as a controlled voltage source, you must include a suitable series resistor between the voltage source and the gate.

Representation by Lookup Table

If using the lookup table representation, you provide tabulated values for anode-cathode current as a function of anode-cathode voltage when in the on state. The main advantages of using this option are simulation speed and ease of parameterization. To further simplify the underlying model, this representation does not model:

- Device triggering due to rate of rise of off-state voltage
- Commuted turn-off time

The turn-on delay is represented by an input gate-cathode capacitor, the value of which is calculated so that the delay between gate voltage rising and the device starting to turn on is equal to the value specified by the **Turn-on delay time** parameter. The turn-on rise time for the load current is implemented by ramping nonlinearly between zero and the current determined by on-state current-voltage profile over a time period specified by the value of the **Turn-on rise time** parameter. Note that the resulting turn-on current profile is an approximation to an actual device.

Thermal Port

The block has an optional thermal port, hidden by default. To expose the thermal port, right-click the block in your model, and then from the context menu select **Simscape > Block choices > Show thermal port**. This action displays the thermal port H on the block icon, and adds the **Thermal Port** tab to the block dialog box.

Use the thermal port to simulate the effects of generated heat and device temperature. For more information on using thermal ports and on the **Thermal Port** tab parameters, see "Simulating Thermal Effects in Semiconductors".

Basic Assumptions and Limitations

The Thyristor block has the following limitations:

- This block does not model temperature-dependent effects. Simscape Electronics simulates the block at the temperature at which the component behavior was measured, as specified by the **Measurement temperature** parameter value. All parameters must be quoted for this temperature.

- If you use the equivalent circuit representation:

  - In sensitive gate circuits (that is, where there is no external gate-cathode resistor R_{GK}), you must set the value of the **Internal shunt resistor, Rs** parameter to ensure correct triggering. If the internal shunt resistance is too high, then the thyristor triggers for currents less than i_{GT}. If the internal shunt resistance is too low, the thyristor does not trigger for an input current of i_{GT}. For details on using simulation to determine the acceptable internal shunt resistance value, see the Thyristor Static Behavior Validation example.

  - Triggering by exceeding the break-over voltage is not modeled.

  - Numerically the thyristor can be demanding to simulate, given the very small gate currents in comparison to the load current, and also the very steep current

gradients during switching. However, for most typical thyristor-based circuits, you can use the default simulation parameters. In some cases you may need to tighten the **Absolute Tolerance** and **Relative Tolerance** parameters on the **Solver** tab of the Configuration Parameters dialog box, to ensure convergence. In such cases, changing the default value of **Absolute Tolerance** from `auto` to `1e-4` or `1e-5` is usually sufficient, because it prevents adaptive changing of this parameter during simulation.

- The leakage currents are approximated by the diodes D1 and D2, as shown in the equivalent circuit. This approach assumes that the leakage via the two transistors is small in comparison. This assumption is not valid for values of v_{GT} that are significantly smaller than the typical forward voltage drop of 0.6 V.

- If you use the lookup table representation:

  - Triggering by exceeding the break-over voltage or by rate of change of off-state voltage is not modeled.

  - Commutated turn-off time is not modeled. Check that your circuit does not violate the stated commutated turn-off time for the thyristor.

  - When you specify a turn-on rise time, the resulting current-time profile is an approximation.

Parameters

- "Main Tab" on page 1-872
- "Gate Triggering Tab" on page 1-874
- "dV/dt Triggering Tab" on page 1-874
- "Time Constants Tab" on page 1-875
- "Advanced Tab" on page 1-876

Main Tab

I-V characteristics defined by

Select the thyristor representation:

- `Fundamental nonlinear equations` — Use an equivalent circuit based on NPN and PNP bipolar transistors. This is the default.

- `Lookup table` — Use a lookup table approximation to the on-state I-V curve.

On-state voltage, V_T

The anode-cathode static voltage drop when in the on-state, and the current flowing is equal to the on-state current I_T. The default value is `1.2` V. This parameter is visible only when you select `Fundamental nonlinear equations` for the **I-V characteristics defined by** parameter.

On-state current, I_T

Static load (or equivalently anode) current that flows when the anode-cathode voltage is equal to the on-state voltage V_T. The default value is `1` A. This parameter is visible only when you select `Fundamental nonlinear equations` for the **I-V characteristics defined by** parameter.

Vector of on-state voltages, V_T

The vector of on-state voltages, to be used for table lookup. The vector values must be strictly increasing, and the first value must be greater than zero. The values can be nonuniformly spaced. The default values, in V, are `[0.75 1 1.25 1.5 1.75 2 2.25]`. This parameter is visible only when you select `Lookup table` for the **I-V characteristics defined by** parameter.

Vector of corresponding currents, I_T

The vector of currents corresponding to the on-state voltages vector values, to be used for 1D table lookup. The two vectors must be of the same size. The default values, in A, are `[0.015 0.22 0.75 1.4 2 2.75 3.45]`. This parameter is visible only when you select `Lookup table` for the **I-V characteristics defined by** parameter.

Off-state current, I_DRM

The off-state anode current I_{DRM} that flows when the anode-cathode voltage is equal to the off-state voltage V_{DRM}. The default value is `0.01` mA.

Corresponding off-state voltage, V_DRM

Corresponding off-state voltage, V_{DRM}. The anode-cathode voltage V_{DRM} applied with the thyristor in the off-state when quoting the off-state current I_{DRM}. The default value is `400` V.

Holding current

Minimum current at which the thyristor stays in the off state when gate voltage is not commanding the on state. The default value is `1` mA. This parameter is visible only when you select `Lookup table` for the **I-V characteristics defined by** parameter.

Measurement temperature

The device simulation temperature. You must specify all block parameter values for this temperature. The default value is 25 °C. This parameter is visible only when you select `Fundamental nonlinear equations` for the **I-V characteristics defined by** parameter.

Gate Triggering Tab

Gate trigger current, I_GT

Critical gate current i_{GT} required to turn the transistor on, resulting in a gate voltage equal to the gate trigger voltage v_{GT}. You must set the value of the **Internal shunt resistor, Rs** parameter on the **Advanced** tab to ensure that the gate triggers at i_{GT}, and not for currents less that i_{GT}. The default value is 3 μA.

Corresponding gate voltage, V_GT

Gate-cathode voltage v_{GT} when the gate current is equal to the gate trigger current i_{GT}. The default value is 0.6 V.

Test voltage, V_D

Supply voltage used when quoting values for v_{GT} and i_{GT}. The default value is 12 V. This parameter is visible only when you select `Fundamental nonlinear equations` for the **I-V characteristics defined by** parameter.

Test load resistor

Load resistor used when quoting values for v_{GT} and i_{GT}. The default value is 120 Ω. This parameter is visible only when you select `Fundamental nonlinear equations` for the **I-V characteristics defined by** parameter.

dV/dt Triggering Tab

Critical rate of rise of off-state voltage, dV/dt

Rate at which the off-state anode-cathode voltage must be increased for the thyristor to turn on. The default value is 150 V/μs. This parameter is visible only when you select `Fundamental nonlinear equations` for the **I-V characteristics defined by** parameter.

Test gate-cathode resistor, R_GK

Gate-cathode resistor used when quoting the critical rate of rise off off-state voltage. The default value is 1 KΩ. This parameter is visible only when you select

`Fundamental nonlinear equations` for the **I-V characteristics defined by** parameter.

Time Constants Tab

NPN device forward transit time parameterization

Select one of the following options:

- `Derive approximate value from gate-controlled turn-on time` — The block calculates the NPN device forward transit time based on the values for the gate-controlled turn-on time and corresponding gate current that you specify.

- `Specify directly` — Provide the value directly by using the **NPN device forward transit time** parameter.

This parameter is visible only when you select `Fundamental nonlinear equations` for the **I-V characteristics defined by** parameter.

Gate-controlled turn-on time

Time for the gate to turn from the off to the on state when a gate current is applied. The default value is 2 µs. This parameter is visible only when you select `Fundamental nonlinear equations` for the **I-V characteristics defined by** parameter and `Derive approximate value from gate-controlled turn-on time` for **NPN device forward transit time parameterization**.

Corresponding gate current

The gate current used when quoting the gate-controlled turn-on time. The gate current and turn-on time are used to calculate an approximate value for the NPN transistor forward transit time on the assumption that all of the input charge is used to raise the gate voltage to the gate trigger voltage v_{GT}. The default value is 10 mA. This parameter is visible only when you select `Fundamental nonlinear equations` for the **I-V characteristics defined by** parameter and `Derive approximate value from gate-controlled turn-on time` for **NPN device forward transit time parameterization**.

NPN device forward transit time

Represents the mean time for the minority carriers to cross the base region from the emitter to the collector of the NPN device [1]. The default value is 0.3 µs. This parameter is visible only when you select `Fundamental nonlinear equations` for the **I-V characteristics defined by** parameter and `Specify directly` for **NPN device forward transit time parameterization**.

PNP device forward transit time parameterization

Select one of the following options:

- `Set equal to NPN device forward transit time` — The block uses the NPN device forward transit time value.

- `Specify directly` — Provide the value directly by using the **PNP device forward transit time** parameter.

This parameter is visible only when you select `Fundamental nonlinear equations` for the **I-V characteristics defined by** parameter.

PNP device forward transit time

Represents the mean time for the minority carriers to cross the base region from the emitter to the collector of the PNP device [1]. The default value is `0.3` μs. This parameter is visible only when you select `Fundamental nonlinear equations` for the **I-V characteristics defined by** parameter and `Specify directly` for **PNP device forward transit time parameterization**.

Turn-on delay time

Time delay before the device starts to turn on following a step in current on the gate from zero to the value specified by the **Gate current for turn-on delay time** parameter. The default value is `0` s. This parameter is visible only when you select `Lookup table` for the **I-V characteristics defined by** parameter.

Gate current for turn-on delay time

The gate current used when measuring the turn-on delay time. The default value is `1` mA. This parameter is visible only when you select `Lookup table` for the **I-V characteristics defined by** parameter.

Turn-on rise time

Time it takes for the thyristor to turn on fully following the turn-on delay event. The default value is `0` s. This parameter is visible only when you select `Lookup table` for the **I-V characteristics defined by** parameter.

Advanced Tab

Internal shunt resistor, Rs

Represents the gate-cathode shunt resistance. It is important to set this parameter value to ensure that the gate triggers at i_{GT}, and not for currents less that i_{GT}. For details, see the Thyristor Static Behavior Validation example. If you are using the thyristor in a circuit where there is an external gate-cathode resistor R_{GK}, then usually

the effect of Rs is small, and it can be set to `inf`. The default value is 87 kΩ. This parameter is visible only when you select `Fundamental nonlinear equations` for the **I-V characteristics defined by** parameter.

Internal series gate resistor, Rg

Represents the resistance associated with the gate connection. A typical value is of the order of a few ohms, and its impact on static and dynamic characteristics is small. Therefore, its precise value is not important, but its presence helps avoid numerical simulation issues if the gate is driven directly by a voltage source. You can specify any positive value. The default value is `10` Ω.

Product of NPN and PNP forward current gains

This is the product of the NPN forward gain BF_{NPN} and the PNP forward gain BF_{PNP}. The value must be greater than one for latching to occur. The smaller the value, the larger the latching current becomes. However, latching current is primarily set by other block parameters, and the total gain has only a small effect. The default value is `10`. This parameter is visible only when you select `Fundamental nonlinear equations` for the **I-V characteristics defined by** parameter.

Ports

The block has the following ports:

G

Electrical conserving port associated with the gate

A

Electrical conserving port associated with the anode

K

Electrical conserving port associated with the cathode

Examples

See the Thyristor Static Behavior Validation and Thyristor Dynamic Behavior Validation examples.

References

[1] G. Massobrio and P. Antognetti. *Semiconductor Device Modeling with SPICE*. 2nd Edition, McGraw-Hill, 1993.

See Also

NPN Bipolar Transistor | PNP Bipolar Transistor

Timer

Model timer integrated circuit behaviorally
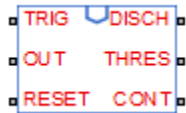
Library

Integrated Circuits

Description

The Timer block is a behavioral model of a timer integrated circuit such as the NE555.

The following figure shows the implementation structure.
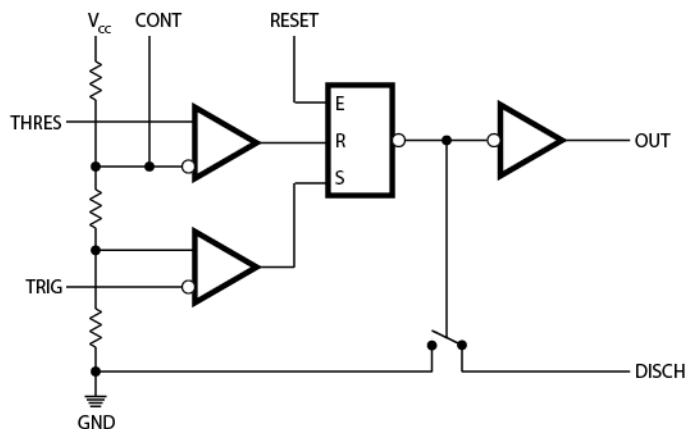
The **Potential divider component resistance** parameter sets the values of the three resistors creating the potential divider. The two comparator inputs have infinite input resistance and zero input capacitance. The S-R Latch block provides the functionality of

1-879

the set-reset latch. It includes an output capacitor and a resistor with values set to match the **Propagation delay** parameter value. The block models the output stage inverter using a CMOS NOT block. You define the output resistance, low-level output voltage, and high-level output voltage for the CMOS gate in the Timer block dialog box. The discharge switch approximates the NPN bipolar transistor on a real timer as a switch with defined switch on-resistance and off-resistance values.

Basic Assumptions and Limitations

This block has the following limitations:

- The behavior is abstracted. Results are not as accurate as a transistor-level model.
- Delay in response to changing inputs depends solely upon the RC time constant of the resistor-capacitor network at the output of the latch. In practice, the delay has a more complex dependency on the device structure. Set this value based on the output-pulse rise and fall times.
- The drop in output voltage is a linear function of output current. In practice, the relationship is that of a bipolar transistor push-pull pair.
- The controlled switch arrangement used by the block is an approximation of an open-collector arrangement.
- The power supply connects internally within the component, and the block assumes that the GND pin is grounded.

Parameters

- "Supply Tab" on page 1-880
- "Outputs Tab" on page 1-881
- "Discharge Tab" on page 1-881
- "Potential Divider Tab" on page 1-881

Supply Tab

Power supply voltage

The voltage value V_{cc} that the block applies internally to the timer component. The default value is 15 V.

Outputs Tab

Low level output voltage

The output voltage when the timer output is low and no output current is drawn. The default value is 0 V.

High level output voltage

The output voltage V_{OH} when the timer output is high and no current is drawn. The default value is 14.1 V.

Output resistance

The ratio of output voltage drop to output current. Set this parameter to

$(V_{OH} - V_{OH1})/I_{OH1}$, where V_{OH1} is the reduced output high voltage when the

output current is I_{OH1}. The default value is 8 Ω.

Propagation delay

Set this value to the input-pulse or output-pulse rise time. The default value is 1e-07 s.

Discharge Tab

Discharge switch on-resistance

A representative value is the discharge pin saturation voltage divided by the corresponding current. The default value is 12 Ω.

Discharge switch off-resistance

A representative value is the discharge pin leakage current divided by the corresponding pin voltage. The default value is 5e+08 Ω.

Potential Divider Tab

Potential divider component resistance

A typical value for a 555-type timer is 5 kΩ. You can measure it directly across the positive supply and control pins when the chip does not connect to a circuit. The default value is 5 kΩ.

Ports

This block has the following ports:

THRES
>   Electrical port corresponding to the threshold pin

TRIG
>   Electrical port corresponding to the trigger pin

CONT
>   Electrical port corresponding to the control pin

RESET
>   Electrical port corresponding to the reset pin

OUT
>   Electrical port corresponding to the output pin

DISCH
>   Electrical port corresponding to the discharge pin

See Also

Comparator | S-R Latch

Transmission Line

Model delay-based or lumped parameter transmission line

Library

Passive Devices

Description

The Transmission Line block lets you choose between the following models of a transmission line:

1. Delay-based and lossless
2. Delay-based and lossy
3. Lumped parameter L-section
4. Lumped parameter pi-section

The first option provides the best simulation performance, with options 2, 3 and 4 requiring progressively more computing power.

Delay-Based and Lossless

This first option, `Delay-based and lossless`, models the transmission line as a fixed impedance, irrespective of frequency, plus a delay term. The defining equations are:

$$v_1(\,t\,) - i_1(\,t\,)\,Z_0 = v_2(\,t - \tau\,) + i_2(\,t - \tau\,)\,Z_0$$

$$v_2(\,t\,) - i_2(\,t\,)\,Z_0 = v_1(\,t - \tau\,) + i_1(\,t - \tau\,)\,Z_0$$

where:

- v_1 is the voltage across the left-hand end of the transmission line.
- i_1 is the current into the left-hand end of the transmission line.
- v_2 is the voltage across the right-hand end of the transmission line.
- i_2 is the current into the right-hand end of the transmission line.
- τ is the transmission line delay.
- Z_0 is the line characteristic impedance.

Delay-Based and Lossy

To introduce losses, the second option, `Delay-based and lossy`, connects N delay-based components, each defined by the above equations, in series via a set of resistors, as shown in the following illustration.
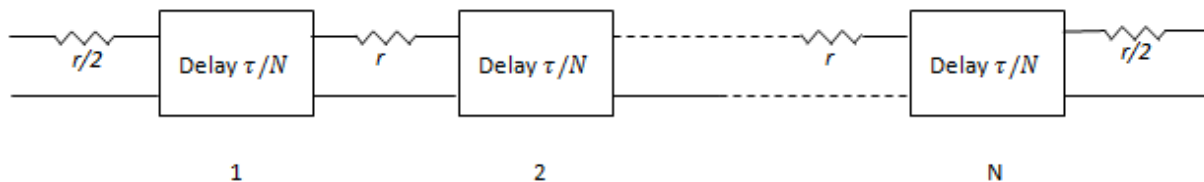
N is an integer greater than or equal to 1. $r = R \cdot LEN / N$, where R is the line resistance per unit length and LEN is the line length.

Lumped Parameter L-Section

The following block diagram shows the model of one L-line segment.
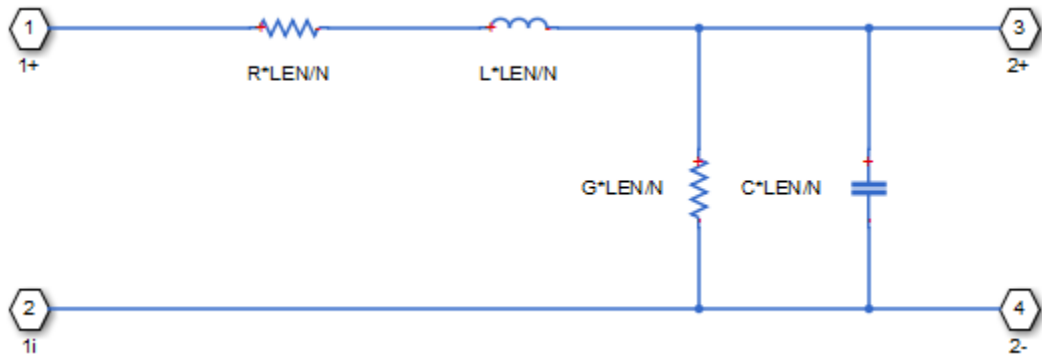
The lumped parameter parameterization uses N copies of the above segment model connected in series.

Parameters are as follows:

- R is line resistance per unit length.
- L is the line inductance per unit length.
- C is the line capacitance per unit length.
- G is the line conductance per unit length.
- LEN is the length of the line.
- N is the number of series segments.

Lumped Parameter Pi-Section

The following block diagram shows the model of one pi-line segment.
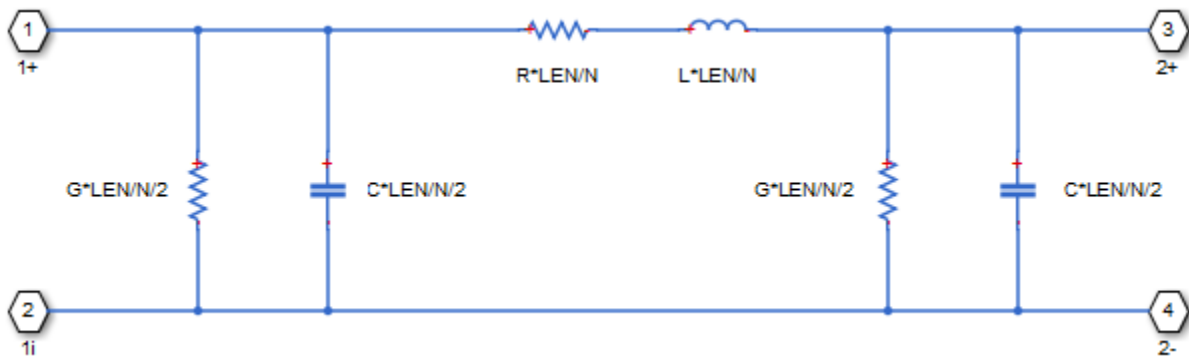
The lumped parameter parameterization uses N copies of the above segment model connected in series. The parameters are as defined for the L-section transmission line model. Unlike the L-section model, the pi-section model is symmetric.

Lumped Parameter Line Model Parameterization

The lumped-parameter models (L-section or pi-section) are the most challenging to simulate, typically needing many more segments (greater N) than for the delay-based and lossy model [1 on page 1-889].

Cable manufacturers do not typically quote an inductance value per unit length, but instead give the characteristic impedance. The inductance, capacitance, and characteristic impedance are related by:

$$L = C \cdot Z_0{}^2$$

The block lets you specify either L or Z_0 when using the lumped parameter model.

Basic Assumptions and Limitations

The Transmission Line model has the following limitations:

- For the lumped parameter options, MathWorks recommends that you use a trapezoidal solver such as `ode23t`. This is because lumped parameter transmission models have very lightly damped internal dynamics, which are best suited to trapezoidal solvers for numerical accuracy.

- The lumped parameter pi-section model has a parallel capacitor at both ends. This means that you should not connect it directly to an ideal voltage source, that is, a source with no internal resistance. The lumped parameter L-section model, however, has a series input resistor, and therefore you can connect it directly to an ideal voltage source.

Parameters

Model type

Select one of the following transmission line models:

- `Delay-based and lossless` — Model the transmission line as a fixed impedance, irrespective of frequency, plus a delay term, as described in "Delay-Based and Lossless" on page 1-883. This is the default method. It provides the best simulation performance.

- `Delay-based and lossy` — Model the transmission line as a number of delay-based components, connected in series via a set of resistors, as described in "Delay-Based and Lossy" on page 1-884.

- `Lumped parameter L-section` — Model the transmission line as a number of L-line segments, connected in series, as described in "Lumped Parameter L-Section" on page 1-884.

- `Lumped parameter pi-section` — Model the transmission line as a number of pi-line segments, connected in series, as described in "Lumped Parameter Pi-Section" on page 1-885.

Transmission delay

The total transmission line delay. This parameter appears for delay-based models only. The parameter value must be greater than zero. The default value is 5 ns, which is a typical value for a one-meter coaxial cable.

Characteristic impedance

The characteristic impedance of the transmission line. This parameter appears for delay-based models, and for lumped parameter models where **Parameterization** is `By characteristic impedance and capacitance`. The parameter value must be greater than zero. The default value is 50 Ω.

Parameterization

This parameter appears for lumped parameter models only. Select the model parameterization method, as described in "Lumped Parameter Line Model Parameterization" on page 1-886:

- `By characteristic impedance and capacitance` — Specify values for the **Characteristic impedance** and **Capacitance per unit length** parameters. This is the default method.
- `By inductance and capacitance` — Specify values for the **Inductance per unit length** and **Capacitance per unit length** parameters.

Inductance per unit length

The effective inductance of the transmission line per unit length. For lumped parameter models where **Parameterization** is `By inductance and capacitance`, this parameter appears instead of the **Characteristic impedance** parameter. The parameter value must be greater than zero. The default value is `220` µH/m.

Capacitance per unit length

The transmission line capacitance per unit length. This parameter appears for lumped parameter models only. The parameter value must be greater than zero. The default value is `90` pF/m.

Resistance per unit length

The total transmission line resistance (that is, the sum of the resistance for the two conducting paths) per unit length. This parameter appears for `Delay-based and lossy` and for lumped parameter models. The parameter value must be greater than zero. The default value is `0.3` Ω/m.

Insulation conductance per unit length

The conductance between the two transmission line conductors per unit length. This parameter appears for lumped parameter models only. The parameter value must be greater than, or equal to, zero. The default value is `5e-6` S/m.

Line length

The total transmission line length. This parameter appears for `Delay-based and lossy` and for lumped parameter models. The parameter value must be greater than zero. The default value is `1` m.

Number of segments

The number of model segments used to represent the transmission line. This parameter appears for `Delay-based and lossy` and for lumped parameter models.

The parameter value must be an integer greater than, or equal to, 1. The default value is 1.

Ports

The block has four conserving electrical ports. For a coaxial cable, the two top ports correspond to the inner conductor, and the two lower ports to the external shielding conductor.

References

[1] Sussman-Fort, S.E. and J.C. Hantgan. "SPICE Implementation of Lossy Transmission Line and Schottky Diode Models." *IEEE Transactions on Microwave Theory and Techniques*. Vol. 36, No. 1, January, 1988.

Unipolar Stepper Motor

Model stepper motor with center taps on phase windings
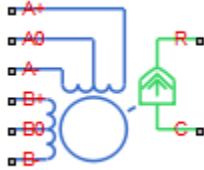
Library

Rotational Actuators

Description

The Unipolar Stepper Motor block represents a stepper motor that has center taps on the two phase windings. The winding currents and mechanical output are defined by the following equations:

$$e_{A+} = -K_m \omega \sin(N_r \theta)$$

$$e_{A-} = K_m \omega \sin(N_r \theta)$$

$$e_{B+} = K_m \omega \cos(N_r \theta)$$

$$e_{B-} = -K_m \omega \cos(N_r \theta)$$

$$\frac{di_{A+}}{dt} = \left(v_{A+} - R i_{A+} - e_{A+}\right) / L$$

$$\frac{di_{A-}}{dt} = \left(v_{A-} - Ri_{A-} + e_{A-}\right)/L$$

$$\frac{di_{B+}}{dt} = \left(v_{B+} - Ri_{B+} - e_{B+}\right)/L$$

$$\frac{di_{B-}}{dt} = \left(v_{B-} - Ri_{B-} + e_{B-}\right)/L$$

$$J\frac{d\omega}{dt} + B\omega = T_e$$

$$T_e = -K_m\left(i_{A+} - i_{A-} - \frac{e_{A+} - e_{A-}}{R_m}\right)\sin\left(N_r\theta\right) + K_m\left(i_{B+} - i_{B-} - \frac{e_{B+} - e_{B-}}{R_m}\right)\cos\left(N_r\theta\right) - T_d\sin\left(4N_r\theta\right)$$

$$\frac{d\theta}{dt} = \omega$$

where:

- e_{A+} is the back emf induced across the A+ to A0 half-winding.
- e_{A-} is the back emf induced across the A- to A0 half-winding.
- e_{B+} is the back emf induced across the B+ to B0 half-winding.
- e_{B-} is the back emf induced across the B- to B0 half-winding.
- i_{A+} is the current flowing from the A+ port to the A0 center tap port.
- i_{A-} is the current flowing from the A- port to the A0 center tap port.
- i_{B+} is the current flowing from the B+ port to the B0 center tap port.
- i_{B-} is the current flowing from the B- port to the B0 center tap port.
- v_{A+} is the voltage at the A+ port relative to the A0 center tap port.
- v_{A-} is the voltage at the A- port relative to the A0 center tap port.
- v_{B+} is the voltage at the B+ port relative to the B0 center tap port.

- v_{B-} is the voltage at the B- port relative to the B0 center tap port.
- K_m is the motor torque constant.
- N_r is the number of teeth on each of the two rotor poles. The **Full step size** parameter is $(\pi/2)/N_r$.
- R is the half-winding resistance. For example, it is the resistance between A+ and A0 ports.
- L is the half-winding inductance. For example, it is the inductance between A+ and A0 ports.
- R_m is the magnetizing resistance.
- B is the rotational damping.
- J is the inertia.
- ω is the rotor speed.
- Θ is the rotor angle.
- T_d is the detent torque amplitude.

If the initial rotor is zero or some multiple of $(\pi/2)/N_r$, the rotor is aligned with the A-phase winding. If a positive current flows from the A+ port to the A0 center tap port, then the stepper acts to stay aligned with the A-phase. Equivalently, a positive current flowing from the A0 center tap port to the A- port also acts on the rotor to stay aligned with the A-phase.

The Unipolar Stepper Motor block produces a positive torque acting from the mechanical C to R ports for either of the following sequences. Both sequences assume the rotor initial angle is zero or some multiple of $(\pi/2)/N_r$.

Sequence	Center taps connected to ground	Center taps connected to positive supply
1	Positive current from A+ to A0	Positive current from A0 to A-
2	Positive current from B+ to B0	Positive current from B0 to B-
3	Positive current from A- to A0	Positive current from A0 to A-
4	Positive current from B- to B0	Positive current from B0 to B-

Averaged Mode

If you set the **Simulation mode** parameter to `Averaged`, both for a Unipolar Stepper Motor block and for the Unipolar Stepper Motor Driver block that controls it, then the

individual steps are not simulated. This can be a good way to speed up simulation. In Averaged mode, under non-slipping conditions, the motor and driver are represented by a second-order linear system that tracks the specified step rate. The demanded step rate is determined directly from voltage across A+ and A-. So, for example, a voltage of +10V across the A+ and A- terminals is interpreted as a step rate demand of ten steps per second. See the Unipolar Stepper Motor Driver reference page for more information on how to connect the Unipolar Stepper Motor Driver to your step angle controller.

Averaged mode includes a slip estimator to predict whether the stepper motor would have slipped if running in Stepping simulation mode. Slip is predicted if the motor torque exceeds the **Vector of maximum torque values** parameter value for longer than one step period, the step period being determined from the current step rate demand. Upon detecting slip, the simulation will proceed or stop with an error, according to the **Action on slipping** parameter value. If you choose the action that lets the simulation continue, note that simulation results may be incorrect: when slipping occurs, the torque generated by the motor will not generally be the maximum available torque; the maximum torque is only achieved if the stepper controller detects slip and adjusts the step rate command accordingly.

The dynamics of the equivalent second-order system are determined from the values that you specify for the **Approximate total load inertia** and **Maximum step rate command** parameters. It is important that you set as accurate values as possible for these parameters, so that the step rate command is tracked, and the block does not generate false slipping warnings or errors.

If you run the motor in Averaged mode with the optional thermal ports exposed (see "Thermal Ports" on page 1-893), then heat is added to the thermal ports assuming that the windings are always powered, even when the step rate command is zero. The block makes adjustments for half stepping and for reduced torque (and winding currents) at higher speeds. For these adjustments to be correct, the **Vector of maximum torque** parameter values must be correct. For half stepping, at zero speed the heat generated by the block is the average of that generated when stopped at a half step and at a full step.

If you simulate or predict slip, MathWorks recommends that you do some validation runs comparing Stepping and Averaged modes before using the averaged model representation for simulation studies.

Thermal Ports

The block has five optional thermal ports, one for each of the four half-windings and one for the rotor. These ports are hidden by default. To expose the thermal ports, right-click

the block in your model, and then from the context menu select **Simscape > Block choices > Show thermal port**. This action displays the thermal ports on the block icon, and adds the **Temperature Dependence** and **Thermal Port** tabs to the block dialog box. These tabs are described further on this reference page.

Use the thermal ports to simulate the effects of copper resistance and iron losses that convert electrical power to heat. For more information on using thermal ports in actuator blocks, see "Simulating Thermal Effects in Rotational and Translational Actuators".

Basic Assumptions and Limitations

The model is based on the following assumptions:

- This model neglects magnetic saturation effects and any magnetic coupling between phases.
- When you select the **Start simulation from steady state** check box in the Simscape Solver Configuration block, this block will not initialize an **Initial rotor angle** value between –π and π.
- All four half-windings are assumed to be identical, and therefore have the same resistance temperature coefficient, alpha, and the same thermal mass.
- To use Averaged mode, the Unipolar Stepper Motor block must be directly connected to a Unipolar Stepper Motor Driver block also running in Averaged mode.
- The Averaged mode is an approximation, and exact step tracking compared to the Stepping mode should not be expected.
- Slip detection in Averaged mode is approximate, and depends on a good estimate for load inertia and maximum step rate. Incorrect values may result in false slip detection.
- When simulating slip in Averaged mode, it is assumed that the stepper motor controller adjusts the step rate command so as to achieve maximum possible torque.

Parameters

- "Electrical Torque Tab" on page 1-895
- "Mechanical Tab" on page 1-896
- "Temperature Dependence Tab" on page 1-897
- "Thermal Port Tab" on page 1-897

Electrical Torque Tab

Simulation mode

Select `Stepping` or `Averaged`. Use `Averaged` only if the block is connected directly to a Unipolar Stepper Motor Driver block also running in Averaged mode. The default value is `Stepping`.

Half-winding resistance

Half of the resistance of the A and B phase windings as measured between the A+ and A-, and the B+ and B- ports. This parameter is visible only if **Simulation mode** is set to `Stepping`. The default value is `0.55` Ω.

Half-winding inductance

Half of the inductance of the A and B phase windings as measured between the A+ and A-, and the B+ and B- ports. This parameter is visible only if **Simulation mode** is set to `Stepping`. The default value is `0.0015` H.

Motor torque constant

Motor torque constant K_m. This parameter is visible only if **Simulation mode** is set to `Stepping`. The default value is `0.19` N*m/A.

Detent torque

The amplitude of the sinusoidal torque variation observed when rotating the shaft of the unpowered motor. This parameter is visible only if **Simulation mode** is set to `Stepping`. The default value is `0` N*m.

Magnetizing resistance

The total magnetizing resistance seen from each of the phase windings, for example across A+ and A0. This parameter is visible only if **Simulation mode** is set to `Stepping`. The value must be greater than zero. The default value is `Inf`, which implies that there are no iron losses.

Vector of rotational speeds

Vector of rotational speeds at which to define maximum torque values, for slip prediction. This parameter is visible only if **Simulation mode** is set to `Averaged`. The default value is `[0 1000 3000]` rpm.

Vector of maximum torque values

Vector of maximum torque values, to be used for slip prediction in conjunction with the **Vector of rotational speeds** parameter. The maximum torque values are often given on a datasheet, and should correspond to the supply voltage and stepping type (half step or full step) specified in the driver. This parameter is visible only if **Simulation mode** is set to `Averaged`. The default value is `[2 2 1]` N*m.

Action on slipping

Select the action for the block to perform during simulation upon detecting slip:

- `none` — Continue simulation, limiting the load torque according to the **Vector of maximum torque values**.
- `warn` — Continue simulation, limiting the load torque according to the **Vector of maximum torque values**, and generate a warning that the rotor is slipping.
- `error` — Stop the simulation and generate an error message that the rotor is slipping.

Note that if you choose an action that lets the simulation continue, simulation results may be incorrect: when slipping occurs, the torque generated by the motor will not generally be the maximum available torque; the maximum torque is only achieved if the stepper controller detects slip and adjusts the step rate command accordingly.

This parameter is visible only if **Simulation mode** is set to `Averaged`.

Approximate total load inertia

The approximate total load inertia, including the rotor inertia. This value is used to help predict when slipping will occur due to rapid acceleration demands. This parameter is visible only if **Simulation mode** is set to `Averaged`. The default value is `1e-4` kg*m^2.

Maximum step rate command

The maximum step rate that your system will command. It is used to determine a suitable bandwidth for the second order system approximation to the stepper motor and driver. This parameter is visible only if **Simulation mode** is set to `Averaged`. The default value is `10` Hz.

Full step size

Step size when changing the polarity of either the A or B phase current. The default value is `1.8°`.

Mechanical Tab

Rotor inertia

Resistance of the rotor to change in motor motion. The default value is `4.5e-05` kg*m^2. The value can be zero.

Rotor damping

Energy dissipated by the rotor. The default value is `8e-04` N*m/(rad/s). The value can be zero.

Initial rotor speed

Speed of the rotor at the start of the simulation. The default value is `0` rpm.

Initial rotor angle

Angle of the rotor at the start of the simulation. The default value is `0` rad.

Temperature Dependence Tab

This tab appears only for blocks with exposed thermal ports. For more information, see "Thermal Ports" on page 1-893.

Resistance temperature coefficient

Parameter α in the equation defining resistance as a function of temperature, as described in "Thermal Model for Actuator Blocks". It is assumed that all windings are made of the same material, and therefore have the same resistance temperature coefficient. The default value is for copper, and is `0.00393` 1/K.

Measurement temperature

The temperature for which motor parameters are defined. The default value is `25` °C.

Thermal Port Tab

This tab appears only for blocks with exposed thermal ports. For more information, see "Thermal Ports" on page 1-893.

Half-winding thermal mass

The thermal mass for half of either the A or B winding. The thermal mass is the energy required to raise the temperature by one degree. It is assumed that all four half-windings have the same thermal mass. The default value is `100` J/K.

Half-winding initial temperatures, [T_A+ T_A- T_B+ T_B-]

A 1 by 4 row vector defining the temperature of the four half-windings at the start of simulation. The default value is `[ 25 25 25 25 ]` °C.

Rotor thermal mass

The thermal mass of the rotor, that is, the energy required to raise the temperature of the rotor by one degree. The default value is `50` J/K.

Rotor initial temperature

The temperature of the rotor at the start of simulation. The default value is 25 °C.

Percentage of magnetizing resistance associated with the rotor

The percentage of the magnetizing resistance associated with the magnetic path through the rotor. It determines how much of the iron loss heating is attributed to the rotor thermal port HR, and how much is attributed to the four winding thermal ports. The default value is 90%.

Ports

The block has the following ports:

A+

Top A-phase electrical connection

A0

A-phase center tap connection

A-

Lower A-phase electrical connection

B+

Top B-phase electrical connection.

B0

B-phase center tap connection

B-

Lower B-phase electrical connection

C

Mechanical rotational conserving port

R

Mechanical rotational conserving port

HA+

Thermal port for winding between A+ and A0. For more information, see "Thermal Ports" on page 1-893.

HA-

Thermal port for winding between A- and A0. For more information, see "Thermal Ports" on page 1-893.

HB+

Thermal port for winding between B+ and B0. For more information, see "Thermal Ports" on page 1-893.

HB-

Thermal port for winding between B- and B0. For more information, see "Thermal Ports" on page 1-893.

HR

Thermal port for rotor. For more information, see "Thermal Ports" on page 1-893.

Examples

See the Unipolar Stepper Motor with Control and Unipolar Stepper Motor Averaged Mode examples.

References

[1] M. Bodson, J. N. Chiasson, R. T. Novotnak and R. B. Rekowski. "High-Performance Nonlinear Feedback Control of a Permanent Magnet Stepper Motor." IEEE Transactions on Control Systems Technology, Vol. 1, No. 1, March 1993.

[2] P. P. Acarnley. *Stepping Motors: A Guide to Modern Theory and Practice*. New York: Peregrinus, 1982.

[3] S.E. Lyshevski. *Electromechanical Systems, Electric Machines, and Applied Mechatronics*. CRC, 1999.

See Also

Stepper Motor | Unipolar Stepper Motor Driver

Unipolar Stepper Motor Driver

Driver for unipolar stepper motor
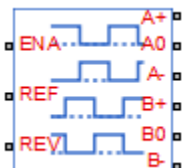
Library

Drivers

Description

The Unipolar Stepper Motor Driver block represents a driver specifically configured for use with the Unipolar Stepper Motor block. It connects the two winding center-tap connections A0 and B0 to the positive supply with a voltage equal to the value you provide for the **Output voltage amplitude** parameter. The A+, A-, B+, and B- ports are grounded in the appropriate sequence to create the stepping motion. The block initiates a step each time the voltage at the ENA port rises above the **Enable threshold voltage** parameter value.

If the voltage at the REV port is less than or equal to the **Reverse threshold voltage** parameter value, pulse A leads pulse B by 90 degrees. If the voltage at the REV port is greater than the **Reverse threshold voltage** value, pulse B leads pulse A by 90 degrees and the motor direction is reversed.

At time zero, A- and B+ are grounded.

If you set the **Stepping mode** parameter to Half stepping, the Unipolar Stepper Motor Driver block can produce the output waveforms required for half stepping. In this mode, there is an intermediate state between the full steps, in which just one of the A or the B half-windings is powered. As a result, the step size is half of the stepper motor's full

step size. At half steps, windings that are not powered are short-circuited. This approximates the effect of a freewheeling diode connected across the windings.

Averaged Mode

If you set the **Simulation mode** parameter to `Averaged`, both for a Unipolar Stepper Motor Driver block and for the Unipolar Stepper Motor block connected to it, then the individual steps are not simulated. This can be a good way to speed up simulation. The Averaged mode assumes that the external controller provides a step rate demand. This step rate demand is determined from the voltage applied between the ENA and REF ports on the Unipolar Stepper Motor Driver block, by multiplying this voltage by the value of the **Step rate sensitivity** parameter. The rotation direction is set by the REF port in the same way as for the Stepping mode.

Averaged mode needs to communicate the step rate demand and also output voltage amplitude information to the Unipolar Stepper Motor block. To do this, the step rate demand is applied as an equivalent voltage across the A+ and A- ports. Similarly the output voltage amplitude information is conveyed by applying a steady-state voltage across the B+ and B- ports with value equal to the **Output voltage amplitude** parameter.

Basic Assumptions and Limitations

The model is based on the following assumptions:

- To use Averaged mode, the Unipolar Stepper Motor Driver block must be directly connected to a Unipolar Stepper Motor block also running in Averaged mode.
- When changing from Stepping to Averaged mode and back, you will need to modify your upstream blocks that provide the input voltages to the Unipolar Stepper Motor Driver. One way to achieve this easily is to use Simulink variant subsystems.

Parameters

Simulation mode

Select `Stepping` or `Averaged`. Use `Averaged` only if the block is connected directly to a Unipolar Stepper Motor block also running in Averaged mode. The default value is `Stepping`.

Enable threshold voltage

When the voltage at the ENA port rises above this threshold, the Unipolar Stepper Motor Driver block initiates a step. This parameter is visible only if **Simulation mode** is set to `Stepping`. The default value is `2.5` V.

Step rate sensitivity

This parameter converts the voltage presented across the ENA and REF ports into a step rate demand. This parameter is visible only if **Simulation mode** is set to `Averaged`. The default value is `10` steps-per-second per volt.

Reverse threshold voltage

When the voltage at the REV port rises above this threshold, pulse B leads pulse A by 90 degrees, and the motor direction is reversed. The default value is `2.5` V.

Output voltage amplitude

Amplitude of the output pulse trains. The default value is `10` V.

Stepping mode

Select `Full stepping` or `Half stepping`. The default value is `Full stepping`.

Ports

The block has the following ports:

A+

Top A-phase electrical connection

A0

A-phase center tap connection

A-

Lower A-phase electrical connection

B+

Top B-phase electrical connection

B0

B-phase center tap connection

B-

Lower B-phase electrical connection

ENA

Triggering input step voltage

REF

Input floating reference voltage

REV

Input voltage that controls motor direction

Examples

See the Unipolar Stepper Motor with Control and Unipolar Stepper Motor Averaged Mode examples.

See Also

Controlled PWM Voltage | Unipolar Stepper Motor

Universal Motor

Model electrical and torque characteristics of a universal (or series) motor
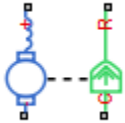
Library

Rotational Actuators

Description

The Universal Motor block represents the electrical and torque characteristics of a universal (or series) motor using the following equivalent circuit model.
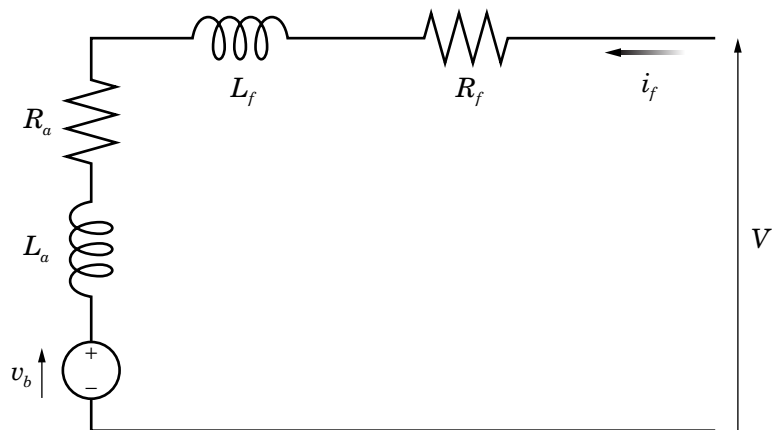
Where:

- R_a is the armature resistance.

- L_a is the armature inductance.
- R_f is the field winding resistance.
- L_f is the field winding inductance.

When you set the **Model parameterization** parameter to `By equivalent circuit parameters`, you specify the equivalent circuit parameters for this model. The Universal Motor block computes the motor torque as follows:

1 The magnetic field in the motor induces the following back emf v_b in the armature:

$$v_b = L_{af} i_f \omega$$

where L_{af} is a constant of proportionality and ω is the angular velocity.

2 The mechanical power is equal to the power reacted by the back emf:

$$P = v_b i_f = L_{af} i_f^{\ 2} \omega$$

3 The motor torque is:

$$T = P / \omega = L_{af} i_f^{\ 2}$$

The torque-speed characteristic for the Universal Motor block model is related to the parameters in the preceding figure. When you set the **Model parameterization** parameter to `By DC rated power, rated speed & maximum torque` or `By DC rated power, rated speed & electrical power`, the block solves for the equivalent circuit parameters as follows:

1 For the steady-state torque-speed relationship when using a DC supply, L has no effect.

2 Sum the voltages around the loop:

$$V = (R_f + R_a) i_f + v_b = (R_f + R_a + L_{af} \omega) i_f$$

3 Solve the preceding equation for i_f and substitute this value into the equation for torque:

$$T = L_{af} \left(\frac{V}{R_f + R_a + L_{af} \omega} \right)^2$$

The block uses the rated speed and power to calculate the rated torque. The block uses the rated torque and rated speed values in the preceding equation plus the corresponding electrical power to determine values for $R_f + R_a$ and L_{af}.

When you set the **Model parameterization** parameter to `By AC rated power, rated speed, current & electrical power`, then the block must include the inductive terms L_a and L_f in the model. This requires information about the RMS rated current and voltage for the total inductance.

The block models motor inertia J and damping B for all values of the **Model parameterization** parameter. The output torque is:

$$T_{load} = L_{af} \left(\frac{V}{R_f + R_a + L_{af}\omega} \right)^2 - J\dot{\omega} - B\omega$$

The block produces a positive torque acting from the mechanical C to R ports.

Thermal Ports

The block has two optional thermal ports, one per winding, hidden by default. To expose the thermal ports, right-click the block in your model, and then from the context menu select **Simscape** > **Block choices** > **Show thermal port**. This action displays the thermal ports on the block icon, and adds the **Temperature Dependence** and **Thermal Port** tabs to the block dialog box. These tabs are described further on this reference page.

Use the thermal ports to simulate the effects of copper resistance losses that convert electrical power to heat. For more information on using thermal ports in actuator blocks, see "Simulating Thermal Effects in Rotational and Translational Actuators".

Parameters

- "Electrical Torque Tab" on page 1-907
- "Mechanical Tab" on page 1-909
- "Temperature Dependence Tab" on page 1-910
- "Thermal Port Tab" on page 1-910

Electrical Torque Tab

Model parameterization

Select one of the following methods for block parameterization:

- By `equivalent circuit parameters` — Provide electrical parameters for an equivalent circuit model of the motor.

- By DC rated power, rated speed & maximum torque — Provide DC power and speed parameters that the block converts to an equivalent circuit model of the motor. This is the default method.
- By DC rated power, rated speed & electrical power — Provide AC power and speed parameters that the block converts to an equivalent circuit model of the motor.
- By AC rated power, rated speed, current & electrical power — Provide AC power and speed parameters that the block converts to an equivalent circuit model of the motor.

Total armature and field winding resistance

Total resistance of the armature and field winding. This parameter is only visible when you select By equivalent circuit parameters for the **Model parameterization** parameter. The default value is 132.8 Ω.

Rated speed (at rated load)

Motor speed at the rated mechanical load. This parameter is only visible when you select By DC rated power, rated speed & maximum torque, By DC rated power, rated speed & electrical power, or By AC rated power, rated speed, current & electrical power for the **Model parameterization** parameter. The default value is 6.5e+03 rpm.

Rated load (mechanical power)

The mechanical load for which the motor is rated to operate. This parameter is only visible when you select By DC rated power, rated speed & maximum torque, By DC rated power, rated speed & electrical power, or By AC rated power, rated speed, current & electrical power for the **Model parameterization** parameter. The default value is 75 W.

Rated DC supply voltage

The DC voltage at which the motor is rated to operate. This parameter is only visible when you select By DC rated power, rated speed & maximum torque or By DC rated power, rated speed & electrical power for the **Model parameterization** parameter. The default value is 200 V.

Electrical power in at rated load

The amount of electrical power the motor uses at the rated mechanical power. This parameter is only visible when you select By DC rated power, rated speed & electrical power or By AC rated power, rated speed, current & electrical power for the **Model parameterization** parameter. The default value is 160 W.

Maximum (starting) torque

Maximum torque the motor produces. This parameter is only visible when you select By DC rated power, rated speed & maximum torque for the **Model parameterization** parameter. The default value is 0.39 N*m.

Total armature and field winding inductance

Total inductance of the armature and field winding. If you do not have information about this inductance, set the value of this parameter to a small, nonzero number. This parameter is only visible when you select By equivalent circuit parameters, By DC rated power, rated speed & maximum torque, or By DC rated power, rated speed & electrical power for the **Model parameterization** parameter. The default value is 0.525 H.

Note You can set the **Total armature and field winding inductance** value to zero, but this only makes sense if you are driving the motor with a DC source.

RMS rated voltage

RMS supply voltage when the motor operates on AC power. This parameter is only visible when you select By AC rated power, rated speed, current & electrical power for the **Model parameterization** parameter. The default value is 240 V.

RMS current at rated load

RMS current when the motor operates on AC power at the rated load. This parameter is only visible when you select By AC rated power, rated speed, current & electrical power for the **Model parameterization** parameter. The default value is 0.8 A.

AC frequency

Frequency of the AC supply voltage. This parameter is only visible when you select By AC rated power, rated speed, current & electrical power for the **Model parameterization** parameter. The default value is 50 Hz.

Mechanical Tab

Rotor inertia

Rotor inertia. The default value is 2e-04 kg*m^2. The value can be zero.

Rotor damping

Rotor damping. The default value is `1e-06` N*m/(rad/s). The value can be zero.

Initial rotor speed

Speed of the rotor at the start of the simulation. The default value is `0` rpm.

Temperature Dependence Tab

This tab appears only for blocks with exposed thermal ports. For more information, see "Thermal Ports" on page 1-906.

Field to armature resistance ratio, Rf/Ra

The ratio of the field to the armature resistance. This parameter is required only when showing the field and armature thermal ports. It is used to determine individual resistance values for the field and armature windings so that the thermal heat generated by the two resistors can be apportioned correctly. The default value is `1`.

Resistance temperature coefficients, [alpha_f alpha_a]

A 1 by 2 row vector defining the coefficient α in the equation relating resistance to temperature, as described in "Thermal Model for Actuator Blocks". The first element corresponds to the field winding, and the second to the armature. The default value is for copper, and is `[ 0.00393 0.00393 ]` 1/K.

Measurement temperature

The temperature for which motor parameters are defined. The default value is 25 °C.

Thermal Port Tab

This tab appears only for blocks with exposed thermal ports. For more information, see "Thermal Ports" on page 1-906.

Thermal masses, [Mf Ma]

A 1 by 2 row vector defining the thermal mass for the field and armature windings. The thermal mass is the energy required to raise the temperature by one degree. The default value is `[ 100 100 ]` J/K.

Initial temperatures, [Tf Ta]

A 1 by 2 row vector defining the temperature of the field and armature thermal ports at the start of simulation. The default value is `[ 25 25 ]` °C.

Ports

The block has the following ports:

+

    Positive electrical port.

-

    Negative electrical port.

C

    Mechanical rotational conserving port.

R

    Mechanical rotational conserving port.

Hf

    Field winding thermal port. For more information, see "Thermal Ports" on page 1-906.

Ha

    Armature winding thermal port. For more information, see "Thermal Ports" on page 1-906.

References

[1] Bolton, W. Mechatronics: *Electronic Control Systems in Mechanical and Electrical Engineering* 3rd edition, Pearson Education, 2004.

See Also

DC Motor | Induction Motor | Servomotor | Shunt Motor

Variable Capacitor

Model linear time-varying capacitor
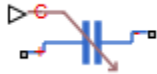
Library

Passive Devices

Description

The Variable Capacitor block represents a linear time-varying capacitor. The block provides two options for the relationship between the current i through the capacitor and the voltage v across the device when the capacitance at port C is C. The **Equation** parameter determines which of the following equations the block uses:

-
  $$i = C\frac{dv}{dt}$$
  Use the preceding equation when the capacitance is defined as the local gradient of the charge-voltage curve for a given voltage:
  $$C(v) = \frac{dQ(v)}{dv}$$

-
  $$i = \frac{dC}{dt}v + C\frac{dv}{dt}$$
  Use the preceding equation when the capacitance is defined as the ratio of the charge Q to the steady-state voltage:
  $$C(v) = \frac{Q(v)}{v}$$

The block includes a resistor in series with the variable capacitor. You can use this resistor to represent the total ohmic connection resistance of the capacitor. You may need to use this resistor to prevent numerical issues for some circuit topologies, such as where a Variable Capacitor block is connected in parallel with another capacitor block that does not have a series resistance.

Parameters

Equation

Select one of the following options for block capacitance:

- `I = C*dV/dt` — This equation assumes the capacitance is defined as the local gradient of the charge-voltage curve for a given voltage. This option is the default.
- `I = C*dV/dt + dC/dt*V` — This equation assumes the capacitance is defined as the ratio of the charge to the steady-state voltage.

Minimum capacitance C>0

The lower limit on the value of the signal at port C. This limit prevents the signal from reaching a value that has no physical meaning. The default value is `1e-09` F.

Series resistance

The value of the resistance placed in series with the variable capacitor. The default value is `1e-06` Ω.

Initial charge

The charge at the start of the simulation. This parameter is only visible when you select `I = C*dV/dt + dC/dt*V` for the **Equation** parameter. The default value is `0` c.

Initial voltage

The output voltage at the start of the simulation. This parameter is only visible when you select `I = C*dV/dt` for the **Equation** parameter. The default value is `0` V.

Ports

The block has the following ports:

C

Capacitance physical signal port (C must be finite and greater than zero)

+

    Positive electrical port

-

    Negative electrical port

See Also

Variable Inductor | Variable Resistor

Variable Inductor

Model linear time-varying inductor
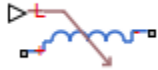
Library

Passive Devices

Description

The Variable Inductor block represents a linear time-varying inductor. The block provides two options for the relationship between the voltage *v* across the device and the current through the inductor *i* when the inductance at port L is *L*. The **Equation** parameter determines which of the following equations the block uses:

-

$$v = \frac{dL}{dt} i + L \frac{di}{dt}$$

   Use the preceding equation when the inductance is defined as the ratio of the magnetic flux Φ to the steady-state current:

$$L(i) = \frac{\Phi(i)}{i}$$

-

$$v = L \frac{di}{dt}$$

   Use the preceding equation when the inductance is defined as the local gradient of the flux-current curve for a given current:

$$L(i) = \frac{d\Phi(i)}{di}$$

The block includes a conductance in parallel with the variable inductor. You can use the conductor to represent the total insulation conductance of the inductor. You may need to use the conductor to prevent numerical issues for some circuit topologies, such as where a Variable Inductor block is connected in series with another inductor block that does not have a parallel conductance.

Parameters

Equation

Select one of the following options for block inductance:

- `V = L*dI/dt + dL/dt*I` — This equation assumes the inductance is defined as the ratio of the magnetic flux to the steady-state current. This option is the default.

- `V = L*dI/dt` — This equation assumes the inductance is defined as the local gradient of the flux-current curve for a given current.

Minimum inductance L>0

The lower limit on the value of the signal at port L. This limit prevents the signal from reaching a value that has no physical meaning. The default value is `1e-06` H.

Parallel conductance

The value of the conductance placed in parallel with the variable inductor. The default value is `1e-09` 1/Ω.

Initial magnetic flux

The magnetic flux at the start of the simulation. This parameter is only visible when you select `V = L*dI/dt + dL/dt*I` for the **Equation** parameter. The default value is `0` Wb.

Initial current

The output current at the start of the simulation. This parameter is only visible when you select `V = L*dI/dt` for the **Equation** parameter. The default value is `0` A.

Ports

The block has the following ports:

L

Inductance physical signal port (L must be finite and greater than zero)

+

Positive electrical port

-

Negative electrical port

See Also

Variable Capacitor | Variable Resistor

Varistor

Voltage-dependent resistor
Library: Simscape / Simscape Components / Passive Devices /
                 Fundamental Components

Description

The Varistor block represents a voltage-dependent resistor (VDR). This component is also commonly known as a metal-oxide varistor (MOV). The block exhibits high resistance at low voltages and low resistance at high voltages.

You can protect parts of an electrical circuit from high-voltage surges by placing this block in parallel with them. When a surge occurs, the resistance of the varistor drops significantly, causing the current to be shunted through the varistor rather than through the circuit.

Use the **Parameterization** parameter to choose between two different behaviors for this block. The Linear option focuses on the on- and off-states of the varistor and uses a linear relationship between current and voltage in both regions. The Power-law option uses an exponential relationship between current and voltage in the initial on-state. This option also adds a third, linear region at higher voltages.

Linear Parameterization

This parameterization option separates the voltage-current relationship into two linear regions:

- Off-region — resistance is high and current increases slowly with increasing voltage.
- On-region — resistance is low and current increases rapidly with increasing voltage.

This figure shows the voltage-current relationship across the on- and off-regions.
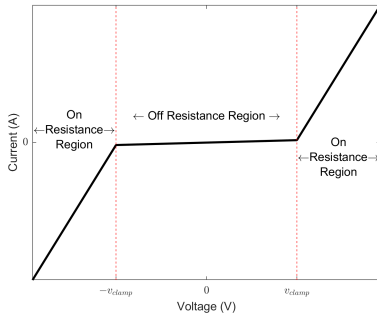
Use linear parameterization in one of these scenarios:

- You are modeling voltage surges close to the threshold voltage
- You expect your varistor to behave linearly in all regions

The voltage-current relationship for the linear varistor is:

$$
i_{varistor} = \begin{cases} \dfrac{v_{varistor}}{R_{off}} & |v_{varistor}| < v_{clamp} \\[3mm] \dfrac{v_{varistor}}{R_{on}} + c_1 \operatorname{sgn}(v_{varistor}) & |v_{varistor}| \geq v_{clamp} \end{cases}.
$$

where:

- $v_{varistor}$ and $i_{varistor}$ are the varistor voltage and current, respectively.
- v_{clamp} is the threshold voltage that separates the two regions of operation. Set this value using the **Clamping voltage** parameter.
- R_{on} and R_{off} are the resistances in the on- and off-regions. Set these values using the **On resistance** and **Off resistance**, respectively.
- c_1 is a constant used to enforce current continuity between the two regions:

$$
c_1 = v_{clamp} \left(\frac{1}{R_{off}} - \frac{1}{R_{on}} \right).
$$

Power-Law Parameterization

This parameterization option separates the voltage-current relationship into three regions:

- Leakage region — Resistance is high and current increases slowly with increasing voltage.
- Normal region — Resistance decreases exponentially with increasing voltage.
- Upturn region — Resistance is low and current increases rapidly with increasing voltage.

This figure shows the three regions of operation in log-log-scale.
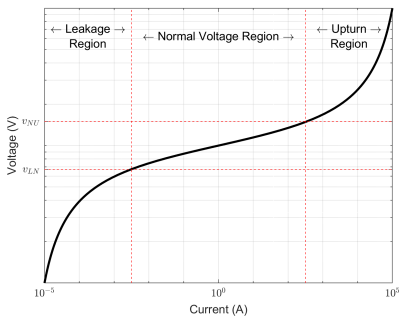
Use power-law parameterization in one of these scenarios:

- You are modeling voltage surges across a large range of voltages
- You expect your varistor to behave exponentially in the first on-region

The voltage-current relationship for the power-law varistor is:

$$i_{varistor} = \begin{cases} \dfrac{v_{varistor}}{R_L} & |v_{varistor}| < v_{LN} \\[2ex] k\left(v_{varistor}\right)^\alpha + c_1 & v_{LN} \le |v_{varistor}| \le v_{NU} \\[2ex] \dfrac{v_{varistor}}{R_U} + c_2 & |v_{varistor}| > v_{NU} \end{cases}$$

where:

- $v_{varistor}$ and $i_{varistor}$ are the varistor voltage and current, respectively.

- α is the power-law exponent which determines the rate of current increase with voltage increase in the normal region. Set this value using the **Normal-mode power-law exponent** parameter.

- v_{LN} and the v_{NU} are the threshold voltages corresponding to the leakage-normal and normal-upturn transition points. Set these values using the **Leakage to normal voltage transition** and **Normal to upturn voltage transition** parameters, respectively.

- R_L and R_U are the resistances in the leakage- and upturn-regions. Set these values using the **Leakage-mode resistance** and **Upturn-mode resistance** parameters, respectively.

- k, c_1, and c_2 are constants used to enforce current continuity between the regions:

$$k = \frac{1}{\alpha R_U v_{NU}^{\alpha-1}},$$

$$c_1 = \frac{v_{LN}}{R_L} - \frac{v_{LN}^{\alpha}}{\alpha R_U v_{NU}^{\alpha-1}},$$

and

$$c_2 = \frac{1}{\alpha R_U v_{NU}^{\alpha-1}} (v_{NU}^{\alpha} - v_{LN}^{\alpha}) - \frac{v_{NU}}{R_U} + \frac{v_{LN}}{R_L}.$$

Equivalent Circuit

In addition to the varistor equations, you can also specify a constant terminal resistance R_t and device capacitance C. This figure shows the equivalent circuit for the varistor in either parameterization mode.
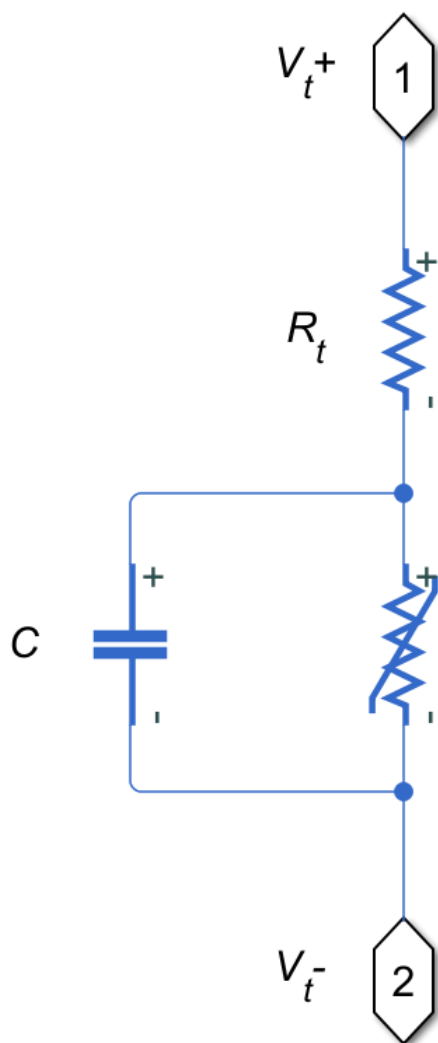
Ports

Conserving

+ — Positive terminal
electrical

Electrical conserving port associated with the varistor positive terminal.

- — Negative terminal
electrical

Electrical conserving port associated with the varistor negative terminal.

Parameters

Parameterization — Varistor operation mode
Linear (default) | Power-law

Choose how the varistor resistance changes with increasing voltages:

- Linear — Two regions. The low-voltage region has high resistance and the high-voltage region has low resistance.
- Power-law — Three regions. The leakage region has high resistance. The normal region has exponentially decreasing resistance. The upturn region has low resistance.

Clamping voltage — Threshold voltage
260 (default) | positive number

Transition point voltage, v_{clamp}, between the off- and on-states of the linear varistor.

Dependencies

Enabled when the **Parameterization** parameter is set to Linear.

Off resistance — Low-voltage resistance
3e8 (default) | positive number

Low-voltage resistance, R_{off}, of the varistor in the off-state.

Dependencies

Enabled when the **Parameterization** parameter is set to `Linear`.

`On resistance` — **High-voltage resistance**
1 (default) | positive number

High-voltage resistance, R_{on}, of the varistor in the on-state.

Dependencies

Enabled when the **Parameterization** parameter is set to `Linear`.

`Leakage to normal voltage transition` — **First threshold voltage**
130 (default) | positive number

Transition point voltage, v_{LN}, between the leakage and normal regions of the power-law varistor.

Dependencies

Enabled when the **Parameterization** parameter is set to `Power-law`.

`Normal to upturn voltage transition` — **Second threshold voltage**
300 (default) | positive number

Transition point voltage, v_{NU}, between the normal and upturn regions of the power-law varistor.

Dependencies

Enabled when the **Parameterization** parameter is set to `Power-law`.

`Leakage-mode resistance` — **Low-voltage resistance**
3e8 (default) | positive number

Low-voltage resistance, R_L, of the varistor in the leakage region.

Dependencies

Enabled when the **Parameterization** parameter is set to `Power-law`.

`Normal-mode power-law exponent` — **Mid-voltage resistance**
45 (default) | positive number

Exponent that determines rate of current increase with voltage increase of the varistor in the normal region.

Dependencies

Enabled when the **Parameterization** parameter is set to `Power-law`.

Upturn-mode resistance — High-voltage resistance
`0.07` (default) | positive number

High-voltage resistance, R_U, of the varistor in the upturn region.

Dependencies

Enabled when the **Parameterization** parameter is set to `Power-law`.

Terminal resistance — Terminal resistance
`100e-6` (default) | non-negative number

Small, constant resistance in series with the varistor. Set this value to zero to remove the resistance from the equivalent circuit.

Capacitance — Parallel capacitance
`4.4` (default) | non-negative number

Capacitor in parallel with the varistor. Set this value to zero to remove the capacitor from the equivalent circuit.

See Also

Resistor | Variable Resistor

Introduced in R2018a

Voltage-Controlled Oscillator

Behavioral model of voltage-controlled oscillator
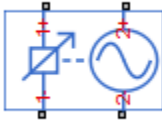
Library

Integrated Circuits

Description

The Voltage-Controlled Oscillator block provides a behavioral model of a voltage-controlled oscillator (VCO). The output voltage is defined by the following equations:

$$v_{\lim} = \begin{cases} v_{\min} & \text{for } v_{\text{in}} < v_{m\,\text{in}} \\ v_{\text{in}} & \text{for } v_{m\,\text{in}} \le v_{\text{in}} \le v_{\max} \\ v_{\max} & \text{for } v_{\text{in}} > v_{\max} \end{cases}$$

$$\dot{\Phi} = 2\pi F\left(v_{\lim}\right)$$

$$v_{out} = A\sin\left(2\pi f_{nom}t + \Phi\right) - i_{out}R_{out}$$

where:

- v_{in} is the voltage applied across the 1+ and 1– ports.
- v_{out} is the voltage across the 2+ and 2– ports.
- f_{nom} is the oscillator frequency when the input control voltage is v_{nom}.

- F is a linear function of v_{lim} or a lookup table function of v_{lim}.
- A is the output voltage peak amplitude.
- t is simulation time.
- i_{out} is the output current.
- R_{out} is the output resistance.

If you choose `Linear` for the **Frequency dependence on input voltage** parameter, then the function F is given by:

$$F = f_{nom} + k \left(v_{lim} - v_{nom} \right)$$

where k is the rate of change of frequency with input voltage.

If you choose `Tabulated` for the **Frequency dependence on input voltage** parameter, then the function F is defined by the vectors of input voltages and corresponding output frequency deviations from nominal that you supply. The values for v_{min} and v_{max} are the first and the last values of the input voltage vector.

You can model the time delay between a change in the input control voltage and the oscillator frequency. Do this by modeling a first-order dynamic between v_{lim} and the value passed to the function F.

Parameters

- "Frequency Tab" on page 1-927
- "Electrical Characteristics Tab" on page 1-929
- "Dynamics Tab" on page 1-929

Frequency Tab

Frequency dependence on input voltage

Select one of the following methods for block parameterization:

- `Linear` — Define a linear function by specifying the rate of change of frequency with input voltage. This is the default option.

- `Tabulated` — Provide the vectors of input voltages and corresponding output frequency deviations from nominal. The block determines the frequency deviation by table lookup based on these values.

Nominal frequency

The oscillator frequency when the input control voltage is at the nominal value. The default value is `1000` Hz.

Input voltage corresponding to nominal frequency

The input voltage corresponding to the oscillator nominal frequency. This parameter is visible only if you select `Linear` for the **Frequency dependence on input voltage** parameter. The default value is `0.5` V.

Rate of change of frequency with input voltage

The linear coefficient defining the rate of change of frequency depending on input voltage. This parameter is visible only if you select `Linear` for the **Frequency dependence on input voltage** parameter. The default value is `1` Hz/V.

Minimum input voltage

The minimum input voltage that affects VCO frequency. This parameter is visible only if you select `Linear` for the **Frequency dependence on input voltage** parameter. The default value is `0` V.

Maximum input voltage

The maximum input voltage that affects VCO frequency. This parameter is visible only if you select `Linear` for the **Frequency dependence on input voltage** parameter. The default value is `1` V.

Input voltage vector

The vector of voltages for the tabulated VCO frequency. This parameter is visible only if you select `Tabulated` for the **Frequency dependence on input voltage** parameter. The default value is `[0 0.2 0.4 0.6 0.8 1]` V.

Frequency deviation from nominal

The corresponding vector of VCO frequencies relative to the nominal frequency. This parameter is visible only if you select `Tabulated` for the **Frequency dependence on input voltage** parameter. The default value is `[-1000 -329 -51 162 342 500]` Hz.

Electrical Characteristics Tab

Output voltage peak amplitude

The peak amplitude of the voltage across the 2+ and 2– terminals. The default value is `1` V.

Input resistance

The resistance seen at the 1+ and 1– terminals. The default value is `Inf` Ohm.

Output resistance

The value of the series output resistance. The default value is `0` Ohm.

Dynamics Tab

Dynamics

Select one of the following methods for specifying dynamics:

- `No dynamics` — Do not model the time delay between a change in the input control voltage and the oscillator frequency. This is the default option.
- `Model frequency tracking dynamics` — Model a first order dynamic between the input control voltage and the oscillator frequency.

Frequency tracking time constant

Time constant for the first-order filter that delays the measured input control voltage, to model the lag between a change in VCO demanded frequency and the resulting VCO frequency. This parameter is visible only if you select `Model frequency tracking dynamics` for the **Dynamics** parameter. The default value is `0.001` s.

Initial frequency

The initial VCO output frequency. This parameter is visible only if you select `Model frequency tracking dynamics` for the **Dynamics** parameter. The default value is `1000` Hz.

Ports

The block has the following ports:

1+

Positive input voltage

1-

  Negative input voltage

2+

  Positive output voltage

2-

  Negative output voltage

Voltage-Controlled Switch

Model voltage-controlled switch with hysteresis
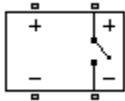
Library

SPICE-Compatible Components/Passive Devices

Description

The Voltage-Controlled Switch block represents the electrical characteristics of a switch whose state is controlled by the voltage across the input ports (the controlling voltage):

- When the controlling voltage is greater than the sum of the **Threshold voltage, VT** and **Hysteresis voltage, VH** parameter values, the switch is closed and has a resistance equal to the **On resistance, RON** parameter value.

- When the controlling voltage is less than the **Threshold voltage, VT** parameter value minus the **Hysteresis voltage, VH** parameter value, the switch is open and has a resistance equal to the **Off resistance, ROFF** parameter value.

- When the controlling voltage is greater than or less than the **Threshold voltage, VT** parameter value by an amount less than or equal to the **Hysteresis voltage, VH** parameter value, the voltage is in the crossover region and the state of the switch remains unchanged.

Basic Assumptions and Limitations

The block output resistance model is discontinuous during switching. The discontinuity might cause numerical issues. Try the following actions to resolve the issues:

- Set the **On resistance, RON** and **Off resistance, ROFF** parameter values to keep the ratio *RON/ROFF* as small as possible, and less than 1e+12.

- Increase the **Hysteresis voltage, VH** parameter value to reduce switch chatter.
- Decrease the **Max step size** parameter value (in the Configuration Parameters block dialog box).

Note This increases the simulation time.

Parameters

Threshold voltage, VT

> The voltage above which the block interprets the controlling voltage as HIGH. The default value is 0 V.

Note The controlling voltage must differ from the threshold voltage by at least the **Hysteresis voltage, VH** parameter value to change the state of the switch.

Hysteresis voltage, VH

> The amount by which the controlling voltage must exceed or fall below the **Threshold voltage, VT** parameter value to change the state of the switch. The default value is 0 V.

On resistance, RON

> The resistance of the switch when it is closed. The default value is 1 Ω.

Off resistance, ROFF

> The resistance of the switch when it is open. The default value is 1e+12 Ω.

Initial switch state

> Select one of the following options for the state of the switch at the start of the simulation:

- On — The switch is initially closed and its resistance value is equal to the **On resistance, RON** parameter value. This is the default option.
- Off — The switch is initially open and its resistance value is equal to the **Off resistance, ROFF** parameter value.

Ports

The block has the following ports:

+

    Positive electrical input and output ports.

-

    Negative electrical input and output ports.

Voltage Source

Simulate voltage source with DC, AC, and noise components

Library

Sources

Description

The Voltage Source block implements a voltage source with DC, AC, and noise components. The voltage across the + and – terminals is given by:

$$v = v_{DC} + v_{AC}\sin\left(2\pi f t + \phi\right) + v_N$$

where:

- v_{DC} is the steady-state DC voltage component.
- v_{AC} is the amplitude of the AC voltage component.
- f is the frequency of the AC component.
- ϕ is the phase offset of the AC component.
- v_N is the noise voltage.

You can configure your source as DC-only, AC-only, or a combination of both. By default, both AC and DC components are set to 0. Define the AC/DC voltage by specifying nonzero parameter values after placing the block in your model.

The noise component is also optional. If you set the **Noise mode** parameter to `Enabled`, then the added noise voltage is given by:

$$v_N = \sqrt{P_v/2}\, \frac{N\,(0,1)}{\sqrt{h}}$$

where:

- P_v is the single-sided noise power spectral density for a 1 ohm load, in V^2/Hz.
- N is a Gaussian random number with zero mean and standard deviation of one.
- h is the sampling interval.

By default, the **Noise mode** parameter is set to `Disabled`, and the voltage source generates no thermal noise.

Noise Options

The block generates Gaussian noise by using the Random Number source in the Simscape Foundation library. You can control the random number seed by setting the **Repeatability** parameter:

- `Not repeatable` — Every time you simulate your model, the block resets the random seed using the MATLAB random number generator:

  `seed = randi(2^32-1);`
- `Repeatable` — The block automatically generates a seed value and stores it inside the block, to always start the simulation with the same random number. This auto-generated seed value is set when you add a Voltage Source block from the block library to the model. When you make a new copy of the Voltage Source block from an existing one in a model, a new seed value is generated. The block sets the value using the MATLAB random number generator command shown above.
- `Specify seed` — If you select this option, the additional **Seed** parameter lets you directly specify the random number seed value.

Basic Assumptions and Limitations

Simulating with noise enabled slows down simulation. Choose the sample time (h) so that noise is generated only at frequencies of interest, and not higher.

Parameters

- "DC & AC Components Tab" on page 1-936
- "Noise Tab" on page 1-936

DC & AC Components Tab

DC voltage

The DC component of the output voltage. The default value is `0` V. Enter a nonzero value to add a DC component to the voltage source.

AC voltage peak amplitude

Amplitude of the AC component of the output voltage. The default value is `0` V. Enter a nonzero value to add an AC component to the voltage source.

AC voltage phase shift

Phase offset of the AC component of the output voltage. The default value is `0` degrees.

AC voltage frequency

Frequency of the AC component of the output voltage. The default value is `60` Hz.

Noise Tab

Noise mode

Select the noise option:

- `Disabled` — No noise is produced by the voltage source. This is the default.
- `Enabled` — The voltage source generates thermal noise, and the associated parameters become visible on the **Noise** tab.

Power spectral density

The single-sided spectrum noise power. Strictly-speaking, this is a density function for the square of the voltage, commonly thought of as a power into a 1 ohm load, and therefore the units are V^2/Hz. To avoid this unit ambiguity, some datasheets quote noise voltage as a noise density with units of V/√Hz. In this case, you should enter the square of the noise density quoted in the datasheet as the parameter value. The default value is `0` V^2/Hz.

Sample time

Defines the rate at which the noise source is sampled. Choose it to reflect the frequencies of interest in your model. Making the sample time too small will unnecessarily slow down your simulation. The default value is `1e-3` s.

Repeatability

Select the noise control option:

- `Not repeatable` — The random sequence used for noise generation is not repeatable. This is the default.
- `Repeatable` — The random sequence used for noise generation is repeatable, with a system-generated seed.
- `Specify seed` — The random sequence used for noise generation is repeatable, and you control the seed by using the **Seed** parameter.

Auto-generated seed used for repeatable option

Random number seed stored inside the block to make the random sequence repeatable. This parameter is visible only if you select `Repeatable` for the **Repeatability** parameter. The parameter value is automatically generated using the MATLAB random number generator command. You can modify this parameter value, but it gets overwritten by a new random value if you copy the block to another block in the model. Therefore, if you want to control the seed of the random sequence, use the `Specify seed` option for the **Repeatability** parameter and specify the desired seed value using the **Seed** parameter.

Seed

Random number seed used by the noise random number generator. This parameter is visible only if you select `Specify seed` for the **Repeatability** parameter. The default value is `0`.

Ports

The block has the following ports:

+

Positive electrical port

-

Negative electrical port

See Also

Current Source | Resistor

Functions — Alphabetical List

elec_calculateFluxPartialDerivatives

Calculate flux partial derivatives for FEM-Parameterized PMSM block

Syntax

```
[dFdA,dFdB,dFdC,dFdX] = elec_calculateFluxPartialDerivatives(A,B,C,
X,F)
[dFdA,dFdB,dFdC,dFdX,D,Q] = elec_calculateFluxPartialDerivatives(A,
B,C,X,F)
```

Description

`[dFdA,dFdB,dFdC,dFdX] = elec_calculateFluxPartialDerivatives(A,B,C,
X,F)` calculates the partial derivatives from flux linkage. For improved numerical
performance, the FEM-Parameterized PMSM block works with flux linkage partial
derivatives, rather than directly with flux linkage. If your finite-element motor design tool
does not have an option to output partial derivatives, then you can use this function to
calculate the partial derivatives from the flux linkage. The flux linkage `F` must be a four-
dimensional matrix with the first three dimensions corresponding to the `A`, `B`, and `C` phase
currents, and the fourth dimension corresponding to the rotor angle `X`. The function
returns four-dimensional matrices for the four partial derivatives. Use this syntax in
conjunction with the 4-D Data variant of the block.

`[dFdA,dFdB,dFdC,dFdX,D,Q] = elec_calculateFluxPartialDerivatives(A,
B,C,X,F)` returns two additional output arguments corresponding to *d*-axis and *q*-axis
currents, respectively. In this case, the four partial derivatives are three-dimensional, the
first two dimensions corresponding to the *d*-axis and *q*-axis currents, and the third
dimension corresponding to the rotor angle. Use this syntax in conjunction with the 3-D
Data variant of the block.

Examples

Calculate 4-D flux linkage partial derivatives

Suppose F is a four-dimensional matrix containing flux linkage data, exported by your finite-element motor design tool. The matrix dimensions correspond to the three phase currents and the rotor angle, respectively. The data is cyclical in the fourth dimension, corresponding to the rotor angle.

Either directly import or recreate the current vectors. For example, if recreating a current vector with evenly spaced values between -250 and 250 A and 5 A increments, then:

```
iA = linspace(-250,250,5);
iB = iA;
iC = iA;
```

Import the rotor angle vector or recreate it based on the number of pole pairs.

```
X = pi/180*linspace(0,360/N,180/N+1);
```

Calculate the flux linkage partial derivatives.

```
[dFdA,dFdB,dFdC,dFdX] = elec_calculateFluxPartialDerivatives(iA,iB,iC,X,F);
```

The function returns four 4-D matrices for the flux linkage partial derivatives. The four matrices correspond to the three phase currents and the rotor angle, respectively. The matrix dimensions also correspond to the three phase currents and the rotor angle.

Calculate 3-D flux linkage partial derivatives

Suppose F is a four-dimensional matrix containing flux linkage data, exported by your finite-element motor design tool. The matrix dimensions correspond to the three phase currents and the rotor angle, respectively. The data is cyclical in the fourth dimension, corresponding to the rotor angle.

Either directly import or recreate the current vectors. For example, if recreating a current vector with evenly spaced values between -250 and 250 A and 5 A increments, then:

```
iA = linspace(-250,250,5);
iB = iA;
iC = iA;
```

Import the rotor angle vector or recreate it based on the number of pole pairs.

```
X = pi/180*linspace(0,360/N,180/N+1);
```

Calculate the flux linkage partial derivatives.

```
[dFdA,dFdB,dFdC,dFdX,iD,iQ] = elec_calculateFluxPartialDerivatives(iA,iB,iC,X,F);
```

The function returns four 3-D matrices for the flux linkage partial derivatives and two vectors for the *d*-axis and *q*-axis current values. The four matrices correspond to the three phase currents and the rotor angle, respectively. The matrix dimensions correspond to the *d*-axis and *q*-axis currents and the rotor angle.

Input Arguments

A — A-phase current, in amperes
vector

A-phase current, in amperes, specified as a vector. The vector must be monotonically increasing and two-sided (contain both positive and negative values). Best practice is to include zero current as one of the points.

Data Types: `double`

B — B-phase current, in amperes
vector

B-phase current, in amperes, specified as a vector. The vector must be monotonically increasing and two-sided (contain both positive and negative values). Best practice is to include zero current as one of the points.

Data Types: `double`

C — C-phase current, in amperes
vector

C-phase current, in amperes, specified as a vector. The vector must be monotonically increasing and two-sided (contain both positive and negative values). Best practice is to include zero current as one of the points.

Data Types: `double`

X — Rotor angle, in radians
vector

The rotor angle, in radians, specified as a vector. The values must be in the range from zero to 2π/*N*, where *N* is the number of pole pairs.

Data Types: `double`

F — Flux linkage, in weber-turns
four-dimensional matrix

The flux linkage, in weber-turns, specified as a four-dimensional matrix, with dimensions corresponding to the three phase currents and rotor angle. The data must be cyclic in the fourth (rotor angle) dimension, that is, for all *i*, *j*, and *k*, $F(i,j,k,0) = F(i,j,k,2π/N)$, where *N* is the number of pole pairs.

Data Types: `double`

Output Arguments

dFdA — Flux linkage partial derivative with respect to the A-phase current
matrix

Flux linkage partial derivative with respect to the A-phase current, returned as a matrix. For syntax used with the 4-D Data variant of the block, the matrix is four-dimensional. For syntax used with the 3-D Data variant of the block, the matrix is three-dimensional, the first two dimensions corresponding to the *d*-axis and *q*-axis currents, and the third dimension corresponding to the rotor angle.

dFdB — Flux linkage partial derivative with respect to the B-phase current
matrix

Flux linkage partial derivative with respect to the B-phase current, returned as a matrix. For syntax used with the 4-D Data variant of the block, the matrix is four-dimensional. For syntax used with the 3-D Data variant of the block, the matrix is three-dimensional, the first two dimensions corresponding to the *d*-axis and *q*-axis currents, and the third dimension corresponding to the rotor angle.

dFdC — Flux linkage partial derivative with respect to the C-phase current
matrix

Flux linkage partial derivative with respect to the C-phase current, returned as a matrix. For syntax used with the 4-D Data variant of the block, the matrix is four-dimensional. For syntax used with the 3-D Data variant of the block, the matrix is three-dimensional, the

first two dimensions corresponding to the *d*-axis and *q*-axis currents, and the third dimension corresponding to the rotor angle.

dFdX — Flux linkage partial derivative with respect to the rotor angle
matrix

Flux linkage partial derivative with respect to the rotor angle, returned as a matrix. For syntax used with the 4-D Data variant of the block, the matrix is four-dimensional. For syntax used with the 3-D Data variant of the block, the matrix is three-dimensional, the first two dimensions corresponding to the *d*-axis and *q*-axis currents, and the third dimension corresponding to the rotor angle.

D — D-axis current, in amperes
vector

D-axis current, in amperes, returned as a vector. This is an optional output argument, to be used when you want to generate 3-D flux linkage partial derivatives. The vector defines the *d*-axis current values at which the partial derivatives are determined.

Q — Q-axis current, in amperes
vector

Q-axis current, in amperes, returned as a vector. This is an optional output argument, to be used when you want to generate 3-D flux linkage partial derivatives. The vector defines the *q*-axis current values at which the partial derivatives are determined.

Algorithms

The function calculates partial derivatives using Akima splines, the same method that is used for `smooth` interpolation in the Simscape language `tablelookup` function. For more information, see "Smooth Interpolation Algorithm" (Simscape). Akima splines are suitable for estimating partial derivatives due to their smooth nature and tendency not to introduce local gradient reversals.

See Also
`elec_generateIdealPMSMfluxData`

Introduced in R2017a

elec_generateIdealPMSMfluxData

Generate tabulated flux linkage data for ideal PMSM

Syntax

```
[F,T,dFdA,dFdB,dFdC,dFdX] = elec_generateIdealPMSMfluxData(PM,Ld,Lq,
L0,A,B,C,X)
[F,T,dFdA,dFdB,dFdC,dFdX] = elec_generateIdealPMSMfluxData(PM,Ld,Lq,
L0,D,Q,X)
```

Description

`[F,T,dFdA,dFdB,dFdC,dFdX] = elec_generateIdealPMSMfluxData(PM,Ld,Lq, L0,A,B,C,X)` generates 4-D flux linkage data for an ideal permanent magnet synchronous motor (PMSM).

Use this function to create test data for the FEM-Parameterized PMSM block, either for validation purposes or to set up a model before the actual flux linkage data is available.

`[F,T,dFdA,dFdB,dFdC,dFdX] = elec_generateIdealPMSMfluxData(PM,Ld,Lq, L0,D,Q,X)` generates 3-D flux linkage data for an ideal PMSM.

Examples

Generate 4-D Flux Linkage Data

Specify the motor parameters.

```
PM = 0.1;      % Permanent magnet flux
N = 6;         % Number of pole pairs
Ld = 0.0002;   % D-axis inductance
Lq = 0.0002;   % Q-axis inductance
L0 = 0.00018;  % Zero-sequence inductance
Rs = 0.013;    % Stator resistance
```

Define the phase current vectors.

```
iA = linspace(-250,250,5);
iB = iA;
iC = iA;
```

Specify the rotor angle vector based on the number of pole pairs.

```
X = pi/180*linspace(0,360/N,180/N+1);
```

Tabulate flux linkage partial derivatives and torque in terms of A-,B-,C-currents and rotor angle.

```
[F,T,dFdA,dFdB,dFdC,dFdX] = elec_generateIdealPMSMfluxData(PM,Ld,Lq,L0,iA,iB,iC,X);
```

The function returns a 4-D flux linkage matrix F, a 4-D torque matrix T, and four 4-D matrices for the flux linkage partial derivatives. The four partial derivative matrices correspond to the three phase currents and the rotor angle, respectively. The matrix dimensions correspond to the three phase currents and the rotor angle.

Generate 3-D Flux Linkage Data

Specify the motor parameters.

```
PM = 0.1;      % Permanent magnet flux
N = 6;         % Number of pole pairs
Ld = 0.0002;   % D-axis inductance
Lq = 0.0002;   % Q-axis inductance
L0 = 0.00018;  % Zero-sequence inductance
Rs = 0.013;    % Stator resistance
```

Define the *d*-axis and *q*-axis current vectors.

```
iD = linspace(-250,250,5);
iQ = iD;
```

Specify the rotor angle vector based on the number of pole pairs.

```
X = pi/180*linspace(0,360/N,180/N+1);
```

Tabulate flux linkage partial derivatives and torque in terms of in terms of *d*-axis and *q*-axis currents and rotor angle.

```
[F,T,dFdA,dFdB,dFdC,dFdX] = elec_generateIdealPMSMfluxData(PM,Ld,Lq,L0,iD,iQ,X);
```

The function returns a 3-D flux linkage matrix F, a 3-D torque matrix T, and four 3-D matrices for the flux linkage partial derivatives. The four partial derivative matrices correspond to the three phase currents and the rotor angle, respectively. The matrix dimensions correspond to the *d*-axis and *q*-axis currents and the rotor angle.

- HEV PMSM Drive Test Harness
- PMSM Iron Losses

Input Arguments

PM — Peak permanent magnet flux linkage, in weber-turns
scalar

Peak permanent magnet flux linkage, in weber-turns, specified as a scalar.

Data Types: `double`

Ld — D-axis inductance, in henries
scalar

D-axis inductance, in henries, specified as a scalar.

Data Types: `double`

Lq — Q-axis inductance, in henries
scalar

Q-axis inductance, in henries, specified as a scalar.

Data Types: `double`

L0 — Zero-sequence inductance, in henries
scalar

Zero-sequence inductance, in henries, specified as a scalar.

Data Types: `double`

A — A-phase current, in amperes
vector

A-phase current, in amperes, specified as a vector. The vector must be monotonically increasing and two-sided (contain both positive and negative values). Best practice is to

include zero current as one of the points. Use this input argument to generate 4-D flux linkage data.

Data Types: `double`

B — B-phase current, in amperes
vector

B-phase current, in amperes, specified as a vector. The vector must be monotonically increasing and two-sided (contain both positive and negative values). Best practice is to include zero current as one of the points. Use this input argument to generate 4-D flux linkage data.

Data Types: `double`

C — C-phase current, in amperes
vector

C-phase current, in amperes, specified as a vector. The vector must be monotonically increasing and two-sided (contain both positive and negative values). Best practice is to include zero current as one of the points. Use this input argument to generate 4-D flux linkage data.

Data Types: `double`

D — D-axis current, in amperes
vector

D-axis current, in amperes, specified as a vector. The vector must be monotonically increasing and two-sided (contain both positive and negative values). Best practice is to include zero current as one of the points. Use this input argument to generate 3-D flux linkage data.

Data Types: `double`

Q — Q-axis current, in amperes
vector

Q-axis current, in amperes, specified as a vector. The vector must be monotonically increasing and two-sided (contain both positive and negative values). Best practice is to include zero current as one of the points. Use this input argument to generate 3-D flux linkage data.

Data Types: `double`

X — Rotor angle, in radians
vector

The rotor angle, in radians, specified as a vector. The values must be in the range from zero to 2π/*N*, where *N* is the number of pole pairs.

Data Types: `double`

Output Arguments

F — Flux linkage
matrix

The flux linkage, in weber-turns, returned as a matrix. The matrix can be four-dimensional or three-dimensional, depending on the syntax used to call the function. In a four-dimensional matrix, the first three dimensions correspond to the phase currents, and the fourth dimension corresponds to the rotor angle. In a three-dimensional matrix, the first two dimensions correspond to the *d*-axis and *q*-axis currents, and the third dimension corresponds to the rotor angle.

T — Torque
matrix

Torque, in N*m, returned as a matrix. The matrix can be four-dimensional or three-dimensional, depending on the syntax used to call the function. In a four-dimensional matrix, the first three dimensions correspond to the phase currents, and the fourth dimension corresponds to the rotor angle. In a three-dimensional matrix, the first two dimensions correspond to the *d*-axis and *q*-axis currents, and the third dimension corresponds to the rotor angle.

dFdA — Flux linkage partial derivative with respect to the A-phase current
matrix

Flux linkage partial derivative with respect to the A-phase current, returned as a matrix. The matrix can be four-dimensional or three-dimensional, depending on the syntax used to call the function. In a four-dimensional matrix, the first three dimensions correspond to the phase currents, and the fourth dimension corresponds to the rotor angle. In a three-dimensional matrix, the first two dimensions correspond to the *d*-axis and *q*-axis currents, and the third dimension corresponds to the rotor angle.

dFdB — Flux linkage partial derivative with respect to the B-phase current
matrix

Flux linkage partial derivative with respect to the B-phase current, returned as a matrix. The matrix can be four-dimensional or three-dimensional, depending on the syntax used to call the function. In a four-dimensional matrix, the first three dimensions correspond to the phase currents, and the fourth dimension corresponds to the rotor angle. In a three-dimensional matrix, the first two dimensions correspond to the d-axis and q-axis currents, and the third dimension corresponds to the rotor angle.

dFdC — Flux linkage partial derivative with respect to the C-phase current
matrix

Flux linkage partial derivative with respect to the C-phase current, returned as a matrix. The matrix can be four-dimensional or three-dimensional, depending on the syntax used to call the function. In a four-dimensional matrix, the first three dimensions correspond to the phase currents, and the fourth dimension corresponds to the rotor angle. In a three-dimensional matrix, the first two dimensions correspond to the d-axis and q-axis currents, and the third dimension corresponds to the rotor angle.

dFdX — Flux linkage partial derivative with respect to the rotor angle
matrix

Flux linkage partial derivative with respect to the rotor angle, returned as a matrix. The matrix can be four-dimensional or three-dimensional, depending on the syntax used to call the function. In a four-dimensional matrix, the first three dimensions correspond to the phase currents, and the fourth dimension corresponds to the rotor angle. In a three-dimensional matrix, the first two dimensions correspond to the d-axis and q-axis currents, and the third dimension corresponds to the rotor angle.

Algorithms

The flux linking each winding has contributions from the permanent magnet plus the three windings. Therefore, the total flux is given by [1] on page 2-13:

$$\begin{bmatrix} \psi_a \\ \psi_b \\ \psi_c \end{bmatrix} = \begin{bmatrix} L_{aa} & L_{ab} & L_{ac} \\ L_{ba} & L_{bb} & L_{bc} \\ L_{ca} & L_{cb} & L_{cc} \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} \psi_{am} \\ \psi_{bm} \\ \psi_{cm} \end{bmatrix}$$

$$L_{aa} = L_s + L_m \cos\left(2\theta_r\right)$$

$$L_{bb} = L_s + L_m \cos\left(2\left(\theta_r - 2\pi/3\right)\right)$$

$$L_{cc} = L_s + L_m \cos\left(2\left(\theta_r + 2\pi/3\right)\right)$$

$$L_{ab} = L_{ba} = -M_s - L_m \cos\left(\theta_r + \pi/6\right)$$

$$L_{bc} = L_{cb} = -M_s - L_m \cos\left(\theta_r + \pi/6 - 2\pi/3\right)$$

$$L_{ca} = L_{ac} = -M_s - L_m \cos\left(\theta_r + \pi/6 + 2\pi/3\right)$$

$$\psi_{am} = \psi_m \cos\theta_e$$

$$\psi_{bm} = \psi_m \cos\left(\theta_e - 2\pi/3\right)$$

$$\psi_{\boldsymbol{b}m} = \psi_m \cos\left(\theta_e + 2\pi/3\right)$$

Here, Θ_e is the electrical angle, which is related to rotor angle Θ_r by $\Theta_e = N \cdot \Theta_r$. The function assumes that the permanent magnet flux linking the A-phase winding is at the maximum for $\Theta_e = 0$.

The function output F corresponds to ψ_a tabulated as a function of A-phase current, B-phase current, C-phase current, and rotor angle.

Ls, *Lm*, and *Ms* are related to input arguments Ld, Lq, and L0 by:

$$L_s = \frac{L_0}{3} + \frac{L_d}{3} + \frac{L_q}{3}$$

$$M_s = \frac{L_d}{6} - \frac{L_0}{3} + \frac{L_q}{6}$$

$$L_m = \frac{L_d}{3} - \frac{L_q}{3}$$

References

[1] Anderson, P.M. *Analysis of Faulted Power Systems*. 1st Edition. Wiley-IEEE Press, July 1995, p.187.

See Also

elec_calculateFluxPartialDerivatives

Topics
HEV PMSM Drive Test Harness
PMSM Iron Losses

Introduced in R2017a

elec_getEfficiency

Calculate efficiency as function of dissipated power losses

Syntax

```
efficiency = elec_getEfficiency('loadIdentifier',node)
efficiency = elec_getEfficiency('loadIdentifier',node,startTime,
endTime)
[efficiency,lossesTable] = elec_getEfficiency('loadIdentifier',node)
```

Description

`efficiency = elec_getEfficiency('loadIdentifier',node)` returns the efficiency of a circuit based on the data extracted from a Simscape logging node.

Before you call this function, you must have the simulation log variable in your current workspace. Create the simulation log variable by simulating the model with data logging turned on, or load a previously saved variable from a file. If `node` is the name of the simulation log variable, then the table contains the data for all semiconductor blocks in the model. If `node` is the name of a node in the simulation data tree, then the table contains the data only for the blocks within that node.

Checking efficiency allows you to determine if circuit components are operating within their requirements. All blocks in the Semiconductor Devices library, as well as some other blocks, have an internal variable called `power_dissipated`, which represents the instantaneous power dissipated by the block. This instantaneous dissipated power includes only the real power (not the reactive or apparent power) that the block dissipates. When you log simulation data, the time-value series for this variable represents the power dissipated by the block over time. You can view and plot this data using the Simscape Results Explorer. The `elec_getPowerLossTimeSeries` function also allows you to access this data.

The `elec_getEfficiency` function calculates the efficiency of the circuit based on the losses for blocks that have a `power_dissipated` variable and that you identify as a load block. The equation for efficiency is

$$Eff = 100 \cdot \frac{P_{load}}{P_{loss} + P_{load}},$$

where:

- *Eff* is the efficiency of the circuit.
- P_{load} is the output power, that is, the power dissipated by load blocks.
- P_{loss} is the power dissipated by nonload blocks.

This equation assumes that all loss mechanisms are captured by blocks containing at least one `power_dissipated` variable. If the model contains any lossy blocks that do not have this variable, the efficiency calculation gives incorrect results.

Some blocks have more than one `power_dissipated` variable, depending on their configuration. For example, the N-Channel MOSFET block has separate `power_dissipated` logging nodes for the MOSFET, the gate resistor, and for the source and drain resistors if they have nonzero resistance values. The function sums all these losses to provide the total power loss for the block, averaged over simulation time. The function uses the loss data to calculate the efficiency of the circuit.

`efficiency = elec_getEfficiency('loadIdentifier',node,startTime, endTime)` returns the efficiency of a circuit based on the `power_dissipated` data extracted from a Simscape logging node within a time interval. `startTime` and `endTime` represent the start and end of the time interval for calculating the efficiency. If you omit these two input arguments, the function calculates the efficiency over the whole simulation time.

`[efficiency,lossesTable] = elec_getEfficiency('loadIdentifier',node)` returns the efficiency of a circuit and the power loss contributions of the nonload blocks in a circuit based on the data extracted from a Simscape logging node.

Examples

Calculate Efficiency for a Circuit

This example shows how to calculate efficiency based on the power dissipated by blocks in a circuit using the `elec_getEfficiency` function.

Open the model. At the MATLAB® command prompt, enter:

```
model = 'elec_converter_dcdc_class_e';
open_system(model)
```


Class E DC-DC Converter
1. Plot output voltage and switching frequency in system (see code)
2. Plot stress voltage on LDMOS (see code)
3. Calculate losses by component (see code)
4. Explore simulation results using sscexplore
5. Learn more about this example

The load in the model is represented by the R Load resistor. No other blocks with power_dissipated variables contain 'Load' in their names. Therefore, you can use the string 'Load' as the 'loadIdentifier' argument.

If no string at least partially matches the names of all load blocks in your circuit, rename the load blocks using a schema that satisfies the matching criteria for the 'loadIdentifier' argument.

This example model has data logging enabled. Run the simulation and create the simulation log variable.

```
sim(model)
```

The simulation log variable simlog_elec_converter_dcdc_class_e is saved in your current workspace.

Calculate efficiency and display the results.

```
efficiency = elec_getEfficiency('Load',simlog_elec_converter_dcdc_class_e)

efficiency =

    90.0326
```

Calculate Efficiency of a Circuit for a Specific Time Period

This example shows how to calculate efficiency based on the power dissipated for a specific time period using the `elec_getEfficiency` function.

Open the model. At the MATLAB® command prompt, enter:

```
model = 'elec_converter_dcdc_class_e';
open_system(model)
```


Class E DC-DC Converter
1. Plot output voltage and switching frequency in system (see code)
2. Plot stress voltage on LDMOS (see code)
3. Calculate losses by component (see code)
4. Explore simulation results using sscexplore
5. Learn more about this example

The load in the model is represented by the R Load resistor. No other blocks with `power_dissipated` variables contain `'Load'` in their names. Therefore, you can use the string `'Load'` as the `'loadIdentifier'` argument.

If no string at least partially matches the names of all load blocks in your circuit, rename the load blocks using a schema that satisfies the matching criteria for the `'loadIdentifier'` argument.

This example model has data logging enabled. Run the simulation and create the simulation log variable.

```
sim(model)
```

The simulation log variable `simlog_elec_converter_dcdc_class_e` is saved in your current workspace.

The model simulation time (*t*) is 1.25e-4 seconds. Calculate efficiency for the interval when *t* is between 1e-4 and 1.25e-4 seconds.

```
efficiency = elec_getEfficiency('Load',simlog_elec_converter_dcdc_class_e,1e-4,1.25e-4)

efficiency =

    90.4899
```

Calculate Efficiency and Power-Loss Contributions

This example shows how using the `elec_getEfficiency` function allows you to calculate both the efficiency of the circuit and the power-loss contributions of the nonload blocks based on the power that they dissipate.

Open the model. At the MATLAB® command prompt, enter:

```
model = 'elec_converter_dcdc_class_e';
open_system(model)
```

Class E DC-DC Converter
1. Plot output voltage and switching frequency in system (see code)
2. Plot stress voltage on LDMOS (see code)
3. Calculate losses by component (see code)
4. Explore simulation results using sscexplore
5. Learn more about this example

The load in the model is represented by the R Load resistor. No other blocks with `power_dissipated` variables contain `'Load'` in their names. Therefore, you can use the string `'Load'` as the `'loadIdentifier'` argument.

If no string at least partially matches the names of all load blocks in your circuit, rename the load blocks using a schema that satisfies the matching criteria for the `'loadIdentifier'` argument.

This example model has data logging enabled. Run the simulation and create the simulation log variable.

```
sim(model)
```

The simulation log variable `simlog_elec_converter_dcdc_class_e` is saved in your current workspace.

Calculate the efficiency and power-loss contributions due to dissipated power.

```
[efficiency,lossesTable] = elec_getEfficiency('Load',simlog_elec_converter_dcdc_class_e)
```

```
efficiency =
```

```
    90.0326


lossesTable =

  7×2 table array

                    LoggingNode                            Power
    _____    _____

    'elec_converter_dcdc_class_e.LDMOS'                    3.6583
    'elec_converter_dcdc_class_e.R_Trans.Resistor'          2.911
    'elec_converter_dcdc_class_e.D2.diode'                 1.9446
    'elec_converter_dcdc_class_e.D1.diode'                  1.837
    'elec_converter_dcdc_class_e.Cs'                      0.27391
    'elec_converter_dcdc_class_e.Ls'                      0.27097
    'elec_converter_dcdc_class_e.Cout'                  0.00044593
```

Input Arguments

'loadIdentifier' — Identify load blocks in the circuit
case-sensitive string

String that is a complete or partial match for the names of load blocks in the circuit. For example, consider a circuit that contains the four semiconductor blocks shown in the table.

Block Name in the Model		IGBT	IGBT1_Load	Diode	Diode1
Block Type		N-Channel IGBT	N-Channel IGBT	Diode	Diode
Block Role in the Model		Source	Load	Load	Load
'loadIdentifier'	'IGBT'	Yes	Yes	No	No
	'Diode'	No	No	Yes	Yes
	'Load'	No	Yes	No	No
	'1'	No	Yes	No	Yes
	'D'	No	No	Yes	Yes
	'd'	No	Yes	Yes	Yes

The `elec_getEfficiency` function returns data just for the three load blocks only when the `'loadIdentifier'` is `'d'`.

A load-block naming schema that gives you better control over the output of the `elec_getEfficiency` function is shown in this table.

Block Name in the Model		IGBT	IGBT1_Load	Diode_Load	Diode1_Load
Block Type		N-Channel IGBT	N-Channel IGBT	Diode	Diode
Block Role in the Model		Source	Load	Load	Load
`'loadIdentifier'`	`'IGBT'`	Yes	Yes	No	No
	`'Diode'`	No	No	Yes	Yes
	`'Load'`	No	Yes	Yes	Yes

Example: `'Load'`

Data Types: `string`

node — Simulation log variable, or a specific node within the simulation log variable
Node object

Simulation log workspace variable, or a node within this variable, that contains the logged model simulation data, specified as a `Node` object. You specify the name of the simulation log variable by using the **Workspace variable name** parameter on the **Simscape** pane of the Configuration Parameters dialog box. To specify a node within the simulation log variable, provide the complete path to that node through the simulation data tree, starting with the top-level variable name.

If `node` is the name of the simulation log variable, then the table contains the data for all blocks in the model that contain `power_dissipated` variables. If `node` is the name of a node in the simulation data tree, then the table contains the data only for:

- Blocks or variables within that node
- Blocks or variables within subnodes at all levels of the hierarchy beneath that node

Example: `simlog.Cell1.MOS1`

startTime — Start of the time interval for calculating the efficiency
0 (default) | real number

Start of the time interval for calculating the efficiency, specified as a real number, in seconds. `startTime` must be greater than or equal to the simulation **Start time** and less than `endTime`.

Data Types: `double`

endTime — End of the time interval for calculating the efficiency
simulation stop time (default) | real number

End of the time interval for calculating the efficiency, specified as a real number, in seconds. `endTime` must be greater than `startTime` and less than or equal to the simulation **Stop time**.

Data Types: `double`

Output Arguments

efficiency — Efficiency of the circuit
percentage

Efficiency of the circuit based on data extracted from a Simscape logging node.

lossesTable — Dissipated power for each nonload blocks
table

Dissipated power losses for each nonload block, returned as a table. The first column lists logging nodes for all blocks that have at least one `power_dissipated` variable. The second column lists the corresponding losses in watts.

Assumptions

- The output power equals the total power dissipated by blocks that you identify as load blocks.
- The input power equals the output power plus the total power dissipated by blocks that you do not identify as load blocks.
- The `power_dissipated` variables capture all loss contributions.

See Also

elec_getPowerLossSummary | elec_getPowerLossTimeSeries

Topics

"About Simulation Data Logging" (Simscape)
"About the Simscape Results Explorer" (Simscape)

Introduced in R2017a

elec_getPowerLossSummary

Calculate dissipated power losses

Syntax

```
lossesTable = elec_getPowerLossSummary(node)
lossesTable = elec_getPowerLossSummary(node,startTime,endTime)
```

Description

`lossesTable = elec_getPowerLossSummary(node)` calculates dissipated power losses for semiconductor blocks in a model, based on logged simulation data, and returns the data for each block in a table.

Before you call this function, you must have the simulation log variable in your current workspace. Create the simulation log variable by simulating the model with data logging turned on, or load a previously saved variable from a file. If `node` is the name of the simulation log variable, then the table contains the data for all semiconductor blocks in the model. If `node` is the name of a node in the simulation data tree, then the table contains the data only for the blocks within that node.

Checking dissipated power is useful for verifying that circuit components are operating within their working envelopes. All blocks in the Semiconductor Devices library, as well as some other blocks, have an internal variable called `power_dissipated`, which represents the instantaneous power dissipated by the block. When you log simulation data, the time-value series for this variable represents the power dissipated by the block over time. You can view and plot this data using the Simscape Results Explorer.

The `elec_getPowerLossSummary` function calculates average losses for each block that has a `power_dissipated` variable. Some blocks have more than one `power_dissipated` variable, depending on their configuration. For example, the N-Channel MOSFET block has separate `power_dissipated` logging nodes for the MOSFET, the gate resistor, and for the source and drain resistors if they have nonzero resistance values. The function sums all these losses and provides the power loss value for the whole block, averaged over simulation time.

```
lossesTable = elec_getPowerLossSummary(node,startTime,endTime)
```
calculates dissipated power losses within a time interval. `startTime` and `endTime` represent the start and end of the time interval for averaging the power losses. If you omit these two input arguments, the function averages the power losses over the whole simulation time.

Examples

Calculate Power Losses by Block for the Whole Model

Open the Solar Power Converter example model.
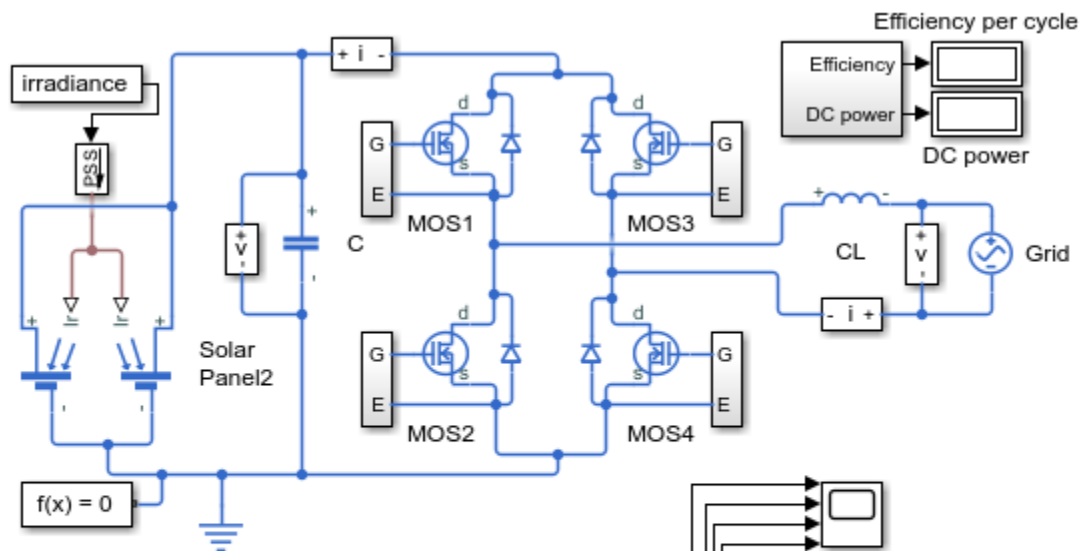
```
elec_solar_converter
```


Solar Power Converter
1. Plot power dissipated by MOSFETs (see code)
2. Calculate losses by component (see code)
3. Explore simulation results using sscexplore
4. Learn more about this example
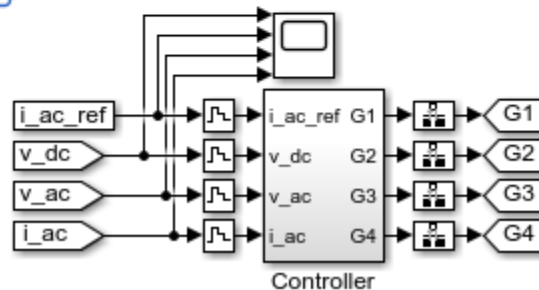
This example model has data logging enabled. Run the simulation to create the simulation log variable `simlog_solar_converter` in your current workspace.

```
sim('elec_solar_converter');
```

Calculate power losses for the whole model and display the results in a table.

```
tabulatedLosses = elec_getPowerLossSummary(simlog_elec_solar_converter)
```

```
tabulatedLosses =

  8x2 table

                LoggingNode                    Power
    _____    _____

    'elec_solar_converter.MOS4'               17.566
    'elec_solar_converter.MOS2'               17.088
    'elec_solar_converter.MOS3'               15.011
    'elec_solar_converter.MOS1'               14.604
    'elec_solar_converter.Diode2.diode'       3.0928
    'elec_solar_converter.Diode4.diode'       2.9378
    'elec_solar_converter.Diode1.diode'       2.0042
    'elec_solar_converter.Diode3.diode'       1.8658
```

The table shows dissipated power losses for each of the four N-Channel MOSFET and four Diode blocks, averaged over the whole simulation time.

**Calculate Power Losses for One Block**

Open the Solar Power Converter example model.
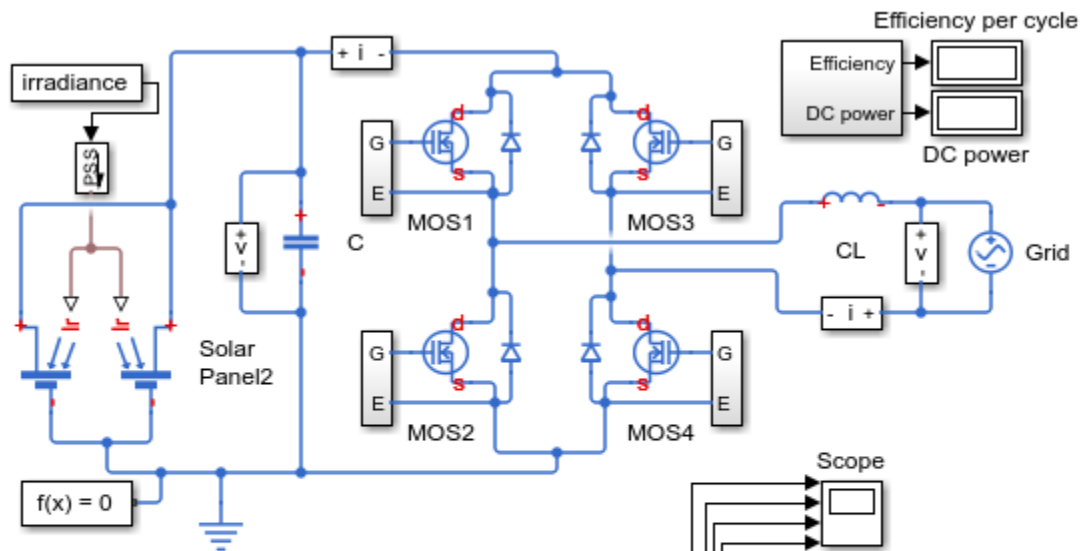
```
elec_solar_converter
```

Solar Power Converter

1. Plot power dissipated by MOSFETs (see code)
2. Calculate losses by component (see code)
3. Explore simulation results using sscexplore
4. Learn more about this example

This example model has data logging enabled. Run the simulation to create the simulation log variable `simlog_elec_solar_converter` in your current workspace.

```
sim('elec_solar_converter');
```
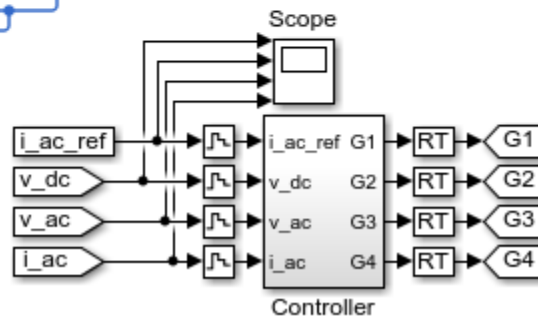
Calculate power losses for the MOS1 block.

```
mosfetLosses = elec_getPowerLossSummary(simlog_elec_solar_converter.MOS1)
```

```
mosfetLosses =

    LoggingNode    Power
    _____    _____

    'MOS1'         15.316
```

The table shows dissipated power losses for the MOS1 block, averaged over the whole simulation time.

Use the `sscexplore` function to further explore the power loss data for the MOSFET block.

`sscexplore(simlog_elec_solar_converter.MOS1)`
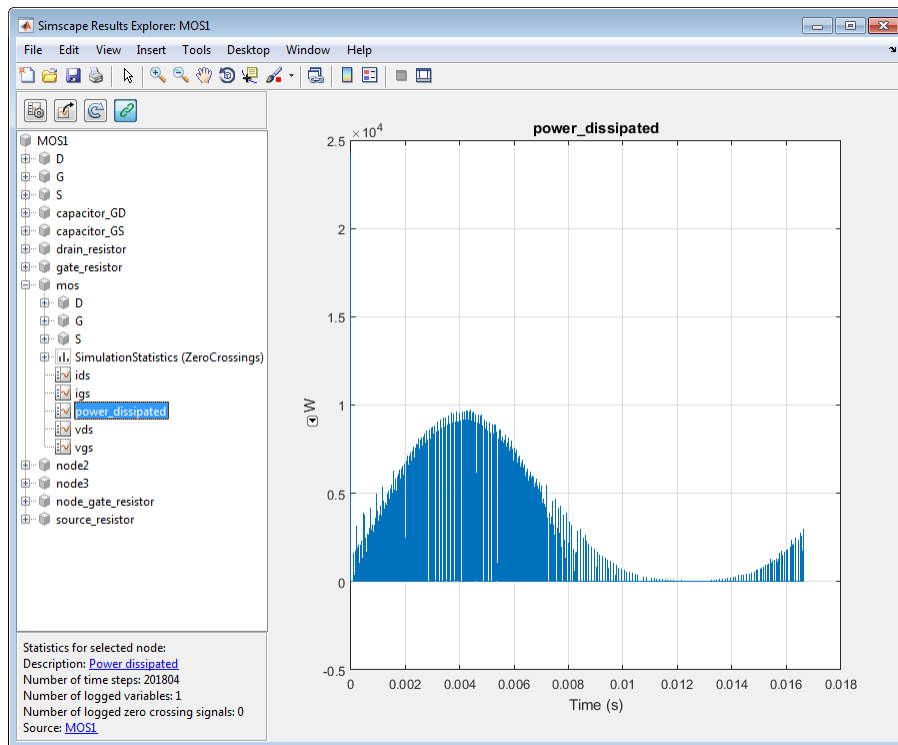
The block has several `power_dissipated` logging nodes: under `drain_resistor`, under `gate_resistor`, under `mos`, and under `source_resistor`. The `15.316` value calculated by the `elec_getPowerLossSummary` function is a sum of all these losses, averaged over the simulation time.

Calculate Power Losses for a Specific Time Period

Open the Solar Power Converter example model.
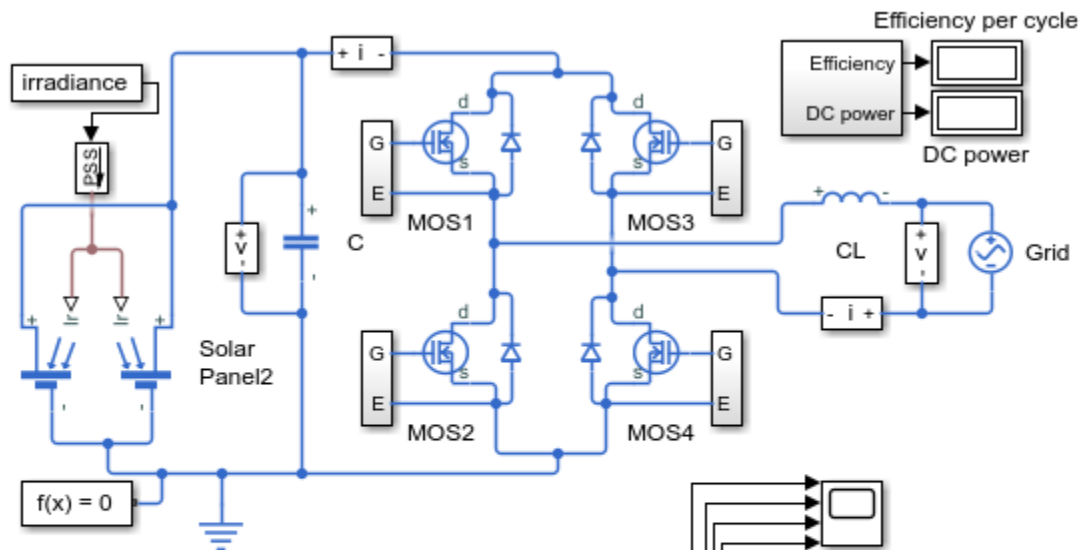
elec_solar_converter

Solar Power Converter

1. Plot power dissipated by MOSFETs (see code)
2. Calculate losses by component (see code)
3. Explore simulation results using sscexplore
4. Learn more about this example
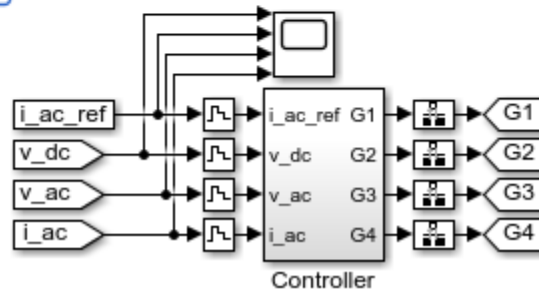
This example model has data logging enabled. Run the simulation to create the simulation log variable `simlog_elec_solar_converter` in your current workspace.

```
sim('elec_solar_converter');
```

The model simulation time is 1/60th of a second. Calculate average power losses for the MOS1 block during the first 1/120th of a second.

```
mosfetLosses1 = elec_getPowerLossSummary(simlog_elec_solar_converter.MOS1,0,1/120)
```

```
mosfetLosses1 =

  1x2 table
```

```
LoggingNode     Power
_____     _____

   'MOS1'       29.202
```

The table shows the average dissipated power losses for the MOS1 block during the first half of the simulation cycle. It is almost twice as high as that for the whole simulation time.

Input Arguments

node — Simulation log variable, or a specific node within the simulation log variable
Node object

Simulation log workspace variable, or a node within this variable, that contains the logged model simulation data, specified as a Node object. You specify the name of the simulation log variable by using the **Workspace variable name** parameter on the **Simscape** pane of the Configuration Parameters dialog box. To specify a node within the simulation log variable, provide the complete path to that node through the simulation data tree, starting with the top-level variable name.

Example: simlog.Cell1.MOS1

startTime — Start of the time interval for averaging dissipated power losses
real number

Start of the time interval for averaging dissipated power losses, specified as a real number, in seconds. startTime must be greater than or equal to the simulation **Start time** and less than endTime.

Data Types: double

endTime — End of the time interval for averaging dissipated power losses
real number

End of the time interval for averaging dissipated power losses, specified as a real number, in seconds. endTime must be greater than startTime and less than or equal to the simulation **Stop time**.

Data Types: `double`

Output Arguments

`lossesTable` — Dissipated power losses for each block
table

Dissipated power losses for each block, returned as a table. The first column lists logging nodes for all blocks that have at least one `power_dissipated` variable. The second column lists the corresponding losses in watts.

See Also

`elec_getEfficiency` | `elec_getPowerLossTimeSeries` | `sscexplore`

Topics
"About Simulation Data Logging" (Simscape)
"About the Simscape Results Explorer" (Simscape)

Introduced in R2015a

elec_getPowerLossTimeSeries

Calculate dissipated power losses and return time series data

Syntax

```
lossesCell = elec_getPowerLossTimeSeries(node)
lossesCell = elec_getPowerLossTimeSeries(node,startTime,endTime)
lossesCell = elec_getPowerLossTimeSeries(node,startTime,endTime,
intervalWidth)
```

Description

`lossesCell = elec_getPowerLossTimeSeries(node)` calculates dissipated power losses for blocks in a model, based on logged simulation data, and returns the time series data for each block.

Before you call this function, you must have the simulation log variable in your current workspace. Create the simulation log variable by simulating the model with data logging turned on, or load a previously saved variable from a file.

The `elec_getPowerLossTimeSeries` function calculates dissipated power losses for each block that has a `power_dissipated` variable. All blocks in the Semiconductor Devices library, as well as some other blocks, have an internal variable called `power_dissipated`, which represents the instantaneous power dissipated by the block. Some blocks have more than one `power_dissipated` variable, depending on their configuration. For example, the N-Channel MOSFET block has separate `power_dissipated` logging nodes for the MOSFET, the gate resistor, and for the source and drain resistors if they have nonzero resistance values. The function sums all these losses and provides the power loss value for all of the blocks as functions of time.

If `node` is the name of the simulation log variable, then the table contains the data for all the blocks in the model that dissipate power (that is, contain at least one `power_dissipated` variable). If `node` is the name of a node in the simulation data tree, then the table contains the data only for the blocks within that node.

`lossesCell = elec_getPowerLossTimeSeries(node,startTime,endTime)` calculates dissipated power losses and returns the time series data for time steps from

startTime to endTime. If startTime is equal to endTime, the interval is effectively zero and the function returns the instantaneous power for the time step that occurs at that moment. If you omit these two input arguments, the function returns data over the whole simulation time.

lossesCell = elec_getPowerLossTimeSeries(node,startTime,endTime, intervalWidth) calculates dissipated power losses and returns the time series data for time steps from startTime to endTime, averaged over the time intervalWidth. If you omit the intervalWidth, or set it to 0, the function returns the instantaneous data, without averaging. If you omit all three optional arguments, the function returns the instantaneous data over the whole simulation time.

Examples

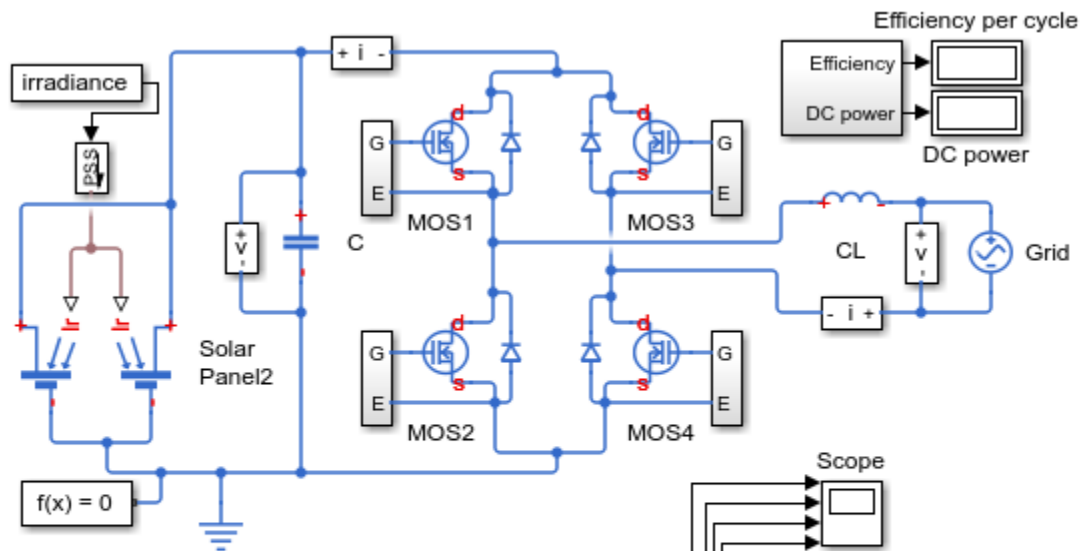### Calculate Dissipated Power Losses for the Entire Simulation Time

This example shows how to calculate instantaneous losses based on the power dissipated and return the time series data for all time steps in the entire simulation time using the elec_getPowerLossTimeSeries function.

Open the model. At the MATLAB® command prompt, enter:

```
model = 'elec_solar_converter';
open_system(model)
```

Solar Power Converter
1. Plot power dissipated by MOSFETs (see code)
2. Calculate losses by component (see code)
3. Explore simulation results using sscexplore
4. Learn more about this example

This example model has data logging enabled. Run the simulation and create the simulation log variable.

```
sim(model)
```
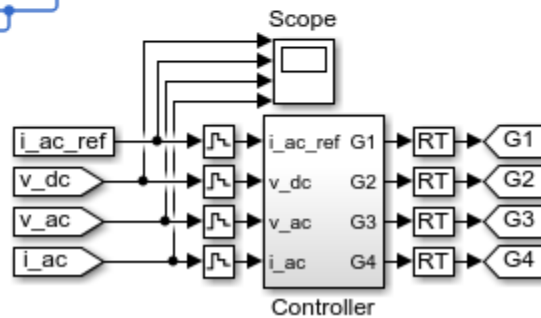
The simulation log variable `simlog_elec_solar_converter` is saved in your current workspace.

Calculate dissipated power losses and return the time series data in cell array.

```
lossesCell = elec_getPowerLossTimeSeries(simlog_elec_solar_converter)
```

```
lossesCell =

  8×2 cell array
```

```
'elec_solar_converter.Diode1.diode'     [201804×3 double]
'elec_solar_converter.MOS1'             [201804×3 double]
'elec_solar_converter.MOS2'             [201804×3 double]
'elec_solar_converter.MOS3'             [201804×3 double]
'elec_solar_converter.MOS4'             [201804×3 double]
'elec_solar_converter.Diode2.diode'     [201804×3 double]
'elec_solar_converter.Diode3.diode'     [201804×3 double]
'elec_solar_converter.Diode4.diode'     [201804×3 double]
```

View the time series data. From the workspace, open the `lossesCell` cell array, then open the `201804x3 double` numeric array for the `elec_solar_converter.Diode1.diode`.

The first two columns contain the interval start and end time. The third column contains the power loss data.

Plot the data.

```
plot(lossesCell{1, 2}(:,end))
title('Dissipated Power')
xlabel('Time Interval')
ylabel('Power (W)')
```
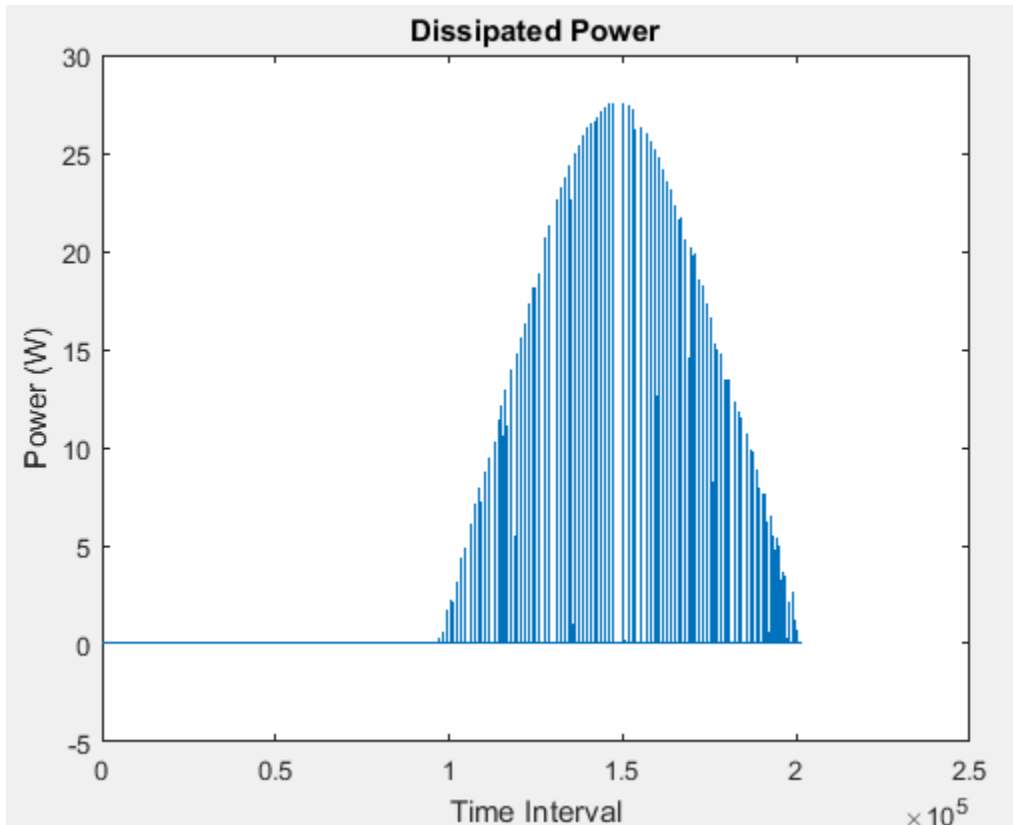
Calculate Dissipated Power Losses for a Specific Time Period

This example shows how to calculate instantaneous losses based on the power dissipated and return the time series data for all time steps in a specific time period using the `elec_getPowerLossTimeSeries` function.
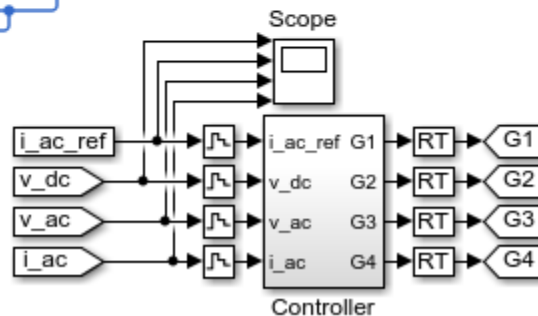
Open the model. At the MATLAB® command prompt, enter:

```
model = 'elec_solar_converter';
open_system(model)
```

Solar Power Converter

1. Plot power dissipated by MOSFETs (see code)
2. Calculate losses by component (see code)
3. Explore simulation results using sscexplore
4. Learn more about this example

This example model has data logging enabled. Run the simulation and create the simulation log variable.

```
sim(model)
```

The simulation log variable `simlog_elec_solar_converter` is saved in your current workspace.

The model simulation time (*t*) is 1/60 seconds. Calculate dissipated power losses and return the time series data in cell array for the second half of the simulation cycle, when *t* is between 1/120 and 1/60 seconds..

```
lossesCell = elec_getPowerLossTimeSeries(simlog_elec_solar_converter,1/120,1/60)
```

```
lossesCell =
```

```
8×2 cell array

    'elec_solar_converter.Diode1.diode'     [105197×3 double]
    'elec_solar_converter.MOS1'             [105197×3 double]
    'elec_solar_converter.MOS2'             [105197×3 double]
    'elec_solar_converter.MOS3'             [105197×3 double]
    'elec_solar_converter.MOS4'             [105197×3 double]
    'elec_solar_converter.Diode2.diode'     [105197×3 double]
    'elec_solar_converter.Diode3.diode'     [105197×3 double]
    'elec_solar_converter.Diode4.diode'     [105197×3 double]
```

View the time series data. From the workspace, open the `lossesCell` cell array, then open the `105197x3 double` numeric array for the `elec_solar_converter.Diode1.diode`.

The first two columns contain the interval start and end time. The third column contains the power loss data.

Plot the data.

```
plot(lossesCell{1, 2}(:,end))
title('Dissipated Power')
xlabel('Time Interval')
ylabel('Power (W)')
```
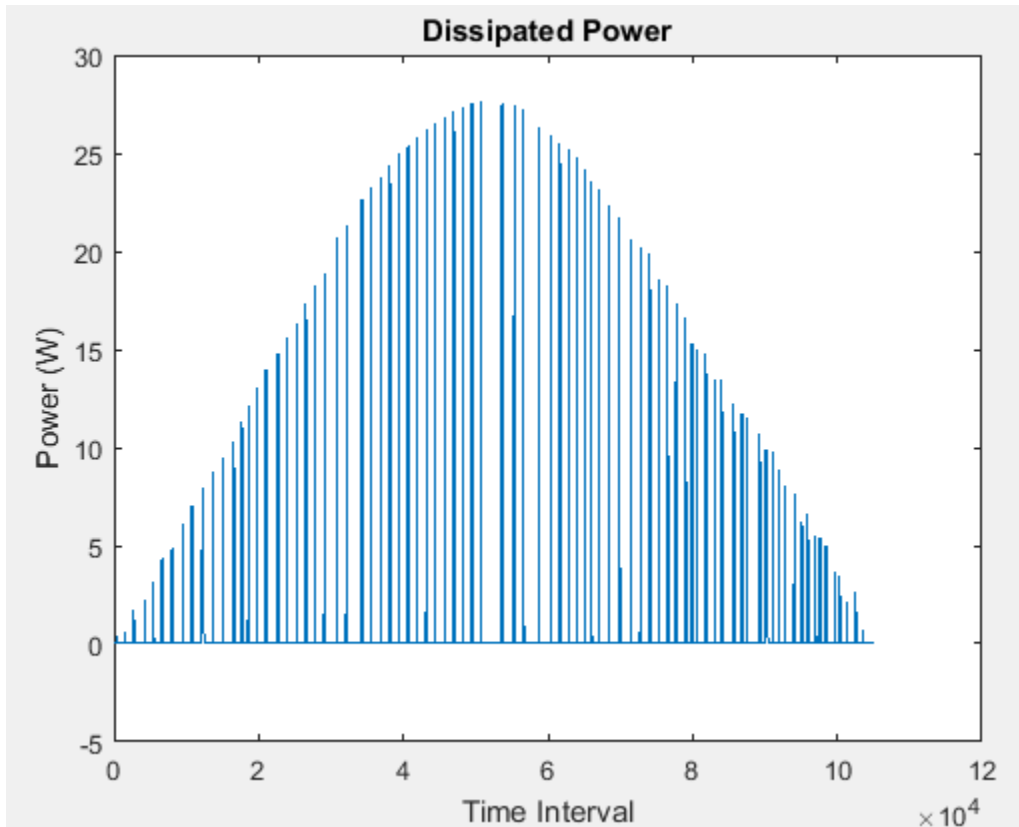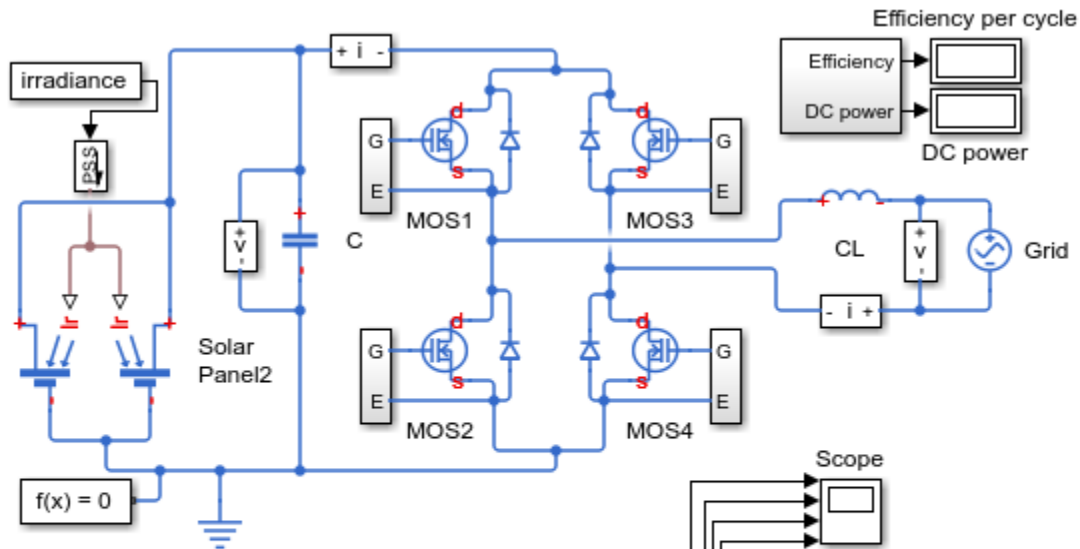
Calculate Dissipated Power Losses Using Specific Interval Widths

This example shows how to calculate losses based on the power dissipated and return the time series data for a specific time period with averaging applied over intervals of a specified width.
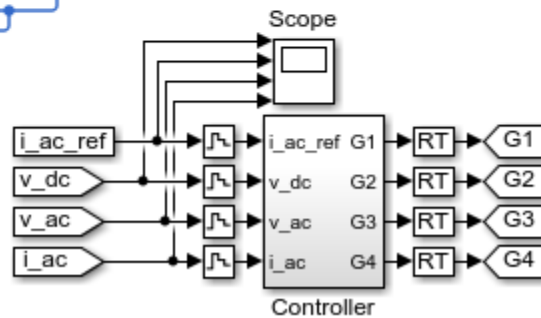
Open the model. At the MATLAB® command prompt, enter:

```
model = 'elec_solar_converter';
open_system(model)
```

Solar Power Converter
1. Plot power dissipated by MOSFETs (see code)
2. Calculate losses by component (see code)
3. Explore simulation results using sscexplore
4. Learn more about this example

This example model has data logging enabled. Run the simulation and create the simulation log variable.

```
sim(model)
```

The simulation log variable `simlog_elec_solar_converter` is saved in your current workspace.

The model simulation time (*t*) is 1/60 seconds. Calculate the average dissipated power losses for 1.1e-4 s intervals and return the time series data in cell array for the period when simulation time, *t,* is 0.008–0.017 seconds.

```
lossesCell = elec_getPowerLossTimeSeries(simlog_elec_solar_converter,0.008,0.016,1.1e-4)
```

```
lossesCell =
```
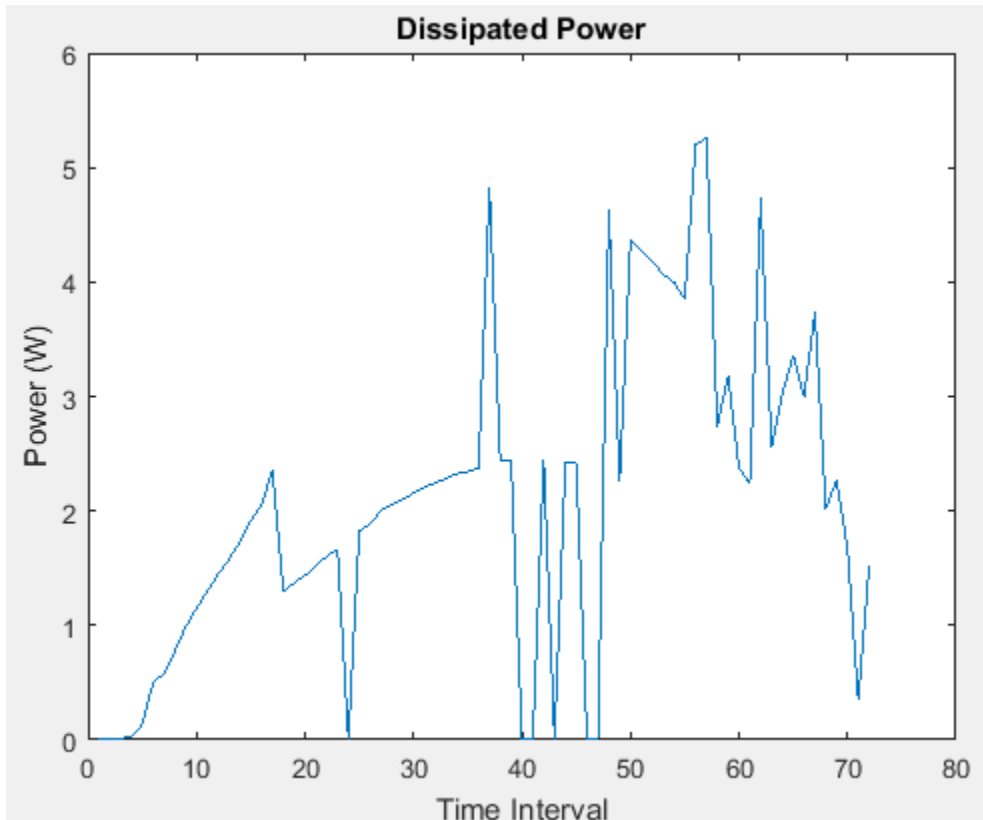
```
8×2 cell array

   'elec_solar_converter.Diode1.diode'     [72×3 double]
   'elec_solar_converter.MOS1'             [72×3 double]
   'elec_solar_converter.MOS2'             [72×3 double]
   'elec_solar_converter.MOS3'             [72×3 double]
   'elec_solar_converter.MOS4'             [72×3 double]
   'elec_solar_converter.Diode2.diode'     [72×3 double]
   'elec_solar_converter.Diode3.diode'     [72×3 double]
   'elec_solar_converter.Diode4.diode'     [72×3 double]
```

View the time series data. From the workspace, open the `lossesCell` cell array, then open the `105197x3 double` numeric array for the `elec_solar_converter.Diode1.diode`.

The first two columns contain the interval start and end time. The third column contains the power loss data. In this case, to use averaging intervals that are equal in width to 1.1e-4 seconds, the function adjusts the start time for the first interval from the specified value of 0.008 seconds to a value of 0.0081 seconds. There are 72 intervals of 1.1e-4 seconds.

Plot the data.

```
plot(lossesCell{1, 2}(:,end))
title('Dissipated Power')
xlabel('Time Interval')
ylabel('Power (W)')
```

Input Arguments

node — Simulation log variable, or a specific node within the simulation log variable
Node object

Simulation log workspace variable, or a node within this variable, that contains the logged model simulation data, specified as a Node object. You specify the name of the simulation log variable by using the **Workspace variable name** parameter on the **Simscape** pane of the Configuration Parameters dialog box. To specify a node within the simulation log variable, provide the complete path to that node through the simulation data tree, starting with the top-level variable name.

If `node` is the name of the simulation log variable, then the table contains the data for all blocks in the model that contain `power_dissipated` variables. If `node` is the name of a node in the simulation data tree, then the table contains the data only for:

- Blocks or variables within that node
- Blocks or variables within subnodes at all levels of the hierarchy beneath that node

Example: `simlog.Cell1.MOS1`

`startTime` — Start of the time interval for calculating the data
0 (default) | real number

Start of the time interval for calculating the power loss time series, specified as a real number, in seconds. `startTime` must be greater than or equal to the simulation **Start time** and less than `endTime`.

Data Types: `double`

`endTime` — End of the time interval for calculating the data
simulation stop time (default) | real number

End of the time interval for calculating the power loss time series, specified as a real number, in seconds. `endTime` must be greater than `startTime` and less than or equal to the simulation **Stop time**.

Data Types: `double`

`intervalWidth` — Size of the time interval for calculating the average power dissipation
0 (default) | real number

Size of the time interval for calculating the average power dissipation, specified as a real number, in seconds. If specified, the function returns data for time steps from `startTime` to `endTime`, averaged over the `intervalWidth`. If you omit the `intervalWidth` argument, or set it to 0, the function returns the instantaneous data, without averaging. If all the optional arguments are omitted, the function returns the instantaneous data over the whole simulation time.

If the time between the specified `startTime` and `endTime` is not an integer multiple of `intervalWidth`, the function adjusts the start time. The figure shows how the function adjusts the start time to ensure that width of each time interval that the dissipated power is averaged over is equal to the specified `intervalWidth`.
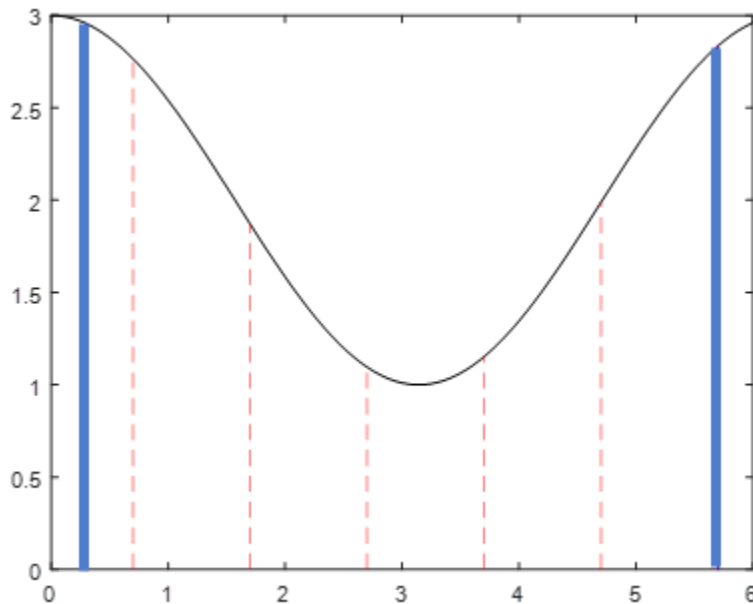
The black line is an example of the instantaneous *power_dissipated* variables summed over all elements in an individual block. The simulation runs for 6 seconds. The `startTime` and `endTime` are indicated by the solid blue lines. The `intervalWidth` is set to 1 second. There are five intervals as indicated by the red dashed lines. The right-most edge of the last interval coincides with `endTime`. The left-most edge of the first interval is always greater than or equal to `startTime`. The edge is equal to `startTime` only if (`endTime` -`startTime`)/`intervalWidth` is an integer. The output in this case consists of five values for the averaged power dissipation, one point for each time period. The function outputs the actual start and stop times in the tabulated output data.

Example: `1.1e-3`

Data Types: `double`

Output Arguments

`lossesCell` — Time series of the dissipated power losses for each block
cell array

Cell array that contains the names of the blocks in the nodes that contain *power_dissipated* variables and, for each block, a three-column array:

- Column one contains the interval start time.
- Column two contains the interval end time.
- Column three contains the dissipated power for the time interval.

If the interval width is 0 seconds, that is, the start time is equal to the end time, then the dissipated power is the instantaneous power loss. If the interval is greater than 0 seconds, the dissipated power is the average power loss for the time of the interval.

See Also
`elec_getEfficiency` | `elec_getPowerLossSummary`

Topics
"About Simulation Data Logging" (Simscape)
"About the Simscape Results Explorer" (Simscape)

Introduced in R2017a